Development of an Augmented Reality Guided Computer Assisted Orthopaedic Surgery System

Nicholas L. Smith

This thesis is submitted in partial fulfilment of the requirments for the Degree of EngD in Medical Devices.

Jan 2016

Declaration

This thesis is the result of the original work of the author. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Word count: 94037

Signed: Date:

Acknowledgments

I would like to offer my sincerest gratitude to all those who have helped and supported me throughout my studies.

Special thanks must be given to my supervisors, Dr Philip Riches and Dr Vladimir Stankovic. Their continued advice and willingness to allow me to explore different ideas throughout this project have helped me develop as a researcher.

I would also like to thank Dr William Kerr for providing an unending source of stimulating discussion, of both work and beyond.

Furthermore, I would like to express my gratitude to my parents, grandparents, and brothers for their unwavering faith in my ability, in spite of my own doubts.

Finally, I am eternally grateful to my wife, Elizabeth, for her love and support throughout everything, and her understanding of my long days and short breaks.

Abstract

This body of work documents the developed of a proof of concept augmented reality guided computer assisted orthopaedic surgery system – ARgCAOS.

After initial investigation a visible-spectrum single camera tool-mounted tracking system based upon fiducial planar markers was implemented. The use of visible-spectrum cameras, as opposed to the infra-red cameras typically used by surgical tracking systems, allowed the captured image to be streamed to a display in an intelligible fashion. The tracking information defined the location of physical objects relative to the camera. Therefore, this information allowed virtual models to be overlaid onto the camera image. This produced a convincing augmented experience, whereby the virtual objects appeared to be within the physical world, moving with both the camera and markers as expected of physical objects.

Analysis of the first generation system identified both accuracy and graphical inadequacies, prompting the development of a second generation system. This too was based upon a tool-mounted fiducial marker system, and improved performance to near-millimetre probing accuracy. A resection system was incorporated into the system, and utilising the tracking information controlled resection was performed, producing sub-millimetre accuracies.

Several complications resulted from the tool-mounted approach. Therefore, a third generation system was developed. This final generation deployed a stereoscopic visible-spectrum camera system affixed to a head-mounted display worn by the user. The system allowed the augmentation of the natural view of the user, providing convincing and immersive three dimensional augmented guidance, with probing and resection accuracies of 0.55±0.04 and 0.34±0.04 mm, respectively.

Table of Contents

List of Abreviations

AC	Articulator cartilage
API	Application program interface
AR	Augmented reality
BM	Block matching
DLL	Dynamically linked library
EM	Electromagnetic
FOV	Field of view
FPS	Frames per second
GAG	Glycosaminoglycan
GLUT	OpenGL utility toolkit
GPU	Graphical processor unit
HJC	Hip joint centre
HKA	Hip-knee-ankle
HLSL	High-level shader language
HMD	Head-mounted display
HUD	Head up display
ICP	Iterative closest point
IPD	Interpupillary distance
IR	Infrared
KA	Knee arthroplasty
KOA	OA of the knee
KR	Knee replacement
KSS	Knee Society Scores
MEMS	Micro-electronic-mechanical system
MTM	Materials testing machine
NOS	Navigated orthopaedic surgery
NSAID	Non-steroidal anti-inflammatory drug
OA	Osteoarthritis
OKS	Oxford Knee Scores
OOP	Object orientated programming
OpenCV	Open computer vision
OpenGL	Open graphics library
OST	Optical see-through
PCL	Posterior curciate ligament
PG	Proteoglycans
P-IR	Polyscopic IR
POI	Points of interest
RANSAC	Random sample concensus
RMSE	Root mean square error
ROM	Range of motion
S-IR	Stereoscopic infrared
S-V	Stereoscopic imaging in the visible spectrum
SAD	Sum of absolute intensity differences
SD	Standard deviation
SDL2	Simple DirectMedia Layer library

SGM	Semi-global block matching
SSD	Sum of squared intensity differences
SVD	Singular value decomposition
TKA	Total knee arthroplasty
TKR	Total knee replacement
TOF	Time of flight
UI	User interface
UKA	Unicondylar knee arthroplasty
UML	Unified modelling language
VR	Virtual reality
VSM	Visible spectrum monoscopic
VST	Video see-through
YLD	Years lost to disability

List of Figures

List of Tables

List of Symbols

A_E	Emitted singal amplitude
A_R	Recived signal amplitude
B_R	Recived background noise
c	Speed of light
d	Distance or disparity
D	Distortion parameters
$\Delta\phi$	Phase delay
f_{mod}	Modulation frequency
F	Fundamental matrix
H	Correlation matrix
$\mathcal{H}$	Hessian matrix
k	Spring constant
K	Intrinsic parameter matrix
P	Projection matrix
Q	Reprojection matrix
R	Rotation matrix
S_E	Emitted light signal
T	Translation matrix
τ	Transmission time
(x, y)	Image coordinates
(x_c, y_c)	Image centre
(X, Y, Z)	World coordinates

1
Introduction

1.1 The need for Knee Arthroplasty

Osteoarthritis (OA) is a degenerative disease that affects the *articulator cartilage* (AC) of joints. AC is fundamental to the function of a joint as it lubricates movement and helps absorb shock forces (Bhosale & Richardson 2008). As OA progresses the upper layers of the AC become increasingly rough and fractured, while underlying bone encroaches into the base layers. This reduces both the lubrication and shock absorbance provided by the AC (Buckwalter & Mankin 1997). The first symptoms noticed by someone afflicted with OA are typically mild pain and stiffness within the joint (Rönn et al. 2011; Litwic et al. 2013). As the disease progresses the pain becomes more severe and the stiffness more restrictive. If allowed to progress, sections of AC may be fully eroded, leading to extremely painful bone-on-bone articulation. The pain and immobility of the joint hugely reduces the *quality of life* (QOL) of those affected. In the case of large fundamental joints such as the hip or knee, those affect by late stage OA may become bed bound.

Current pharmacological and physiological treatments offer no cure for OA. Some treatments may slow the progression of the disease, however none have yet to stop progression, let alone reverse existing damage (Kon et al. 2012; Fraenkel et al. 2004). Therefore, to combat the hugely detrimental effects on QOL seen in late stage major joint OA, surgery is considered the most viable treatment (Lützner et al. 2009; Rönn et al. 2011).

The research presented within this thesis predominately is associated with *OA of the knee* (KOA) over other joints for three reasons. Firstly, 1.8% of all *years lost to disability* (YLDs) are attributed to KOA (Vos et al. 2012). As such, KOA is one of the largest single causes of disability in the world. Furthermore, projections based on aging and increasingly obese populations predict the incidents of KOA to rise significantly (Turkiewicz et al. 2014). Secondly, KOA is a relatively consistent condition with the majority of procedures following a fixed methodology. This is in comparison to OA of regions such as the spine, which may require vastly different approaches on a per-procedure basis. Finally, two systems of a similar nature are already commercially available, which offer a solid basis for the core principles of this research.

The gold standard of KOA joint replacement surgery is *total knee arthroplasty or replacement* (TKA or TKR) (Myers 2005). As indicated by the name, TKA replaces the full knee. Both the distal end of the femur and proximal end of the tibia are removed and replaced with metallic and polymer components (Williams et al. 2010). TKA has a long and distinguished history as an extremely effective treatment for KOA. This history of effectiveness has led to 90% of all current knee replacements, in the UK, being performed as TKA (Wishart et al. 2014).

As TKA replaces the whole joint, all diseased cartilage and bone is remove and the pain is therefore significantly alleviated. However, this approach also results in the removal of healthy tissues including important stabilising ligaments. The operation therefore may sometimes cause unnecessary trauma to the knee, and will affect the stability and functionality of the resulting joint. A number of methods have been developed to preserve one or both of the cruciate ligaments, thus improving joint function and longevity (Rand et al. 2003). One such method limits joint replacement to the separate condyles of the knee, thus preserving the cruciate ligaments. The condylar approach also preserves significantly more bone, reducing the trauma of the procedure. This is particularly true in the case of *unicondylar knee arthroplasty (UKA)*.

UKA offers a considerably more conservative approach than TKA for those affected by OA in only a single compartment of the knee, which is found to occur medially in upto 47.6% of surgical candidates (Willis-Owen et al. 2009). As opposed to the large resection of TKA, UKA requires only a small resection of the diseased compartment, typically the medial compartment. The diseased compartment is replaced while the opposite compartment and cruciate ligaments are preserved. This reduces the trauma of the surgery, improving recovery times, and provides a more functional joint.

1.2 The need for computer guidance

Premature implant failure is the second most common indication for implant revision after infection. For these premature failures it is often observed that the implants were malpositioned or aligned (Keene et al. 2006). Even subtle misalignment has been shown to significantly alter the force distribution across the implant, leading to excessive wear and premature failure. Furthermore, evidence indicates that knee function is significantly linked to component alignment. These findings have resulted in one of the main criteria used to assess knee arthroplasty procedures; *hip-knee-ankle (HKA)* alignment angle. A successful procedure is typically classified as having a HKA alignment within 3° of the planned, typically natural, angle in the coronal plane (Brin et al. 2011). It should be noted that the ideal angle is still a contested and varied value, with a one-size fits all solution being unlikely (Davis et al. 2015; Parratte et al. 2010).

The UKA procedure is particularly susceptible to premature failure as a result of misalignment. As only one condyle is replaced the remaining anatomical condyle is exposed to the imbalanced forces induced by misalignment. Excessive force may lead to rapid degradation of the cartilage, resulting in OA and a need for replacement. Meanwhile, a lack of force may lead to a reduction in osteoblast activity, reducing bone density and weakening the joint.

It is evident that accurate implantation is fundamental to the success of arthroplasty, and as such considerable efforts have been made to ensure it. The bulk of this work saw the development of mechanical alignment guides and jigs. However, recent years have seen work focus on computer assisted alignment devices as part of the field of *computer assisted orthopaedic surgery (CAOS)*. Initial CAOS systems used tracking technologies to guide the position of cutting jigs, which would then ensure accurate resection. Recently however, direct guidance systems have been introduced. These systems directly control resection by tracking the limbs and resection tool. There are two main commercial systems that take this direct guidance approach for UKA procedures, the Blue Belt Navio® and MAKOplasty® RIO®. Both systems combine planning with a type of semi-active constraint to guide the surgeon through resection and ensure accurate implantation. Semi-active constraint allows the surgeon to be in full control as the system passively observes during normal resection. However, if the surgeon mistakenly attempts to resect bone not indicated in the planning stage, the system intervenes. By either stopping or retracting the cutting tool or resisting the movement of the surgeon the CAOS system ensures accurate resection and in turn a well-aligned implant.

Despite CAOS systems offering improved alignment and reduced outliers uptake has remained limited (Brin et al. 2011; Davey et al. 2012). Initial cost and precived lake of advantages present the primary barriers to increased uptake (Mattess & Decking 2014; Picard 2014). However, physical and practical aspects of the systems themselves, such as size and complexity, raise opposition (Davey et al. 2012).

1.3 Thesis Overview

This thesis follows the design, development and testing of an augmented reality (AR) guided CAOS system. An initial literature review summarises relevant research and describes the challenges faced by existing systems. The aims and objectives of the research are defined and then analysed using a problem analysis based system design methodology. This analysis provides several problem and design diagrams, which are referenced throughout the development of the system.

Two initial alternative tracking systems are explored in Chapter 5, based upon the original aim of this research. The main body of the thesis then focuses on the development of the AR guided CAOS system as it evolves through three key generations.

Chapter 6 documents the first generation system built upon the ARToolKit library. The ARToolKit library, originally planned to provide tracking only, served as the inspiration for an AR based solution. A tracking accuracy test that made use of a basic augmented overlay demonstrated the incredibly intuitive nature of AR guidance and thus shifted the focus of this research.

Due to the limitations of the ARToolKit library identified in Chapter 6, the system was redeveloped from the ground up under the *Open Computer Vision* (OpenCV) library. The development of this second generation is described in Chapter 7. The primitive and outdated fixed pipeline graphics support offered by ARToolKit was replaced by a custom graphics engine developed using the programmable pipeline of modern *Open Graphics Library* (OpenGL). This allowed the system to provide much more robust and intuitive AR guidance.

Chapter 8 details the development of the third generation system, ARgCAOS. This final generation system endeavoured to fully embrace the potential of AR by augmenting the full view of the user and displaying the resulting mixed reality using a *head-mounted display* (HMD). Although several elements of the second generation system were used for the third generation, the core structure of the system was redeveloped to better suit interfacing with a HMD and the stereoscopic rendering now required to produce a full 3D experience.

This thesis concludes with a discussion of the work presented and proposed future work based upon the ARgCAOS platform.

2

Literature Review

2.1 Introduction

The following literature review begins by discussing the pathology and epidemiology of osteoarthritis, as the most common indication for knee replacement. Treatments of the condition are then explored, initially non-surgical approaches before total and partial knee arthroplasty. Navigated arthroplasty in then introduced, with several systems investigated. Various tracking technologies are investigated, as tracking is core to any potential navigated system. Lastly, augmented reality is described, investigating the display technologies and the systems in which they have seen use.

2.2 Osteoarthritis

2.2.1 Pathology

OA is a degenerative disease of joints, associated with a loss of AC causing pain and reduced mobility. Historically OA was seen as a 'wear and tear' disease (Aigner et al. 2004), whereby, the relatively inert AC covering of a joint’s surface is mechanically worn away, through friction, by continuous use (Buckwalter & Mankin 1997). However it is now evident that AC is very biologically active, and far from the inert material it was once considered (Mankin & Lippiello 1979). AC is made up of four main components: 70-80% water; 12-19% collagen (90-95% of which is Type II) (Matyas et al. 2002); 4-8% *proteoglycans* (PGs) (majority of which are aggrecan core proteins with long chondroitin sulphate and keratin sulphate *glycosaminoglycan* (GAG) chains (Carney & Muir 1988)); and 0.01-0.1% chondrocytes (Trattnig 1997), with the remaining 5% consisting of additional proteins (Eric M Darling & Athanasiou 2003; Novelli et al. 2012).

AC is divided into three main layers or zones (Wu & Herzog 2002). The top most superficial zone or surface layer is typically 10-20% of the total AC thickness. It consists of very densely packed collagen fibres aligned parallel to the joint surface (Waldschmidt et al. 1997). This dense alignment of fibres, combined with their very low permeability, results in a thin surface very well adapted to distributing forces across the joint (Novelli et al. 2012).

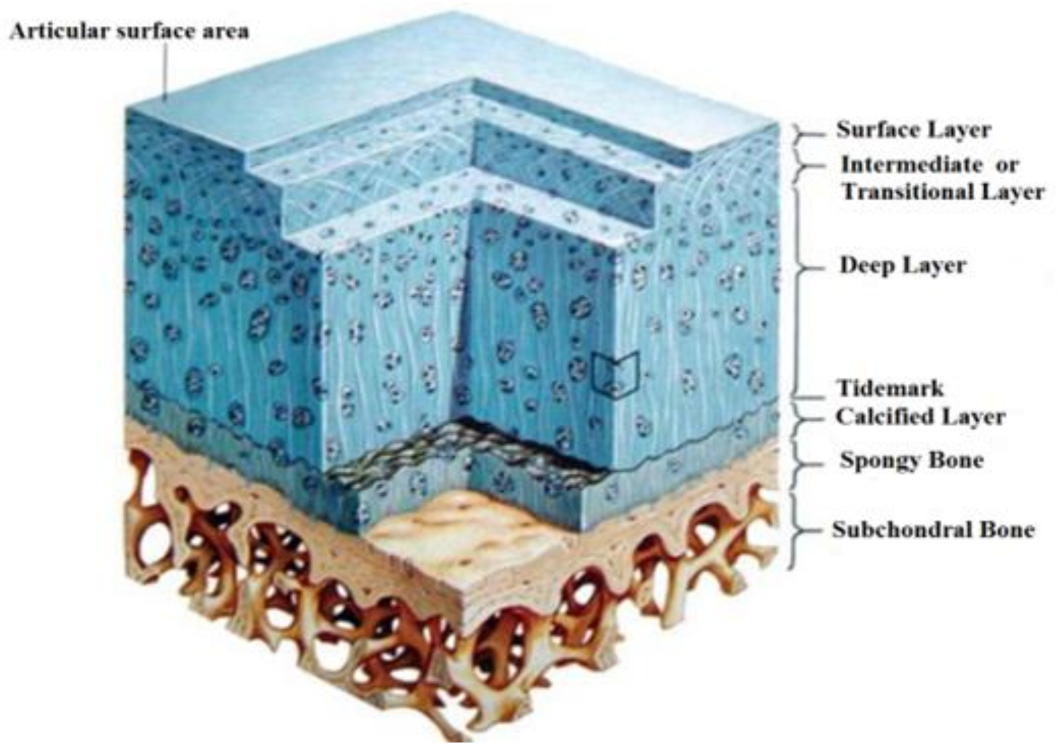


Figure 2.1: Layers of AC (S. et al. 2012)

The middle, or intermediate, and deep zones – 40-60% and 20-50% total thickness respectively – have collagen fibres aligned perpendicular to the joint surface with an increased concentration of PGs macromolecules compared to the superficial zone (Athanasiou et al. 2013). The highly negative nature of sulphate and carboxylate groups found in PG causes it to strongly imbibe water through the Donnan effect (Bhosale & Richardson 2008). When the AC is compressed, as through the impact of walking, the imbibed water is pressurised and forced out. However, due to the low permeability of the collagen matrix - and its high tensile strength preventing it from breaking - the water experiences high levels of hydrodynamic friction (Ateshian et al. 2003). This allows the pressure of the impact to be distributed and dissipated throughout the AC of the joint (Novelli et al. 2012).

Proximal to the condylar bone is a thin calcified region separated by the tidemark – sometimes classified as a separate forth layer (Waldschmidt et al. 1997). It is proposed that this produces a more gradual stiffness gradient, reducing stress (Radin & Rose 1986).

In the early stages of OA there is a marked decrease in the presents of PG (DeGroot et al. 1999). This in turn leads to reduced osmotic pressure, reducing the hydration of the AC (Bhosale & Richardson 2008). This may lead to a roughening of the AC, increasing its susceptibility to mechanical wear causing fibrillation of the surface. This wear may then propagate deeper into the AC, producing fissures into the middle and deep layers (Buckwalter & Mankin 1997). The reduced hydration of AC also affects its ability to absorb shock, as discussed above.

The reduction of PG seen in aging subjects is a natural effect which may be amplified by a sedentary life style. Anabolic activity reduces with age leading to a natural drop in PG concentrations (DeGroot et al. 1999). However, older subjects tend to be significantly less active. Due to the avascular nature of AC it derives its nutrients from the fluid within and around it. The compression of AC, caused by the joint's use, is essential to ensure that the fluid is cycled throughout the AC, sustaining adequate nutrient supply for metabolic activity (Eric M. Darling & Athanasiou 2003).

Chondrocytes also play a significant role in the development of OA (Fassbender 1987). The collagen matrix is routinely damaged through its use. However, in a normal subject this damage is repaired by chondrocytes (Goldring 2000), which are activated by damage and other abnormal forces, and respond by releasing inflammatory mediators and increasing catabolic activities. They release proteolytic enzymes, such as collagenase, gelatinase and stromelysin, which break down the damaged collagen matrix (Mohamed-Ali 1991). In healthy subjects this breakdown causes the release of feedback growth factor mediators which inhibit the catabolic activities of chondrocytes and up regulate their anabolic functions leading to the growth of new matrix. However, in older subjects, the base anabolic rate of chondrocytes is reduced and they show reduced sensitivity to growth factors (Loeser

2006). This can result in unregulated catabolic activity of chondrocytes, leading to the continuous destruction of AC.

The active destruction of the collagen matrix combined with the mechanical wear of the exposed fibres leads to a thinning of the AC (Buckwalter & Mankin 1997). This reduces the joint's ability to absorb shock, primarily leading to an increased rate of AC destruction. However, it also allows the disease to propagate into the subchondral bone (Salter 1999). The abnormally high forces now experienced by the bone of the joint causes the bone to become sclerotic. Furthermore, due to the reduced distribution of forces by the AC, proportions of the bone distal from the point of contact may become rarefied (Salter 1999).

Further bone growth may develop in the form of osteophytes - small irregular bone spurs typically forming at the cartilage bone interface (Hashimoto et al. 2002). These are likely to be the result of increase anabolic cytokines, as the body attempts to repair the damaged joint. However, they tend to exacerbate the disease as their rough surfaces abase surrounding tissues causing pain and increased wear. This is particularly true of central osteophytes which form on articulating surfaces. Some osteophytes may also physically reduce the range of motion of a joint by obstructing the movement of articulating bones (Felson et al. 2005).

In the most developed forms of OA a joint may suffer a total loss of AC. This causes extreme pain as subchondral bone articulates directly (Rosneck et al. 2007).

2.2.2 Epidemiology

OA of the knee is a very prolific disease with studies showing that between 4.9 and 6.9% of the adult population have symptomatic OA - that is they experience regular pain (Lawrence & Felson 2008; Andrianakos et al. 2006; Woolf et al. 2007). As discussed above, OA is strongly linked to age, as has been demonstrated by several studies (Felson et al. 1987; Felson 2000; Litwic et al. 2013), with one study reporting symptomatic rates as high as 18.4% in those over 70 (Guccione et al. 1994).

Another significant risk factor associated with knee OA is obesity (Felson 1988; Coggon et al. 2001; Litwic et al. 2013). One particularly insightful study by Blagojevic *et al.*, based upon the pooled results of 85 studies, showed that obesity to normal weight has an odds ratio of 2.63 (95% Cl 2.28-3.05) with regards to OA of the knee. Furthermore it was shown that being overweight (BMI of 25-30) compared to normal weight still produced an odds ratio of 2.18 (95% Cl 1.86-2.55) (Blagojevic et al. 2010).

Gender has also been frequently highlighted by literature, with females having a significantly increased risk (Nigel Arden et al. 2006; Litwic et al. 2013). The Dutch Institute for Public Health found prevalence rates of 15.6% and 30.5% in men and women above 55 years of age respectively (Woolf et al. 2007). This is consistent with the odds ratio for female gender of 1.84 (95% Cl 1.32-2.55) reported by Blagojevic *et al* (Blagojevic et al. 2010). It is noted that these rates are higher than those previously reported as they are based upon the radiographic definition of the disease while previous figures were symptomatic classification. While there is a link between radiographic and symptomatic OA it is not absolute (Hannan et al. 2000). Other reported risk factors include genetic predisposition, ethnicity, bone mineral density, activity, joint misalignment, and previous injury (Woolf & Pfleger 2003; Litwic et al. 2013).

Musculoskeletal (MSK) disorders are the second largest contributor to YLD. OA of the knee is the third largest contributor to that group, responsible for 206 YLD per 100,000 (95% Cl 142-290) in 2010, making OA of the knee responsible for 1.83% of the total YLD in 2010 globally (Vos et al. 2012). The YLD of OA of the knee increased 26.8% from the 163 YLD per 100,000 (95% Cl 112-232) of 1990. This is much higher than the 2.5% increase of total YLD per 100,000 (Vos et al. 2012). OA of the knee's increasing effect on lives is predicted to continue, with prevalence rates expected to increase by 40% by the year 2025 (London et al. 2011). This predicted increase is predominately the result of an aging and increasingly overweight population (Christensen et al. 2009; Lutz & K C 2010; de Onis et al. 2010; Wang et al. 2011).

Economically, studies have estimated the combined direct and indirect cost of OA in the United States to be $60-65 billion annually, representing approximately 0.5% of their GDP (Buckwalter et al. 2004; Jackson et al. 2001).

2.2.3 Treatment

Treatment options for OA fall into three broad categories: non-pharmacological, pharmacological, and surgical (Zhang et al. 2010; Kon et al. 2012). Non-pharmacological treatments consist of different forms of exercise and life style changes as well as acupuncture, massage and the use of TENS devices, often helping to alleviate the pain or slow the progression of cartilage loss (van Baar et al. 1999; Bennell & Hinman 2011). Meanwhile, typical pharmacological treatments use paracetamol and *non-steroidal anti-inflammatory drugs* (NSAIDs) or other analgesics such as Cox-2 inhibitors and opioids predominantly to reduce pain (Kon et al. 2012). Surgical options may involve lavage operations, although various forms of joint or joint surface replacement are significantly more common (Zhang et al. 2010).

Some treatments do offer significant improvements to quality of life at an acceptable cost to health care providers. For example water based exercise or NSAID in combination with misoprostol have inflation adjusted cost per Quality-Adjusted Life Year (QALY) of $11,530 and $17,011 respectively (Zhang et al. 2010). However, as the disease progresses and the pain becomes increasingly debilitating *knee arthroplasty or replacement* (KA or KR), is typically seen as the most viable treatment. Costs per QALY may be as low as $428 for suitable subjects (Zhang et al. 2010).

2.3 Knee Arthroplasty

2.3.1 Total knee Arthroplasty

2.3.1.1 Introduction

TKA was first introduced, in its current form, in the early 1970s and has remained the most prevalent treatment (Ranawat & Ranawat 2012). Earlier examples exist, such as the ivory TKA implanted by Theophilus Gluck in the 1890s (Ranawat & Ranawat 2012). However, these were hinged devices as opposed to the condylar designs seen almost exclusively today; hinged devices account for less than one percent of bicondylar KA procedures (Porter et al. 2013a). Although, new rotating hinged implants have recently emerged as an alternative option for more complicated replacements, such as those required in tumour surgery where there is significant bone loss (Berend & Lombardi 2009).

TKA is the most common form of knee arthroplasty, being used in approximately 90% of primary knee replacements between 2006 and 2012 (Porter et al. 2013a). This popularity is likely the result of the high survival rate, improved function and effective pain relief of TKA implants (Healy et al. 2002). Koskinen *et al.* reported a 15 year survival rate of 80% (95% CI 79-81), while Rand *et al.* and Schai *et al.* reported ten year survival rates of 91% (95% CI 90-91) and 92% (95% CI 87-95) respectively (Koskinen et al. 2009; Rand et al. 2003; Schai et al. 1998).

Over the years, a myriad of TKA implants have been developed, with 37 different implants produced by 14 different companies available in the UK in 1997 (Liow & Murray 1997). This figure has increased to 62 implants produced by 26 companies as reported by the NJR in 2012 (Porter et al. 2013b). Although, 79% of the market share is held by just five implants by DePuy (36%), Zimmer (16%), Stryker (11%), Smith & Nephew(8%), and Biomet (8%). Only six other implants hold a share greater than one percent (Porter et al. 2013b). In 86% of primary TKAs bone cement fixation was used (Porter et al. 2013a). The use of cement is reported to significantly reduce the likelihood of implant loosening (Duffy et al. 1998; Chockalingam & Scott 2000). Chockalingam and Scott reported loosening rates of 9.8% and 0.6% ($P<0.05$) at six years for cementless and cemented fixation respectively in a study of 351 implants (Chockalingam & Scott 2000). Similarly, Duffy *et al.* reported ten year survival rates of 72% for 55 cementless and 94% for 51 cemented implants (Duffy et al. 1998). However, Himanen *et al.* reports no significant ten year survival rate difference – 93.6% (95% CI 91.6-95.6) of 5203 cementless implants vs 93.5% (95% CI 87.6-96.6) of 1103 cemented implants (Himanen et al. 2005). This observation is supported by Baker *et al.* who found 15 year survival rates of 75.3% (95% CI 63.5-84.3) and 80.7% (95% CI 71.5-87.4) for cementless and cemented implants respectively, in a randomised control trial of 501 implants (Baker et al. 2007). These different observations may have resulted from the specific implant designs used. An additional advantage presented by cemented implants is the ability to use antibiotic-loaded cement, to reduce the risk of infection (Engesaeter et al. 2003). However, there is some controversy concerning the immuno-effectiveness of the routine use of antibiotic-loaded cement (Bolognesi, Namba, et al. 2009).

Another key differentiating feature of TKA implants is their constraint classification. The NJR groups implants as either; constrained, posterior stabilised (semi-constrained) or unconstrained. The latter two groups may also be further differentiated as having either fixed or mobile tibia bearings (Porter et al. 2013a). Unconstrained implants are considered the preferred choice as they offer the most natural kinematics and gait in addition to reduced wear and increased survival rates (Kaplan 2008; Skwara et al. 2008; Pang et al. 2013). This approach is evident in the NJR data where 73-75% of arthroplasties performed between 2006 and 2012 were unconstrained (Porter et al. 2013a). Semi-constrained and constrained implants become necessary when the soft tissues of a subject – ligaments and muscles – are unable to offer sufficient stability to the joint, allowing excess rotation or translation (Vince & Malo 2008). Semi-constrained implants are preferred over constrained as they still offer more natural gait and proprioception and superior survival rates (Barrett et al. 1991; Pour et al. 2007). However, constrained implants may be the only option in a joint with significant instability, either as a result of soft tissue inadequacies or significant bone defects or loss (Tateishi 2000).

The NJR reports that fixed bearing implants are more prevalent to mobile, representing 86-93% and 13-6% of implants, respectively, between 2006 and 2012, with the higher fixed bearing prevalence being seen in the most recent years (Porter et al. 2013a). There is no clear reason for this prevalence as the body of evidence is inconclusive with regards to one bearing being superior to the other (Myers 2005; Huang et al. 2007; Bistolfi et al. 2013; Kim et al. 2007; Jacobs et al. 2004; Aglietti et al. 2005; van der Voort et al. 2013). A limited number of studies have highlighted slight differences between the two implant types: improved *Oxford Knee Scores* (OKS) in mobile-bearing implants at one year follow up (Price et al. 2003); possible risk of mobile-bearing subluxation and dislocation (Bhan et al. 2005); increased osteolysis prevalence in mobile-bearing implants (Huang et al. 2002). Finally, the review paper of Huang *et al.* concludes that fixed-bearings are recommended for more inactive elderly subjects, while mobile-bearings may be better suited to younger, more active subjects due to the possible wear reduction (Huang et al. 2007).

2.3.1.2 Procedure

There is an enormous range of TKA procedures, far exceeding the scope of this review, with procedures being created for specific disease or anatomical defects. New implants typically detail their own procedures which are modified by centres and surgeons (Kao et al. 2009). Furthermore, new assistive devices are continuously being developed which bring their own modifications to established operating procedures (Bonnin & Chambat 2008; Hanssen & Scott 2009; Bellemans et al. 2005; Sculco & Martucci 2001; Thienpont 2012; Munzinger et al. 2004; Bonnin et al. 2011; Scuderi & Tria 2006; Kaplan 2008). Williams *et al.* offers a concise summary of the generic operating procedure which is paraphrased below (Williams et al. 2010).

The patient is draped and a proximal thigh tourniquet is used to reduce blood loss before the knee joint is exposed using a medial parapatellar approach (Lombardi 2004). Once exposed, several soft tissues are excised, including the *posterior curciate ligament* (PCL) if a non-preserving implant is to be used. At this stage any osteophytes are removed before completing the exposure of the joint. A jig (physical cutting guide) is then used to remove approximately 10 mm of the tibial plateau, using a cut that is perpendicular to the mechanical axis. A femoral intramedullary jig is used to remove approximately 10 mm from the distal aspect of the femur at 5-7° valgus from the anatomical axis – perpendicular to the mechanical axis. Further jigs are used to perform the remaining femoral cuts and any peg holes required by the implants are drilled. Trial implants are then used to check the alignment and soft tissue balancing. Any revisions are made before the permanent implants are secured, typically with antibiotic-loaded cement, and the tibial insert is positioned. The tourniquet is removed for haemostasis, despite evidence suggesting this leads to increased blood loss (Jorn et al. 2009). Finally, the incision is closed at deep and superficial levels, completing the operation.

Most procedures broadly follow this method with minimal changes, such as a more medial initial incision or resecting the femur prior to the tibia (Hanssen & Scott

2009; Bellemans et al. 2005; Munzinger et al. 2004). However, the major procedural differences result from the choice of implant and its associated alignment jigs. Some implants make use of the PCL to improve stability and preserve the kinematics of the knee (Sorger et al. 1997). Therefore, care must be taken to ensure its retention. However, some implants opt to remove the PCL and substitute it with a semi-constrained design, as it may provide better results in some instances (Simmons et al. 1996). Due to the designs of implants different sections of bones may require resection and the fixation methods may require peg holes or deeper stem holes.

Joint alignment is considered one of the most significant indications of a successful TKA, with poor alignment being associated with reduced function and premature implant wear, loosening and failure (Bolognesi, Fang, et al. 2009; Jeffery et al. 1991; Longstaff et al. 2009; Choong et al. 2009; Fang & Ritter 2009). However, at least one study questions this long held concept (Parratte et al. 2007). To improve alignment accuracy many implant systems employ jigs and other guide devices. Simple cutting jig systems, that are often placed by eye using local bony features such as Whiteside’s line, are predominately intended to ensure that multiple resections of a single bone are sufficiently accurate, relative to each other, to allow correct implant fitting (Fargie et al. 1988; Stuart L. Axelson et al. 1999). More complex alignment systems may instead be used that mechanically couple the positioning of a cutting jig to non-local anatomical features, such as the talus of the ankle, to optimise alignment (Petersen 1985; Cooke et al. 1986). Alignment jigs may be classified as either intramedullary or extramedullary for both tibial and femoral component placement. Intramedullary devices are typically reported as producing good alignment more consistently than extramedullary devices (Engh & Petersen 1990; Brys et al. 1991; Maestro et al. 1998). However, Dennis *et al.* reported to the contrary that extramedullary devices offered a slight increase in consistence (88% vs 72% within 2° of goal). Furthermore, Dennis *et al.* reported several other concerns associated with intramedullary devices, including fat embolization and intraoperative fractures (Dennis et al. 1993).

2.3.2 Unicompartmental Knee Arthroplasty

2.3.2.1 Introduction

UKA was first attempted by McKeever in 1952 after postulating that osteoarthritic damage could be present in only a single compartment of the knee while the remaining compartments remained healthy (Meneghini & Sheinkop 2004; McKeever 1960). McKeever designed and implanted a metallic unicondylar implant in a procedure that preserved significantly more of the natural anatomy of the patient than the alternative TKA procedure (McKeever 1960). Stukenborg-Colsman *et al.* concluded that the literature supports UKA as a superior treatment option compared to osteotomy for unicompartmental arthritis of the knee (Stukenborg-Colsman et al. 2001). The accelerated degeneration of the healthy condyle, often experienced after osteotomy due to unloading of forces from the damaged side may be avoided by instead replacing the damaged condyle and allowing normal force distribution (Stuart et al. 1990).

The concept of UKA presents a significantly less invasive alternative to TKA for subjects presenting unicondylar symptoms. Less bone is removed and both cruciate ligaments are preserved, resulting in more natural kinematics and undisrupted proprioception (Jeer et al. 2004; Isaac et al. 2007). Furthermore, the smaller implants and reduced need for exposure allow for smaller incision and less soft tissue trauma, resulting in reduced blood loss and improved recovery time, when compared to TKA (Jorn et al. 2009; Lombardi et al. 2009). These effects culminate in a cost effective alternative to TKA (Peersman et al. 2014). Figure 2.2 below illustrates the size difference of the implanted components.

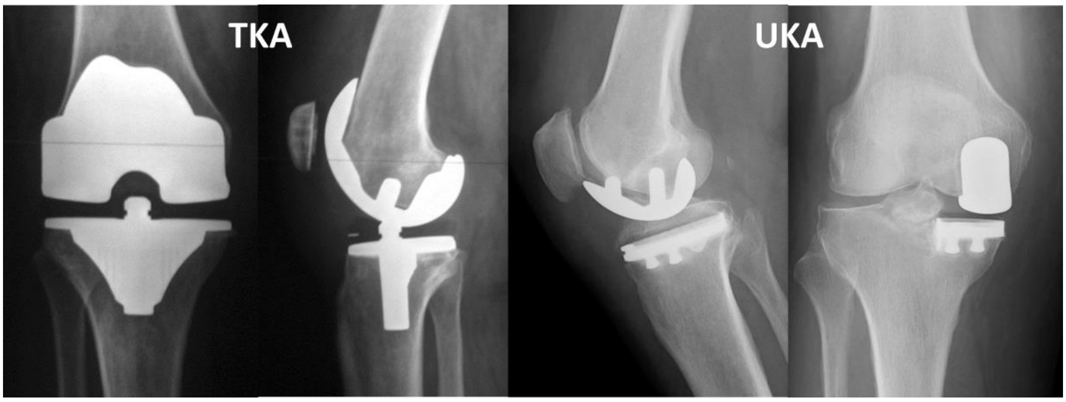


Figure 2.2: Comparison between TKA and UKA implant components (Jacquot 2007; Magnus Manske 2012)

It is strongly emphasised, however, that proper patient selection is vital to the success of a UKA procedure (Miettinen et al. 2015). Geller *et al.* suggest that lax patient selection in the early, enthusiastic, period of UKA procedures may account for the poor results reported by early studies, which had a detrimental effect on the popularity of UKA (Geller et al. 2008). However, more recent studies, where patient criteria were more strictly followed, have produced survival rates on par with those of TKA, with 10 year survival rates as high as 98% reported by two studies on different implants (Murray et al. 1998; Berger et al. 2005). Miettinen *et al.* reporting from a small volume centre upon 52 of 95 UKAs performed over a 10 year period reported similar Kaplan-Meier 9-year estimated survival rates of 88.9% (95% CI = 78.7-99.1%) at a mean follow up of 6.5 years (Miettinen et al. 2015). Lim *et al.* also reported high survival rates based upon 602 UKA patients and matched TKA patients over a 12 year period (Lim et al. 2014). At 12 years the UKA survival rate was 93.7% compared to 97.0% in TKA. The main cause of failure for UKA was dominated by aseptic loosening. This observation was confirmed by Epinette *et al.* who found 45% of 418 failed UKAs across multiple centres were attributed to loosening (Epinette et al. 2012). OA progression and implant wear were implicated in 15 and 12% of failures respectively. These survival rates are compared to that of only 73% (95% CI = 70-76%) and 60% (95% CI = 54-66%) at 10 and 15 years respectively reported by Koskinen *et al.* based upon 1,886 primary UKAs from the Finnish Arthroplasty Register performed between 1980 and 2003 (Koskinen et al. 2009).

As stated above, patient selection is fundamental to UKA success. Howell *et al.* looked at whether these optimally selected patients would still achieve better outcomes than standard TKA patients if converted to TKA (Howell et al. 2015). 55 planned UKA patients that were converted intra-operatively to TKA (UKA-TKA) were compared to 110 UKA patients (UKA-UKA) and 142 TKA patients (TKA-TKA). It was reported that both UKA-TKA and UKA-UKA groups had superior preoperative ROM, Knee Society Pain and functional scores, when compared to the TKA-TKA group. Post-operatively UKA-UKA patients presented the best ROM (P = 0.03), but were closely followed by UKA-TKA which were significantly better than the TKA-TKA group (P = 0.002). Both UKA-UKA and UKA-TKA groups were found to show better post-operative functional scores (P = 0.007 and P = 0.05). However, there were no significant differences in post-operative pain scores between the three groups. The author concludes that patient selection for UKA does produce a bias in outcomes. However, they advocated the continued use of UKA due to its lower risk profile and faster recovery times and activity levels.

2.3.2.2 Procedure

The UKA procedure follows the same basic structure as the TKA procedure described above in Section 2.3.1.2. Argenson and Flecher provide a detailed documentation of their medial UKA procedure (Argenson & Flecher 2004). This procedure is typical of UKA procedures and as such is paraphrased with other sources below.

The leg is placed in 90° of flexion, although it is often moved throughout the procedure. A tourniquet is used to restrict blood flow throughout the procedure. An off-centred parapatellar approach is used to maximise the exposure of the diseased condyle. A skin incision of between 60 and 80 mm is typically used (Mullaji et al. 2007). The synovial cavity is opened and obscuring sections of the fat pad removed. A retractor is used to keep the joint exposed while the joint is manipulated to evaluate the ACL and joint surfaces. Osteophytes are removed and the main resection of the procedure may now begin. Intramedullary or extramedullary guides may be used to perform the required resections (Geller et al. 2008; Jonna & Tria 2009). The tibial cut is aligned such that there is a 5 to 7° posterior slope and so that the resulting height matches that of the opposite plateau. A sagittal cut is made along the tibial spine, freeing the resected section of bone. If used, the intramedullary femoral guide may now be drilled and inserted. This guide is used to set the desired angle between the mechanical and anatomical axes. The desired angle is quoted as similar to that used in TKA at between 4° and 6°. The final femoral cuts are made such that the two implants will meet at their centres and away from their edges. With all major cuts performed and additional osteophytes removed, spacing checks are performed through the full range of flexion to size the tibial component. After this, final pin or keel cuts are made and trial implants fitted. Final sizing for the thickness of the tibial liner is performed, before the final components are cemented in place. After final testing the incision is then closed and the procedure completed.

Correct implant alignment is extremely important during the UKA procedure (Kasodekar et al. 2006; Valenzuela et al. 2013; Jung et al. 2010). In addition to the complications observed in TKA, poorly aligned UKA may result in excessive loading of the preserved compartment leading to accelerated degeneration (Hernigou & Deschamps 2004).

2.3.3 Navigated Knee Arthroplasty

2.3.3.1 Introduction

In both TKA and UKA sections, implant position and alignment are fundamental to the performance and longevity of the joint (Ritter et al. 2011; Harvie et al. 2012). To this end, a myriad of alignment guides and cutting jigs have been designed. However, despite this, it is reported that alignment outside of the accepted range routinely occurs (Stulberg et al. 2002). Tingart *et al.* report misalignment rates as high as 26% during TKA procedures (Tingart et al. 2008). In an effort to combat these inconsistencies, the field of *navigated orthopaedic surgery* (NOS) was introduced (Stulberg et al. 2002). NOS, often used interchangeably with the phrase Computer Assisted Orthopaedic Surgery (CAOS), utilises computerised systems to guide the surgeon, typically during resection, to ensure accurate implant positioning and the desired joint alignment (Sparmann et al. 2003). Since the first reported use of NOS to assist with a TKA procedure in 1997 the field has rapidly grown (Krackow et al. 1999). Due to the diverse nature of this growth, Picard *et al.* developed a clinical classification protocol to categorise systems (Picard et al. 2004). Systems were classified by two criteria. Firstly by imaging type: preoperative, intraoperative or image-free. Secondly by clinical action: active, semi-active or passive.

Preoperative imaging based systems typically use Computed Tomography (CT) models. These allow implants to be preoperatively size and positioned virtually. Intra-operatively the CT model, and associated cutting plan, is referenced to the physical bone using local landmark features (Picard et al. 2004). The requirement of CT models is not typical for KA and therefore represents an additional cost when using a preoperative imaging CAOS system. For the NHS this cost is reported as being between £53 and £268 per scan (National Audit Office 2011). Furthermore, and perhaps more importantly, CT imaging causes significant ionising radiation exposure. A typical leg image exposes the subject to a dose of between 0.7 mSv and 2.7 mSv, equivalent to 26-100% of the average annual dose of an individual (Henckel et al. 2006; Chauhan et al. 2004; Hughes et al. 2005)

Intra-operative imaging based systems use fluoroscopy after marker fixation. This allows bone to marker referencing to occur automatically, potentially minimising human error (Picard et al. 2004). Again, however, the use of fluoroscopy is not standard during KA and as such represents an increased cost and risk of radiation exposure, all be it less than preoperative CT (Ma et al. 2009).

Image-free systems require no pre or intra-operative imaging. Instead a series of landmark features are identified using a tracked probe, and further features such as joint centres may be found using kinematic algorithms (Bae & Song 2011). This data is then used to select or morph standard non-patient-specific models stored within the system library (Bae & Song 2011). The image-free approach removes the cost and risk associated with CT and fluoroscopy imaging. However, the non-patient-specific nature may decrease precision (Picard et al. 2004).

With regards to clinical action, active refers to a system that is able to perform at least part of the procedure autonomously, without human intervention. This will typically involve one or more of the cutting stages (Picard et al. 2004). Early NOS systems tended to favour this approach, often using repurposed industrial robots (Siebert et al. 2002). This approach reduces the risk of human errors. However, the highly autonomous nature does not often sit well with surgeons or patients alike (Sackier & Wang 1994). Semi-active systems are the middle ground of CAOS systems. The surgeon is in control at all times. However, their input may be filtered by the system. The filtering is often in the form of resection limiting. A resection plan is made, and bone for removal is identified. If the surgeon attempts to remove bone from outside of these areas the system will intervene by stopping the cutting tool (Picard et al. 2004). This approach puts the surgeon back in charge of the operation while still granting substantial navigation. Finally, passive systems provide only information and guidance to the surgeon, such as resection planes. However, they are unable to intervene if the surgeon deviates from these recommendations (Bae & Song 2011). These include systems that mimic traditional KA using navigation only to assist in the placement of conventional cutting jigs (Seon et al. 2009). This conventional analogue may be seen favourably by surgeons put off by the autonomous nature of more active systems.

A further classification groups systems as being either closed, in that they are designed with support for a specific prosthesis or technique, or open whereby the system supports a range of prosthesis produced by different manufacturers (Bae & Song 2011).

2.3.3.2 NOS Systems

One of the earliest examples of NOS was presented by Krackow *et al.* in 1999 (Krackow et al. 1999). The system used an Optotrak (Northern Digital Inc., Ontario, Canada) *Infrared* (IR) stereoscopic marker tracking system to kinematically determine the position of the *hip joint centre* (HJC). Further landmarks of the knee and ankle were captured directly allowing alignment of the knee to be calculated throughout the procedure. Under the classification system of Picard *et al.* this system would be considered an image-free passive system operating in a closed fashion. Krackow *et al.* fail to report overall joint alignment accuracy. However, it is stated that the HJC was located to within 2-4 mm, which corresponds to an angular accuracy below 1° (Krackow et al. 1999).

Another early system, CASPAR (Computer Assisted Surgical Planning And Robotics) - (URS Ortho, Rastatt, Germany), requires tibial and femoral bone pins to be fitted prior to CT imaging (Siebert et al. 2002). This allows the bone geometry to be referenced to the markers, reducing the intra-operative registration. Anatomical landmarks present in the CT model are then identified by the surgeon and alignment axes, such as the mechanical axis, are calculated. The surgeon is able to pre-operatively size and position the implants using the virtual model. Intra-operatively the knee is exposed with the standard median parapatellar approach. A rigid frame, with retroreflective markers, is fixed over the joint, which is in high flexion. The robot, a modified industrial system, is then moved into position and bone resection is performed using a water-cooled milling head. The surgeon is required to operate a dead man's switch throughout the cutting procedure (Siebert et al. 2002). Siebert *et al.* reported a tibiofemoral planned alignment deviation of 0.8° (0.0-4.1°) for the CASPAR system (70 patients) versus a deviation of 2.6° (0.0-7.0°) in a manual historical control group (50 patients) (Siebert et al. 2002). However, Decking *et al.* reported a deviation of only 0.2° (95% CI: -0.1-0.5°) and component alignment and position accuracies of ±1.2° and ±1.1 mm respectively. However, this was a relatively small study with only 13 subjects (Decking et al. 2004). The literature suggests that the system improved implant accuracy. However, it also requires pre-operative CT imaging and pre-imaging bone pins. Furthermore, operating times are also increased (Siebert et al. 2002). Under Picard *et al.* classification this system is defined as being preoperative active. Jaramaz *et al.* suggests that the use of these large, repurposed industrial robots in early active systems may have resulted in their limited uptake and the preference for semi-active or passive systems (Jaramaz et al. 2006).

ROBODOC (Integrated Surgical Systems, Sacramento, CA) was another common early system, similar to CASPAR. It too required bone pin fixation prior to CT-imaging, and allowed preoperative planning via landmark referenced CT-images (Börner et al. 2004). The main difference from the CASPAR system was that ROBODOC was mechanically coupled to a subject and did not use an optical tracking system (Wiesel & Boerner 2004). As with the CASPAR system ROBODOC is classified as being an active system using preoperative imaging (Jaramaz et al.

2006). Börner performed the first 100 ROBODOC procedures and stated that 97% of procedures produced ideal mechanical axes, with the remaining 3% being within 1° of ideal. However, five of the 100 procedures required intra-operative conversion to conventional techniques due to system issues (Börner et al. 2004). Two studies by Song *et al.* investigated the post-operative mechanical axis alignment produced by the ROBODOC system compared to conventional TKA. These studies both concluded that the ROBODOC system produced improved mechanical axes alignment of 0.2±1.6° and 0.5±1.4° compared to those of 1.2±2.1° and 1.2±2.9° for conventional procedures (Song et al. 2011; Song et al. 2013). Both studies also found ROBODOC to produce significantly fewer mechanical axis alignments outside of the ±3° acceptable range. However, both studies also showed that the ROBODOC procedure had a mean duration 25 minutes greater than conventional arthroplasty.

In contrast to the industrial nature of the robotics used by the ROBODOC and CASPAR systems, Wolf *et al.* presented a small-scale custom-designed robotic device (Wolf et al. 2005). The Mini Bone-Attached Robotic System (MBARS) was designed to be directly mounted onto the bone of the patient. This had the advantage that it required no external tracking devices, such as those used by the CASPAR system, to reference the cutting device to the bone anatomy. The procedure for femoropatelar replacement was reported by Wolf *et al.* as follows.

The base platform of the system is rigidly fixated to the exposed femoral side of the knee joint by three Steinman pins; medial and lateral epicondylar pins and a metadiaphyseal pin. A standard milling drill (TPS, Stryker Instruments, Kalamazoo, MI) is supported above the platform using six linear actuators in the classic Stewart-Gough six-degrees-of-freedom arrangement (Stewart 1966). Procedurally, the joint is exposed as standard before the robot is fixated. Using a probe tip, inserted into the drill, the surgeon may use a combination of manual and automatic surface registration. By manually controlling the position of the probe the surgeon may select specific landmarks. In the case of the femoropatelar replacement presented, the surgeon marks the patellar tracking line. The system then traces the probe over the bone surface using force feedback to record the surface topology. This data is used to produce a bone model from which the system may automatically position the implant. The surgeon inspects and verifies this intra-operative plan. The robot then automatically resects the bone surface in accordance to the cutting plan. Once complete, the robot is removed and the implant is inserted before the procedure is completed in the conventional fashion.

This procedure would classify the device as an active, image-free system. Wolf *et al.* report that the system is capable of producing cavity milling within 1 mm of the planned location. However, they fall short of discussing the final position and alignment accuracy of the system. In addition to improved accuracy, Wolf *et al.* propose that the system may provide additional advantages, including reduced operation time and reduced exposure compared to traditional approaches (Wolf et al. 2005). The design of the system offers no mechanism to obtain either ankle or hip centres and as such the mechanical and anatomical axes required for accurate knee arthroplasty could not be obtained, limiting the system.

Song *et al.* present a second generation prototype with the intention of improving upon the short comings of the MBARS (Song et al. 2009). Song *et al.* identified the inability to sterilise or protect the MBARS as one of the main issues preventing its clinical application. Furthermore, low accuracy, limited accessibility and excessive complexity were cited as flaws (Song et al. 2009). A three degree-of-freedom – two parallel and one serial – prototype named Hybrid Bone-Attached Robot (HyBAR) was developed and tested on both foam and pig bones. Furthermore, a Certus optical tracking system (Northern Digital Inc., Ontario, Canada) was used for bone relative drill tracking. These system changes allowed improved access while reducing much of the MBARS complexity. Song *et al.* report overall precisions of 0.2-0.37 mm for pig bone resections. However, they highlighted that the system produced large implant displacements, likely as a result of insufficient rigidity in the structure of the device (Song et al. 2009). Song *et al.* also concluded that although the system was designed for femoropatelar resurfacing it could be readily extended to other procedures such a UKA, reclassifying the system as open, active image-free.

The Acrobot® system (Stanmore Implants, Elstree, UK), which first saw clinical use for TKA in 2000, introduces the new concept of active constraint into CAOS system (Rodriguez et al. 2005). Active constraint is a semi-active technology, whereby the system resists the movement of the surgeon if they attempt to remove bone from outwith the planned cutting region. The Acrobot$^{®}$ system facilitates active constraint through two connected modules. The first module is referred to as the gross positioning system. It has three axial movement and is responsible for moving the second active constraint module into position. The gross positioning system is mechanically attached to the operating table using rigid struts. The tibia and femur of the patient are also attached to the operating table using three telescopic rods. These are attached to the bone through 5 mm incisions. The patient is therefore mechanically coupled to the system (Rodriguez et al. 2005). The gross positioning module is moved into place and its joints are locked preventing any system errors from inadvertently moving it mid-procedure. The active constraint module, with three axes of movement, is registered to a preoperative CT model and plan using 20-30 points on the bone surface, marked with a 1mm diameter probe. The surgeon is then able to resect bone aided by the active constraint system. Within the planned cutting region the burr may be moved freely without resistance. However, as the surgeon approaches the edge of the cutting region the resistance to movement, in the direction of the edge, increases. If the surgeon continues to move outside of the cutting region the burr may be stopped. This keeps the surgeon in control of the procedure while still receiving the accuracy advantages of a CAOS system (Picard et al. 2004). Rodriguez *et al.* investigated the accuracy of the Acrobot$^{®}$ system. It was reported that 100% of the 13 procedures produced mechanical axis alignment within ±2° of the planned alignment. However, only 40% (6 of 15) of the conventional procedures met this level of accuracy, a significant difference (P = 0.001) (Rodriguez et al. 2005). Cobb *et al.* expand upon this result numerating the mean alignment of the Acrobot$^{®}$ procedures as 0.65° (SD 0.59, range -1.6 to 0.3°) compared to -0.84° (SD 2.75, range -4.2 to 4.2°) for conventional procedures (Cobb et al. 2006). The Acrobot$^{®}$ system also produced significantly better American Knee Society Scores. The mean operation times were found to be 104±16.6 and 88±16.3 minutes for Acrobot$^{®}$ and conventional procedures respectively. However, this time difference failed to reach significance (Cobb et al. 2006). Karia *et al.* investigated the learning curve of first time surgeons using the Acrobot$^{®}$ under the new name Sculptor RGA. It was found that procedure time using saw bones decreased from 37.5 to 25.7

minutes (P = 0.002) after only three uses over three weeks. Furthermore, when compared to first time surgeons performing conventional arthroplasty the Sculptor RGA cohort produced significantly better positioned implants (Karia et al. 2013). The system operated under open semi-active pre-operative imaging classification. However, the system was able to operate in active mode, which Rodriguez *et al.* suggested may be used for performing rough cuts to remove the bulk of bone before a surgeon performs the detailed resection under semi-active control. Furthermore, Rodriguez *et al.* state that an image-free methodology had been developed (Rodriguez et al. 2005).

2.3.3.2.1 MAKOplasty

One of the most successful modern systems is the RIO® (Robotic-arm Interactive Orthopaedic) MAKOplasty® system (Mako Surgical Corp., Fort Lauderdale, FL, USA) (Hagag et al. 2011). Having seen consistent growth since its first use in 2006 there are currently 174 systems in use around the world. These systems average 6.6 procedures per month, resulting in 3,259 procedures during the third quarter of 2013, and approximately 33,000 procedures since production (MAKO 2013). The operation of the RIO system is similar to that of Acrobot®, in that it also uses a surgeon guided cutting burr controlled by active constraint – referred to as tactile or haptic feedback – to prevent the surgeon from inadvertently resecting bone outside of the planned resection regions (Hagag et al. 2011). However, instead of the mechanical coupling used by Acrobot®, an IR stereoscopic marker tracking system is used (Cobb & Pearle 2013). The RIO® also uses CT imaging to produce a pre-operative plan, much like Acrobot®. The images are rendered as 3D bone models upon which virtual models of the desired implant are positioned by the surgeon. The system aids the surgeon in implant positioning and rotation by displaying anatomical alignment information. This information, such as the predicted mechanical axis in the coronal plane, is updated in real time as the surgeon moves or resizes the virtual implants (Hagag et al. 2011).

Operationally, the theatre is set up to maximise the access for the surgeon and system to the knee that is to be operated upon. Before exposure of the knee joint, tibial and femoral IR markers are fixated using dual bone pins. The knee is then exposed, typically using a *minimally invasive surgery* (MIS) incision of approximately 60 mm in length (Hagag et al. 2011). The surgeon then registers the anatomy of the patient to the CT models and pre-operative plan. This is done by marking defined anatomical landmarks using an optically tracked probe. Checkpoint pins that allow the surgeon to check the consistency of the system throughout the procedure are also inserted into the tibia and femur. The system guides the surgeon through the registration process, displaying the location of the landmarks on the virtual model. Once registration is complete the surgeon is able to refine the pre-operative plan. CT imaging is not sensitive to cartilage, nor is it possible to accurately determine the characteristics of the soft tissues of the knee. Therefore, the alignment information presented during pre-operative planning is only an estimate. The surgeon moves the joint through its full *range of motion* (ROM) while applying appropriate valgus load. This allows the system to update its model and provide a better kinematic model. Soft tissue balancing is also factored by ensuring proper collateral ligament tension throughout flexion (Cobb & Pearle 2013; Hagag et al. 2011). Once the plan has been finalised the system automatically determines resection regions to position the implants. Additional *no-go* regions may be marked to prevent accidental damage of surrounding tissues, for example the collateral ligament. Resection is performed using a high speed 80,000 rpm 6 mm diameter pneumatic burr attached to the RIO (Hagag et al. 2011). The robotic arm has six joints of rotation along its length. These provide the burr with six DOF, allowing easy resection from different approaches (Hagag et al. 2011). Again, as with the Acrobot®, the burr may be moved freely while within the planned cutting region. The system then begins to apply increasing resistance as the surgeon nears the boundary of the resection. The RIO system also provides audible feedback in the form of beeps and a visual feedback as the virtual model is updated in real time showing which regions of bone still require resection (Cobb & Pearle 2013). If the implant requires peg holes the burr may be replaced with either a 1.4 or 2.0 mm router. This router is positioned at the start of the peg hole before the arm is locked in position and only axial motion of the router is permitted (Hagag et al. 2011). This ensures quick and accurate resection of peg holes.

Unlike many CAOS systems, the popularity and business model of the MAKOplasty® system has resulted in a sizable body of literature. Coon *et al.* compared the accuracy of 33 MAKO UKAs to that of 44 conventional MIS UKAs. It was reported that the MAKO system produced superior implant accuracy in both the coronal and sagittal planes (Coon et al. 2010b). The RMS error of the posterior slope of the tibial component was found to be 2.5 times greater for the conventional group (1.4° and 3.5°, MAKO and conventional respectively). Furthermore, the variance for the conventional group was found to be 2.8 times greater (Coon et al. 2010b). Lonner *et al.,* reporting on 31 MAKO assisted and 27 conventional procedures, obtained similar results. Reduced posterior tibial slope RMS errors of 1.9° for MAKO procedures compared to 3.1° for conventional, were reported, with variance 2.6 times greater for conventional (Lonner et al. 2010). In the coronal plane Coon *et al.* reported the average error to be 3.3±1.8° (more varus) for the conventional group, while only 0.1±2.4° for the MAKO group (Coon et al. 2010b). These results concur with those of Lonner *et al. who* reported 2.7±1.2° and 0.2±1.8° respectively (Lonner et al. 2010). Roche *et al.* reported upon their first 43 MAKO procedures. It was stated that of the 344 post-operative radiographic measurements obtained none were considered clinically significant outliers and only four were identified as non-clinically significant outliers, with regard to implant positioning and rotation (Roche et al. 2010a). On average components were placed 0.6° (SD 1.9°) less varus and with 0.1° (SD 1.8°) less posterior slope than planned (Roche et al. 2010a). Mofidi *et al.* reported upon 220 consecutive MAKO assisted UKA procedures performed by two surgeons. A notably large average difference of 2.2-3.6° was reported between the planned and post-operative radiographic alignment of the femoral and tibial components in the coronal and sagittal planes (Mofidi et al. 2012). Coon *et al.* also reported upon patient outcome in relation to the experience of a surgeon with the MAKOplasty system (Coon et al. 2010a). A surgeon with no prior MAKO experience performed 36 MAKO assisted procedures and 45 conventional procedures, with which they had considerable experience. Both *Knee Society Scores* (KSS) (Insall et al. 1989) and Marmor ratings (Marmor 1979) were determined at three, six and twelve week follow-ups. Coon *et al.* found no significant difference using either scoring system at any of the follow-up periods, nor in a specific component of the scoring (Coon et al. 2010a). Coon *et al.* present this as evidence that the MAKO could match the results of an experienced surgeon and that there is no period of poor practice during the learning curve of the MAKO system. Roche *et al.* report upon 14 patients at one year follow-up and found that range of motion ($P < 0.02$), KSS ($P < 0.0001$), WOMAC scores ($P < 0.01$), pain ($P < 0.01$) and stiffness ($P < 0.01$) all showed significant improvement compared to preoperative values (Roche et al. 2010b). Roche *et al.* also report reoperations, as opposed to revisions, on six of 223 implants. One reoperation was the result of a femoral fracture at the site of the bone pin used for intraoperative tracking (Roche et al. 2010b). This increased fracture risk has been raised a number of times (Ossendorf et al. 2006; Bonutti et al. 2008; Li et al. 2008; Wysocki et al. 2008). Sinha *et al.* report a 100% success rate on the initial 20 procedures performed, requiring no conversion to conventional methods (Sinha et al. 2010). However, in another paper Sinha reports that a single conversion to a conventional procedure was required during the following 17 procedures (Sinha 2009). Due to the high cost of the MAKO system - US$793,000 for the robotic platform and US$148,000 for the partial knee software as of August 2010 (Lang et al. 2011) - it is unlikely that research upon the system may be conducted without support. While in no way is the quality of the above studies called into question, it should be noted that many researchers were in receipt of funds from MAKO Surgical Corp.

2.3.3.2.2 Navio Surgical System

Having acquired FDA approval and a CE mark in 2012 the NavioPFS™ (Blue Belt Technologies, PA, USA) is a relatively new entry to the market (Nikou et al. 2013; Beasley 2012; Klatt et al. 2012). The NavioPFS™, standing for Precision Freehand Sculpting, presents a simplified commercial rival to the MAKOplasty system. Like the MAKO the NavioPFS is a semi-active system. However, instead of providing active constraint via tactile feedback a system of either burr retraction or slowing is used (Jaramaz & Nikou 2012). Furthermore, the NavioPFS™ operates under the image-free classification, therefore removing the cost, risk and complexity associated with CT imagery. However, this does limit the system to intra-operative planning. As with the MAKO the procedure begins with bone pin insertion for IR marker fixation. A standard MIS UKA incision and exposure is performed. As the system is image-free the anatomy of the patient must be entered into the system intra-operatively. To facilitate this, a pointed probe is traced across the exposed surface of the joint while being tracking via the IR stereoscopic system. The system records the position of the probe and uses the data to morph a standard knee model to match that of the patient (Fleute & Lavallée 1998; Jaramaz et al. 2013). This procedure is performed for both the femur and tibia. Further anatomical features are gathered to allow alignment calculation. The ankle centre is obtained by marking the medial and lateral malleoli with the probe. The hip centre is kinematically determined by exercising the femur about the hip (Jaramaz et al. 2013). The articulation and the laxity of the knee joint is also assessed kinematically, allowing improved soft tissue balancing during the planning stages (Jaramaz et al. 2013). The system clearly guides the surgeon through the multistage registration process with on screen prompts and graphics indicating how much data has been collected. As with the MAKO, the system then guides and assists the surgeon with positioning the implant using a virtual model of the now patient-specific bone and a virtual implant. The planning procedure is divided into three stages. Firstly, the implant is sized and positioned. Secondly, gap balancing is performed to ensure a positive, ideally uniform, gap throughout flexion. Finally, the contact points between the tibial and femoral components are checked over full flexion (Jaramaz et al. 2013). Unlike the MAKO the NavioPFS™ is an open system. Therefore, the surgeon may be able to use the system with implants they are already accustomed with (Jaramaz et al. 2013). Once the cutting plan has been confirmed the surgeon may begin resection. The resection is performed with a freehand pneumatic cutting burr, a stark contrast to the large RIO of the MAKO system (Nikou et al. 2013). Visual and mechanical feedback is provided to assist the surgeon with correct resection. Visual feedback is provided, much like the MAKO system, via a real-time 3D model. A colour scale from purple to green to red is used to indicate to the surgeon how close the resection is to going outwith the plan (Nikou et al. 2013; Jaramaz et al. 2013). The NavioPFS™ offers two options for mechanical feedback. Firstly, burr speed control feedback. As the surgeon moves towards the edge of the resection region the burr begins to slow, reducing the rate of resection. If the surgeon continues beyond the planned region the burr stops, preventing excess resection. Alternatively, mechanical feedback may be provided via burr retraction. Under this operation a plastic guard is placed over the burr. While the surgeon is within the resection region the burr is exposed and resection is performed as normal. However, as the surgeon exits the planned region the burr rapidly retracts into the plastic guard of the NavioPFS, immediately stopping resection (Nikou et al. 2013; Jaramaz & Nikou 2012). After resection trial implants may be fitted and the knee may be moved through its full range of motion. The NavioPFS™ system continues data capture and presents the surgeon with immediate post-operative kinematic data. Based upon this information the surgeon may decide to resect more bone to improve the implant fit and the resulting kinematics (Jaramaz et al. 2013). As the NavioPFS™ is a relatively new system it has only limited studies with no long term follow-ups. However, Jaramaz and Nikou report the results of three users performing the femoral resection required for UKA on five synthetic bones each (Jaramaz & Nikou 2012). Average implant position and angular differences from plan of 0.54 mm (SD 0.23 mm) and 1.08° (SD 0.53°) were reported, respectively. Smith *et al.* reported upon the accuracy of 20 UKA procedures on synthetic bones, including femoral and tibial components (Smith et al. 2013). RMS angular and positional errors of all components were reported as 1.46° (Max 3.2°) and 0.61 mm (Max 1.18 mm). Smith *et al.* conclude that these errors are comparable to other commercial robotic systems, such as the MAKO (Lonner et al. 2010; Coon et al. 2010a). A follow-on study conducted upon 25 cadaveric implants found mean rotational errors of 1.04 to 1.88° and 1.48 to 1.98° for the femoral and tibial components, respectively. The translational errors were found to be 0.72 to 1.29 mm and 0.79 to 1.27 mm, for the two components (Lonner et al. 2015).

2.4 Tracking Technologies

2.4.1 Introduction

The fundamental aspect of NOS systems is tracking. It is impossible to give direction if the current location is not known. The majority of the modern systems discussed above use IR stereoscopic optical tracking, while those that do not, use mechanical tracking. However, outside of the CAOS field many other tracking approaches have been developed, many of which will be discussed below and summarised in Table 2.1 at the end of this section.

2.4.2 Technologies

2.4.2.1 Optical

Optical systems are perhaps the most diverse type of tracking systems, and as such they have been subdivided into several of the more common techniques. Broadly speaking systems may be either vision-based or sensor-based (Liming et al. 2012). Vision-based systems rely on computer vision techniques to determine tracking information from 2D images. Meanwhile, sensor-based systems often use additional systems to capture tracking information in a more direct fashion.

2.4.2.1.1 Stereoscopic Infrared

Stereoscopic IR (S-IR) optical tracking is by far the most common tracking technology deployed by knee CAOS systems, as is evident by its use in the majority of the systems discussed above. Further to knee arthroplasty it is also the tracking technology of choice in other areas of orthopaedic surgery, such as hip replacement using the MAKOplasty® system or the modiCAS system, presented by Knappe *et al.* (Tarwala & Dorr 2011; Knappe et al. 2003). Additional applications include navigated spinal surgery, such as in the management of spinal metastatic disease and spinal screw positioning (Kalfas 2001; Nolte et al. 1995; Tamura et al. 2005). S-IR has also been used in patient motion tracking and correction during neurological *positron emission tomography* (PET) imaging (Bloomfield et al. 2003; Woo et al. 2003; Montgomery et al. 2006). Other neurological applications include navigated emitter placement during transcranial magnetic stimulation procedures (Fernandez et al. 2002).

S-IR uses two cameras sensitive in the infrared spectrum to detect markers and calculate their 3D location. The stereoscopic system functions under the principles of epipolar geometry (Hartley & Zisserman 2003). Under the pinhole model of a camera, illustrated in 2D in Figure 2.3, a point X in world-space is projected onto the point x on the image plane P, along the line L joining point X to the cameras projection origin O. A tracking system requires the reverse of this projection, going from the 2D point x to the 3D point X. However, this is not possible. The point X may only be defined as being somewhere along the projection line L.

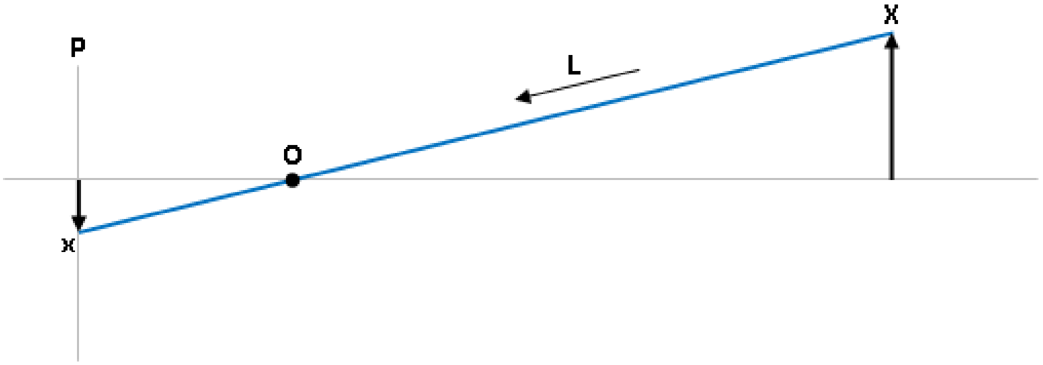


Figure 2.3: Pinhole camera model

Stereoscopic systems, as illustrated in 2D in Figure 2.4, overcome this by introducing a second camera that views the scene from a different position. Therefore, the two points x_1 and x_2, the image point in the left and right image plane respectively, each produce a projection line L_1 and L_2. The point at which these two lines intersect gives the position of the original point X in world-space.

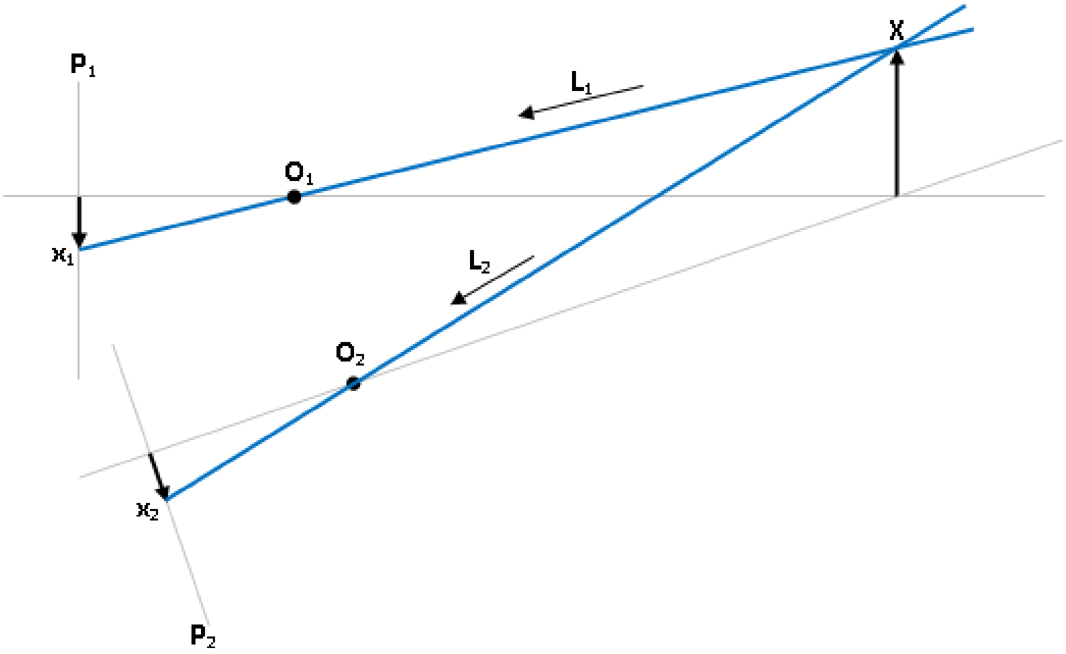


Figure 2.4: Stereoscopic imaging system

For a system to be able to perform this operation it must know the position and orientation of the two cameras relative to each other. This is found through the process of calibration, producing the fundamental matrix containing the positional and rotational separation of the cameras (Hartley & Zisserman 2003; Forsyth & Ponce 2011). A single point found from one marker provides the three positional DOF of an object. Two markers would allow 5 DOF to be determine, while a minimum of three markers are required to determine the full 6 DOF of an object, such as a probe (Ribo et al. 2001).

To simplify marker segmentation, the process of separating the sections of interest from a whole image, S-IR systems use IR cameras. IR cameras are sensitive to wavelengths typically in the near-infrared spectrum, between 700 and 1000 nm, and use filters to remove the majority of the visible spectrum. S-IR cameras typically have two types of marker. Firstly, passive markers are coated in highly retroreflective surfaces. An IR light source, often integrated into the camera, is used to illuminate these markers, making them visible to the camera system (Glossop 2009). Alternatively, active markers are direct IR light sources, often small IR LEDs. Passive markers have the advantage of requiring no separate power source. However, active markers may be turned on and off by a system potentially offering useful applications (Glossop 2009). Furthermore, active markers may be encoded, as with the Qualisys (Qualisys Medical AB, Göteborg, Sweden) motion capture system, to simplify marker matching between stereo images (Josefsson 2002).

Wiles *et al.* reported upon the accuracy of the Polaris optical tracking system (Northern Digital Inc. (NDI), Ontario, Canada). For a single marker within a pyramidal volume no greater than 2400 mm from the S-IR system a RMS error of 0.255 mm was reported. The mean and standard deviation of the error were also reported as 0.193 and 0.167 mm respectively. Finally, it was determined that 95% of single marker positions were within 0.451 mm of their ground truth value (Wiles et al. 2004). The report also investigated the accuracy of the system at determining the position and orientation of multi-marker rigid bodies. The first body, consisting of three passive markers, was found to have a positional RMS error of 0.231 mm ($\bar{x}$ = 0.185, SD = 0.137, 95% CI = 0.462 mm) and an orientation error of 0.383° ($\bar{x}$ =

0.208, 95% CI = 0.713°). The second body consisted of four active markers and produced a positional RMS error of 0.223 mm ($\bar{x}$ = 0.190, SD = 0.135, 95% CI = 0.417 mm) and an orientation mean error of 0.362° ($\tilde{x}$ = 0.256, 95% CI = 0.598°). The difference in active and passive rigid body accuracies did not reach significance. However, the improved accuracy of the bodies compared to single markers, resulting from error averaging, reached significance (Wiles et al. 2004). Elfring *et al*. investigated the accuracy of three optical tracking systems; the Polaris P4, the Polaris Spectra (in active and passive modes) and the Stryker Navigation System II (Stryker, Kalamazoo, MI, USA). The Stryker system had the smallest reported RMS error of 0.067 mm ($\bar{x}$ = 0.058, SD = 0.033, 99% CI = 0.204 mm), followed by the Polaris Spectra in active and passive modes at 0.108 mm ($\bar{x}$ = 0.089, SD = 0.061, 99% CI = 0.285 mm) and 0.192 mm ($\bar{x}$ = 0.170, SD = 0.090, 99% CI = 0.417 mm). The Polaris P4 reported the poorest results with an RMS error of 0.478 mm ($\bar{x}$ = 0.272, SD = 0.394, 99% CI = 2.222 mm) (Elfring et al. 2010). Elfring *et al.* used a 1000 mm diameter spherical measurement volume as opposed to the larger pyramidal volume used by Wiles *et al*. which may account for the reduced errors reported by Elfring *et al*. However, while Wiles *et al.* reported no difference between the active and passive modes of the Polaris Spectra, Elfring *et al.* reported passive mode to produce an RMS error 1.78 ($\bar{x}$ = 1.91, SD = 1.48, 99% CI = 1.46) times that of active mode (Wiles et al. 2004; Elfring et al. 2010). This large difference cannot be readily explained by the difference in measurement volumes used. Khadem *et al.* also produced a report on several tracking systems. The *jitter* – defined as the standard deviation of a set of measurements about their mean – of the Polaris system in active and passive modes was investigated alongside three versions of the Flashpoint (Image Guided Technology, Boulder, Colorado, USA); 300 mm, 580 mm and 1 m (Khadem et al. 2000). For the active and passive Polaris systems jitter was calculated to be 0.053±0.037 mm and 0.115±0.075 mm respectively. The Flashpoint systems produced results of 0.028±0.012 mm, 0.051±0.038 mm and 0.059±0.047 mm in order of ascending camera separation (Khadem et al. 2000). It should be noted that there is limited validity in comparing systems as they were tested across different measurement volumes based upon the intended use of each system. However, the majority of measurement volume dimensions where matched for both modes of the

Polaris system so comparison may be applicable. The findings of Khadem *et al*. concur with those of Elfring *et al.* in that the passive system produces a significantly larger error than the active system. Khadem *et al.* reported that jitter increased significantly for the passive system when a multi marker rigid body was presented at an orientation of greater than 40° (Khadem et al. 2000). This is possibly the result of the spherical passive markers partially obscuring each other, an effect which may be avoided by the lower profile active emitters.

All the studies discussed above report sub-millimetre accuracy. This may explain the preference for S-IR systems for navigated surgery where accuracy is important. Furthermore, S-IR allows tracking of objects without adding excessive bulk or complexity. For example tools as small as needles may be tracked (Berthold & Richter 2009). S-IR is also capable of supporting a large number of markers making it possible to track multiple objects without substantial changes to the system (Josefsson 2002). The main drawback to S-IR systems is the requirement of line-of-sight between camera and markers (Glossop 2009). This is a considerable concern during NOS as it restricts access to the patient during a potentially complex procedure. Although not directly comparable to the knee arthroplasty procedures of interest to this report, Hofer *et al.* reported an average of three (maximum 12) line-of-sight issues per procedure of navigated skull surgery with a duration of approximately 30 minutes (Hofer et al. 2009). Glossop also highlights the risk of tool flexion (Glossop 2009). As the tool markers must be within the line-of-sight of the S-IR system at all times they are often positioned away from the working tip of the device. If the tool flexes the accuracy at the working tip will be greatly reduced.

2.4.2.1.2 Polyscopic Infrared

Polyscopic IR (P-IR) systems use the same principle of point matching between 2D images to derive 3D position as S-IR systems. However, polyscopic systems may use considerably more than two cameras. Systems using as many as 20 cameras are commonly reported (Woellner et al. 2010; Yoganandan et al. 2011; Wixted et al. 2011; Yoganandan et al. 2012). While P-IR systems use multiple cameras they are able to determine a markers position from a minimum of two images. Therefore, P-IR systems are more resistant to occlusion as a complete 360° view of the subject is possible, unlike S-IR systems which provide a single view. However, marker occlusion is still reported as a problem during complex situations, such as tracking finger movement (Zhao et al. 2012). Despite the popularity of P-IR systems in motion analysis there have only been a limited number of publications reporting on their accuracy. Windolf *et al.* demonstrated a systematic accuracy and precision analysis procedure on a four camera Vicon-460 system (Vicon Motion Systems Ltd., Oxford, UK) (Windolf et al. 2008). An overall accuracy of 63±5 µm and precision of 15 µm were reported. It was also noted that the precision was worst in the direction normal to the projection plane of the cameras. Furthermore, the arrangement of the cameras affected accuracy, with different arrangements producing mean accuracies between 76 and 129 µm. A very important aspect of the investigation was the measurement volume – 180x180x150 mm. The effect of measurement volume and camera arrangement on system accuracy may account for the limited published values. The systems are used for a large array of investigations over a broad spectrum of scales. Examples include very small movements such as tooth displacement under load up to total body kinematics over a 10 m walkway (Liu et al. 2007; Van de Walle et al. 2012). Due to this large range of uses, accuracy data produced by any one setup may be irrelevant to another. Jensenius *et al.* investigated the performance of a nine camera Qualisys Oqus 300 system (Qualisys Medical AB, Göteborg, Sweden) and an eight camera Optitrak FLEX:V 100 system (NaturalPoint Inc., Corvallis, OR, USA) on a room size scale (5x7x3 m and 6x8x3.5 m) (Jensenius et al. 2012). The spatial range – defined as the rooted sum of squares of the full range of each axis – was calculated for a stationary passive marker placed in full view of all cameras. Spatial ranges of 0.57 and 3.9 mm were reported for the Qualisys and Optitrak systems respectively. While it is not possible to directly compare the results of Windolf *et al.* and Jensenius *et al.*, due to the two studies using different camera systems and result types, the effect of measurement volume is still suggested as results move from the micrometre to millimetre scale (Windolf et al. 2008; Jensenius et al. 2012). Jensenius *et al.* also report on factors influencing the spatial range. Marker size is reported as having no effect of spatial range, as long as the marker is sufficiently large to be fully detected by all cameras. Sampling rate was also found to have little effect on the results, as did lighting conditions. However, partial occlusion was found to affect spatial range. As more cameras are obscured the system has a smaller data set from which to determine the position of the marker. Furthermore, the calibration process may result in better calibration for some cameras than others. Therefore, occlusion of well calibrated cameras may notably increase the spatial range produced by the system (Jensenius et al. 2012).

P-IR systems offer impressive accuracy, even over a large measurement volume. Furthermore, the problem of marker occlusion is considerably improved upon compared to S-IR systems. P-IR systems are also able to provide very high sample rates. The Vicon T10S system (Vicon Motion Systems Ltd., Oxford, UK) advertises a sample rate of 1000 Hz with a one megapixel resolution image, far higher than required for surgical guidance. A major disadvantage of P-IR systems is the high cost, with a 12 camera Vicon system costing approximately £200,000. Furthermore, the tracking cameras are either attached to a rigid frame around the room, which is unlikely to be practical in an operating theatre, or are on movable tripods which would require considerable space and frequent calibration.

2.4.2.1.3 Time of Flight

Time of flight (TOF) cameras are sensor-based systems that provide distance estimations directly without the need for additional computer vision techniques (Lee et al. 2012). Fundamentally, TOF cameras operate under the principle that light travels at a fixed rate ($c \approx 3.0e8 \text{ ms}^{-1}$) and therefore the distance (*d*) to an object may be calculated by measuring the time (τ) taken for light to travel from the object to the camera, as shown by Equation 2.1 (Kahlmann et al. 2006).

$$d = c\tau \tag{2.1}$$

In practice the light is emitted from the camera and as such the calculation must allow for light transmission and reflection time.

$$d = \frac{c\tau}{2} \tag{2.2}$$

Despite the apparent simplicity, as Mutto *et al.* note, the involvement of the speed of light presents considerable technological challenges (Mutto et al. 2012). Light travels one millimetre in 3.3 ps. Therefore, for a system to have millimetre depth resolution it would require a sampling frequency of 0.3 THz. To overcome this challenge three main approaches have been developed; continuous wave intensity modulation, optical shutter and single-proton avalanche diodes (Kahlmann et al. 2006). Due to its almost exclusive use in commercial systems only continuous wave intensity modulation shall be discussed here (Mutto et al. 2012).

The emitted light signal (S_E), typically in the near IR spectrum, has its maximum amplitude (A_E) modulated by a sinusoidal signal with a modulation frequency (f_{mod}).

$$S_E(t) = A_E[1 + \sin(2\pi f_{mod} t)] \qquad (\ 2.3\ )$$

This signal is partially reflected by the object and is incident upon a receiver. Due to partial absorbance at the object and general propagation attenuation the amplitude of the signal is attenuated (A_R). The reflected signal also acquires undesirable background noise from external sources (B_R). Most importantly, the reflected signal shows a phase delay ($\Delta\phi$) proportional to the distance over which the signal propagated (Kahlmann et al. 2006; Lee et al. 2012).

$$S_R(t) = A_R[1 + \sin(2\pi f_{mod} t + \Delta\phi)] + B_R \qquad (\ 2.4\ )$$

The phase difference is given by the propagation time multiplied by the modulation frequency. Therefore, using the equations above it is possible to determine the distance from the reflected signal (Mutto et al. 2012).

$$d = \frac{c}{4\pi f_{mod}} \Delta\phi \qquad (\ 2.5\ )$$

Attempts to directly obtain the phase shift would again encounter the technical limitations of extremely high sampling rates. Instead the reflected signal is sampled at a minimum of four times the modulation frequency. These samples are then used, as described by Lee *et al.*, to obtain an estimate for the phase shift which in turn provides an estimated object distance (Lee et al. 2012).

TOF cameras extend upon this principle by using a matrix of detectors (Mutto et al. 2012). This allows the systems to simultaneously obtain depths for multiple points producing a depth image. Ideally each detector would have an emitter at the same location. However, this is not readily possible. Instead, most systems use a charge-coupled device (CCD) matrix and a separate emitter array. Several algorithms are then used to compensate for the separation of the detector and emitter and attempt to simulate the ideal situation (Mutto et al. 2012).

Dorrington *et al.* investigated the accuracy and precision of two commercial TOF systems across a measurement volume of 0.8x0.8x0.6 m. The SR4000 (Mesa Imaging, Zurich, Switzerland) was found to have average accuracies of 21.956, 12.667 and 3.909 mm in the x, y, and z axes respectively, while producing respective average precisions of 5.98, 2.24 and 6.54 mm. The XZ422 (Canesta, California, USA) produced x, y and z average accuracies and precisions of 3.111, 3.033 and 3.988 mm and 2.08, 1.48 and 2.97 mm respectively (Dorrington et al. 2010). Kahlmann *et al.* report on the effect of additional calibration on the accuracy of an SR3000 system (Mesa Imaging, Zurich, Switzerland). Prior to calibration the performance of the sensors was highly non-linear. The central region of the system produced standard deviations between 5 and 10 mm. However, moving away from the centre of the sensor the deviation increased to approximately 50 mm for a planar target at 2.45 m (Kahlmann et al. 2006). Kahlmann *et al.* concluded that the change in accuracy was predictable and as such proposed several calibration methodologies to improve accuracy. After calibration, deviation was reported as approximately 10 mm across the entire image.

TOF cameras are able to produce depth images directly at high frame rates. For example the CamBoard nano (pmdtechnologies gmbh, Siegen, Germany) operates at approximately 75 fps. However, TOF cameras have several drawbacks. Predominately, as seen by the above reports, accuracies are relatively low. Furthermore, system accuracies are susceptible to several effects, such as phase wrapping, harmonic distortion and photon-shot noise, as well as more general noise sources (Mutto et al. 2012). TOF cameras are also currently limited to relatively low resolutions with pixel counts often in the tens-of-thousands as opposed to the millions seen in conventional camera systems (Kahlmann et al. 2006; Van den Bergh & Van Gool 2011). This in turn leads to a limited spatial resolution again limiting the maximum accuracy possible.

2.4.2.1.4 Visible Spectrum Monoscopic

Visible spectrum monoscopic (VSM) systems use a single camera, operating in the visible spectrum, to determine positional information. One of the original example of such a system was presented by Kato and Billinghurst in 1999 (Kato & Billinghurst 1999). Kato and Billinghurst later expanded this system, releasing their work as the ARToolKit. The system, and later similar systems, uses high contrast square fiducially markers, as shown in Figure 2.5, to allow robust point detection using relatively simple computer vision algorithms (Kato & Billinghurst 1999). However, advances in computing power and computer vision have allowed more recent systems to operate under a markerless principle, instead extracting natural features from the scene (Lee & Hollerer 2007; Nguyen et al. 2013; El-Sana & Billinghurst 2013).

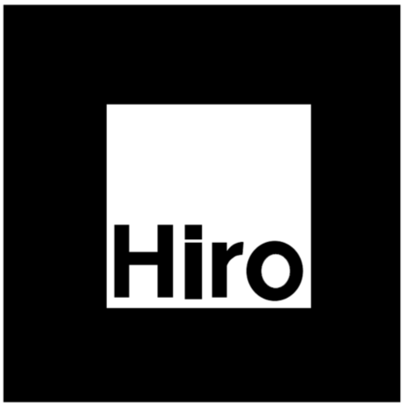


Figure 2.5: Example high contrast fiducial marker

ARToolKit and similar monocular systems use the pinhole model of a camera to determine the relative pose of markers from the camera (Kato & Billinghurst 1999). The corners of known-size square markers are extracted from the colour image using a process of thresholding and contour extraction. The marker is then identified using its central pattern. In the case of ARToolKit template matching is used to compare the image to preloaded user defined images (Kato & Billinghurst 1999). However, other systems tend to opt for an encoded 2D barcode like image, such as the modified Hamming code used by the ArUco system (Munoz-Salinas 2012). Marker pose is estimated by taking the cross product of the marker edges as parallel pairs, producing direction vectors. The cross product of these two vectors produces a third direction vector. These three direction vectors are normalised and represent the rotational element of the marker pose. The translational aspect of the marker pose is then calculated using the known marker dimensions and the corners detected from the image. This estimate is often iteratively improved upon by projecting the calculated marker corners into the image plane and comparing them to the detected corners (Kato & Billinghurst 1999).

Kato and Billinghurst reported upon the accuracy and rotational error using an 80 mm target imaged at 100 to 600 mm. Positional error was found to increase significantly with increasing distance. A maximum error of approximately 5 mm was reported at 100 mm while a maximum error of over 25 mm was reported at 600 mm. Rotational error also increased with distance. Both positional error and rotational error were affected by marker orientation. Greater orientations were found to produce larger positional errors. However, markers parallel to the image plane were found to produce the largest angular error, exceeding 15° at 500 mm distance (Kato & Billinghurst 1999). Abawi *et al.* confirm the findings of Kato and Billinghurst, reporting on a 55 mm marker over the range of 200 mm to 1000 mm (Abawi et al. 2004). A mean accuracy at different orientations of 20 mm was reported at 200 mm distance compared to greater than 100 mm at 1000 mm. Furthermore, angular accuracies of approximately 5° are reported when the marker is orientated above 40°. However, errors increase to over 15° below this orientation.

VSM systems may be produced using low cost, readily available, web cameras. There are also several open source libraries, such as ARToolKit and ArUco, available (Kato & Billinghurst 1999; Munoz-Salinas 2012). This makes VSM systems a very cost effective technology. However, as both Kato and Billinghurst and Abawi *et al.* highlighted, accuracy is lacking. Accuracy may potentially be improved through the use of higher resolution camera and more robust calibration. However, there is no published data to support this. A further advantage of VSM systems is to operate in a markerless manner (Comport et al. 2003; Lee & Höllerer 2009). This enables tracking of natural objects without the need of additional markers or other devices.

2.4.2.1.5 Visible Spectrum Stereoscopic

Stereoscopic imaging in the visible spectrum (S-V) has two main categories. Firstly there are systems that perform very similarly to S-IR systems. That is, matching markers between two images from calibrated cameras to determine the 3D position of the markers. However, unlike S-IR, S-V images are real colour images and a number of computer vision techniques are required to segment and extract the markers from the images. A process made much simpler in S-IR systems by the use of IR illumination and filtering, often reducing the problem to simple thresholding as markers are easily segmented from the background. The second form of S-V is a markerless technique producing depth images of the observed scene. This discussion will focus on the latter categories, as marker based systems do not differ majorly from S-IR systems already discussed.

Depth information is produced through stereo correspondence, the process of matching points between two or more images (Szeliski 2010). Broadly speaking, stereo correspondence has two main forms; sparse and dense. Sparse correspondence was typically the choice in earlier vision systems, predominantly due to limited computational power (Szeliski 2010; Schauwecker et al. 2012). Sparse systems initially detect interesting features in both images. These features may be simple edges (regions of high contrast) or more complex features such as corners or geometric shapes (Hsieh et al. 1992). These features are then matched between image pairs, using standard epipolar constraints, and a range of matching algorithms such as the fast and high accuracy algorithm exFAST (Schauwecker et al. 2012). The disparities, and in turn depths, of matched features are then calculated as described in stereoscopic IR above. This approach produces a sparse depth image where only a limited number of points or regions have known values. However, significantly less computational power is required (Szeliski 2010).

Most modern research is now focused on dense correspondence solutions (Szeliski 2010; Schauwecker et al. 2012). These methods produce depth values for the majority of image pixels. As a consequence dense algorithms are typically significantly more computationally intensive (Szeliski 2010). Many standard algorithms require several seconds per frame to compute a total depth map

(Mattoccia 2010). However, recent advances in computing power, particularly the massive parallelism afforded by new graphics hardware, have allowed algorithms to perform at real time rates. Yao *et al.* report a high accuracy algorithm operating at 35 Hz on an image pair with a resolution of 384x288 using a commodity CUDA graphics card (Yao et al. 2012). Dense correspondence algorithms may be further categorised as local or global. Local methods use only data from small sample windows to determine disparity values. However, global methods utilise the whole image to optimise the disparity values obtained (Szeliski 2010). One of the most basic local methods is block-matching. A small region, for example a 3x3 square, is formed around a pixel with coordinates (x, y) in the first stereo image. The values of this region are then compared using a cost function to those of a second region centred on the coordinates (x-d, y) in the second image. The disparity (d) is chosen such that the cost function is minimised, returning the most likely match between images (Radke 2012). Development of cost functions is an active field of research and as such many algorithms have been reported. Szeliski presents an excellent list of algorithms, ranging from the popular and simple *sum of squared intensity differences* (SSD) and *sum of absolute intensity differences* (SAD) algorithms to more complex and specialised functions such as the dense feature descriptor DAISY algorithm (Szeliski 2010; Hannah 1974; Kanade et al. 1995; Tola et al. 2010).

Examples of sparse matching, whereby the system uses markers much like S-IR have also been reported upon (Fan et al. 2014), producing mean errors of 1.15±2.51 mm and 0.74±1.41 mm measured parallel and perpendicular to the optical axis of the camera, respectively.

2.4.2.2 Acoustic

Acoustic tracking systems use the propagation of sound to estimate position. The majority of systems operate under the time-of-flight principle (Welch & Foxlin 2002). A short pulse of sound, typically in the ultrasound range, is emitted and the time taken for it to reach a receiver is recorded. This time may then be used to calculate the distance between transmitter and receiver (Srinivasan et al. 2009). An alternative phase coherence method uses the phase shift of a continuous ultrasonic wave to estimate relative motion (Schreer et al. 2005).

Both Time-of-flight and pulse coherence systems provide distance estimates based upon the standard propagation equation, as already seen for optical TOF systems in Equation 2.1, whereby c now represents the speed of sound and d and τ represent distance and time respectively. A single transmitter receiver pair may only determine distance, i.e. the radius of a sphere projected around the receiver upon the surface of which the transmitter must be. To obtain 3 DOF three receivers are required, although three transmitters could instead be used (Schreer et al. 2005). In this way each receiver may determine the distance to the transmitter and a system that knows the relative positions of the three receivers may then calculate the position of the transmitter. As with several of the optical systems, to extend the acoustic system to full 6 DOF a minimum of three markers – the devices attached to tracked object, typically transmitters - are required (Foxlin & Harrington 2000).

Despite their relatively low cost and non-invasive nature, acoustic systems have several major disadvantages. Chiefly, large latency and low frequency of measurement data (Schreer et al. 2005; Welch & Foxlin 2002). Due to the relatively slow propagation of sound in air (343.2 ms^{-1}) there is an unavoidable latency of approximately 3 $ms\cdot m^{-1}$ (Lide 2004). The measurement frequency suffers substantially in an attempt to combat the effect of reflections (Welch & Foxlin 2002). Distance is calculated from the time taken for the sound to travel from transmitter to receiver. However, as sound waves are so readily reflected by hard surfaces such as walls and floors not all signals incident upon the receiver will have taken the direct route. While these reflections will not affect the current signal – the shortest time must be the most direct and thus correct distance – the system must pause long enough to ensure all reflected signals have attenuated before transmitting the next signal. Welch and Foxlin state that this wait may be between 5 and 100 ms depending on room acoustics and size (Welch & Foxlin 2002). This alone may cause measurement frequencies to be as low as 10 Hz. Accuracy is a further concern of acoustic systems. One of the main issues is the effect of environmental conditions upon the speed of sound. Temperature, humidity, pressure and air currents all affect the speed of sound in air (Welch & Foxlin 2002; Novák-Marcinčin & Fečová 2010; Lide 2004). As such any change in these properties will affect the accuracy of a system. Again, Welch and Foxlin quantify this problem as causing approximately 1 mm of error per degree Fahrenheit at 1 m distance (Welch & Foxlin 2002). Despite background noise containing only limited ultrasound frequencies, due to their relatively rapid attenuation, noise interference still presents a problem. Welch and Foxlin describe how jingling keys tend to send pure acoustic systems berserk (Welch & Foxlin 2002).

Due to these short comings there is a lack of literature reporting on purely acoustic systems. With many researchers instead opting to produce hybrid systems typically combining an acoustic system with inertial systems (Foxlin et al. 1998; Gilson et al. 2006). However, Foxlin *et al.* report upon the acoustic accuracy of their hybrid system independently of the additional inertial tracking. The acoustic system was reported to produce a distance error of approximately 1 $mm \cdot m^{-1}$ of transmitter-receiver separation (Foxlin et al. 1998).

2.4.2.3 Magnetic

Magnetic tracking is similar to acoustic tracking in that it too requires a transmitter-receiver pair (Glossop 2009). However, as opposed to acoustic systems, magnetic systems typically use a static transmitter with the receivers instead attached to the tracked object (Glossop 2009). The receiver detects the magnetic vector produced by the transmitter. This data is used to determine the position and pose of the receiver.

The transmitter contains three coils wound orthogonally about a single core. These three coils are activated sequentially using either an AC current, producing three dynamic magnetic fields, or a DC current, producing three quasi-static magnetic fields (Welch & Foxlin 2002). In the case of an AC transmitter the receiver also consists of three orthogonally wound coils (Raab et al. 1979). Each dynamic magnetic field induces small currents in each of the three coils, proportional to the orthogonal components of each of the fields produced, giving a total of nine components. These signals are sent via wires to a control system which calculates the relative orientation and position of the receiver based upon *electromagnetic* (EM) theory (Chabay & Sherwood 2011). DC transmitters require fluxgate magnetometer receivers which are able to measure static magnetic fields (Blood 1989; Nabighian et al. 2005). These receivers analyse the orthogonal components of the three magnetic fields produced and the control system may again use EM theory to estimate the position and pose of the receiver.

Nafis *et al.*, while defining a method for the estimation of the dynamic accuracy of EM tracking systems, presented a detailed analysis of several systems (Nafis et al. 2006). The paper focuses on dynamic accuracy as it is highlighted that few manufacturer or researchers attempted to quantify dynamic accuracy, instead opting only to discuss static values, a point supported by much of the literature in this review. However, for completeness Nafis *et al.* also report upon static accuracy of the EM systems investigated, using a system described as 'a high accuracy composite robot, which would be inaccessible to most'. The 2004 1.8 mm microBIRD system (Ascension, Burlington, VT, USA) produced an RMS accuracy of 1.10 mm (SD = 0.63 mm, 95% CI = 2.09 mm). However, a maximum error of 11.93 mm was also reported for this system. The later 2005 1.8 mm microBIRD system improved notably with an RMS of 0.90 mm (SD = 0.46 mm, 95% CI = 1.77), and most notably a maximum error of only 3.61 mm. The InstaTrak Gold (GE Healthcare Navigation, Lawrence, MA, USA) provided a very low RMS of only 0.40 mm (SD = 0.17 mm, 95% CI = 0.67 mm, Max = 1.07 mm). The Aurora 6DoF (Northern Digital Inc., Ontario, Canada) system provided a relatively high RMS of 1.07 mm (SD = 0.61 mm, 95% CI = 2.15 mm, Max = 4.56 mm) (Nafis et al. 2006). Although the results are too numerous to include here, Nafis *et al.* also investigated the distortive effects of several metals common to the operating theatre environment, such as titanium and several types of stainless steel. Nafis *et al.* conclude that distortion from conductive materials is a problem. It was highlighted that metals near the transmitter produce the greatest measurement errors. It was noted that most devices have systems in place to detect distorted fields which attempted to minimise the transmission of distorted readings (Nafis et al. 2006). Mascott compared the use of a magnetic Compass Cygnus-PFS system (Compass International, Rochester, MN, USA) with that of an optical StealthStation (Medtronic SNT, Louisville, CO, USA) system for cranial surgery (Mascott 2005). Mascott determined the magnetic system provided an RMS accuracy of 1.4±0.6 mm. This was described as comparable to the 1.4±0.8 mm accuracy provided by the optical system. Mascott also noted that interference of the magnetic field resulting from conductive objects was rarely observed during procedures (Mascott 2005).

The main concern with magnetic tracking is inaccuracy due to distortion of the magnetic field (Welch & Foxlin 2002; Preim & Botha 2013). Distortions may be induced by ferromagnetic materials, such as magnets used in electric drills or older display devices. Furthermore, and more difficult to avoid, distortions may be created by induced fields in conductive materials (Chabay & Sherwood 2011). In accordance to Lenz's and Faraday's laws a change in a magnetic field will induce a current in a conductive material which will in turn produce a magnetic field which opposes the original field (Chabay & Sherwood 2011). This effect accounts for the preference of DC transmitters over AC. AC transmitters continuously change the magnetic field and thus continuously induce distortive magnetic fields from conductive materials. However, DC transmitters only briefly change the magnetic field when switching between active coils. Therefore, if a brief pause is taken between activating a coil and recording the resulting field the effect of induced conductive material distortion may be minimised (Welch & Foxlin 2002). However, Faraday's law also demonstrates that moving a conductive material through a static magnetic field produces the same inductive effect as a static object in a dynamic magnetic field (Chabay & Sherwood 2011). Therefore, conductive materials are still a source of distortion if they are moving. Magnetic tracking systems also suffer from limited range due to the field strength of a dipole magnetic field decaying with distance in accordance to the inverse cube law (Chabay & Sherwood 2011). Strong magnetic fields are not readily produced and also incur a major risk of distortion, and pose a general hazard not suited to a standard operating theatre (Chaljub et al. 2001).

2.4.2.4 Mechanical

Mechanical tracking uses a physical connection to determine the relative position and orientation of the tracked object from a fixed reference point (Welch & Foxlin 2002). There are two types of mechanical tracking which may be used separately or combined in a single system. Firstly, rigid tracking is performed using rigid sections joined by transducers, such as potentiometers or rotary encoders. Potentiometers are voltage dividers that convert a mechanical angle into an electrical signal through variable electrical resistance. Rotary encoders convert mechanical angle into an electrical signal by inspecting the pattern displayed by a special disk often encoded with Grey code (Barnes 2003). As the tracked object moves, the transducers rotate and the resulting electrical signals are used in combination with the known geometry of the tracking device to determine the relative position of the object. The alternative tracking mechanism uses semi-rigid sections with bending transducers that have a resistive response dependent upon the degree to which they are bent (Stefanescu 2011). These systems work as with rigid systems whereby a computer uses the electrical transducer signals in combination with device geometry to estimate the positions of tracked objects.

Panerai *et al.* reported upon the positional accuracy of a 6 DOF head tracking device developed for active vision experimentation (Panerai et al. 1999). The system measured the angle of five rotational joints using optical rotary encoders and one translational joint using a magnetic transducer. The system was tested over a volume of 700x500x250 mm and produced a mean accuracy of 0.6 mm. The system was also reported to have a very high sampling rate in excess of 1000 Hz (Panerai et al. 1999).

Mechanical systems offer a good degree of accuracy. However they present several major drawbacks which hamper their use in many clinical settings. Primarily mechanical systems typically have a limited range of motion (Zhou & Hu 2008). It is difficult to increase this range while maintaining accuracy as larger and longer tracking arms are more susceptible to unintentional bending or flexing. Furthermore, mechanical systems are arguably the most intrusive tracking system as they require physical coupling to the tracked object at all times (Zhou & Hu 2008). Mechanical systems are also limited by complex expansion to multiple object tracking. With the majority of the systems discussed within the review additional objects may be tracked with the simple addition of extra markers or sensors (Welch & Foxlin 2002). However, mechanical systems require substantial modification to allow additional object tracking, often an entire secondary device.

2.4.2.5 Inertial

Inertial tracking systems based upon Newtonian physics first found use in the mid-1900s in submarines and aircraft (Welch & Foxlin 2002). However, due to their reliance on heavy flywheel gyroscopes their use in small scale tracking, such as that of the medical field, was found unfeasible. However, the last two decades have seen the growing availability of *micro-electronic-mechanical system* (MEMS) type devices (Bao & Wang 1996; Brown et al. 2001). MEMS inertial devices are extremely lightweight and self-contained making them a strong candidate for object tracking (Welch & Foxlin 2002).

Inertial devices contain two types of element; gyroscopes and accelerometers. Three orthogonal gyroscopes are used to determine the orientation of the device. The full explanation of the physics behind gyroscopes is quite lengthy and beyond the scope of this review. An excellent mathematical description is given by Armenise *et al.* (Armenise et al. 2010). The basic principle of a traditional gyroscope requires a heavy wheel to be spun at high speed to produce a large angular momentum – along the axis of rotation with direction dictated by the right hand screw rule. When an external force is applied, i.e. the submarine changes course, Newton's third law states that the system must produce an equal and opposite reaction. The system presents this reaction via an additional angular momentum. This angular momentum must be provided by a new rotational velocity. In the case of an external force acting perpendicular to the original axis of rotation the axis of rotation of the new induced rotation is antiparallel to the external force. Furthermore, the velocity of the induced rotation is proportional to the force that induced it. By integrating this velocity it is possible for the system to determine the change in orientation of the gyroscope. With three orthogonal gyroscopes it is possible to obtain orientation to the full 3 DOF. New MEMS gyroscopes replace the heavy rotating wheel with an oscillating beam on the micrometre scale (Acar & Shkel 2008). Accelerometers, the second component of an inertial tracker, utilise Newton's second law to determine the position of the device. Traditionally, a mass (m) is held in place by a spring with spring constant (k). If the device is moved it will induce a force (F) which will cause the mass to move by an amount (x), compressing or expanding the spring. Hook's law, shown by Equation 2.6, allows the calculation of the induced force.

$$F = kx \tag{2.6}$$

Newton’s second law, Equation 2.7, allows this force to be converted to acceleration, the acceleration of the movement of the device.

$$F = ma \tag{2.7}$$

A system may then double integrate the acceleration of the device to determine its position relative to an initial reference point. Inertial devices first calculate orientation so that the effect of gravity may be removed from the accelerometers calculation of position (Foxlin et al. 1998).

Zhou *et al.* investigated the use of two commercial 6 DOF MT9B inertial sensors (Xsens, The Netherlands) in the tracking of upper limb joints for rehabilitative purposes (Zhou et al. 2008). The accuracy of the system was evaluated through co-tracking by a CODA motion tracking system (Charnwood Dynamic Ltd., UK). The limb was moved through a range of random and planned motions and the initially estimated joint positions were compared to the ground true values of the CODA system. The RMS positional and angular accuracies of the inertial sensors were concluded to be <10 mm and 2.5-4.8° respectively. Ren and Kazanzides report upon an endoscopic surgical tool tracked via inertial sensors (Ren & Kazanzides 2012). An additional magnetic sensor was used to alleviate the effect of drift experienced by traditional inertial sensors. The custom inertial device contained an LIS331DLH three DOF accelerometer (STMicroelectronics, Geneva, Switzerland), an IDG300 two DOF and ISZ300 one DOF gyroscope (InvenSense, San Jose, CA, USA) and a HMC1043 three DOF magnetometer (Honeywell, Morristown, NJ, USA). This device was rotated through three DOF for 100 seconds while simultaneously being tracked by a Polaris optical tracking system (Northern Digital Inc., Ontario, Canada) to provide ground truth data. The orientation RMS accuracies of the inertial system were found to be 0.96, 0.76 and 1.06° for roll, pitch and yaw respectively (Ren & Kazanzides 2012). No attempt was made within the report to quantify the positional accuracy of the device. Foxlin investigated the use of inertial sensors for wide range tracking of human ambulation (Foxlin 2005). An InertiaCube3 (InterSense, Billerica, MA, USA), with triaxial accelerometer, gyroscope and magnetometer, was attached to a subject’s foot and recorded their movement through a two storey house. The subject covered 118.5 m within the house over a 322 second period. As a measure of accuracy the subject returned to their initial position at the end of the experiment. The inertial system reported a final positional difference of 0.32, 0.10 and 0.06 m in the x, y and z directions respectively. This equated to a positional drift of approximately 0.3%. A further outdoor experiment over a 741 m path also resulted in a drift of approximately 0.3% (Foxlin 2005). These initial experiments used the magnetometers and an extended Kalman filter to minimise gyroscopic drift through comparison with the magnetic field of the earth. The system was later extended by incorporation of a GPS system allowing long range outdoor tracking. Yun *et al.* also report on the use of inertial systems for long range tracking (Yun et al. 2007). The system determines relative distance walked by a subject using a foot mounted inertial tracker. The effect of drift is minimised by assuming the tracked foot is stationary during gait phase and thus instructing the system to ignore signals at this time. Walking and running tests were performed over 120 m track and produced maximum displacement errors of 3.3 and 4.75 % respectively.

Due to MEMS technology very small scale inertial devices are now available (Woodman & Harle 2008). For example the MTx triaxial tracker (Xsens Technologies, Enschede, the Netherlands) measures just 38x53x21 mm (Saber-Sheikh et al. 2010). Inertial devices are also self-contained units, requiring no additional sensors or signal generators (Schreer et al. 2005). Furthermore, as a result of this there are no line-of-sight or measurement volume restrictions as seen with the technologies discussed above. Some systems, such as the InertiaCube3 used by Foxlin, have integrated wireless data communication, allowing a wire free battery powered system with a potentially unlimited measurement volume (Foxlin 2005). These advantages present inertial tracking as a near ideal solution, leading to statements such as, “Inertial trackers might appear to be the closest thing to a silver bullet” (Welch & Foxlin 2002). However, inertial trackers have one major flaw: drift. As highlighted by much of the literature reviewed above the main challenge faced by those designing inertial systems is overcoming the effect of drift. This explains why the majority of systems combine inertial trackers with other sensors, such as magnetometers or GPS (Foxlin 2005; Yun et al. 2007; Zhou et al. 2008; Ren & Kazanzides 2012). Positional drift, as observed in most of the aforementioned studies, is the result of the system’s accelerometers giving a slightly incorrect value.

As inertial systems are only able to detect relative displacements, these errors compound with each reading and cause the gradual increase in positional error. Welch and Foxlin quantify how a single accelerometer producing an error of only 1 mG would, in only 30 seconds, produce a positional error of 4.5 m (Welch & Foxlin 2002). The drift error is not typically caused by the accelerometers directly, but is instead an effect of improper compensation of gravity due to an orientation error. Even high-end gyroscopes experience slight drift which over a short period can reach 1 mrd of error, sufficient to cause the 1 mG error previously discussed (Welch & Foxlin 2002).

2.4.3 Comparison

The table below summarises several of the key aspects of the above technologies based upon the literature previously cited.

Technology	Accuracy	Precision	Sampling Frequency	Measurement Volume	Interference Susceptibility	Cost
Stereoscopic IR	High (~0.25 mm)	High (<0.15 mm)	Medium (60 Hz)	medium (~8 m3)	Limited interference due to use of IR. However, highly susceptible to line-of-sight	high
Polyscopic IR	v.high 0.5 - <0.1 mm)	v.high (~0.015 mm)	v.high (1000 Hz)	high (>300 m3)	Limited interference due to use of IR. Slight susceptibility to line-of-sight	v.high
Time of Flight	Low (~5 mm)	Low (~3 mm)	Medium (90 Hz)	medium (~6 m3)	Slight interference risk from optical noise	low
Visible Monoscopic	Low (~5 mm)	NA	Medium (60Hz)	medium (~8 m3)	Some interference risk due to lighting susceptibility and marker-like objects	v.low
Visible Stereoscopic	Medium (~1 mm)	Low (~2 mm)	Medium (40 Hz)	medium (~8 m3)	Some interference risk due to lighting susceptibility and marker-like objects	v.low
Acoustic	Medium (~1 mm)	NA	Low (10 Hz)	low (~2 m3)	High interference risk from background noise susceptible to temperature, humidity and air pressure	low
Magnetic	High (<1mm)	High (<0.5 mm)	Medium (40 Hz)	low (~1 m3)	Large interference risk from magnetic and metallic objects	high
Mechanical	High (<1mm)	NA	v.high (1000 Hz)	low(~1 m3)	Susceptible to limb bending and errors over larger volumes	medium
Inertial	High (<1 mm)	NA	v.high (>1000 Hz)	v.high (>1000 m3)	Susceptible to drift leading to increased error over time	low

Table 2.1: Tracking technologies comparison

2.5 Augmented Reality

2.5.1 Introduction

Augmented reality (AR) is the addition of virtual elements to the real world. Unlike *virtual reality* (VR), where a user is entirely immersed in a virtual world, AR systems blend the virtual and real to produce an augmented world (Duh & Billinghurst 2008). Azuma *et al.* define AR independently of hardware and software implementations as shown in Table 2.2 (Azuma et al. 2001).

Combine real and virtual objects in a real environment
Run interactively in real time
Register objects in 3D

Table 2.2: Augmented reality system requirements

The first two points are self-explanatory while the third point iterates that 2D image or text overlays do not constitute AR, even if interactive at real time rates (Azuma 1997). For example, live subtitles or the small graphic often used to name and describe a person on the news do not constitute AR. It should also be noted that while this report pertains to visual AR the definition does not apply this limit. Aural AR is also a possibility, augmenting audio data into a real environment, with applications in assistive devices for the visually impaired and tourism (Blum et al. 2012; Vazquez-Alvarez et al. 2011). Although considerably less reported, AR may be applied to any sense including olfactory and gustatory (Krevelen & Poelman 2010).

The first example of AR was presented by Sutherland in the 1960s (Sutherland 1965; Sutherland 1968). The system relied on a large, bulky and fairly impractical see-through HMD. Little progress in the field is reported until the 1990s as which point small light weight computing systems became increasingly available (Krevelen & Poelman 2010). Around this time Caudell and Mizell are attributed with first using the term ‘augmented reality’, while developing a device to assist with aircraft construction (Caudell & Mizell 1992). By the late 1990s AR had gained recognition as a distinct field with several conferences dedicated to the topic (Azuma et al. 2001; Krevelen & Poelman 2010). As computer power increases and lightweight portable devices, such a smartphones, become increasingly popular and common place so too will augmented reality.

2.5.2 Displays

Augmented reality requires both accurate tracking and real time display. Tracking allows the system to relate virtual elements to the real world position of the user. These virtual elements then require a real time display to allow the augmentation of the user's view. Krevelen and Poelman apply two levels of categorisation to AR display devices (Krevelen & Poelman 2010). Firstly, positioning separates displays into either head mounted, hand held or spatial. HMD, like those of the first AR system presented by Sutherland (Sutherland 1965), attach a display device in front of one or both eyes of the user, typically using a helmet, goggles or glasses type frame (Cakmakci & Rolland 2006). Hand held displays are small displays which a user may use, almost like a magnifying glass, to see into an augmented world. Smartphones are commonly used as hand held displays (Schmalstieg & Wagner 2007). This has caused a particularly rapid rise in popularity of hand held displays due to high powered smartphones with integrated cameras becoming relatively common place (Duh & Billinghurst 2008). Spatial displays are placed statically in the environment such that they are a window into the augmented world. Krevelen and Poelman suggests that spatial displays are better suited to presentation type activities, with limited interactivity, due to the ease with which a large group may use spatial displays as opposed to hand held or head mounted devices (Krevelen & Poelman 2010).

The second categorisation of display devices is based upon the mode of visualisation. Displays typically operate as either *optical see-through* (OST), *video see-through* (VST) or projective (Duh & Billinghurst 2008; Krevelen & Poelman 2010). OST display devices allow the user a direct view of the real world while overlaying virtual information using the display device. Typically half-silvered mirrors or prisms are used to merge the light of the virtual image, projected from video displays, with the light directly from the real world (Duh & Billinghurst 2008; Azuma et al. 2001). OST displays allow the most natural display of real elements, as there are no resolution limits, contrast limits, frame rate limits or latency problems – beyond those of natural human vision. However, the lack of control over the real imagery may complicate virtual overlay (Azuma et al. 2001). Latency is a major concern in OST displays. The real imagery is instant and without video capture and redisplay it is not possible to introduce a delay. However, the virtual image will always have some delay due to processing times. If this delay becomes too great there will be a notable lag of the overlay relative to the real world (Hilliges et al. 2012). Also, if the real world is too bright, for example, it may not be possible to display the virtual image with sufficient contrast such that it is visible (Rolland & Fuchs 2000). OSTs also require per-user calibration for best results, particularly when used in HMDs. Eye separation and position are variable among different users, therefore for an accurate overlay the system must be calibrated to ensure proper registration of real and virtual elements (Rolland & Fuchs 2000).

Video see-through display devices present the user with a full video display, where both virtual and real imagery is displayed using a monitor type device (Duh & Billinghurst 2008). The real world is captured using one or more cameras and the virtual image is then digitally overlaid before the final image is displayed to the user. As both the real and virtual images are processed problems such as contrast and latency differences may be overcome. However, VST displays still suffer latency problems as the image the user sees is delayed relative to their motion, a situation which may lead to an effect similar to motion sickness, known as simulation sickness (Howarth & Costello 1997). VST displays also provide reduced depth cues (Liu et al. 2010). Depth cues are pieces of information used by the brain to determine the depth of an object. The convergence cue is the relative angle of the eyes so that an object is centred in the image of both eyes. The less reported focus cue is the amount of lens distortion applied to focus an object (Liu et al. 2010). As all objects, irrelevant of their true depth, are displayed on the 2D plane of the display device it is not possible for the eyes to effect object focus by adjusting the lenses. Therefore this cue is lost and depth perception is reduced (Liu et al. 2010). VST displays also reduce the resolution and frame rate of the real world, subtracting from its clarity and authenticity (Rolland & Fuchs 2000).

Finally, projective displays project the virtual information directly onto the surface to be augmented (Choi & Kim 2012). This modality is similar to that of OST in that the physical world is preserved. As such it also presents the delay issues discussed with OST systems. Projective systems offer improved group interaction, as multiple users observe the same augmented reality without additional hardware (Samosky et al. 2012). The appearance of the projected image is dependent upon the topology of the surface onto which it is projected. Therefore, projective systems must measure the surface topology and transform the projected image as required. An additional challenge of projective systems is shadowing caused by objects, such as user's hands, coming between projector and surface. However, similar effects are seen with the other display modalities where the augmented object is rendered above nearer objects.

2.5.3 Examples

Augmented reality has found use in many walks of life. Several medical uses are discussed below as well as a number of interesting non-medical uses.

2.5.3.1 Medical

Nicolau *et al.* present an AR system for thermal ablation of the liver (Nicolau et al. 2009). The system uses an S-V marker tracking systems to track both the abdominal skin of the patient and the needle tool. Internal organs, obtained through preoperative CT, are registered to the skin markers. Additional algorithms manipulate the virtual organs to compensate for the patient's breathing intraoperatively. The augmented image is displayed using a spatial display not aligned with the patient. Additional, pure virtual, imagery is also shown, such as a magnified view of the needle tip. The system allows the user to accurately position the tip of the needle relative to the liver without requiring any incision or exposure of the liver. Nicolau *et al.* concluded that the system provided quick and accurate needle positioning. Although, it was noted that a number of effects, such as needle bending and marker registration, reduced the accuracy to an average value of approximately 4.5 mm (Nicolau et al. 2009). Furthermore, due to the computationally heavy processes involved the system produced a peak frame rate of only 10 Hz.

Navab *et al.* modified a Siremobile Iso-C 3D C-arm system (Siemens Medical Solutions, Erlangen, Germany) to provide augmented imagery (Navab et al. 2010). A C-arm is an intraoperative X-ray imaging device capable of quickly providing X-ray images from a wide range of positions and orientations which may be processed to produce CT-like images (Strobel et al. 2009). The standard C-arm provided only X-ray images to assist the surgery with orthopaedic procedures. Navab *et al.* connected an optical camera with a two-mirror system, to align the X-ray and video image centres. Using image registration techniques the X-ray and video images were merged and displayed on a spatial monitor. This greatly simplified the task of relating features in the X-ray image to real world visible features visible to the surgeon. Navab *et al.* report on the systems use for procedures such as pedicle screw placement and vertebroplasty, concluding that the augmented C-arm system produced promising results (Navab et al. 2010). Navab *et al.* determined an image overlay accuracy of 0.92-1.05 mm. However, Chen *et al.* recently extended upon this work, applying additional distortion correction, and reported a final overlay accuracy of 0.53 mm (Chen et al. 2013). Importantly, unlike many AR systems, the augmented C-arm does not require any additional tracking systems. Furthermore, one-time calibration is sufficient to obtain the overlay accuracies reported above (Navab et al. 2010; Chen et al. 2013).

Blum *et al.* developed an augmented reality ultrasound simulator to improve ultrasound training (Blum et al. 2009). The system is designed to simplify the very difficult task of learning how to use an intraoperative ultrasound imaging device. A fake ultrasound probe is tracked, using IR markers, relative to a human torso phantom. A CT data set, upon which the phantom was designed, is then used to generate virtual ultrasound images based upon the probes position. The user wears a VST HMD which applies several augmentations to their view. Firstly two virtual monitors are produced. These display the simulated ultrasound image and the corresponding CT slice. An internal view of the phantom is augmented onto the image showing the simulated beam of the ultrasound probe. This provides contextual in-situ visualisation of the procedure to help the user understand where they are looking and what the ultrasound images are showing (Blum et al. 2009). Bichlmeier *et al.* also report on a proposed AR training system (Bichlmeier et al. 2007). A VST

HMD allows the user to see internal views of a subject merged with external video imagery. Head and tool tracking relative to the target is provided by a combination of monoscopic IR marker tracking and P-IR marker tracking, providing a reported accuracy of 0.35 mm (Bichlmeier et al. 2007). The report highlights how standard AR overlays can be unconvincing at producing internal objects. Instead of appearing internal these objects often appear to be above the skin. To prevent this Bichlmeier *et al.* developed an algorithm to vary the transparency around the virtual overlay producing a much more convincing internal effect (Bichlmeier et al. 2007).

Mondal et al. reported upon an AR fluorescene application, whereby the scene is captured by a single lens camera with both colour and NIR sensors. The NIR fluorescene image may then be augmented onto the colour image. This was used in mice and humans to image tumor location onto a HMD to guide surgical resection.

AR is presented as improving surgery through increasing accuracy and intuitiveness through guidance. However, Dixon *et al.* presented the concept of augmented blindness, whereby the augmented overlay obstructs the detection of otherwise noticed complications. In two separate studies performed with an AR endoscopic system used by surgeons (n = 32 and 50) 6.7 and 32.0% of pure AR users detected a foreign body, compared to 41.2 and 60% of standard monitor users (Dixon et al. 2013; Dixon et al. 2014). Both studies were found to reach significance (P = 0.04 and P = 0.02).

2.5.3.2 Non-Medical

Kato and Billinghurst discuss the use of their very popular ARToolKit software to produce a AR conferencing system (Kato & Billinghurst 1999). The system displays virtual monitors to a user through an OST display. The position and orientation of these virtual monitors are dictated via fiducial markers which are tracked using visible spectra monoscopic techniques as previously discussed. The system also features a virtual whiteboard which is shared by the different users allowing an additional form of communication.

Schoenfelder and Schmalstieg describe an AR system for building acceptance (Schoenfelder & Schmalstieg 2008). The system allows the user to compare a physical building with its intended design. Typically this is a very time consuming job, requiring highly skilled personnel. However, the AR system proposed allows a much quicker inspection. The system uses a mobile camera-monitor platform which is tracked as it moves throughout the factory by four IR cameras using passive IR markers. Using carefully positioned markers the virtual model is registered to the physical building. The User moves the device throughout the environment and uses the VST spatial display to inspect regions of interest. Any deviations from the original design are easily detected as the physical features will not properly align with the virtual overlay. Schoenfelder and Schmalstieg note that the accuracy of the system is distance dependent. While an object at 2 m may have an optimal accuracy of 3.5 mm an object at 10 m has an accuracy of only 17.5 mm. Further errors, resulting from the orientation accuracy of the tracking system, may result in additional inaccuracies of 70 mm at 10 m. However, it is reported that these values are below those typically required for building acceptance (Schoenfelder & Schmalstieg 2008).

The AR sandbox developed by Reed *et al.* is an interesting application of projective AR (Reed et al. 2014). A Microsoft Kinects system is used to image a sandpit from above. The Kinects determines the topology of the surface and a calibrated projector is then used to project this topology back onto the sandpit using colour coded elevation gradients. Users are able to dig in the sandpit and build mounds and the system responds in real time. Additionally virtual water may be added to the system allowing pools, rivers, and lochs to be simulated by manipulating the physical sand.

2.6 Summary and Rationale

This review clearly indicates that OA of the knee is already a global health concern; and with the projected rises in risk factors such as obesity and an aging population the severity of the problem is predicted to worsen. As arthroplasty is still considered the only viable option for those affected by the most severe forms of the disease it is becoming increasing important to ensure the best possible surgical outcomes. The importance of accurate implant alignment and positioning has been highlighted with regards to functional quality and the potential longevity of the new joint. Navigated systems have been shown to consistently allow the implantation of knee prostheses with exceptionally high accuracy. Furthermore, it has been shown how navigated systems permit less invasive procedures, reducing patient recovery and the associated costs of lengthy hospital stays. This makes navigated systems a prime candidate for improving knee arthroplasty procedures and providing the consistently good results demanded.

However, current navigated systems are not ideal, with cost, size, and increased complexity being cited as reasons for opposition to adoption, resulting in limited uptake. Therefore, alternative tracking technologies were investigated with the intention of identifying systems able to provide the accuracy required for UKA, while reducing both size and potential costs of the system.

Furthermore, the concept of AR was explored to reduce the complexity of the system through intuitive guidance. The underlying principles of AR were reported upon as well as the various display devices that are fundamental to the concept. Examples of the application of AR in both the medical and wider community have shown how the technology may simplify complex tasks by presenting the user with a combination of real and virtual information. Therefore, the application of AR to UKA guidance may reduce the complexity of the system, the learning curve of surgeons, and the operation time.

It is hypothesised that combining a compact, small scale, yet still accurate tracking system with AR guidance would allow the delivery of a CAOS system for UKA system that would better meet the requirements of surgeons and medical facilities, thus increasing uptake of CAOS technologies.

3
Aims and Objectives

The original research aim proposed to develop a small scale, markerless, tracking system to replace the Polaris tracking camera used by existing commercial systems.

To meet this aim a tool-mounted markerless optical tracking system, which directly tracked bone, was envisioned. This approach combines the tracking and resection systems, eliminating the bulk of a standalone tracking camera. Additionally, by placing the tracking system on the tool the region of space that must be kept clear to allow constant line-of-sight would be considerably reduced.

Another concern of existing CAOS systems was the increased complexity, resulting from additional processes and tools required by the system. This may slow the learning curve of surgeons and increase operational time and associated costs (Garvin et al. 2011; Zhang et al. 2011). Therefore, the provision of intuitive guidance was considered key to the proposed system to minimise these issues.

Initial research, discussed in Chapter 5, suggested a markerless approach would struggle to meet several of the requirements of the system, including accuracy, precision, and latency. Therefore, the markerless requirement was omitted and a broader range of tracking techniques was investigated.

In light of this omission, the following project aim was defined. Develop a proof of concept CAOS system for UKA procedures based upon a miniaturised, less bulky and intrusive, tool-mounted tracking system, which provides consistently accurate resection via intuitive and immersive guidance.

To fully define the above aim it was necessary to define accurate resection. The majority of literature reports upon the accuracy of CAOS system based upon the post-operative alignment of the limb, or implant components, by comparison to the pre or intra-operative plan. A threshold of success of 3° difference is typically applied (Brin et al. 2011). For the development of a novel system this measure is non-ideal as it requires a complete system to allow analysis. Therefore, this 3° threshold was converted to a resection and probing accuracy threshold. Figure 3.1 below shows a schematic of the knee in the coronal and sagittal planes. Assuming an average inter-midcondylar distance of 60 mm, posterior-anterior tibial length of 50 mm (Terzidis et al. 2012) and a single component uniaxial error, trigonometry defines a 3° error as resulting from a 3.1 or 2.6 mm translational error, respectively.

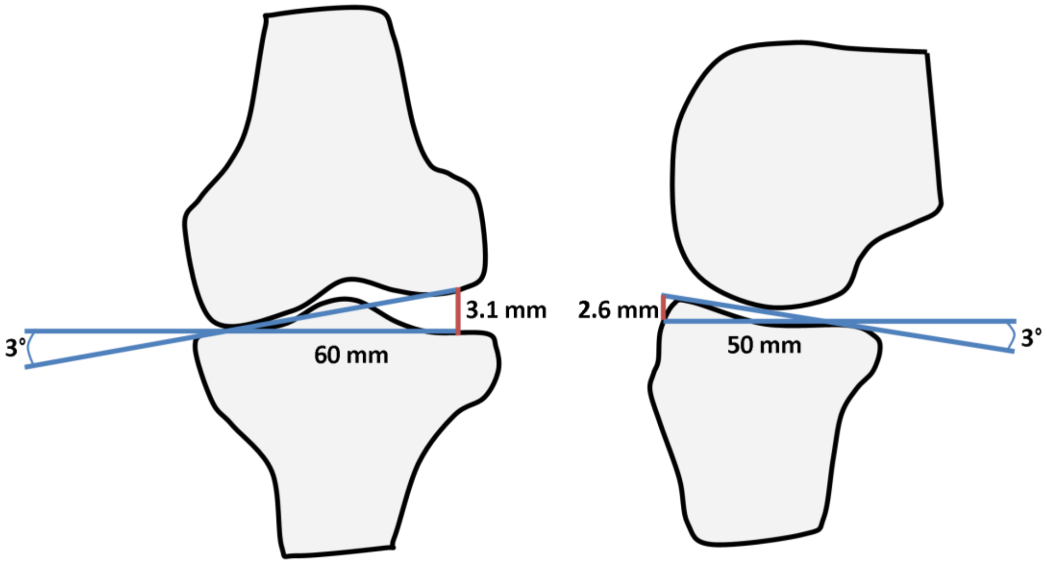


Figure 3.1: Approximate translational error required to induce a rotational error of 3°

These translational errors are divided by two to allow for an error in each component or distal ends of each component. Therefore, a maximum translational or resection error of 1.3 mm is defined. This agrees with the component translational accuracy reported by Lonner *et al.* for the commercial Blue Belt Navio® system (Lonner et al. 2015). Due to the common use of cemented implants a final probing and resection accuracy of 1.0 mm was defined, to allow for final implant fitting error.

4 System Analysis and Requirements

4.1 System Development Methodology

An iterative and incremental development methodology was used throughout the project. This combined methodology incrementally develops the system, adding new sections or features, gradually increasing its complexity and scope. In addition to the incremental development, existing system sections are improved through iterative cycles of development and evaluation. This general methodology was chosen as it allows the rapid development of a basic prototype, which may then be expanded and refined as is deemed necessary. Alternative methodologies may opt to fully develop each section of a system before starting the next. This carried a higher risk due to the explorative nature of this research.

System development started with the problem analysis, discussed in Section 4.2 below. This defined the core requirements for the system, as well as several additional features. A basic prototype consisting of the core elements was developed in a stage known as initialisation. The core elements identified in the chapter below were developed to a minimal working level in an order based upon their importance and perceived risk. As such, important, high-risk, elements were developed first. This ensured the maximum freedom for developing an alternative solution in case of a problem. Typically the development of features is time-boxed, however, due to a lack of experience in programming and vision systems a feature-boxed approach was taken for the majority of core elements. This permits development of a section until completion, as opposed to working to a fixed time frame.

Once a basic prototype was achieved iterative and incremental stages were undertaken. Incrementally added sections often required iterative improvement of existing work as new requirements were presented. Iterative improvements were also performed as new techniques and approaches were discovered through incremental progress or continuous literature review.

Three generations of system are presented within this thesis. Transitioning from generation one to generation two saw a complete rebuild of the system from the ground up, followed by a series of important incremental additions. Generation three was able to borrow some modules from generation two. However, the base architecture of the system was redeveloped to better suite interfacing with a head-mounted display, a process described fully in Chapter 8.

4.2 Problem Analysis

Before designing a system it is fundamental to fully understand the problem that the system is intended to resolve. This problem analysis takes a cause and effect approach, in that after identifying the general problem the distinct causes of this problem are then determined. This method is repeated for each cause until only root causes remain. A root cause is defined as a cause that has no cause of its own.

From its most general perspective the system serves as a bridge between surgeon and patient. As discussed in Chapter 3 the broad aim of the system is to guide a surgeon through a UKA procedure. This is shown as a *unified modelling language* (UML) use case diagram in Figure 4.1.

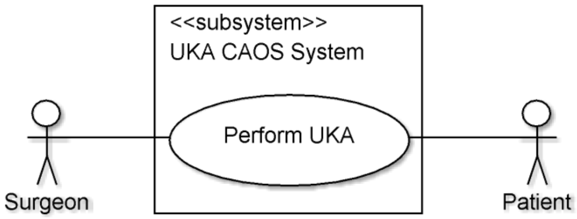


Figure 4.1: Basic UKA CAOS system use case diagram

It is proposed that a guided UKA procedure consists of three core stages, each resulting in a separate, yet connected use case. Planning is often the first stage of a guided procedure. However, in a desire to avoid the costs and risks associated with CT and other pre-operative imaging systems, an intra-operative planning approach shall be taken. Therefore, prior to planning it is necessary to intra-operatively generate a model, upon which a plan may be constructed. This provides the first core stage and use case, model generation. This is evidently followed by the second core stage, implant planning. Lastly, the plan is carried out in the final core stage, bone resection. This is illustrated by the UML use case diagram below in Figure 4.2. It is notated that both the surgeon and patient interact bi-directionally with all use cases, excluding implant planning.

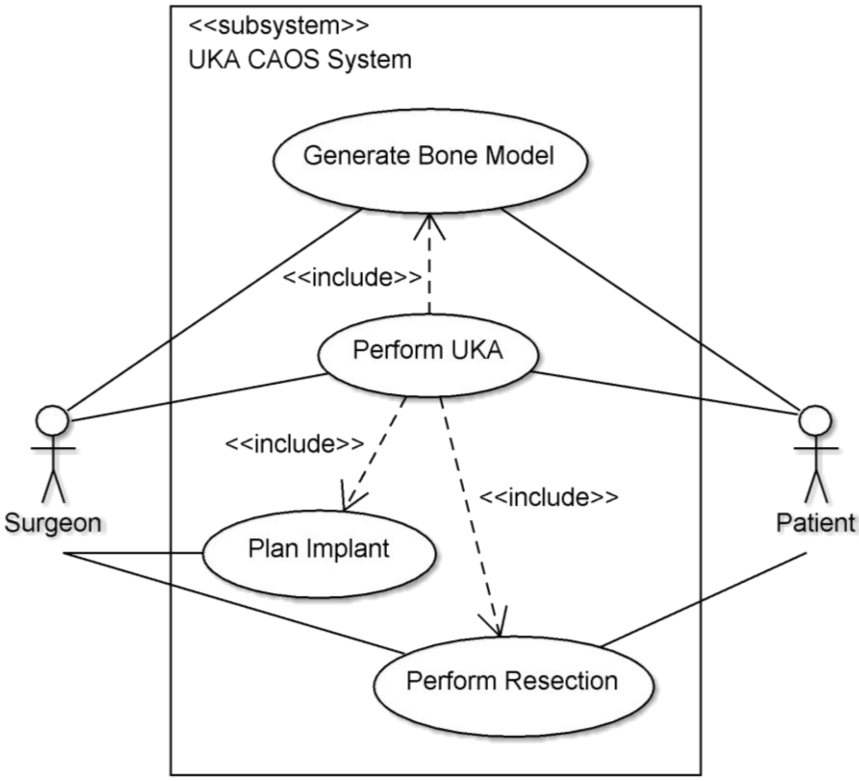


Figure 4.2: Expanded use case diagram of proposed UKA CAOS system

Each of the three core stages are further divided into sub-processes. The generation of a bone model requires several processes. Initially, a virtual model of the articulating surfaces of the joint of the limb must be generated. This requires model generation for both the tibial and femoral surfaces. In addition to the surface models, full bone model generation requires joint centres to be located. As noted in Chapter 2 the HKA angle is fundamental to the success and therefore planning of a UKA procedure. Due to the proposed imageless approach, both surface generation and joint centre location shall be performed intra-operatively. To enable this, the patient and any tools, such as probes, must be tracked throughout the procedure. This is illustrated by the use case diagram shown below in Figure 4.3. Dashed arrows indicate the dependency of one use case upon another, while arrows with closed heads indicate a generalisation of use cases.

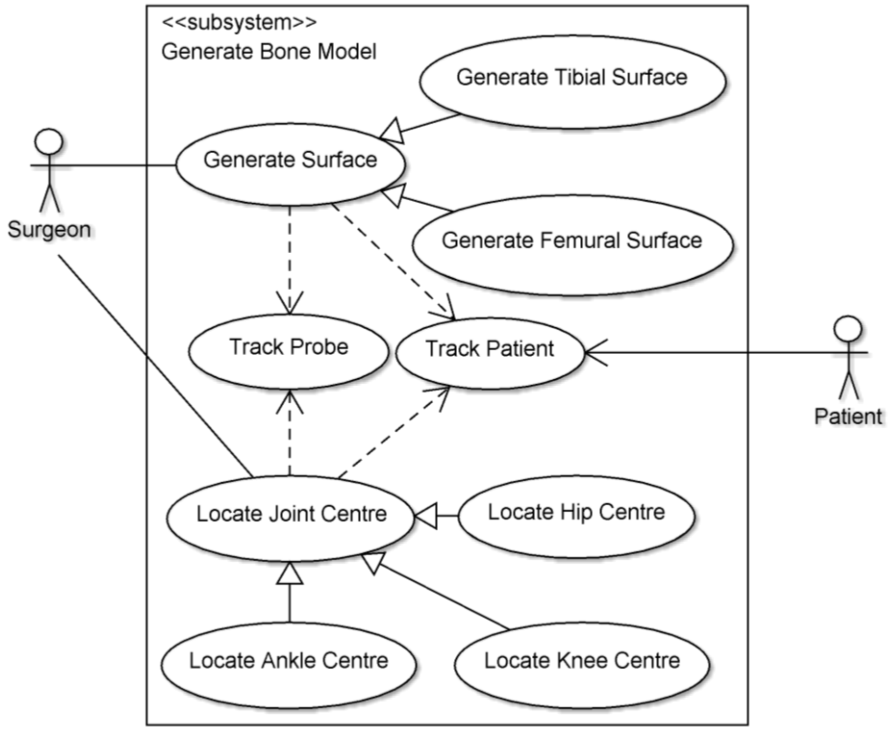


Figure 4.3: Generation bone model use case diagram.

As discussed, the planning stage shall utilise the surface models and joint centres captured by the bone model generation subsystem above. Planning entails the surgeon positioning virtual implant models onto the joint surface models. The joint centres will be used to determine the HKA angle resulting from the implant size and position, which is relayed to the surgeon. Once planning is complete the bone model is updated to indicate the bone regions requiring resection. This is illustrated by the bidirectional relation between the implant planning and bone model generation use cases shown below in Figure 4.4.

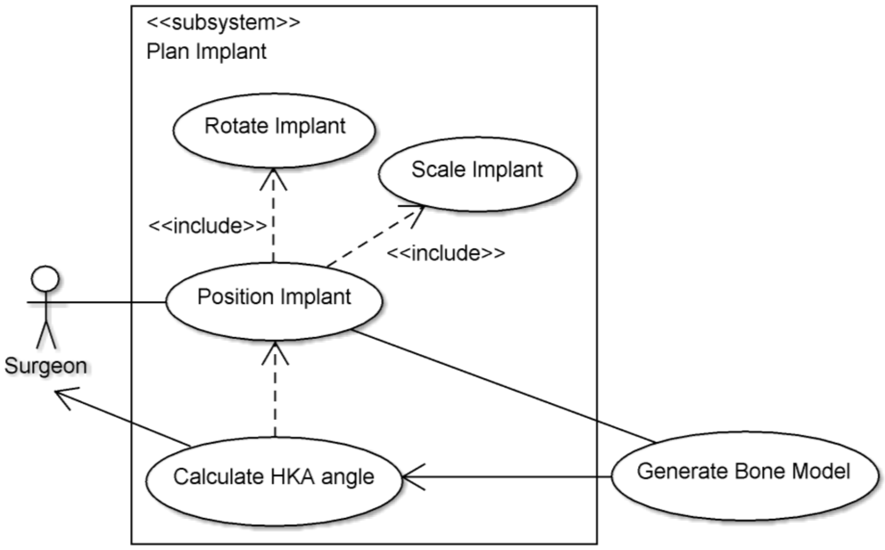


Figure 4.4: UML use case diagram of Plan Model use case.

The final resection stage, arguably the core system stage, utilises the implant plan to control resection. Resection is controlled to stay within the planned cutting region through semi-active constraint of a high-speed cutting burr used by the surgeon. This semi-active constraint is illustrated by the bidirectional relationship between surgeon and the resect use case in Figure 4.5. Again, as with bone model generation, the tool and patient are tracked throughout the process.

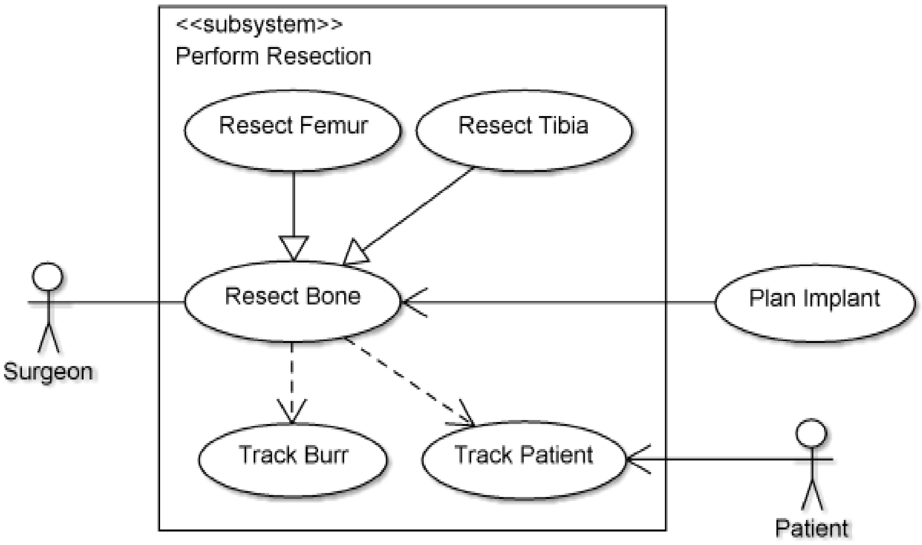


Figure 4.5: UML use case diagram of Perform Resection use case.

Figure 4.3, Figure 4.4, and Figure 4.5 describe the general operation of the system. Therefore, they form the basis of the problem analysis of the system discussed below.

Both Figure 4.3 and Figure 4.5 identify tracking as a core problem. As discussed in Chapter 2, there are many potential tracking technologies available. However, at the time of this research optical tracking methodologies presented the most viable solution and were defined as the suggested approach in the original project proposal. Optical tracking offers two distinct approaches, namely marker or markerless tracking. The problem of tool and patient tracking varies little between these two approaches. However, the requirements and implementations have major core difference. A problem hierarchy for tracking approaches is illustrated below in Figure 4.6. It is seen that the two approaches share much of the same hierarchy.

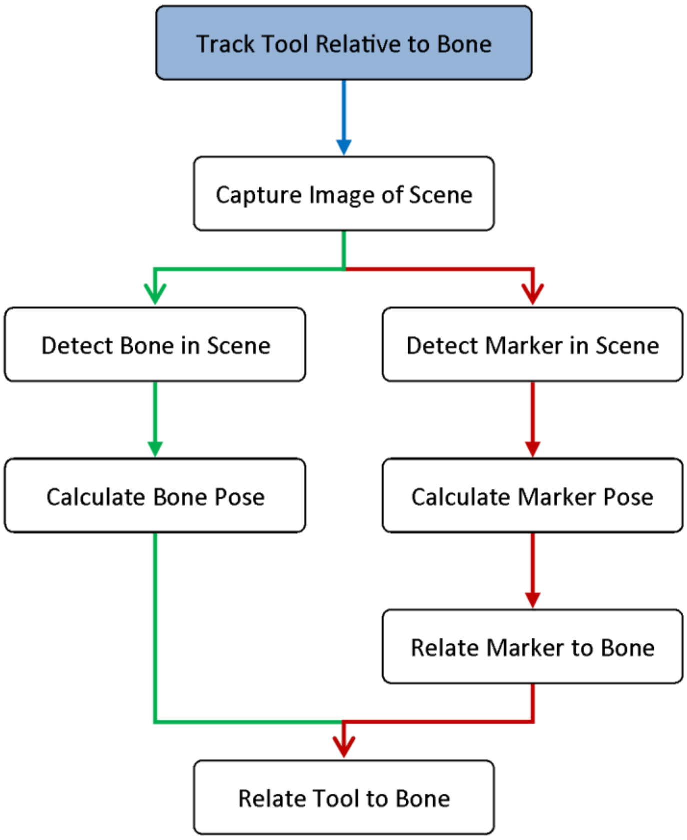


Figure 4.6: Core problems of marker and markerless tracking system.

Both systems require an image of the scene to be captured. The markerless approach, shown by green arrows, will then be required to detect the bone within the scene. As the surrounding soft tissue may move independently of the bone it would be necessary to track the bone itself. Once detected the pose of the bone, relative to the origin of the optical tracking system would be calculated. Finally, the spatial relationship between the tool and tracking system would be used to relate the position and orientation of the tool to the bone, thus allowing guided resection. Marker based tracking would follow a similar approach, deviating between image capture and tool relation. The system would be required to detect the marker within the captured image. Depending on the type of marker, the detection method and pre-detection filters would vary. Once detected, the pose of the marker would be calculated. This is similar to the approach taken in markerless tracking. However, an additional step is required to relate the pose of the marker to that of the actual bone. After this the tool may be related to the bone position, as with markerless tracking, allowing guided resection.

The problem hierarchy of Figure 4.6 will be expanded upon in Section 4.3 below, whereby specific system elements required to overcome the above problems are detailed. This is then further expanded upon in Chapters 6, 7, and 8 where the implementation specifics are described.

A second reoccurring element highlighted by Figures Figure 4.3 to Figure 4.5 is the bone model, which features in all three figures. With the imageless approach intended the model must be intra-operatively generated, used for planning, and used to guide resection. Again, as with tracking, there are many different approaches that may be taken. However, this problem analysis looks at the implementation independent specifics and attempts to generalise the problem. The following three figures illustrate the problem hierarchy associated with the bone model during the three core stages illustrated by Figures Figure 4.3, Figure 4.4, and Figure 4.5 respectively.

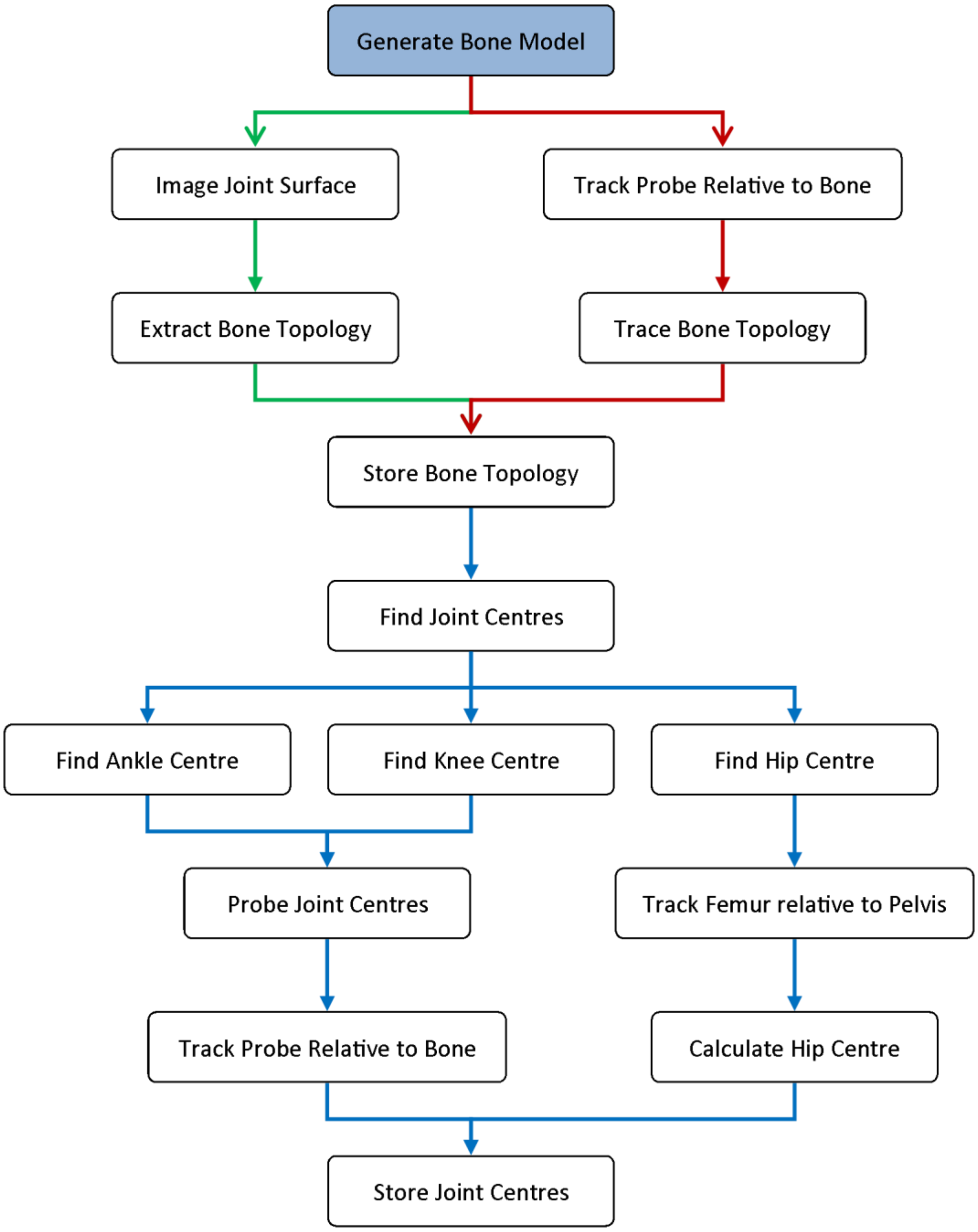


Figure 4.7: Problem hierarchy of bone model during generation

As with the tool tracking illustrated by Figure 4.6, the problem of bone model generation diverges due to markerless or marker based tracking. It is proposed that a markerless approach may directly capture the joint surface topology. The system would be required to image the joint surface and extract the surface topology to produce the bone model. Meanwhile, a marker approach would require a probe tool to be tracked, using the marker based approach defined by Figure 4.6. The probe would be traced across the bony surface to generate the joint topology. In addition to the joint surface topology, the bone model requires joint centres to allow the calculation of HKA angles. Both marker and markerless approaches would obtain these centres using a tracked probe. Ankle and knee centres may be directly probed. However, the hip centre is not easily probed. As such the hip centre is found by tracking the movement of the femur relative to the pelvis. With the joint centres stored the bone model is complete.

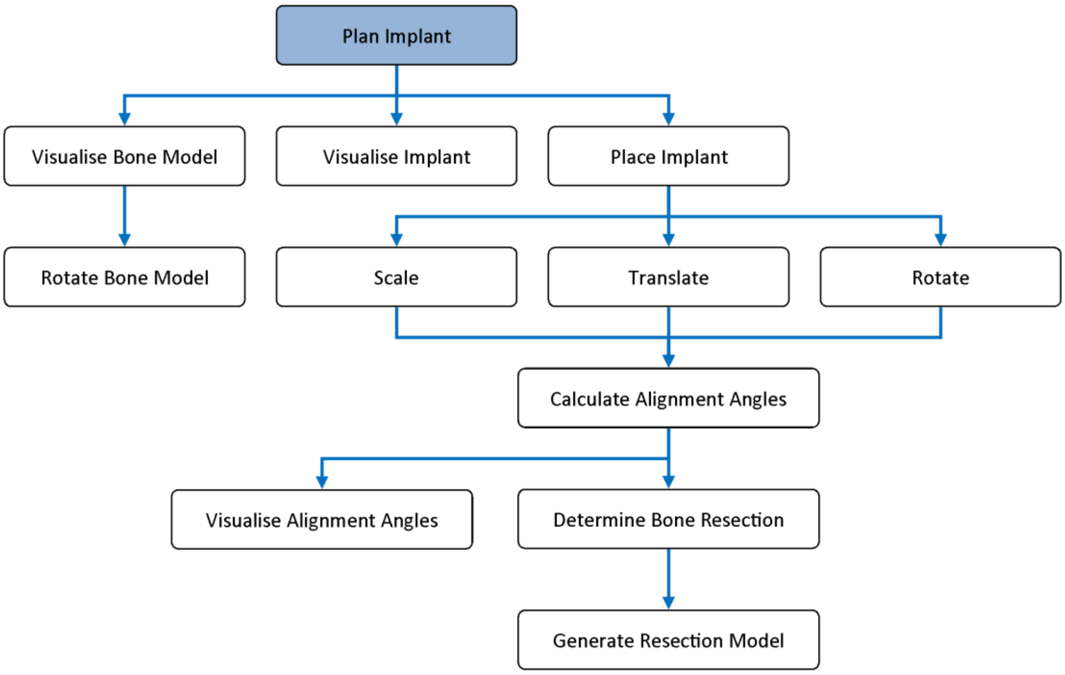


Figure 4.8: Problem hierarchy of bone model during planning

Figure 4.8 shows the bone model associated problem hierarchy during the implant planning stage. To enable the planning of the implant the bone model and implant components should be visualised. To place the implant the system should be able to scale (select size), translate, and rotate the implant, relative to the bone model in three dimensions. To guide planning the system shall calculate the joint alignment angles based upon the bone model and implant placement. These angles should be displayed for the user to act upon. Once the implant placement is finalised the bone model is adapted to indicate regions requiring resection.

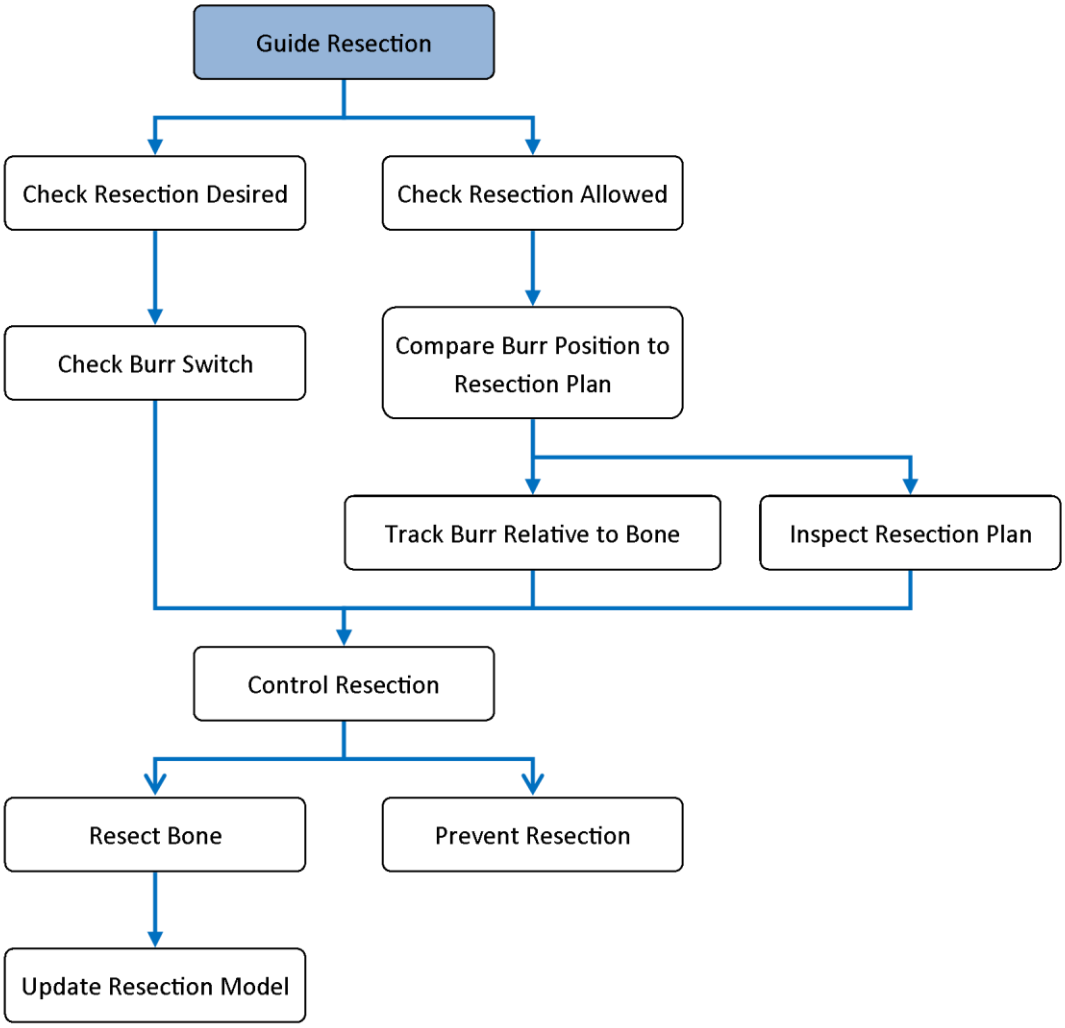


Figure 4.9: Problem hierarchy of bone model during resection.

Figure 4.9 shows the problem hierarchy for the final function of the bone model, resection. For the system to provide semi-active control of resection two conditions must be inspected. Firstly, the surgeon must want to resect, i.e. the burr trigger or foot pedal must be depressed. Secondly, the system must confirm resection is appropriate by comparison of the current burr position with the resection plan. The system may then decide to either allow resection, thus removing bone and updating the bone resection model to match, or prevent resection.

Figure 4.9 introduces a third important component to the system, previously implied by the other figures, *user interface* (UI). UI is required to allow the user to interact with the system. In the case of Figure 4.9, UI is required to determine if the surgeon wishes to active the cutting burr. In the previously discussed planning stage UI is required to allow the surgeon to translate, rotate, and scale the implant, or bone model. Finally for the model generation stage UI is required to instructed the system when to capture the joint topology. A range of UI options will be required throughout the system, for the specific functions discussed above, as well as more general actions, such as progressing from one procedure stage to the next. Figure 4.10 below illustrates the problem hierarchy for the UI of the proposed system. The hierarchy is considerably generalised, as UI is strongly implementation specific.

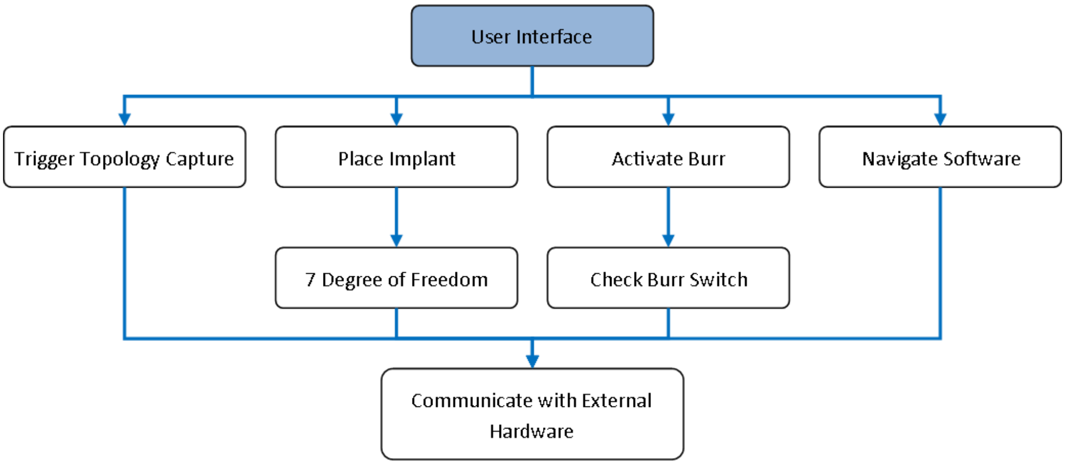


Figure 4.10: Problem hierarchy of generic user interface.

The problems illustrated above in Figure 4.10 all require communication with external hardware. This may be detecting key presses upon the keyboard, mouse clicks and movements used to interact with on screen buttons, or other external hardware such as joysticks or sensors.

UI, along with tracking and the various states of the bone model, encompass the core problems that any potential system must overcome. As such, each shall be further analysed in Section 4.3 below. The problems of each element highlighted by Figures Figure 4.6 through Figure 4.10 shall be expanded upon, detailing the components required to resolve each problem.

4.3 Requirements

The problems presented in Section 4.2 above shall be expanded to produce a set of system requirements. Although more detailed than the problem analysis this section also aims to be implementation independent. A combination of UML activity and conceptual class diagrams shall be used to illustrate the information flow and structure of the potential system, respectively. Activity diagrams expand upon the problem hierarchies used above in that they include decision junctions, whereby information flow between actions is diverted. This allows the expression of greater detail used to define the requirements of a system. Conceptual class diagrams provide a primitive structure of the system. These help define required code elements and are utilised in later stages of development to design code layout and interactions.

Firstly the problem hierarchy of tracking, illustrated in Figure 4.6 above, is expanded upon. Capture image of scene was defined as the first tracking sub-problem. An image is typically captured from a video stream which is a continuous connection with a camera sending sequential frames. This stream must be enabled and typically the camera settings, such as exposure time, require initialisation. Once the stream is established, image frames may be captured. This is illustrated by the activity diagram below.

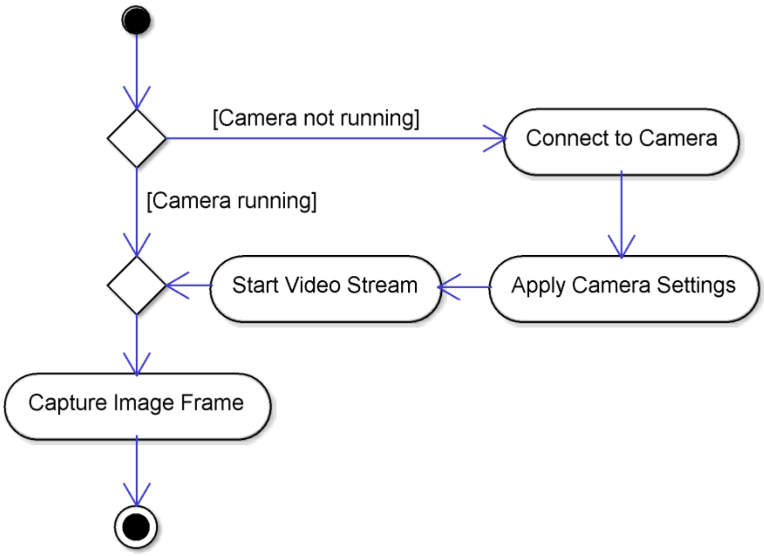


Figure 4.11: Activity diagram of image capture

After image capture Figure 4.6 describes the problems of determining the tool pose relative to the bone surface, for both marker and markerless systems. Figure 4.12 below illustrates the system requirements to overcome these problems.

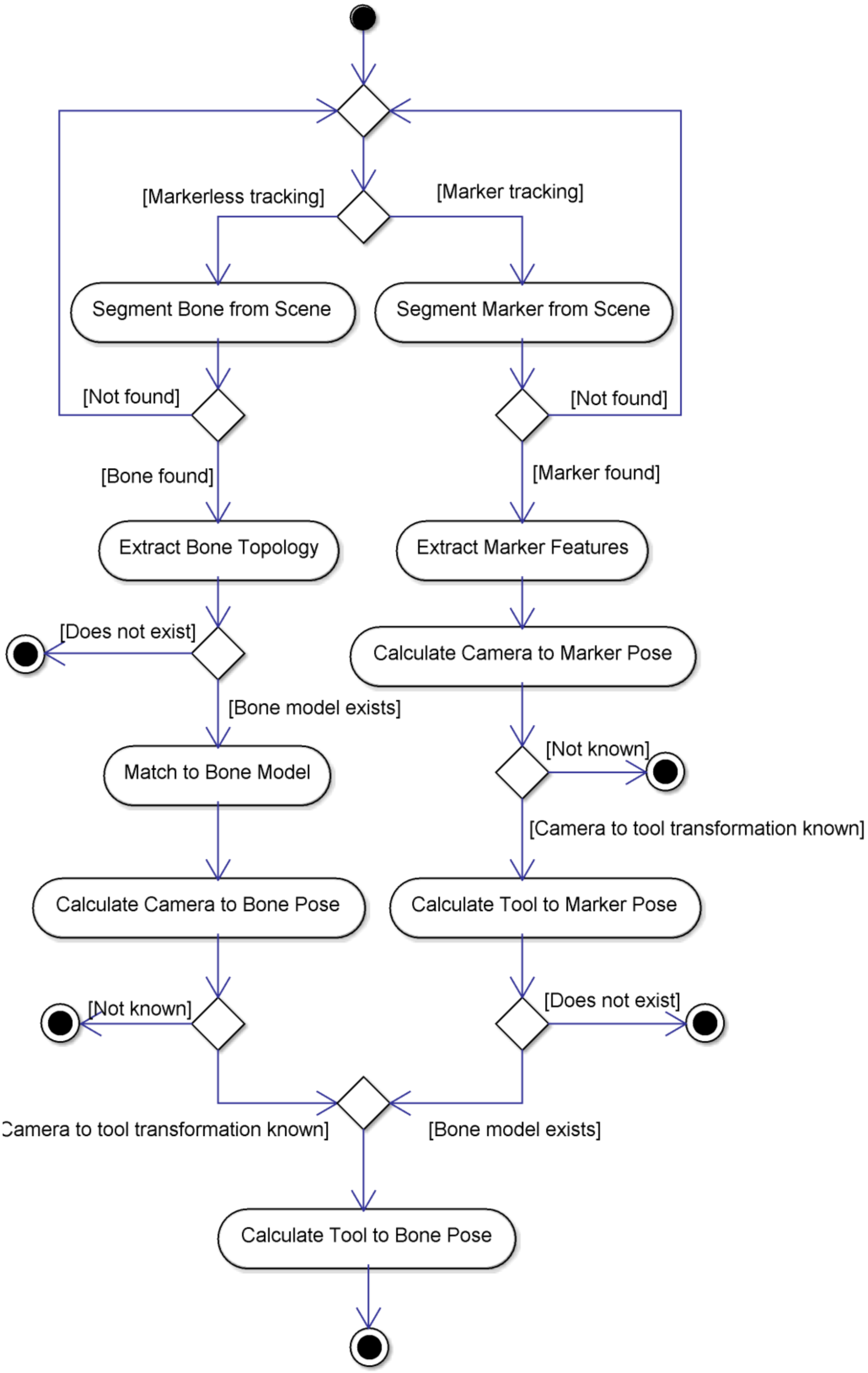


Figure 4.12: Activity diagram of tool tracking

Both methods will require some form of image segmentation to separate the object of interest from the imaged scene. For markerless tracking if the bone is detected its topology will be captured. If the bone model has previously been generated this new topology may be matched against it. This will determine the transformation between the camera and the bone. If the transformation between the camera and the tool is known it will then be possible to calculate the tool pose relative to the bone. The marker based approach requires a similar process. If a marker is detected in the scene its features are extracted and its pose, relative to the camera, may be calculated. If the camera to tool transformation is known the marker to camera pose may be translated to provide the pose of the tool relative to the marker. Finally, if a bone model exists, the tool pose may be translated relative to the bone.

The tool tracking described above is used by both marker and markerless systems during the intra-operative generation of the bone model. The bone model generation problem hierarchy of Figure 4.7 is divided for requirement analysis. Firstly, the requirements of joint topology generation are illustrated in Figure 4.13. The process of locating the three joint centres is then described by Figure 4.14.

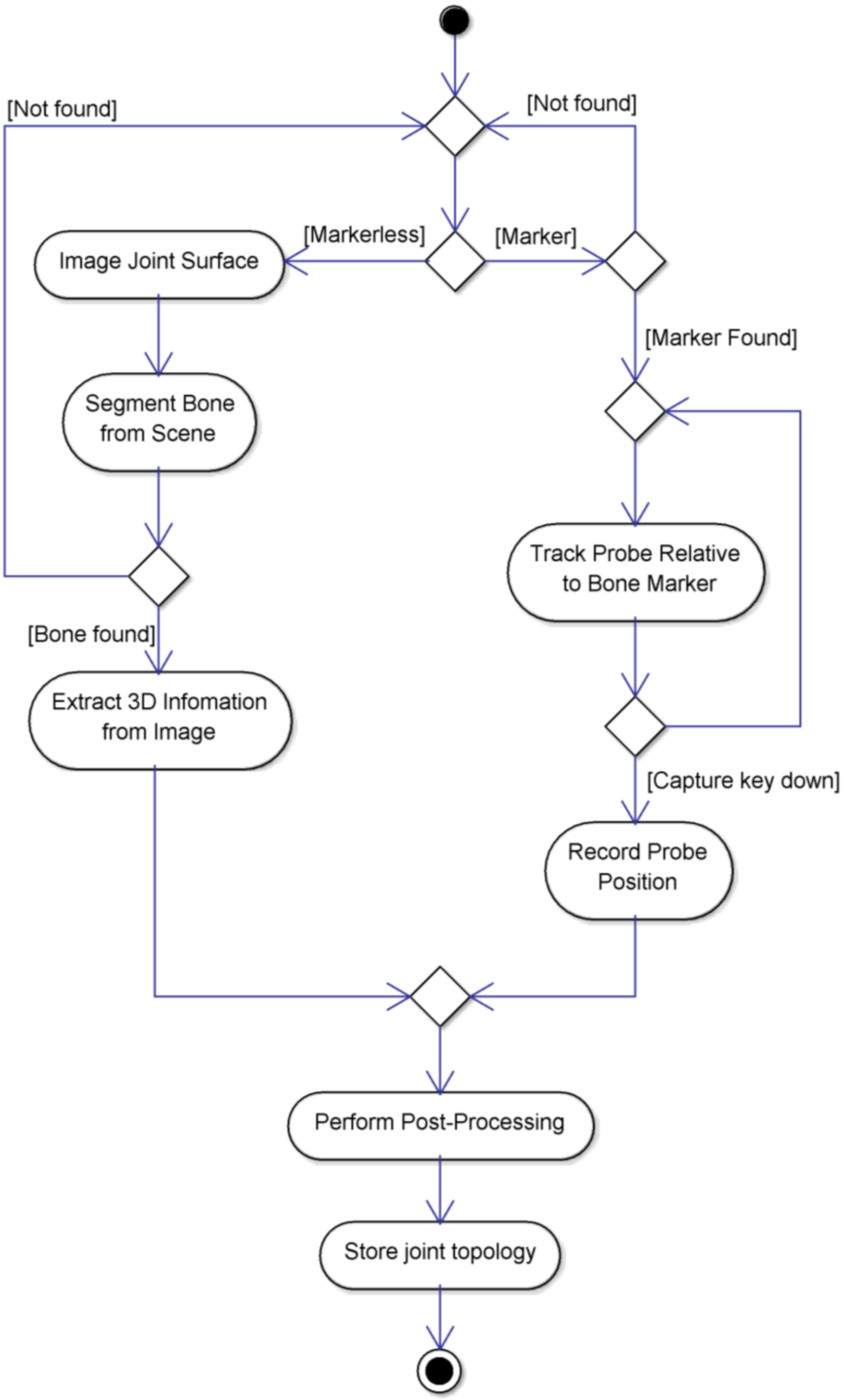


Figure 4.13: Activity diagram of joint topology generation

Looking first at the markerless approach, it may be seen from Figure 4.13 that the system would be required to image the joint surface and segment the bone from the image as required by tool tracking, seen in Figure 4.12 above. If the bone was successfully detected the system would be required to extract the 3D topology of the surface from the acquired image. Chapter 5 discusses two potential implementations of this topology extraction.

The marker approach requires the tool to be tracked relative to the bone marker, as described by Figure 4.12. To ensure the system records only the bone topology some form of user interface will be required to trigger the recording of the probe position.

Once either the markerless or marker approaches have captured the joint topology it is likely some post-processing will be required. For example, the surface may be smoothed and errors removed. Finally, the surface must be stored. As the surface requires rendering during the planning stage and editing during resection, the manner in which the surface is stored must be compatible with the requirements of these actions. This compatibility requirement is discussed in Appendix A1.2, where the implementation of the surface generation of the first generation system is discussed, and again in Appendix A2.5 for the second generation system.

The requirements of the second stage of bone model generation, the locating of joint centres, are illustrated below in Figure 4.14.

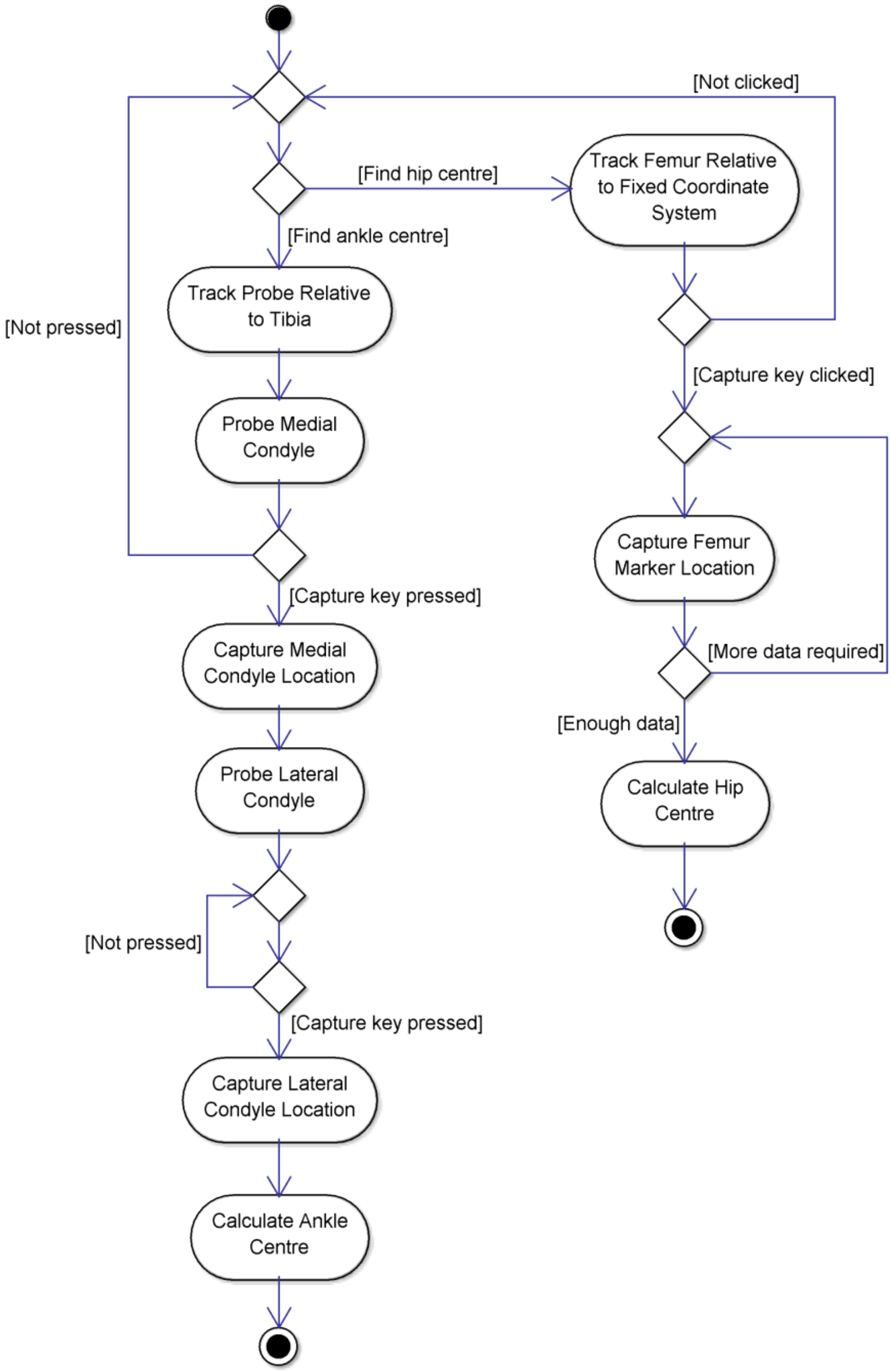


Figure 4.14: Activity diagram of the second stage of bone model generation – joint centres location

As discussed previously, Figure 4.14 illustrates that two different approaches will be required to capture the ankle and hip centres. For the ankle centres it will be required to track the probe, as described by Figure 4.12. As it is not possible to directly probe the centre of the ankle joint the system will be required to probe both the medial and lateral malleoli of the ankle. The system may then define the ankle joint centre by taking the mean of these two locations or alternative standard anatomical definition. As with surface generation UI is required to trigger the system when to capture the probe location.

As mentioned in Section 4.2 above, the hip centre may not be directly probe and is instead found by resolving the centre of rotation of the femur about the hip. To enable this Figure 4.14 identifies the requirement of a fixed coordinate frame relative to which the motion of the femur may be captured. The coordinate frame is required to be fixed relative to the pelvis. The system may be triggered to capture the femur location, as with surface generation and ankle joint location, via a key pressed. However, as a continuous series of measurements are required it may be more desirable to capture multiple measurements after a single key press. This begins to expand the requirements of the UI system. It is seen that it will be desirable for the UI system to differentiate between a key being down, being clicked, or being released. Once sufficient data have been collected the system will be required to calculate the hip centre. The mechanism through which the hip centre is derived from these data is discussed in implementation specific terms in Appendix A1.3.

Figure 4.8 illustrated the problems faced during implant planning. It is noted that this is an extremely limited planning process. A full planning suit would ideally include many more factors. The contact points and pressures of contact would be estimated and the effect of soft tissue and several other elements would be included in the calculation of the estimated results. However, the scope of this project was limited and as such the planning suite consisted of only the fundamental elements required to place an implant.

The requirement analysis of Figure 4.8 is divided into two sections. Firstly, Figure 4.15 depicts the requirements of both bone model and implant rendering. It is seen that these two tasks are very similar, at least in non-implementation specific terms. Figure 4.16 then illustrates the positioning of the implanted and the limited feedback provided to guide the implantation positioning.

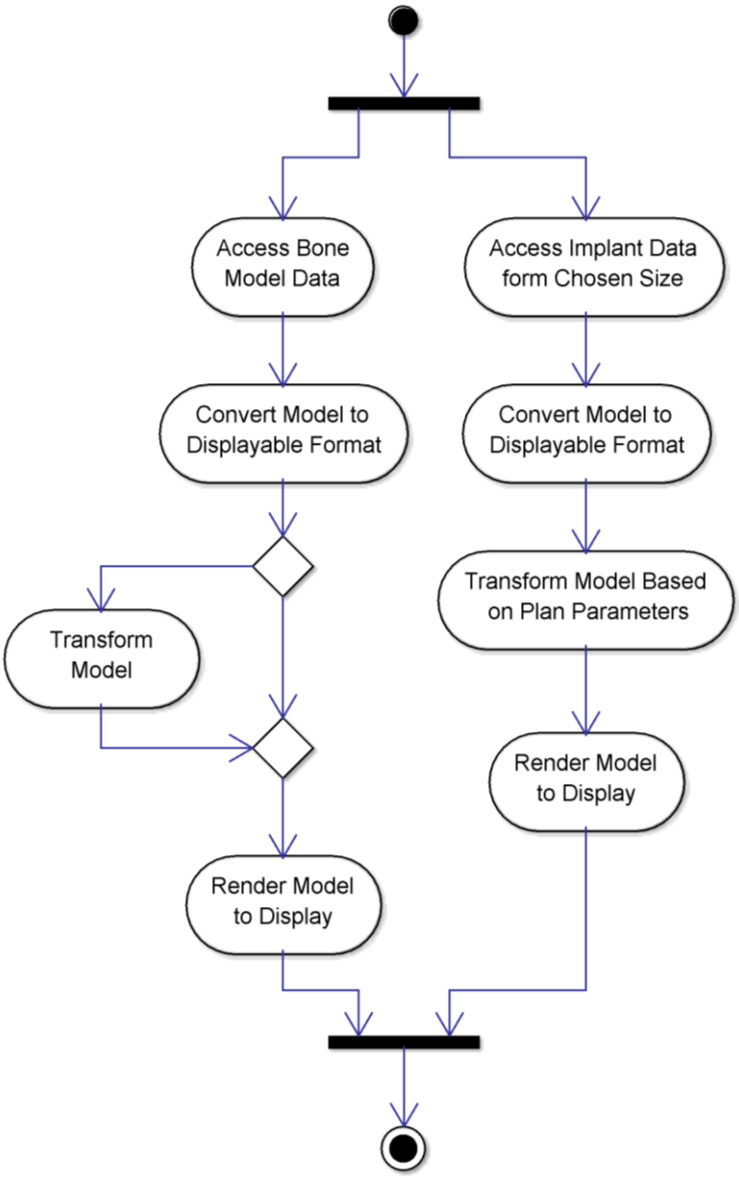

Access Bone Model Data
Access Implant Data form Chosen Size
Convert Model to Displayable Format
Convert Model to Displayable Format
Transform Model
Transform Model Based on Plan Parameters
Render Model to Display
Render Model to Display

Figure 4.15: Activity diagram of model rendering processes of planning stage.

It may be seen that both bone model and implant model display present the same requirements. Both models, which may be stored in different formats, are loaded and potentially may require conversion to a different format more suited to display. The models are then transformed, both translated and rotated, based upon parameters set by the user. Again these setting indicate a UI requirement for either key or mouse interface. The models are then rendered in their final pose. This process is highly implementation-specific, based upon the model formats and rendering API, therefore it shall be discussed in more detail in the relevant sections of Chapters 6, 7, and 8.

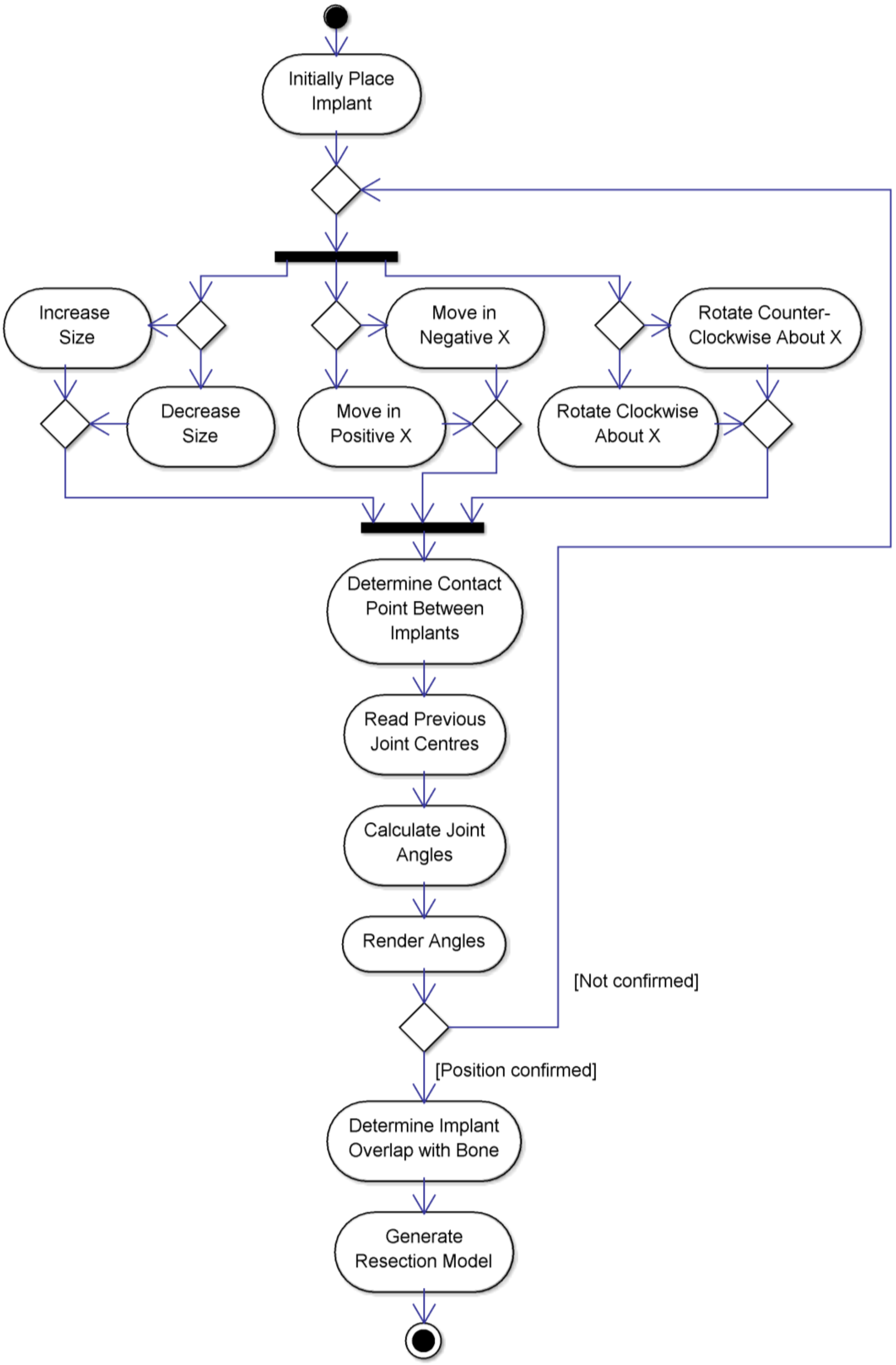


Figure 4.16: Activity diagram of implant positioning processes of implant planning stage.

Figure 4.16 above has been simplified for the sake of clarity. After the system initially places the implant the user is able to: change the size of the implant, move the implant in three dimensions, and rotate the implant in three dimensions. This requires the system to facilitate seven DOF, providing the required UI for each action. The figure only depicts three DOF for a single implant. Each time either of the two implants are moved or sized the system must recalculate the alignment angles of the limb. Although again highly implementation specific, this process will require the three broad stages listed in Figure 4.16. For the simple system proposed, that does not include any advance kinematics, the alignment of the limbs is governed by the contact of the two implant components. The positions of the components are known relative to the bones of the limb. It will then be presumed that the implants are in contact with each other. The previously found hip, knee, and ankle centres are then used to calculate the HKA angle based upon this contact. The system will display the estimated post-operative HKA angle to the user, who may either confirm the implant or continue positioning. Once confirmed the system must generate the resection model. The chosen implant position will be compared to the previously generated bone models. The regions of the bone models that are overlapped by the implants will be marked for resection. As discussed previously, the format of the bone and resection models is implementation specific and as such shall be discussed in the appropriate section.

After planning has been finalised and the resection model generated the system may now enter its core function – guided resection. The problem analysis of Figure 4.9 is expanded upon, emphasising the required decisional structure of the system, in Figure 4.17 below.

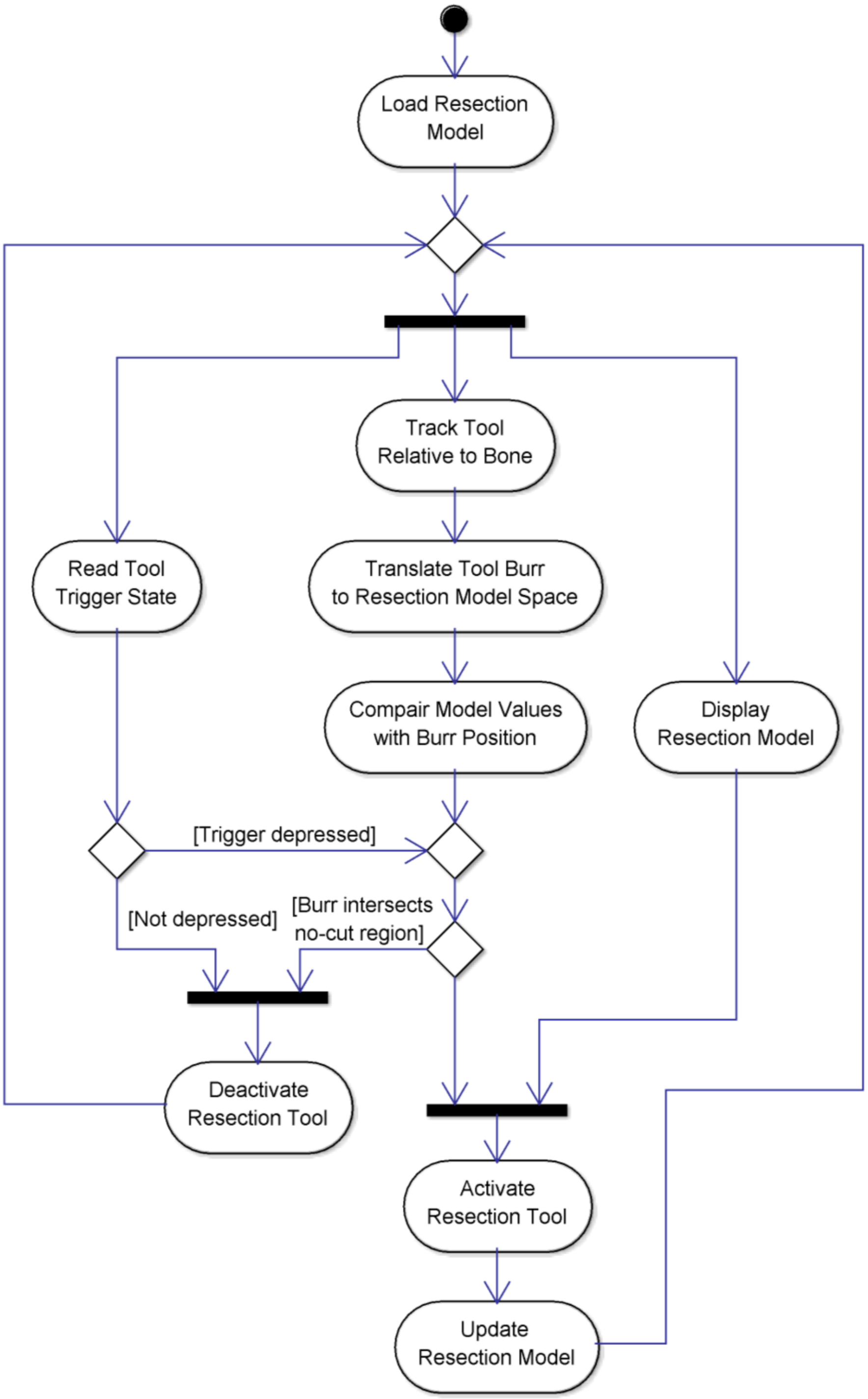

Load Resection Model
Track Tool Relative to Bone
Read Tool Trigger State
Translate Tool Burr to Resection Model Space
Compair Model Values with Burr Position
Display Resection Model
[Trigger depressed]
[Not depressed]
[Burr intersects no-cut region]
Deactivate Resection Tool
Activate Resection Tool
Update Resection Model

Figure 4.17: Activity diagram of guided resection

To passively guide the user through resection the system is required to load and display the resection model as described by Figure 4.15. In parallel with display, as indicated by the fork node in Figure 4.17 above, the system must provide active guidance through the control of semi-active constraint. Whether the system should resect the current section of bone is decided by two factors. Firstly, the trigger of the resection tool is polled, a process discussed in more detail below. If the trigger is not depressed, this indicates the user does not wish to resect. Therefore, the system immediately deactivates the resection tool. However, if the trigger is depressed the system must judge the suitability to resect based upon the resection model. To do this the system is required to track the tool relative to the physical bone, a process covered above by Figure 4.12. The position of the tool must then be translated into the space of the resection model, if it differs from the bone space. The position of the resection tool may then be compared to the values of the resection model. If it is found that the burr will intersect with no-cut regions within the bone model, the system will deactivate the resection tool. If the burr is found to intersect only allowed regions, the system will activate the tool, allowing resection. This combination of checks is required to ensure that bone is only resected when both the user and system agree. If resection occurs it will be necessary to update the resection model to include these changes and provide an updated display to continue guiding the user. This process of guided resection is continued until the user indicates that resection is complete.

As depicted by Figure 4.10 during the problem analysis UI is fundamental to the system and used throughout. The system is required to provide both standard interface, via keyboard and mouse, as well as via additional external hardware, such as switches and motors. Due to the generic nature of this requirement it shall not be discussed in detail.

The major elements of the proposed system and their requirements have now been discussed to the limits of this implementation independent section. As explained previously, the details of the features discussed above will be expanded upon in Chapters 6, 7, and 8. The requirements highlighted by this section are summarised by Table 4.1 below. As opposed to grouping the requirements by procedure stage, as done above, they have been regrouped into four elements: tracking, bone model, resection tool, and UI.

Group	Requirement	Description	Figure
Tracking	Stream images	Connect, configure, and stream images from the physical camera in real-time	Figure 4.11
	Detect Markers or Bones	Locate and identify markers or bones from video image	Figure 4.12
	Calculate Pose	Estimate the pose of the markers or bones to 6DOF	Figure 4.12
Bone Model	Capture Geometry of Bone	Capture and store the geometry of the bone via probing or direct methods	Figure 4.13
	Visualise	Be displayed during several procedural stages	Figure 4.15
	Resection Plan	Used during planning stages and updated to indicate regions to be resected	Figure 4.16
	Resectable	Updated in real-time to match resection of physical bone	Figure 4.17
Resection Tool	Tracked	Tracked relative to the bone allowing correlation of plan and reality	Figure 4.12
	Calibrated	Determine offset between tracking system and tool tip	Figure 4.12
User Interface	External UI	Facilitate input and output external interface	
	Hard	Facilitate keyboard and mouse physical interface	
	Soft	Provide on screen buttons for virtual interface	

Table 4.1: System requirement summary

5

Marker-Free Tracking

5.1 Introduction

As explained in Chapter 3, the initial aim of this research was to develop an alternative marker-free tracking system for use in guided orthopaedic procedures. This would, therefore, eliminate the need for the bone screws used by current systems to attach the markers and reduce the infection and fracture risk associated with these (Vetter et al. 2014; Wysocki et al. 2008). This chapter documents the preliminary research into markerless systems, the tracking approaches investigated and the conclusions made.

5.2 Stereoscopic Matching

5.2.1 Introduction

As described in Section 2.4.2.1.5 of the literature review, stereoscopic matching takes two images of the same scene and then attempts to match image features between the two images, so that it may apply stereo-projection and determine the 3D position of the features.

It was theorised that this approach could be applied to the UKA procedure to provide direct tracking of the limb bones, eliminating the need for markers. As described in Chapter 4, stereoscopic matching would be used to capture the topology of the articulating surface of the joint. This topology would then form the basis of the bone model, upon which the implant procedure would be planned. Then, during the procedure, the system would continually capture the joint surface which would be matched to the original model to determine the position of the resection tool. This would allow computer assistance without the need for markers or bone to marker registration.

5.2.2 Materials and Methods

5.2.2.1 Camera Selection

The body of this thesis focuses on optical based systems, the majority of which utilise standard full colour cameras. Although several different approaches are taken throughout this thesis they share similar requirements as listed below in Table 5.1.

Requirement	Reason
High image resolution	Improve tracking accuracy
Lossless encoding	Preserve image detail
Real-time frame rate	Allow real-time tracking
Low image distortion	Improve tracking accuracy
Wide field of view	Increase tracking volume
Small and light weight	Improve usability

Table 5.1: Camera requirements

Digital cameras quantize the image, in the case of digital cameras into pixels. This reduces the amount of positional information which may be extracted from the image. Figure 5.1 illustrates the effect of image resolution on the spatial information available in an image.

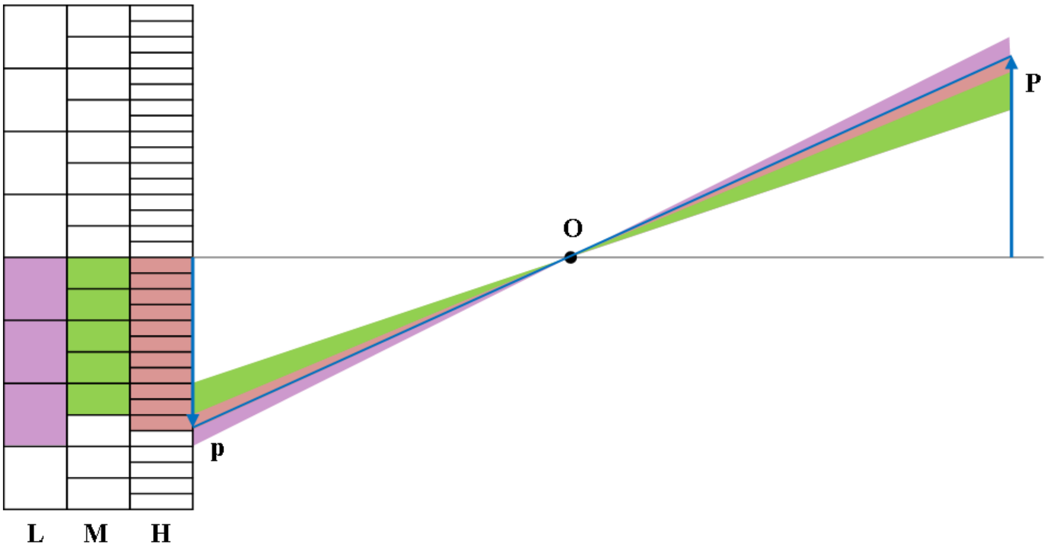


Figure 5.1: Effect of image resolution on spatial resolution

As may be seen from Figure 5.1 each of the three different camera resolutions, represented by the low, medium, and high grids, produces a different image. The coloured regions indicate where the physical object may be to produce each image. As with all signal sampling, the higher the sampling frequency, in this case the higher the resolution, the more accurately the true signal is captured. Therefore, a camera with as high of a resolution as possible was desired.

To ensure the image detail is preserved it is important that the camera provides lossless encoding. Encoding converts the raw pixel values from the image sensor into an image format that may be interpreted by a computer system. Encoding formats are designed for different applications. Some are designed to optimise the rate of video transfer, and as such these often compress the video, reducing the image quality and detail. Other encoders, described as lossless, are designed to maintain the maximum amount of image information and quality. A camera system that provided lossless encoding was required.

The effect of framerate is similar to that of image resolution, only now in the temporal domain. The higher the framerate of the camera the more accurately it may capture the scene. For example if a camera with a framerate of only 10 *frames per second* (fps) were to track an object moving at 1 ms^{-1} that object could move 100 mm between frames. If this system was to be used for semi-active constraint in orthopaedics, the surgeon could perform a considerable amount of erroneous resection between camera samples. Therefore, a high framerate was desired.

Real camera systems induce distortion in the image. This may be compensated for, to some extent, through camera calibration. However, this results in the loss of data, as pixels are stretched or compressed or deleted entirely during the de-distortion process. Furthermore, image distortion correction is typically characterised as radial and tangential. Therefore, distortion that does not fit these models may not be correctly compensated for. Therefore, a camera system with minimal distortion was required.

Chapter 3 defined the initial research aim as developing a tool mounted tracking system. This would considerably reduce the separation between the camera and tracked objects. To compensate for this, and allow the system to maintain a relatively large tracking volume, a large *field of view* (FOV) is required. Additionally, a practical tool mounted system is required to be lightweight and small.

Two further aspects considered during camera selection, but not listed, were cost and ease of connectivity. The system was to be run upon a Toshiba Tecra R850-119 laptop computer. This system provided only a single USB 3.0 and two USB 2.0 interface ports. The system also did not feature a PCIe expansion slot, therefore additional interface ports could not be appended. As the original stereo matching system required two cameras connected simultaneously USB 2.0 was the only viable connection option.

Considering the criterial listed above two USB 2.0 web cameras were proposed, the Microsoft LifeCam Studio (Microsoft Corp., WA, USA) and Logitech HD Pro C920 (Logitech, CA, USA). These are compared below in Table 5.2.

Requirement	Microsoft LifeCam Studio	Logitech HD Pro C920
Maximum resolution (pixels)	1920x1080	1920x1080
Maximum framerate (fps)	30	30
Maximum real-time resolution (pixels)	1280x720	1280x720
Lossless encoding (bits per pixel)	RGB(24), YUY2(16)	RGB(24), I420(12)
Field of vie (diagonal °)	75	83
Image distortion	Low	Low
Dimensions (mm)	30x40x60	92x30x25
Weight (g)	128	162
Connection type	USB 2.0	USB 2.0
Cost (£)	89.99	84.99

Table 5.2: Candidate camera comparison

It may be seen from Table 5.2 above that both cameras meet the requirements well, offering very similar specifications. Ideally, higher framerates and resolutions would be preferred. However, USB 2.0 provides a maximum data transfer rate of 480 Mbps. A full 1920x1080p HD image contains 2.07 MP. Using the smallest I420 format provided by the C920 web camera each pixel produces 12 bits of data. Therefore, a full HD images requires approximately 25 Mb. At the full transfer rate supported by USB 2.0 this would allow a maximum framerate of 19.2 fps. In reality this rate would most likely not be achieved due to the USB not performing optimally. It was found that a 1280x720p image with 16 bit encoding streamed at a rate of approximately 25 fps, despite the theoretical rate of 32.5 fps.

Due to the similarities in the camera specifications, selection was based predominately upon the dimensions of the camera. Figure 5.2 below shows the two candidate camera systems.

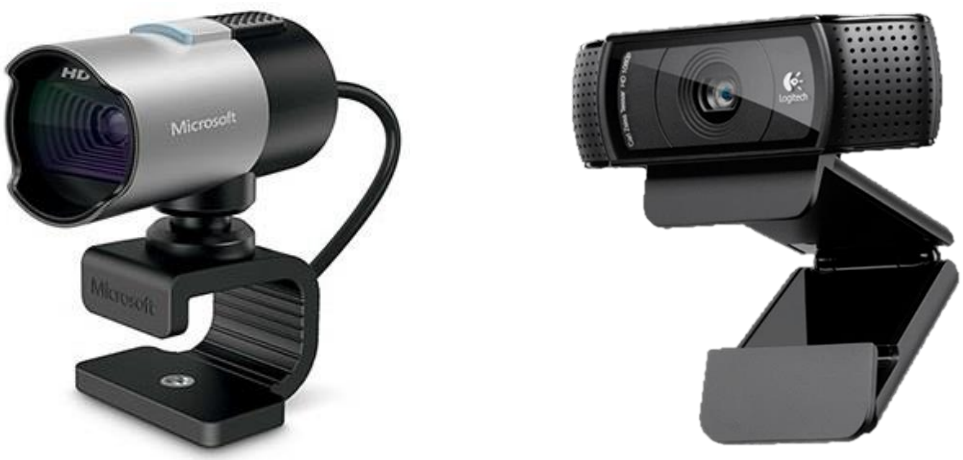


Figure 5.2: Final candidate camera options. Left: Microsoft LifeCam Studio, Right: Logitech C920.

Figure 5.2 shows the LifeCam to be long and narrow, while the C920 is short and wide. The stereoscopic tool mounted design investigated required two cameras to be mounted side-by-side upon a resection tool. The proposed resection tool was an Anspach eMax 2 Plus (Synthes, Inc., PA, USA), pictured below in Figure 5.3.

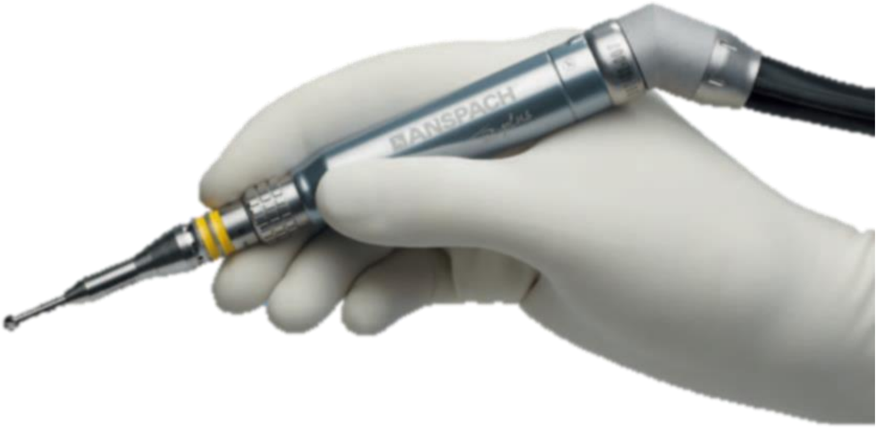


Figure 5.3: Anspach eMax 2 Plus high-speed resection tool.

The Anspach is seen to have a long, narrow, cylindrical design. As such the narrow structure of the LifeCam would better suite stereo attachment to the Anspach tool.

5.2.2.2 Experimental Procedure

In accordance with the tool mounted design a working distance, defined as the average separation between camera and imaged object, of 150 mm was used. To ensure a good image overlap of approximately 4/5th the two LifeCam cameras were placed in parallel with a baseline of 35 mm. The optical axes of the cameras were aligned parallel to preserve image quality as misaligned cameras, such as those arranged in a toe-in configuration, require greater transformation during image processing. This setup is shown for the imaging of a synthetic tibial saw bone in Figure 5.4 below.

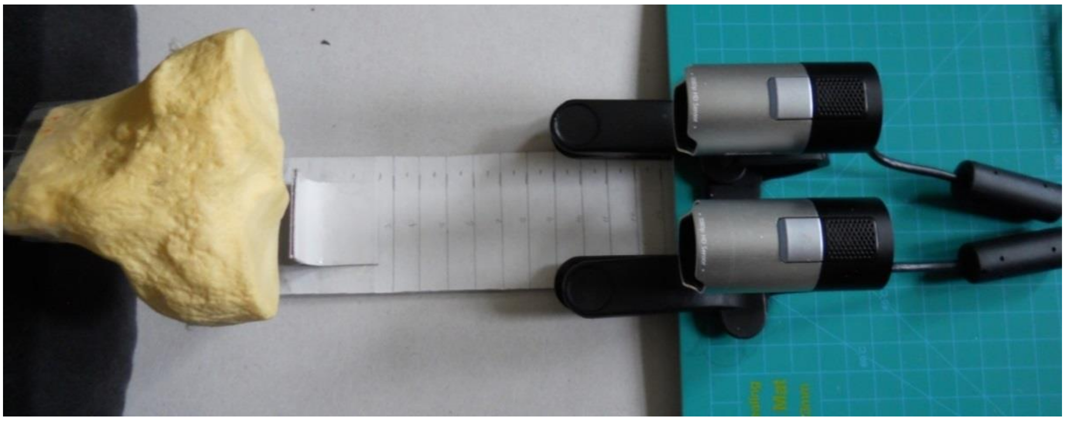


Figure 5.4: Experimental setup for stereo matching imaging of tibial saw bone.

As described in sections 2.4.2.1.1 and 2.4.2.1.5, stereoscopic matching pairs image features between two images to determine the topology of the scene. If this matching were performed directly the system would be required to search large regions of the second image for each feature found in the first image. This would be extremely time consuming and would likely produce many false positives. To improve matching efficiency the two images are first rectified. Image rectification reduces the matching problem from a 2D search to a 1D search. The images are manipulated in such a way that corresponding pixels are within the same image row. This is the ideal case for optimally aligned cameras. However, in reality it is extremely difficult to perfectly align the two cameras to ensure the image pairs show only horizontal displacement. Therefore, the camera systems must be calibrated to produce lookup tables which map each pixel to an idealised location.

A camera calibration program was developed in Matlab. The program connected to both cameras and streamed image frames using the YUY2 16 bit video format. An image was captured from each camera and converted to grey scale to improve matching speed, without causing a notable increase in false matches. *Points of interest* (POI) were extracted from each grey scale image independently.

To extract these POI the Matlab SURF (speeded up robust features) algorithm was used (Bay et al. 2008). SURF algorithms perform two important roles, POI detection and description. To detect features SURF uses a blob detector based on a simplistic Hessian-matrix approximation. This approach is used as it well facilitates one of the key optimisations of the SURF algorithm, the use of box filters and integral images. Previous feature detectors tended to use Gaussian filters to scale the image multiple times to ensure robust feature detection. However, this requires applying a relatively computationally costly Gaussian filter to the image, then applying the filter again to the resulting image, which is again filtered. The SURF algorithm approximates the Gaussian filter with a simple box filter. This is immediately less computationally intensive. However, by increasing the size of the box filter different scales may be achieved directly from the original image, offering the possibility of parallelisation. The efficiency of the box filter is further improved through the use of integral images. The value of each pixel of an integral image is equal to the sum of all pixels within the rectangle formed between that pixel and the image origin of the original image, as expressed by Equation 5.1.

$$I_{\Sigma}(x,y) = \sum_{i=0}^{i\leq x} \sum_{j=0}^{j\leq y} I(i,j) \quad (5.1)$$

Box filters require the sum intensities of rectangular regions of the image. Without integral images these values must be found by summating each pixel within the region. However, with integral images the value may be found for any region with only three additions. Therefore, the box filter used by the Hessian matrix to characterise points may be performed with minimal computational cost. The standard Hessian matrix requires three Gaussian second order derivatives as different scales that are each convoluted with the image at each point. The determinate of these results produce the blob response used to characterise features. The SURF algorithm approximates the Gaussian functions with three box filters in the x, y, and xy direction, D_{xx}, D_{yy}, and D_{xy} respectivly. Therefore the approximated Hessian determinate is given by Equation 5.2, where ω applies weighting.

$$\det(\mathcal{H}_{approx}) = D_{xx}D_{yy} - (\omega D_{xy})^2 \quad (5.2)$$

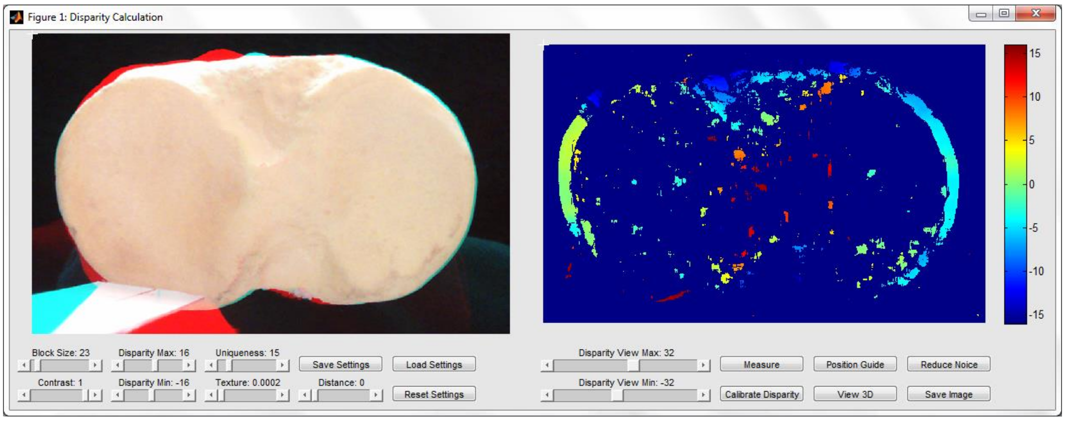
A range of filter sizes are applied to approximate the range of scales traditionally achieved using iterative Gaussian filtering and sub-sampling. Different scales improve the robustness of the feature description, which is particularly important when attempting the match features imaged at different scales. The maxima Hessian determinates are interpolated, both in space and filter scale.

It was found that using a metric threshold value of 800 with 3 octaves for the detectSURFFeatures function produced the most true-positive matches. The SAD matching metric was used over SSD as it was found to have no notable effect on the quality of results, but completed more quickly due to reduced computational complexity (Bull 2014). A matching threshold of five was found to remove the most errors while maintaining a good amount of matches. The GeometricTransformEstimator from the vision library of Matlab was used to remove significant outlying matches. A pixel distance threshold of 50 was found to remove the most significant outliers.

The inlying matches were then used to estimate the fundamental matrix and determine which matches concur with the epipolar geometry. Further outliers were removed using the RANSAC (RANdom SAmple Concensus) method with a threshold of 0.1 and confidence of 99.9 (Fischler & Bolles 1981). The number of trials performed was varied between 500 and 10,000, with 2000 found to produce a good speed/accuracy balance. However, as calibration was performed offline and therefore was less time critical a value of 10,000 was used to ensure the most accurate results. The resulting fundamental matrix was checked to ensure it followed epipolar constraints. This, together with the inlying features, was used to estimate the image rectification transformations. The estimateUncalibratedRectification function was used as it avoided the need to first determine the intrinsic and extrinsic parameters of the two cameras. The resulting transformation matrices were saved for use by the actual imaging program.

Before closing, the calculated rectification transformations were applied to the original images and the resulting rectified images displayed as a sanity check. This process was useful as it was found occasionally that a calibration could process smoothly through all stages only to produce a nonsensical rectified image. The rectified images were also overlaid as a red-cyan anaglyph image to view the quality of depth of the rectified pair.

With the camera system calibrated the system was then used to determine the disparity image of the bony surface. As the algorithms used to determine disparity contained several parameters a GUI was developed to simplify adjusting of the system, as shown in Figure 5.5.



Block Size: 23
Disparity Max: 16
Uniqueness: 15
Contrast: 1
Disparity Min: -16
Texture: 0.0002
Distance: 0
Disparity View Max: 32
Disparity View Min: -32

Figure 5.5: Disparity system GUI

Upon starting the program both camera streams were started at a resolution of 1920x1080. A GUI allowed exposure, focus and white balance to be set for each camera independently. These values were set to maximise image quality based upon current light conditions. Once configured the camera setting GUI was closed and the user was presented with the GUI shown in Figure 5.5 above. A capture button, not shown in the figure above, captured an image from both camera streams and disconnected the cameras. The disparity image was then calculated using the parameters set by the GUI.

The captured images were converted to grey scale and transformed using the previously calibrated rectification matrix. The resulting rectified images were cropped such that both images contain the same section of the scene, i.e. the left part of the left camera image not visible in the right image was removed, and vice versa. The images along with the parameters set by the GUI were passed to the disparity function.

The disparity function offered two SAD based methods to perform disparity estimation, *block matching* (BM) and *semi-global block matching* (SGM). Theoretically, SGM should provide a more complete solution by considering small disparity steps to likely be part of a slanted surface as opposed to discontinuities (Hirschmuller 2005). However, it was found that SGM did not notably improve results over BM but did increase the computational cost. Therefore, BM was used throughout. The BM algorithm uses SAD to match blocks between images. A block is produced about each pixel in the left image. The block is centred about this pixel with square sides of an odd number of pixels, defined by the block size parameter. As discussed previously, without rectification the system would have to perform an exhaustive search across all blocks of the right image. However, rectification ensures that corresponding pixels are found within the same row of the two images, greatly reducing the computational complexity of the search. This search was further reduced by applying a disparity range. This indicates to the algorithm that all imaged objects are within a certain range of distances, and therefore should be within a fixed range of disparities. Therefore, it is only necessary to search a limited number of blocks along a row. The block size defines the amount of pixels used to characterise a feature. Increasing the size increased the quality of the match. However, non-erroneous image differences, such as lighting, are more likely to cause the match to fail. The contrast and uniqueness thresholds define the quality of match required to be accepted as a positive match. The most likely disparity value is determined from the match that produces the lowest SAD value. This SAD value is compared with the other values produced for that pixel. If it is not found to be significantly smaller the match is considered uncertain and removed. The texture threshold ensures that matched blocks contain sufficient contrast to be considered reliable. Finally, the distance threshold is used to compare the forward and backward match. Once a point in the left image is matched with a point in the right image the right image point is then matched to the left image. Ideally the second match should return the original pixel from the left image. However, due to false matches this is often not the case. Therefore the distance threshold sets by how much the original pixel and the second matched pixel are allowed to differ by.

These parameters, their experimental ranges, and typical optimal values are summarised below in Table 5.3.

Parameter	Experimental Range	Optimal Value
Block Size	5 – 255	23
Disparity Range	-128 – 128	-16 – 16
Uniqueness	0 – 100	15
Contrast	0 – 1	1.0
Texture	0 – 1	0.0002
Distance	0 – 100	0

Table 5.3: The experimental and typical optimal values for the six disparity parameters

Disparity values were converted to displacement values using a basic linear regression. A target with three 10 mm square panels with a horizontal displacement of 10 mm was imaged. The panels had a 10 mm displacement parallel to the optical axis of the camera, with the left most panel being the front most. The average disparity of each panel was calculated and a linear regression was performed to produce an equation relating disparity to displacement.

Disparity images were converted to displacement images and then into a 3D point cloud. A laser scan of the images surfaces was obtained which provided a ground truth 3D point cloud. Both the image and laser scan point clouds contained large regions of erroneous points, such as where the table surface had been partially imaged. As such software was designed to allow the manual remove of the large regions of noise. The two models were then matched to each other using the *iterative closest point* (ICP) algorithm, developed by Per Bergström under the BSD license (Bergström & Edlund 2014).

5.2.3 Results

Three synthetic saw bones were imaged: a separate tibia and femur and an elastically jointed femur and tibia that had undergone resection for a medial UKA. However, laser scans could only be obtained for the separate bones. The imaged bones and resultant disparity images are shown below in Figure 5.6.

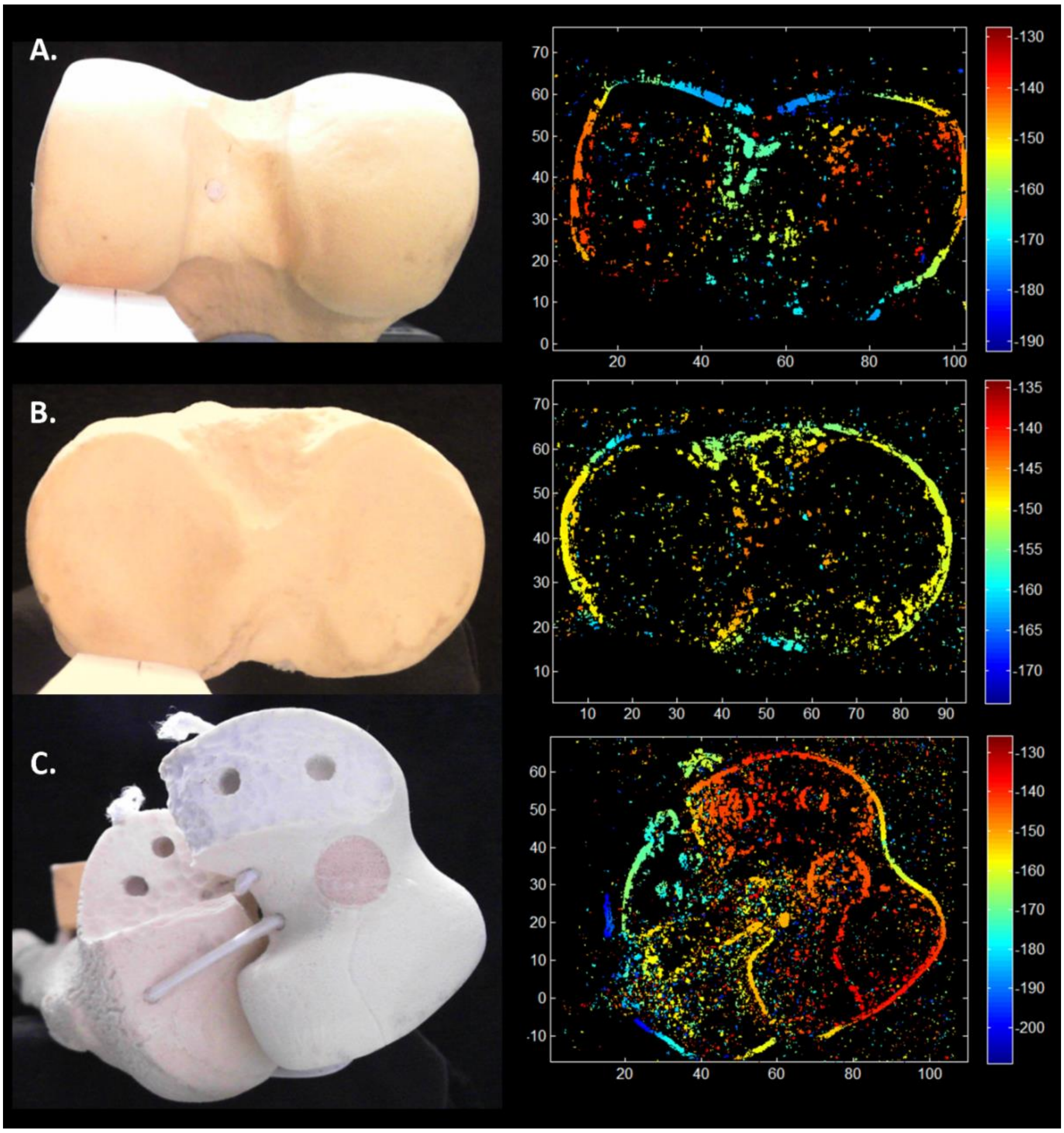


Figure 5.6: Synthetic saw bone images and representative displacement images. A. separate femur, B. separate tibia, C. resection knee (All scales in millimetres)

Several measurements were taken of the physical bones using digital callipers and compared to measurements obtained from the displacement images. This accuracy comparison is shown below in Table 5.4. Both physical and image measurements were repeated several times and incurred a measurement error of ±1 mm due to the inconsistency of edge definition. For all measurements across all samples an RMSE value of 4.6±2.0 mm was obtained.

Bone	Measurement	Physical (mm) ±1	Image (mm) ±1	Accuracy (mm) ±2	Fractional Error
Separate Femur	Medial-lateral	88	94	6	0.068
	Anterior-posterior	51	47	-4	-0.078
Separate Tibia	Medial-lateral	87	86	-1	-0.011
	Anterior-posterior	49	47	-2	-0.041
Jointed Femur	Medial-lateral	75	76	1	0.013
	Inter-peg hole	18	19	1	0.056
Jointed Tibia	Medial-lateral	71	61	-10	-0.14
	Inter-peg hole	13	10	-3	-0.23

Table 5.4: Accuracy comparison of displacement image for three imaged bones

The resulting accuracies are plotted below in Figure 5.7 against the mean displacement from the camera at which they were taken.

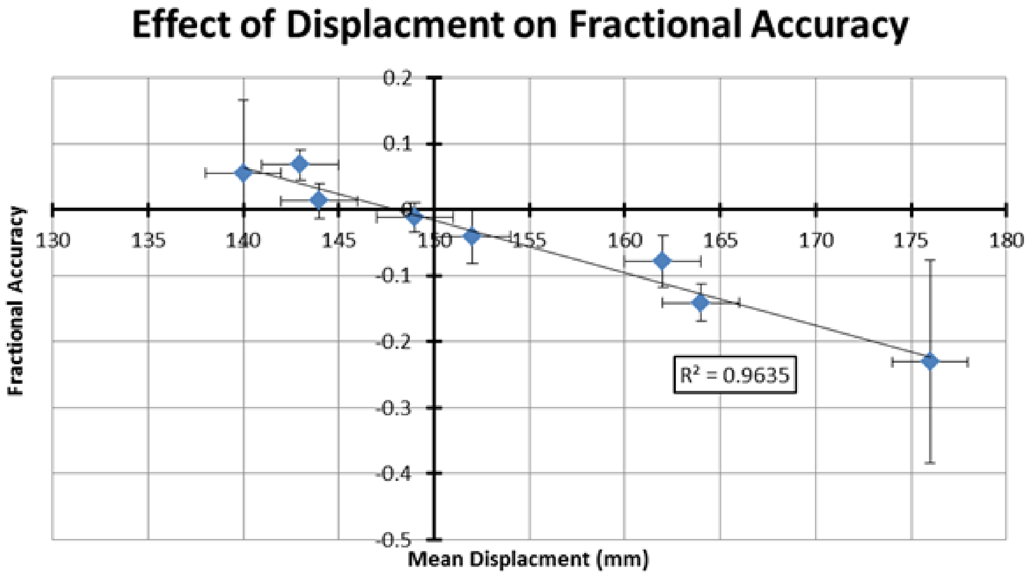


Figure 5.7: The effect of mean measurement displacement along the optical axis upon the accuracy of measurements.

The point clouds produced by imaging the separate tibia and femur models were matched to ground truth data using an ICP algorithm. The resulting matches are shown below in Figure 5.8. The figure also shows the mean error of the matches.

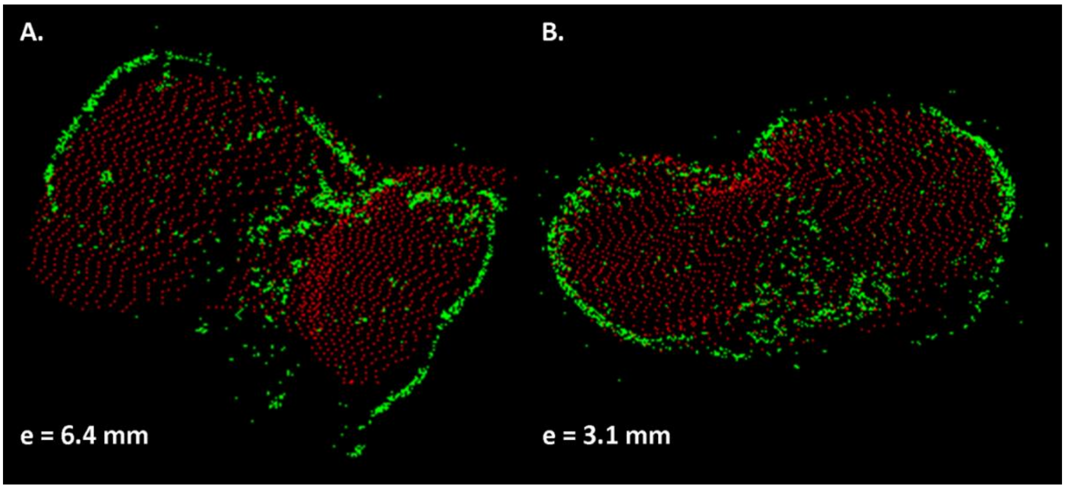


Figure 5.8: ICP matching of femoral (A) and tibial (B) imaged displacement models with laser scan data.

5.2.4 Discussion

Fundamentally, this study has demonstrated the possibility to produce a 3D model from a stereoscopic imaging system, and correlate this with a model gained through more traditional means, thus potentially allowing the relative position between bone and burr to be determined, and for the burr to be controlled in accordance with a pre-operative plan designed on a patient specific virtual model. However, this system is far from ideal and has many problems, some of which are considered to be unavoidable without substantial modification to the system.

The most evident problem with the system is illustrated by Figure 5.6 in which large regions of the depth images are shown as black. Black regions within the depth images represent pixels that were not successfully matched between the two stereo images. Therefore it was not possible to calculate a depth for these pixels. It is seen in all three disparity images that the majority of pixels were unmatched. It is believed that this is a result of the uniform or homogeneous nature of the graphical texture of the saw bones used (Koschan et al. 1996). As discussed in Section 5.2.2 above, the matching algorithm first produces intensity or texture profiles. These are produced for a block, defined by the block size parameter, about each point in the images. The profiles from the left and right images are then compared using the various search reduction schemes discussed above. If a sufficiently similar intensity profile is found the pixels are matched. However, as discussed above a number of parameters are used to filter the results. For each pixel the generated intensity profile must contain enough detail to pass the texture threshold. Due to the highly uniform nature of the saw bone few pixels produced sufficiently detailed intensity profiles. Decreasing the texture threshold reserved more pixels and thus produced more matched points in the disparity image. However, as weak intensity profiles were passed it was difficult for the system to ensure good matches. Therefore, increasing the texture threshold resulted in considerably more noise in the disparity image due to erroneous matches. Similarly, the uniqueness threshold is used to compare the most probable match with other probable matches. As the bone surface is highly uniform many pixels are likely to return very similar intensity profiles and as such many possible matches will be produced. As the system is not able to determine which match is most likely all matches are disregarded. Decreasing the uniqueness threshold reduces the amount by which the best match must be superior to the other possible matches. Therefore, fewer matches are rejected and more pixels are produced in the disparity image. Again, these matches are of low quality and result in a considerable increase in noise. The typical texture and uniqueness thresholds used, approximately 0.0002 and 15 respectively, were found to produce the least sparse disparity images, without excessive noise. The effect of texture uniformity is well demonstrated by Figure 5.6C. Comparison of the anterior aspect of the lateral condyle (lower right corner) and the posterior aspect of the resected medial condyle (upper centre) reveals a considerable difference. The smooth homogeneous lateral condyle has very few matched points with a notable amount of noise. Conversely, the medial condyle has an extremely textured surface and is seen to produce many more matches, with an improved signal to noise ratio. The posterior aspect of the medial condyle has a particularly high match density. As can be seen in the bone image of Figure 5.6C the lighting of the bone produced complex shadows that caused this texture to be particularly pronounced on the posterior aspect. This resulted in a less homogeneous image, and therefore more diverse intensity profiles. The edges of all bones imaged in Figure 5.6 show high match density. This is the result of the sharp contrast between white bone and black background producing very distinct, and therefore easy to match, intensity profiles.

The poor quality of intensity profiles, resulting from the highly homogeneous surface images, was also likely the cause of many of the erroneous matches. Under intended operation if the system is unable to find the true match, for example due to large lighting difference between the two images, the pixel would be unmatched and set as black. However, again due to the homogeneous surface, if the true match is not found it is possible that another pixel produces a sufficiently similar intensity profile to be incorrectly labelled as a match. This is likely the cause of noise such as the sparse blue samples seen on the predominantly red medial (right) condyle in Figure 5.6A.

The accuracy measurements shown in Table 5.4 at first appear erratic, ranging from 6 to -11 mm of error. However, Figure 5.7 suggests a clear trend that the errors are related to the mean displacement along the optical axis at which the measurements were taken. The error is seen to be at a minimum around 150 mm, intercepting the x-axis at 147.8 mm. This is indicative of the disparity to displacement conversion used. At the time of this initial research it was believed that an effective way to convert the disparity image into a displacement image, an effective 2.5D model of the object, was to image a known geometry, at a mean displacement of 150 mm, and produce linear equations relating disparity to displacement in the z-axis and pixel coordinates to displacements in the x and y-axes. The errors seen are at least partially caused by this approach. It is now appreciated that during the rectification process the fundamental matrix of the stereo camera system was calculated. The correct approach to conversion would be to multiply each pixel within the disparity image by this fundamental matrix, as shown below.

$$\begin{pmatrix} X \\ Y \\ Z \\ 0 \end{pmatrix} = F \begin{pmatrix} column \\ row \\ disparity \\ 0 \end{pmatrix}_{pixel} \tag{5.3}$$

The linear regression approach used would have caused model conversion to become increasingly erroneous at greater displacements. This was supported by the findings of the final section of this study. After conversion the displacement point clouds of both the separate tibia and femur were matched against laser scans of their surfaces. It may be seen in Figure 5.8 that the femoral match produced a mean error of 6.4 mm compared to an error of only 3.1 mm for the tibial match. The image of the tibial match may be seen by eye to match considerably better, with sections of the femur model poorly scaled to the ground truth. This is likely, in part, a result of the linear conversion approach. The tibial surface imaged is relatively planer with a range of depth along the optical axis of approximately 10 mm. Therefore, as shown by Figure 5.7, with the tibia at a mean range of 150 mm the conversion error was at a minimum. Conversely, the femur presented a large depth range along the optical axis, approximately 35 mm, resulting in a much larger conversion error.

In conclusion, it is proposed that this study demonstrated the potential for stereoscopic matching of a bony surface. However, major improvements to accuracy are required. As discussed above, using the fundamental matrix to convert the disparity image into a 3D, or more accurately 2.5D, point cloud would offer considerable improvement over the approach presented here. Additionally, performing a full calibration of the camera system and correcting for lens distortion would also improve the accuracy of the system. An additional concern with the system is processing time. As described in Chapter 4, the tracking system requires a real-time acquisition frequency of approximately 30 Hz, or less than 35 ms per acquisition. The system presented above is far from real time. Assuming appropriate filters were developed to automate noise removal the system requires approximately 60 seconds per acquisition. Disparity calculations account for between 5 and 10 seconds, while the ICP model correlation requires 45 seconds on average. It is believed that both of these times could be reduced to a degree. For real-time applications model correlation could be improved by using the solution of the previous acquisition as an initial guess to the ICP algorithm. Additionally, the number of points used within the correlated data sets could be reduced. Both of these measures would decrease the computational cost of the correlation algorithm and therefore reduce the time required. Within recent years considerable effort has been made into developing real-time disparity algorithms and several approaches have produced impressive results (Mroz & Breckon 2012).

Even with these potential improvements it is suggested that the pure stereoscopic matching approach discussed above would be currently unable to provide an accurate and densely populated disparity image. As described previously, the surfaces of the bone are extremely homogenous when imaged, with a relatively uniform colour and minimum topology to cause shadows. One approach to improve the disparity image is to project a pattern onto the surface of the bone. This reduces the homogeneity of the image and therefore provides stronger intensity profiles, resulting in an increase of true matches. Furthermore, this approach may be used to speed up image matching, as discussed in Section 2.4.2.1.5. However, with the tool mounted approach investigated here, projected light approaches would require a projector to be mounted to the tool in addition to the two cameras. While very light weight projectors are available, the tool may become overly cumbersome. Additionally, the projected patterns and dim lighting conditions often required by such structured light systems would be unsuitable for the surgical environment. Although, an infrared approach, using a projected pattern outside of the visible spectrum of the surgeon, could eliminate these two issues. This shares some similarity to the next imaging system investigated, time of flight.

5.3 Time of Flight

5.3.1 Introduction

The principles of TOF cameras are described in Section 2.4.2.1.3 of the literature review. They offer the potential to directly extract depth information from a surface, independent of the surfaces texture, or lack thereof. By emitting a pulse of light and calculating the time taken for the light to reflect off the imaged object and return to the image sensor, TOF cameras are able to determine the distance to each object point for each pixel. Importantly, TOF cameras operate in the infrared spectrum, therefore, there is no visible light to distract the user, nor does the object require specific lighting conditions – at least not to the same degree as visible spectrum systems. These advantages motivated the following experimentation into TOF cameras.

5.3.2 Materials and Methods

A PMD[vision]® CamBoard nano (PMDTechnologies GmbH, Siegen, Germany) TOF camera was selected for these experiments for several reasons. The CamBoard measured 37x30x25 mm and weighed only 34 g, due to utilising only a single image sensor (pmd PhotonICs® 19K-S3) and 850 nm LED (OSRAM Dragon SFH4235). This was ideal for the proposed tool mounted tracking system. Furthermore, the CamBoard had an advertised acquisition rate of 90 Hz, providing depth or full 3D Cartesian coordinates directly. The CamBoard connected via a single USB 2.0 port and provided a Matlab API. One notable disadvantage of the CamBoard nano was the limited image sensor resolution of 160x120. The image sensors used by TOF cameras are extremely expensive and therefore resolutions are typically limited.

A program was developed in Matlab to connect to the CamBoard, apply settings and stream the depth image to display in real time. Initial tests showed the acquisition rate as approximately 45 Hz, half the advertised rate. The Matlab API function used for frame capture was found to be a bottleneck in the system. A third party Matlab Mex file provided an alternative capture function that exploited multiple processor threads (Lutz & Stefan 2012). This raised the acquisition rate to 80 Hz. The Matlab program developed was used to tune the settings of the CamBoard to produce optimal images. The CamBoard API provided both spatial and temporal filters. The spatial bilateral filter applied an edge preserving smoothing filter. The bilateral filter was found to reduce some noise and as such was used throughout experimentation. The default parameters appeared to produce the best error reduction across a range of materials. Temporal averaging was implemented as a moving average with adjustable frame range. Values between 0 and 10 frames were used throughout the experimentation in an attempt to reduce noise. The API also allowed setting of the integration time between 12 and 2000 μs. The integration time is analogous with the exposure time of a regular camera in that a longer integration time increases the intensity of the image. The appropriate integration time was found to be highly dependent on the material imaged.

After initial testing, an experimental protocol to test the accuracy and precision of the CamBoard was developed. The CamBoard was placed 150 mm from a planar matt surface and orientated such that its optical axis was perpendicular to the surface. A two stage acquisition and analysis program was developed. The first stage connected to the CamBoard, activated the bilateral filter with default parameters, and set the integration time to 150 μs. The program then performed 1000 acquisitions each using five different temporal averaging values: 1 (no averaging), 2, 3, 5, and 10. Once captured additional spatial averaging was performed using a box filter of sides: 1 (no averaging), 2, 3, 4, and 5. This resulted in 25 data sets of 1000 frames each, a total of 25,000 frames or 480 MP. The analysis program stage performed statistical analysis upon these data and provided a GUI interface to inspect the results of desired frames or pixels. For each spatial pixel within each data set temporal statistics were calculated and stored: minimum, maximum, range, mean, variance, standard deviation, and interquartile range. These statistics were also calculated for the frame as a whole and for each of the total data sets.

Figure 5.9 shows the displacement of a single pixel at each frame. The initial reading is seen to be over 10 mm below the steady state value. The displacement increased over the first 150 to 200 frames. This was indicative of an initial warm-up period of the TOF camera system. Therefore, the above acquisition program was modified to capture 1500 frames, excluding the initial 500 frames from storage, to ensure the system had reached a steady state.

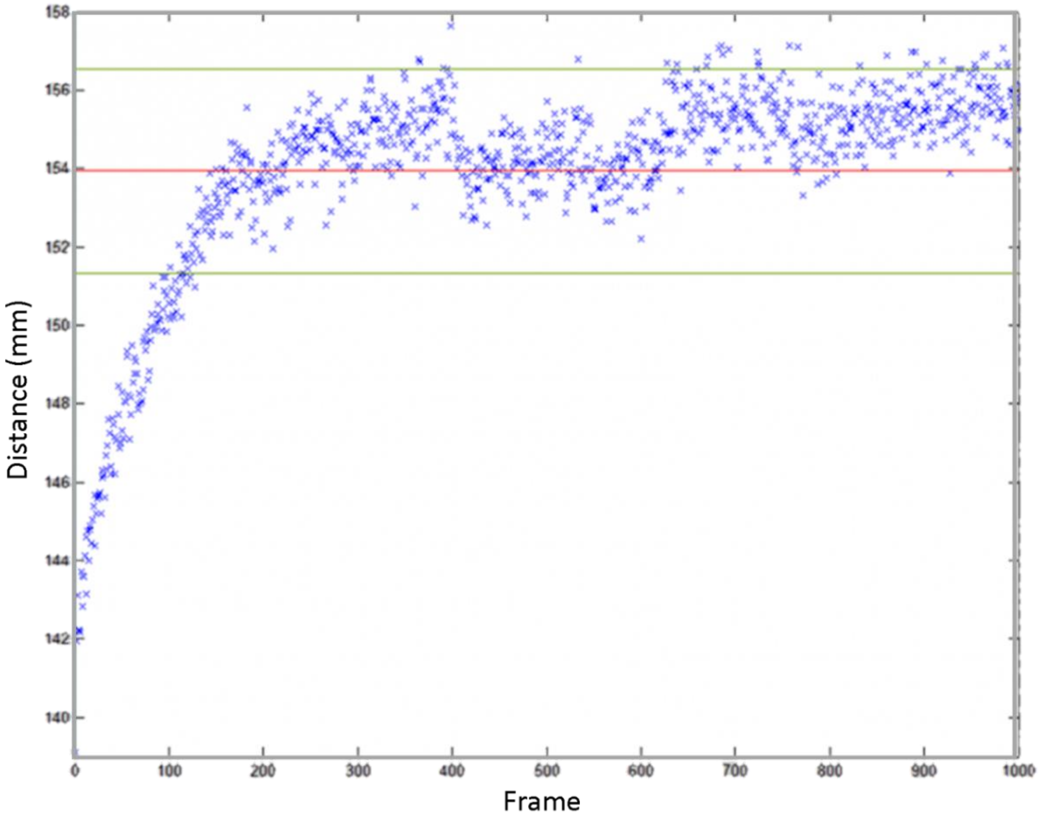


Figure 5.9: CamBoard nano warm-up time observed for a single pixel

5.3.3 Results

Figure 5.10 shows an example distance image produced without spatial or temporal averaging.

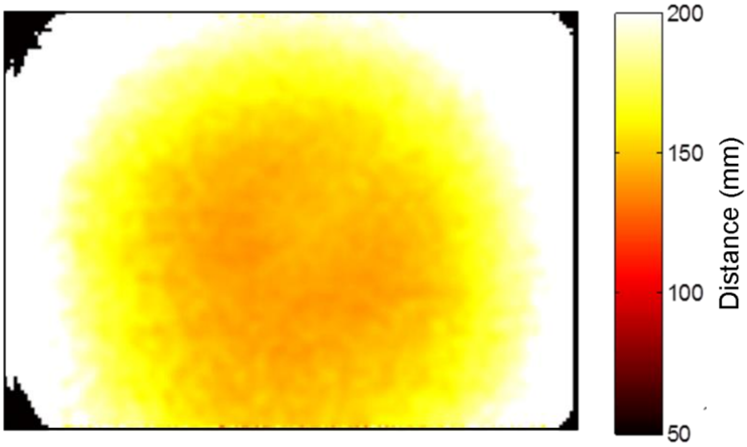


Figure 5.10: Example distance image produce by CamBoard nano system without spatial or temporal averaging of planar matt surface.

From this un-averaged data set Figure 5.11 below shows the distance value returned for a single, near-centre pixel (80,60), across the full 1000 frames captured. The central horizontal solid red line indicates the mean distance of the pixel, while the two dashed green lines show the positive and negative standard deviations, 150.68±1.44 mm for the pixel shown.

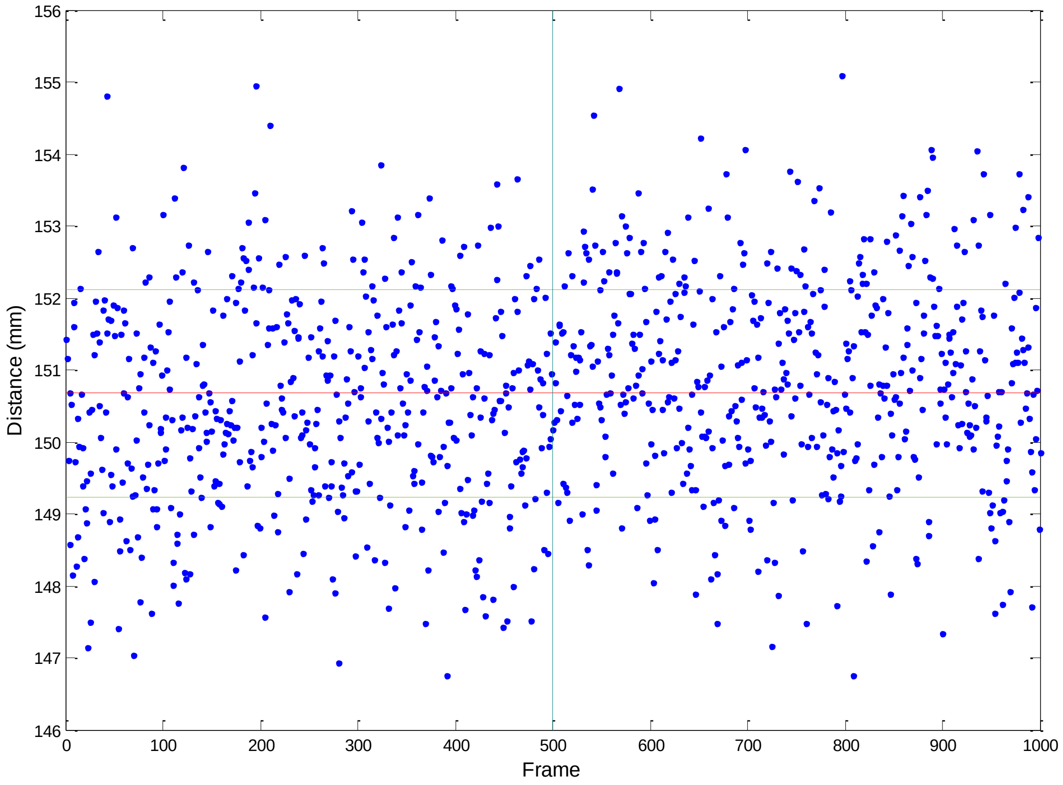


Figure 5.11: Example temporal distance profile for a single un-averaged pixel across all 1000 frames.

The effect of spatial and temporal averaging on the distance image is shown below in Figure 5.12.

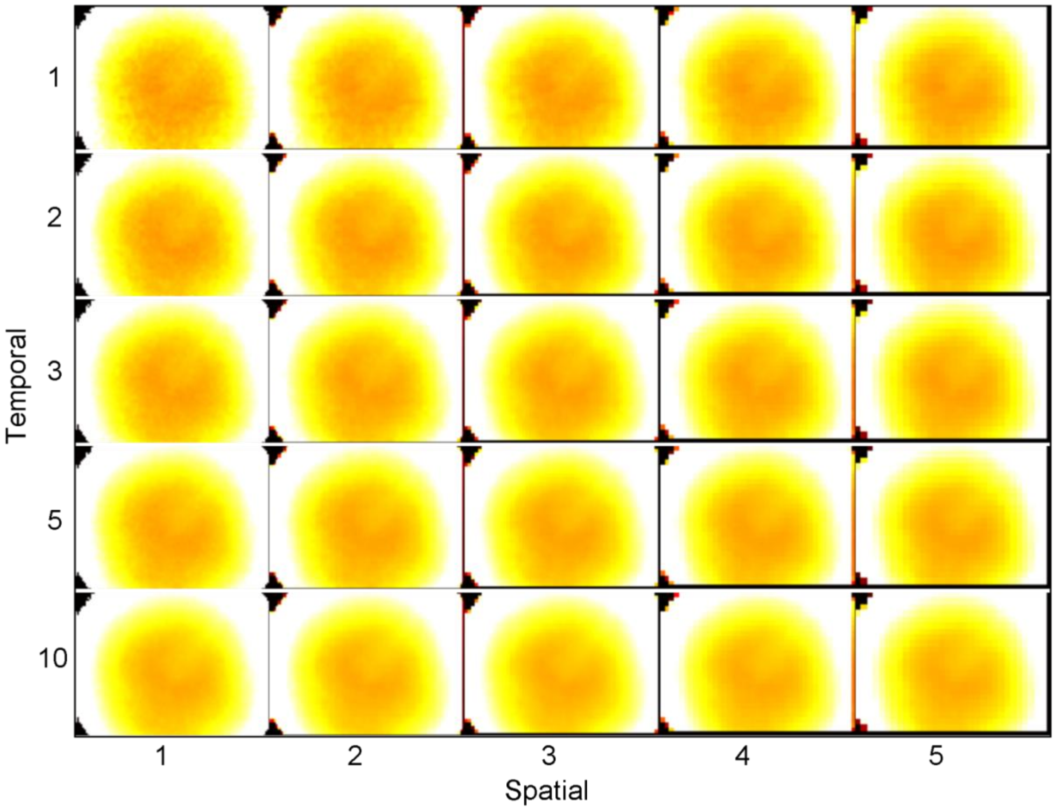


Figure 5.12: Effect of spatial and temporal averaging of distance image.

The range and standard deviation were calculated for each pixel from the 1000 frames. These values were then averaged across the full image. These values are plotted for each spatial and temporal averaging below in Figure 5.13 and Figure 5.14, respectively. A mean distance value was not calculated as it would be expected to change across the image.

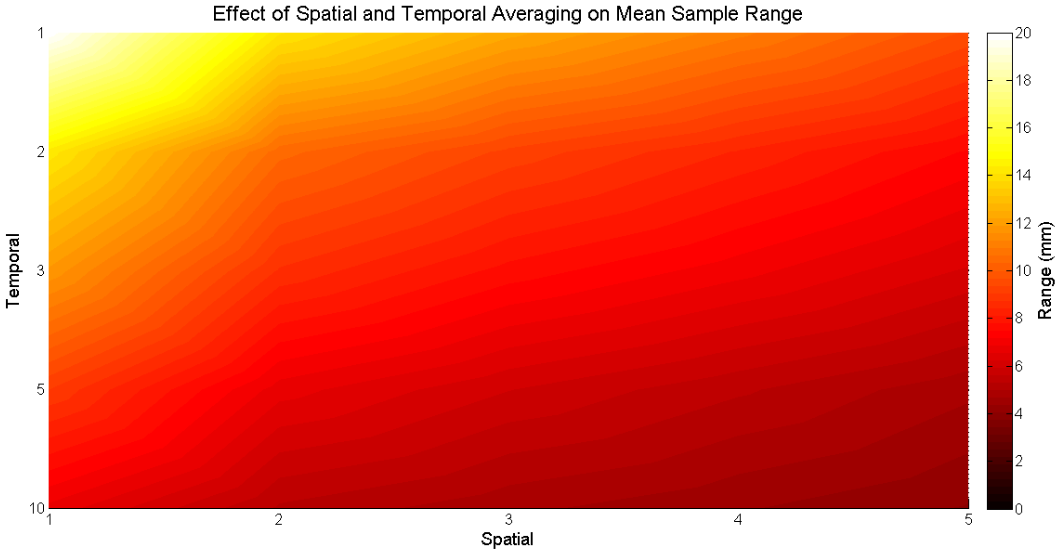


Figure 5.13: Effect of spatial and temporal averaging on the mean sample range.

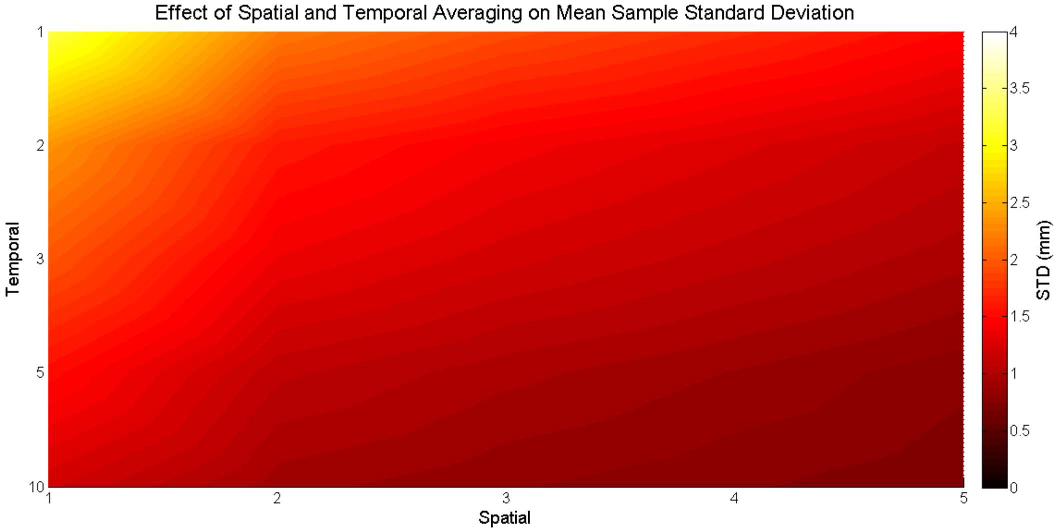


Figure 5.14: Effect of spatial and temporal averaging on the mean sample standard deviation.

5.3.4 Discussion

When using the distance function the CamBoard returns the distance from the camera centre for each pixel. Therefore, Figure 5.10 presents the expected image pattern, whereby the centre of the image of the planar surface is closer than the peripheries. The black regions in the corners of the image are indicative of error filtering. If pixels have insufficient intensity they are flagged as having a high chance of being erroneous, and are thus removed. It is noted that the central region of the image shows visible noise. During data acquisition this noise was more pronounced, as pixels changed value despite the camera being stationary. Figure 5.11 shows this clearly for a single pixel near the centre of the image. The range for this single pixel was found to be greater than 8 mm. It may be shown that there was no trend to the data, a running mean remained relatively constant as did the range.

In an attempt to reduce this variance the effects of spatial and temporal averaging were investigated. Figure 5.12 shows the full range of averaging combinations investigated. In the figure is may be observed that the noise of the central region decreases as progressively larger averaging are applied. It may be seen how spatial averaging reduces the resolution of the image. In the static situation shown here temporal averaging does not decrease the image quality. However, Figure 5.15 below shows a comparison of the same action, waving a hand slowly, with and without temporal averaging.

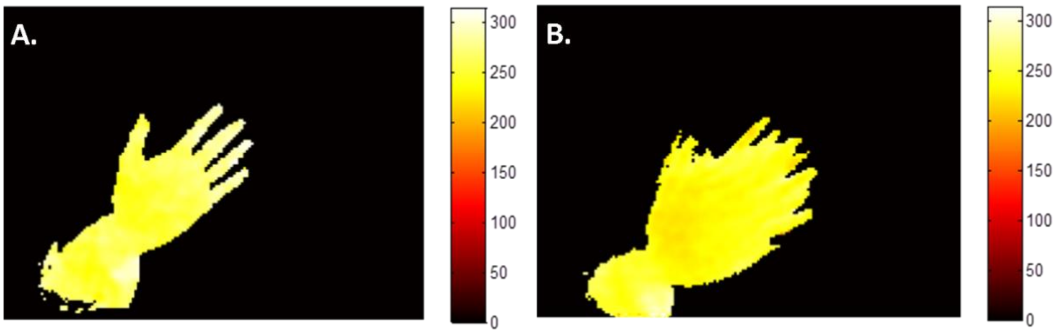


Figure 5.15: Effect of temporal averaging on a moving object. A – Zero temporal averaging. B- Three frame temporal averaging.

As expected, as temporal averaging combines multiple frames, moving subjects are considerably distorted, with the distortion increasing with increasing temporal averaging or object speed.

It may be seen from Figure 5.13 and Figure 5.14 that range and standard deviation, respectively, are both decreased by the application of either spatial or temporal averaging. For spatial averaging the range and standard deviation are decreased from 19.9 and 3.24 mm to 9.62 and 1.49 mm, respectively, a reduction of 51.7 and 54.0% from five times spatial averaging. Meanwhile, five times temporal averaging provided a reduction of 54.5 and 52.8%, in range and standard deviation respectively. With both spatial and temporal averaging set to maximums a total reduction of 79.7 and 78.4% for range and standard deviation, respectively, were produced. At this averaging the images present a mean standard deviation of 0.7 mm. The signals were tested for a normal distribution using a Jarque-Bera test which failed to reject the null hypothesis at 1% significance. Therefore, it may be said that approximately 31.6% of samples will fall more than 0.7 mm from the mean with maximum averaging and more than 3.24 mm without averaging.

As demonstrated by Figure 5.15, such high temporal averaging is not practical for the tracking of a moving object, such as would be required during guided UKA. Furthermore, spatial averaging considerably reduces the potential accuracy of the system. At the 150 mm working distance used for these experiments the horizontal FOV was measured to be approximately 250 mm. Therefore, with the limited horizontal resolution of only 160 pixels this equates to a physical resolution of 1.6 mm/pixel. This value increases with spatial averaging, with even only two times averaging doubling the value to 3.2 mm/pixel.

Without averaging the system is unlikely to meet the accuracy and precision requirements of the application. The physical accuracy limit imposed by the resolution of the camera exceeds the accuracy requirement. Furthermore, the precision indicates that even with a perfect mean accuracy a third of samples will be more than 3 mm off. The precision may be improved to more acceptable levels through averaging. However, this is at the cost of either spatial or temporal accuracy.

Therefore, despite the advantages of the CamBoard nano, such as its small size, high sample rate, and ability to provide direct 3D information, it was concluded that it would not provide sufficient accuracy or precision to successfully guide a UKA procedure.

5.4 Conclusions

Both the stereoscopic matching and TOF systems examined above demonstrated potential. However, it was concluded they both failed to meet the requirements of the investigated application. Both systems lacked accuracy and precision and would potentially require excessive processing time, due in part of model matching. Several improvements were suggested. However it was still suggested they would struggle to meet the requirements of a practical intra-operative tool mounted tracking system, as outlined in Chapter 4. In addition to the problems highlighted it is proposed the systems fall foul of a fundamental flaw of the concept. A core advantage of the UKA procedure, as described in Section 2.3.2.1 is the reduced recovery time, partially attributed to the smaller incision required due to the unicondylar exposure. The two tracking systems described above both directly track the bone. To perform this task accurately, even with idealised systems, would require a considerable amount of bone to be visible to the tracking system at all times. This alone presents a challenge, however, much of the condyle being operated upon would be frequently obscured by resected bone particles and fluids, suction tools, and the resection tool itself. Therefore, it is probable that a full exposure would be required, exposing the opposite condyle to allow tracking. This would defeat one of the key advantages of the UKA approach. While one of these two systems may be able to meet the original aims of the research and provide an alternative, less bulky and intrusive tracking system, it was evident that this should not be done at the expense of the patient.

6
First Generation – ARToolKit

6.1 Introduction

Chapter 5 concluded that marker-free tracking did not present an ideal solution to UKA procedures, and as such marker based approaches were investigated. One such approach, based upon the ARToolKit library (Kato & Billinghurst 1999), utilised planar fiducial markers imaged by a single visible-spectrum camera. This provided a very compact, lightweight, and cost-effective solution. The ARToolKit library was also investigated due to its original function, of providing an AR overlay. AR presented an extremely intuitive form of guidance, and as such the ARToolKit library could potentially meet both the accuracy and guidance aims of the project as outlined by Chapter 3.

In line with the iterative and incremental development approach, the initial design aimed to implement only the elements required to test the suitability of the system. These were defined as: tracking, of bones and tool, bone model generation and modification, including joint centres, and basic user interface. At this stage a cutting tool was not to be included.

This chapter first describes the working of the ARToolKit library. The design of the system is then discussed from a hardware perspective, describing the physical arrangement of the system, and then detailing the software design, making use of UML diagrams.

6.2 ARToolKit

ARToolKit is described by its developer as a software library for building augmented reality applications. It was designed to increase access to AR to those without computer vision expertise and allow developers to focus on their application and not the underlying algorithms. The library is based upon the detection and pose estimation of square planar fiducial markers, such as that shown by Figure 6.1, from a standard 2D colour video stream, in real-time. This pose information is then used to render virtual objects onto the video stream, such that they align and move with physical objects, as though real themselves.

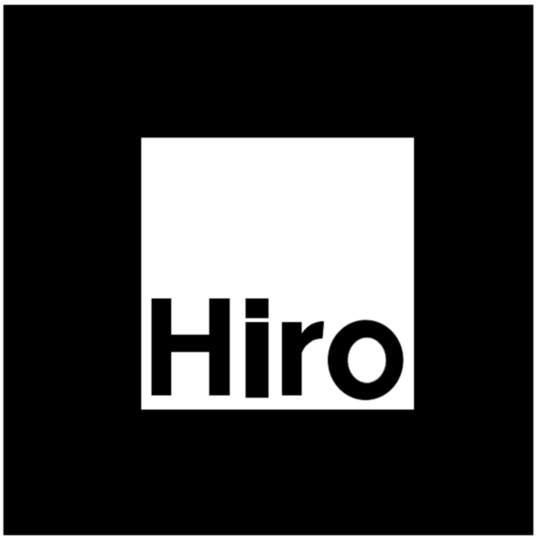


Figure 6.1: Example ARToolKit marker

ARToolKit interfaces with a standard web camera and performs the four key functions of a tracking system identified in Section 4.3, namely marker: detection, identification, feature extraction, and pose estimation. The first three of these functions are based upon standard computer vision techniques, centred on contour detection, and are optimised for real-time performance. The pose estimation is based upon the pin-hole project model of a camera.

By locating the four corners of a marker within in the video frame the library is able to estimate the pose of the marker. To do this ARToolKit inverts the pinhole camera projection model. The pinhole model states that a single ray of light from an object point travels in a straight line and passes through the pinhole to produce a single image point upon the image plane. Therefore, by reversing the situation it is possible to say that a single image point must have been produced by a ray of light that passed, in a straight line, through the pinhole from the object point in the world. This concept is illustrated by Figure 6.2 below.

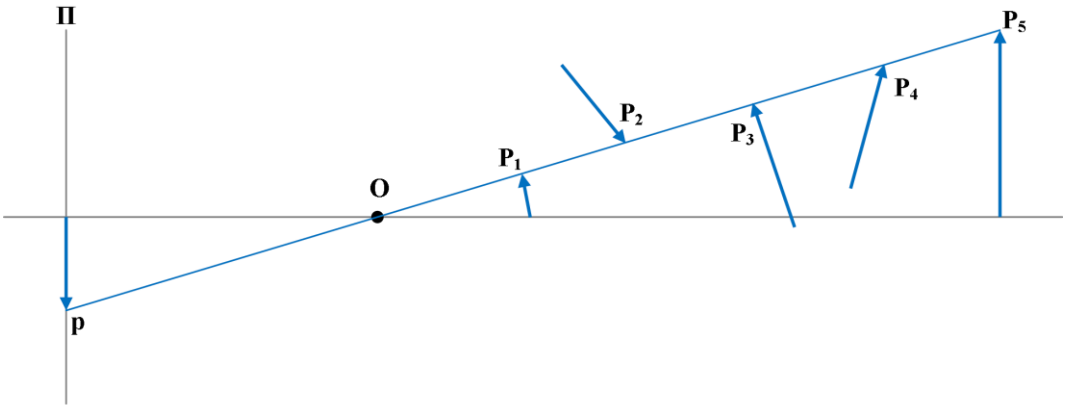


Figure 6.2: Reverse pinhole projection, extrapolating a line from image point p from which the object point P which produced the image must lie.

Several object points P_{1-5} and the objects to which they are attached are shown in Figure 6.2. As the image point is only two dimensional it is not possible to extract depth information about the object point. Therefore, it is only possible to define the object point as lying somewhere upon the projection line shown. Additionally, with only one image point it is not possible to define the orientation or any other information about the object that produced it. However, the ARToolKit located four image points, providing four projection lines for each marker. Importantly the system also knows the relative position of each corner to the rest in the coordinate system of the marker. This additional information produces the scenario shown in Figure 6.3.

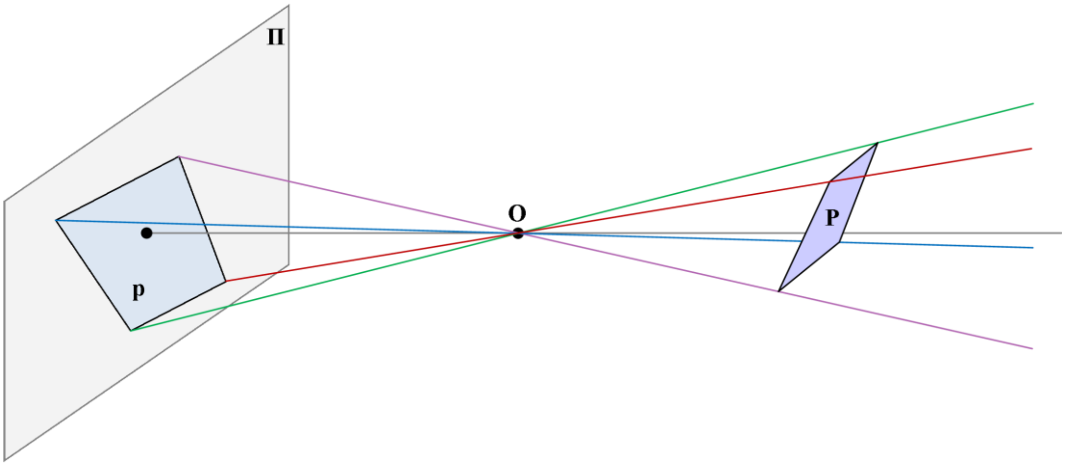


Figure 6.3: Reverse pinhole projection of the four corners of a fiducial marker.

The four projection lines are each known to correspond to a certain corner and the relative positions of each corner is known. Therefore, there is only one position and orientation that the marker object P may have had to produce the observed image p.

ARToolKit determines the marker location through a two stage process. Firstly, a geometric approach is used to derive an approximate location. This location is then refined through an iterative projection process. The geometric stage treats the corners in pairs, to produce two lines on opposite sides of the marker image. These image lines are projected into world space using the projection matrix of the camera that is obtained through calibration. This produces two pairs of planes. The cross product of the first vertical pair of planes, V_1 and V_2 is taken. The cross product of two planes provides the vector of intercept of those two planes. The process is repeated for the second horizontal pair, H_1 and H_2. This provides a second vector, perpendicular to the first. Taking the cross product of these two lines provides a third vector, again, perpendicular to the previous two. The three perpendicular vectors provide an orthogonal axis, which is equivalent to the orientation of the marker in world space relative to the camera. With the orientation known it is now possible to derive the relative position of the marker through simple geometry, based upon the know dimensions of the marker.

The non-ideal nature of the system makes the geometry unlikely to produce a closed solution. Therefore, the second iterative stage uses the geometrically derived pose and projects the solution onto the image plane using the calibrated projection matrix. It compares this virtual image with the real image and iteratively adjusts the pose to minimise the difference.

Once each imaged marker is identified and has had its pose estimated ARToolKit allows augmented overlays to be placed, based upon each markers pose and the projection matrix of the imaging camera.

6.3 Design

This section first briefly discusses the hardware design of the first generation system. Next implementation specifics of the requirements outlined in Section 4.3 shall be described during software design.

6.3.1 Hardware

As discussed in Chapter 3, one of the original aims of this research was to reduce the footprint of the stereoscopic optical tracking cameras used by existing CAOS systems. To this end the first generation system proposed a tool mounted tracking system. As noted in Chapter 5 this effectively removes the space requirements of a separate tracking system. ARToolKit was well suited to a tool mounted approach as the system required only a single light weight camera to provide 6 DOF tracking of multiple markers. The camera requirements of this first generation system well matched those outlined in Table 5.1. Therefore, the same LifeCam Studio web camera was used.

Several different markers were used throughout the development of this system, although, most common were 20 mm markers. The size quoted for markers defines the length of the outer black edge of the marker. To improve segmentation from the scene background markers included a 5 mm border around all sides. Therefore, a 20 mm marker had physical dimensions of 30x30 mm. For the first generation system markers were printed using a Konica Minolta bizhub C353 multi-function printer (Konica Minolta, Inc., Osaka, Japan) onto standard A4 copier paper at 600 dpi. These were then mounted, unless otherwise stated, to two layers of 1.5 mm stock card, laminated using super glue to produce a rigid backing.

To allow camera calibration and testing, as discussed in sections 6.4.2 and 6.4.3 respectively, an adjustable camera holder was designed. As shall be discussed below camera calibration under the ARToolKit library followed a two stage process. The first distortion stage required the camera to be fixed at a range of poses relative to a planar target. The second stage required the camera to be displaced along its optical axis at fixed increments. To ensure accurate calibration, it was necessary to precisely align the optical axis of the camera with the vector of this displacement. Therefore the camera holder was designed to allow gradual adjustment of the alignment of the camera, relative to the holder, that would be mounted during calibration. The calibration procedure and use of the holder shall be discussed in detail in Section 6.4.2.

The general design of the holder held the camera in a metal surround. To allow angular adjustment in the X and Y-axes four nylon screws were placed at both ends of the holder. Although only small angular adjustments were expected to be required during calibration the bore of the camera hole was set at 40 mm. This left 5 mm of clearance around the camera, allowing for approximately 9° of angular adjustment from the centre axis. This larger range of movement would allow greater flexibility of the system. The M6 ISO threaded socket with 10 mm depth was placed at the front of the camera holder to allow attachment during calibration. The final camera holder is shown below in Figure 6.4. The additional holes seen in the corners of the holder shall be discussed in Chapter 7.

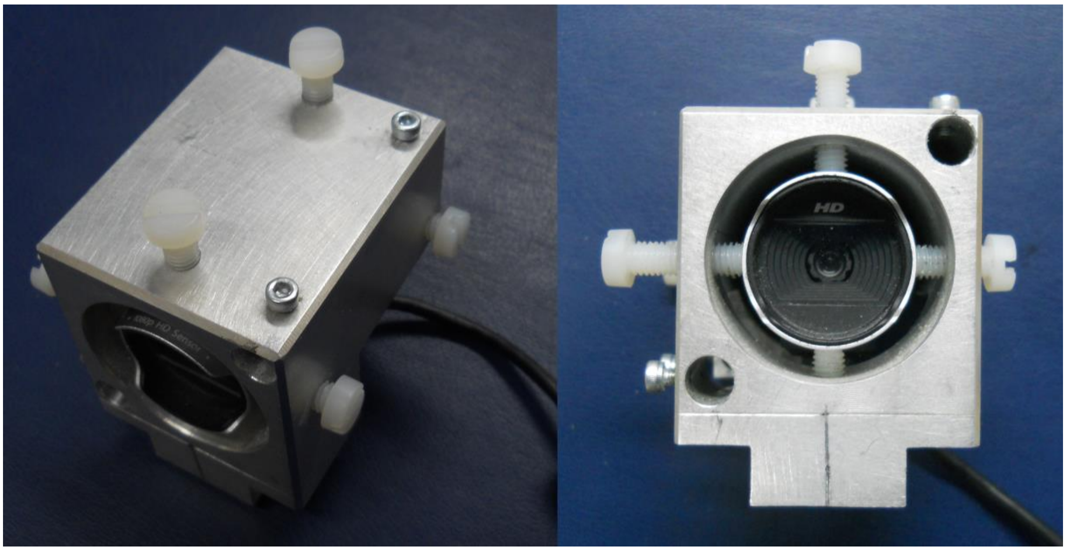


Figure 6.4: Adjustable camera holder

To test the performance of the system a series of probing tests were performed. The probe used was a 125x8 mm scalpel handle with a 40 mm pointed probe attached producing a probe of 150 mm total length. This was attached directly to the underside of the camera using a 10 mm screw hose clamps and two cable-ties,as shown in Figure 6.5 below. Once attached the tip of the probe was set 120 mm from the forward most part of the camera, roughly aligned with the optical axis. The probe was able to withstand practical levels of force without deforming or moving.

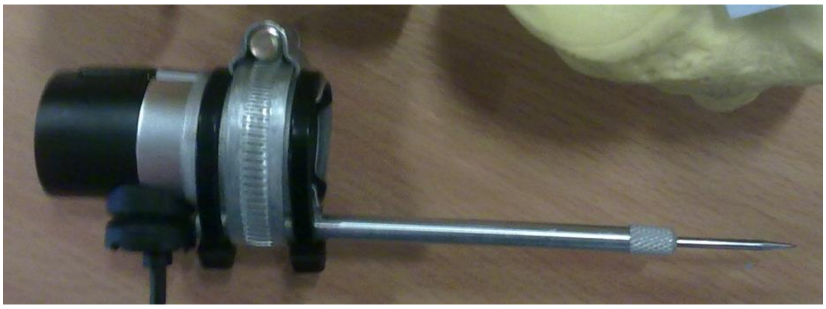


Figure 6.5: Probe fixation method

This probe was also used for joint and surface probing during joint model generation. A resection tool was not developed for this generation. Therefore, the probe was used to simulate the tasks that would be performed by the resection tool in a practical system.

UI, defined as a core system requirement in Section 4.3, was provided by standard keyboard and mouse devices. Therefore, no additional UI hardware was required.

6.3.2 Software

The design of the system was based upon the simple application examples included with the ARToolKit library. However, unlike the library and examples which were based upon standard C it was decided to use an object orientated C++ approach. As demonstrated in Section 4.3 the system divided neatly into distinct elements, which complements the class structure available with C++. In addition to basic classes C++ provides inheritance which was used to extend general classes into more specific classes, as shown later. The strong typing of C++ was also found to be beneficial as it caught and identified many errors at compile time, as opposed to much more difficult to diagnose runtime errors.

Figure 6.6 below illustrates the general structure of the first generation system. The application was built upon the ARToolKit library, which itself was built upon the OpenGL API and *OpenGL utility toolkit* (GLUT) API that interface with the Windows operating system (Kilgard 1996).

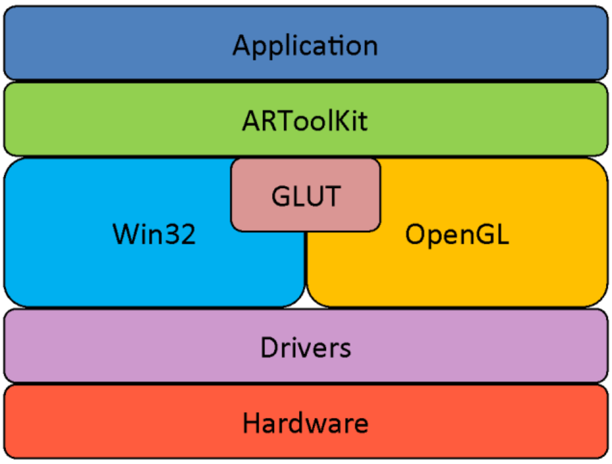


Figure 6.6: General architecture of first generation system.

The GLUT library communicates with the operating system to provide window control and basic user interface.

The ARToolKit application used a continuous loop to repeatedly capture and process video frames. To allow user interaction the application included two additional functions that checked for mouse or keyboard events between each iteration of the main loop. Figure 6.7 illustrates the general control flow of the application. Also included are initialisation and termination functions which are performed outside the main loop.

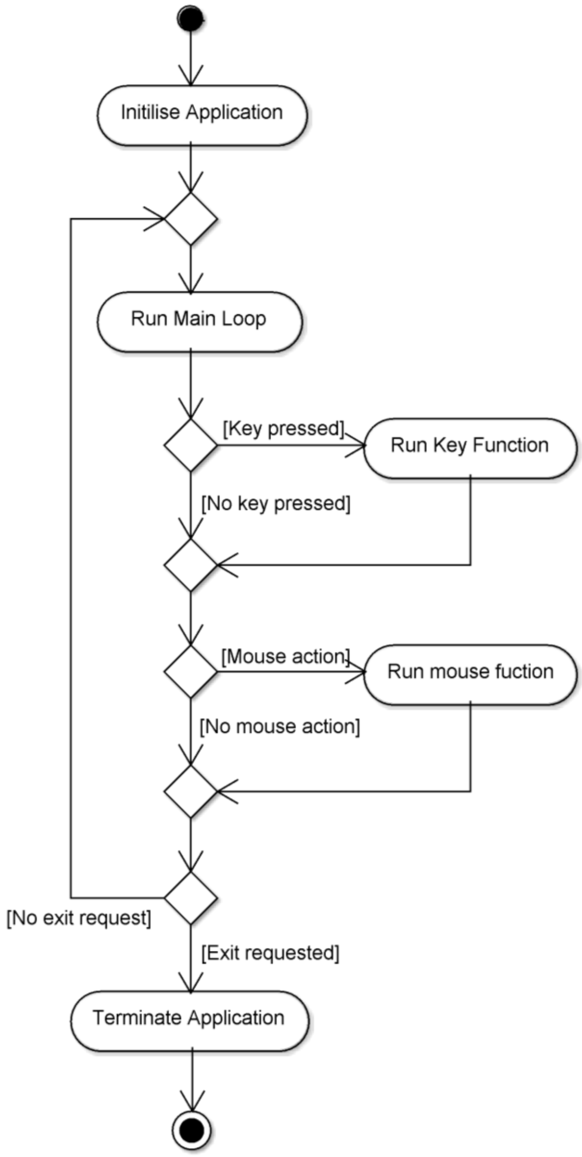


Figure 6.7: General activity diagram of first generation system

The five functions shown in Figure 6.7 were placed within the main source file, as suggested by the example ARToolKit applications.

To design the remaining architecture of the system a top down approach was taken based upon the requirements defined in Section 4.3 and the physical arrangement of the UKA procedure. This designed in summerised by Figure 6.8 below.

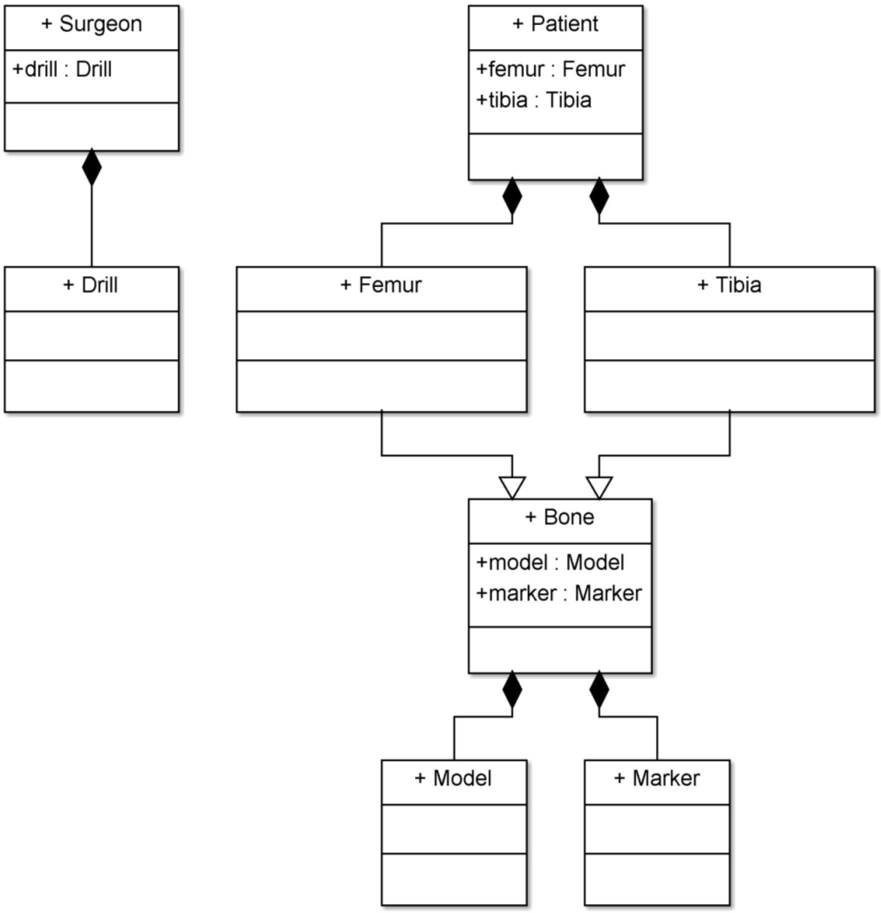


Figure 6.8: Simplified class layout of first generation system

In addition to the physically derived classes shown above in Figure 6.8 a number of supporting classes were required by the system. These classes chiefly provided graphical and UI functionality. Further details of the first generation system may be found in Appendix A1.

6.4 System Analysis

6.4.1 Introduction

Chapter 3 identifies accuracy as a core goal of this research, as such accuracy and precision analysis were performed upon the system throughout its development.

Analysis of the system took two forms. Firstly, the ability of the system to determine the pose of a marker, relative to the camera was investigated. This pose estimation forms the basis for all other tracking used by the system, such as the tool tracking described in Appendix A1.1 and the dual marker tracking used to estimate the hip centre, described in Appendix A1.3.

The second analysis form looked at the probe functionality of the system. The majority of system functions, such as surface generation and joint registration, were performed using the probe system. As such, the accuracy of the probe tracking strongly defines the accuracy of the final system. Additionally, although not implemented in this generation, resection, which lies at the heart of a complete system, would be implemented in almost an identical fashion as the probe.

This section first discusses the calibration of the first generation system. The calibration method and the equipment used evolved over time and a number of the key changes are discussed below. After calibration, camera-marker and then probe experiments will be described and discussed. Finally a brief analysis of the hip centre estimation method is performed.

6.4.2 System Calibration

As described above the ARToolKit library utilised a two stage calibration process. The first stage attempted to characterise the distortion and image centre of the camera systems, while the second stage determined the remaining intrinsic properties.

6.4.2.1 Distortion

The distortion produced by the lens system was characterised by taking multiple images of a planar pattern from multiple angles using the first stage calibration program $calib\_dist$. The pattern consisted of 24 circles arranged in a 6x4 grid pattern. This was scaled to 100x60 mm with 6 mm point diameter, to allow good image coverage at the average working distance, a printed at a resolution of 600 dpi using a standard office laser printer. Calliper measurements indicated an acceptable printing error of 0.1%.

Calibration patterns were mounted to standard 2 mm float glass, due to being highly planar and rigid. Figure 6.9 below shows the calibration targets used for both stages of system calibration.

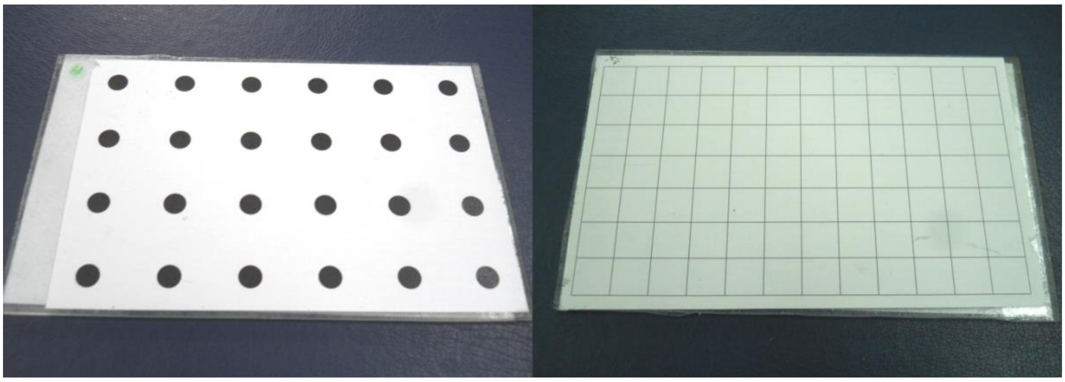


Figure 6.9: Calibration targets

As previously discussed, distortion and intrinsic parameter calibration is dependent upon focal length. Therefore, the focal length was set and locked for calibration, and the same setting used for experimentation. The focus value at which focus was reached was found to change occasionally when the camera was disconnected for longer periods of time. As such the focus of the camera was set using the above method each time.

To ensure a reliable calibration the camera must image the pattern at a range of orientations. The maximum 20 images allowed by the software were used for calibration.

Originally calibration was performed one of two ways. Either, the camera was mounted to a fixed surface and the calibration pattern was position by hand, or vice versa, with the camera instead held. This was found to produce relatively inconsistent calibrations, due to small movements during frame capture. Therefore, calibration was performed with both camera and pattern mounted in adjustable clamps. This resulted in a 25% reduction in the range of inter-calibration camera centres, with dual-fixed calibration producing image centres and ranges of 376.0 and 199.8 and 58.0 and 69.5 pixels, respectively.

The ARToolKit library characterises radial distortion using the image centre and two floating point values, the distortion and scaling factors. This distortion factor is used to move observed pixels to their ideal location. The distortion is applied based on Equations 6.1 to 6.3.

$$d^2 = (x_O - x_0)^2 + (y_O - y_0)^2 \qquad (6.1)$$

$$p = 1 - fd^2 \tag{6.2}$$

$$x_I = p(x_O - x_0) + x_0 \quad y_I = p(y_O - y_0) + y_0 \tag{6.3}$$

Equation 6.1 defines d, the distance between the observed pixel (x_O, y_O) and the image centre (x_0, y_0). Equation 6.2 uses this distance to convert the distortion factor f into the distortion product, p. Finally Equation 6.3 applies the distortion product to both coordinates of the observed pixel, producing the ideal pixel coordinate (x_I, y_I).

All distortion calibrations, including both semi-fixed methods, returned a distortion factor of zero. Initially, this was suspected as an error, either with the calibration method or library itself. However, after investigation it was concluded that zero was a probable value for the camera used. Values of zero were reported by other users of the library, with the consensus being that the cameras produced insufficient distortion to be detected by the calibration method. An archived email by one of the developers of ARToolKit confirmed that ARToolKit used a relatively primitive calibration method. The image distortion produced by the LifeCam is shown below in Figure 6.10, where a printed grid was imaged. A straight digital grid was then overlaid upon the image.

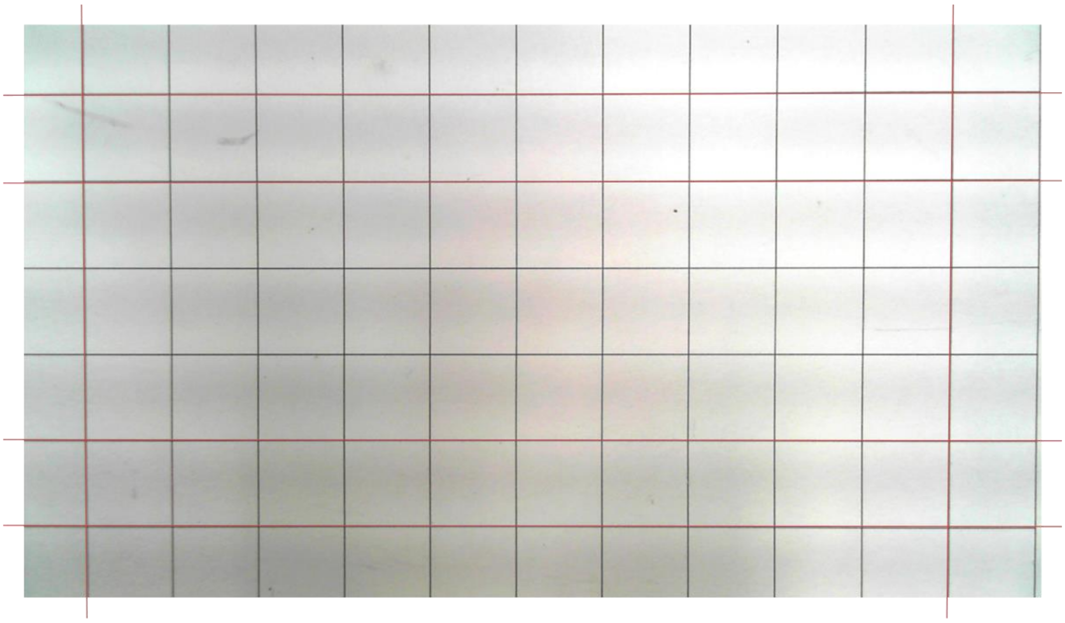


Figure 6.10: Image showing the low distortion induced by the LifeCam optical system.

It is seen that the red virtual grid matches the physical grid across the full range of the image very well. Therefore, it may be seen that the LifeCam produced very little radial distortion, which may account for the zero value returned by the distortion calibration method.

6.4.2.2 Intrinsic Parameters

The intrinsic parameters of the camera were determined using the second stage calibration program, *calib_param*. Multiple images were taken of a 13x8 grid pattern, each time increasing the separation between camera and pattern. The pattern was printed at 120x70 mm with 10 mm grid spacing as shown by Figure 6.9.

To ensure an accurate calibration the movement of the camera must be extremely accurate, both in magnitude and direction. Initial calibration attempts using conventional measuring equipment, such as callipers, returned poor results. An Instron E10K *materials testing machine* (MTM) was used to provide an extremely accurate linear displacement. While the Instron provided very accurate linear displacement it was necessary to ensure that this was parallel to the optical axis of the camera. As discussed in Section 6.3.1 a camera holder was designed to allow minute adjustments to the alignment of the camera. The camera was placed within the holder, and approximately aligned to the bore of the holder using callipers to provide an equal clearance around both ends of the camera. The camera was then fixed into the vertical jaws of the Intron using a 100x20x5 mm aluminium plate, attached using the threaded connector of the camera holder. A bullseye spirt level was then used to roughly align the camera to the vertical.

With the system running the calibration pattern was placed below the camera on the planar test bed of the Intron. The displacement of the camera was set to 90 mm using the coarse adjustment of the Instron, with the fine adjustment set to its minimum of -30.0 mm. The pattern was centred and aligned to the software crosshair. The fine adjustment of the Instron was then set to its maximum value of +30.0 mm, using a ramping speed of 2 mms^{-1}. As the camera separation increased the crosshair was seen to move across the calibration pattern. An iterative approach was used to remove this displacement, which was indicative of a misalignment of the camera.

With the system fully aligned intrinsic calibration could be performed. For each calibration result obtained from the distortion stage intrinsic calibration was performed. The camera was set, using the LifeCam software, to the same settings as those used for the distortion calibration of that set.

An image was taken at 2 mm intervals between -30 and 30 mm, resulting in 31 images. Once an image was taken the system presented a white line. This was positioned using the keyboard to match the grid lines, first the horizontal then the vertical. The arrow keys were used to move and rotate the line. To enable more accurate and faster calibration the *calib_param* program was modified to allow the step size of the movement and rotation to be reduced or increased. The line marking process is shown in Figure 6.11 below.

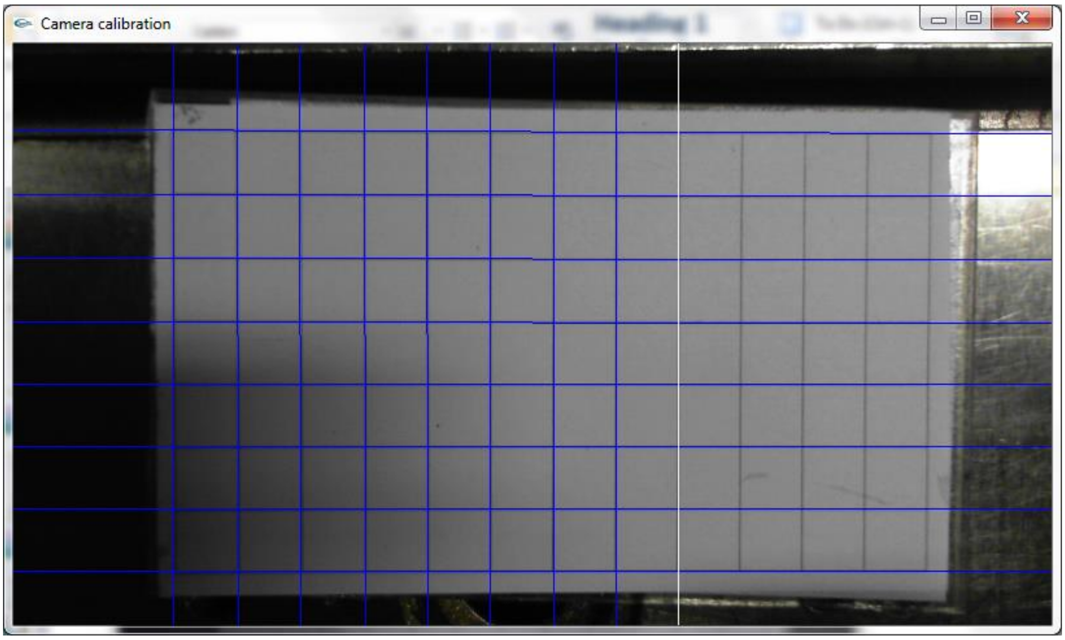


Figure 6.11: Intrinsic parameter camera calibration grid marking.

The consistency of the calibrations is discussed below. For each calibration the camera was focused to the same distance and the other settings were kept constant. Therefore, the five fully fixed calibrations discussed below are expected to be very similar.

Image centre form two of the five intrinsic parameters used by ARToolKit. The values are given in pixels and are expected to be approximately equal to half the resolution. Therefore, 400 and 224 pixels in the x and y-axes respectively.

The image centre means were found to be 400.06 and 228.44 pixels for x and y-axes respectively, which is within the expected range. The difference of each calibration from these mean values is shown below in Figure 6.12.

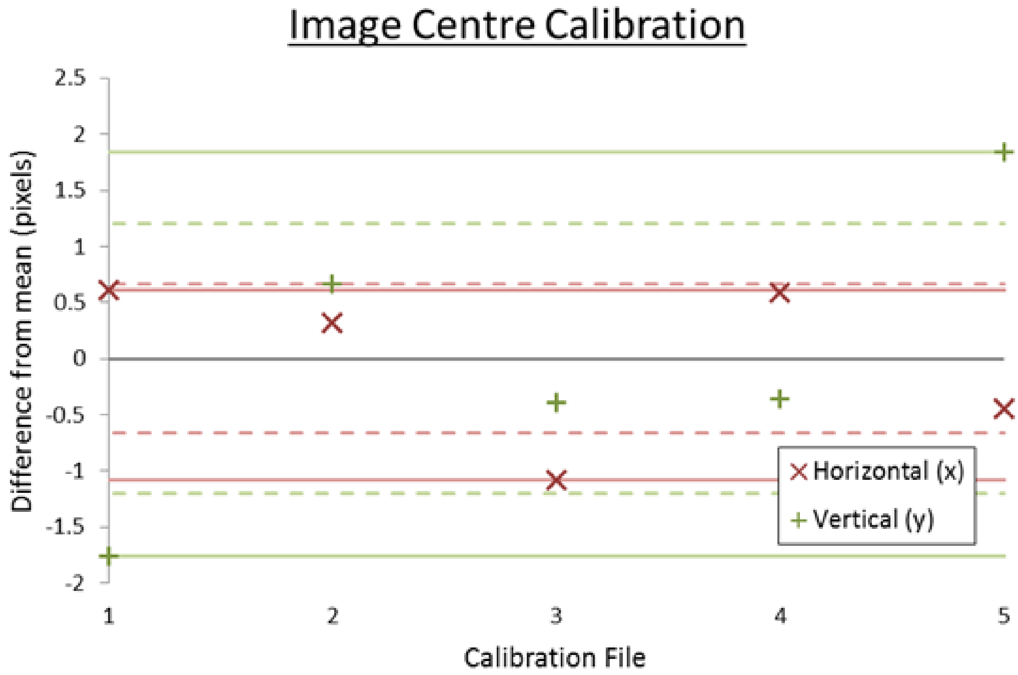


Figure 6.12: Image centre results for intrinsic parameter calibration. The range is shown by the solid lines while dashed lines show the standard deviation.

The image centre results are seen to have relatively similar results with a range of 1.69 and 3.60 pixels for the x and y-axis respectively. Standard deviation values are 0.66 and 1.20 pixels respectively. In terms of percentage the range shows a percentage difference of 0.42 and 1.58% for the two axes respectively.

The next two intrinsic parameters are the scaled focal lengths in the x and y-axes. The focal length of a camera determines the image size of an object in accordance to Equations 6.4 and 6.5 respectively.

$$u = \frac{Xf}{Z} \tag{6.4}$$

$$v = \frac{Yf}{Z} \tag{6.5}$$

Solving for the focal length f it is possible to derive an estimate based upon the object postion (X, Y, Z) and its image (u, v). Measurments performed at a separation of 90 mm produced a focal length estimate of 589 and 548 pixels for the x and y-axis respectively.

The five calibrations produced mean focal lengths of 616.55 and 616.60 pixels for the x and y-axis respectively, shown in Figure 6.13 below.

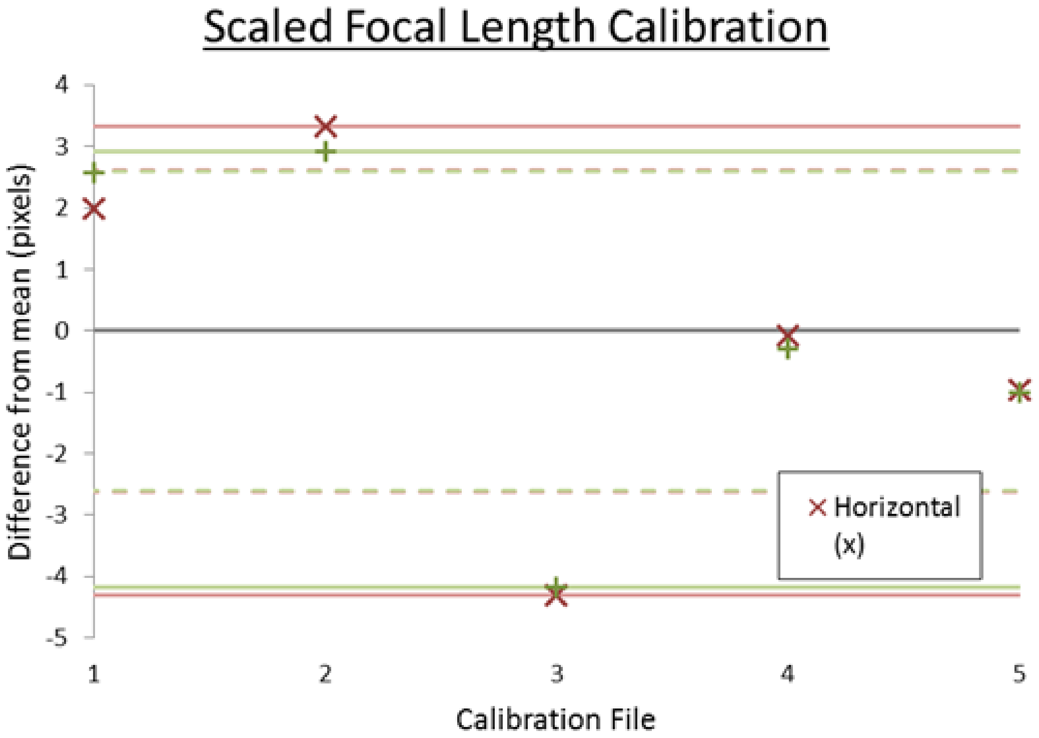


Figure 6.13: Scaled focal length results for intrinsic parameter calibration. The range is shown by the solid lines while dashed lines show the standard deviation.

Again, Figure 6.13 shows all calibrations produce relatively similar results. The x and y-axis focal lengths produced a range of 7.63 and 7.10 pixels, and a standard deviation of 2.62 and 2.60 pixels, respectively. The range translates to a percentage difference of 1.24 and 1.15%, respectively.

The final intrinsic parameter is the skew factor. This factor is often assumed and set to be zero. However, the calibration method used by ARToolKit calculated a value. Unlike the previous intrinsic parameters it is not possible to formulate an estimate, only to state that the skew value should be close to zero. The mean skew was calculated to be 1.03°, as shown in Figure 6.14.

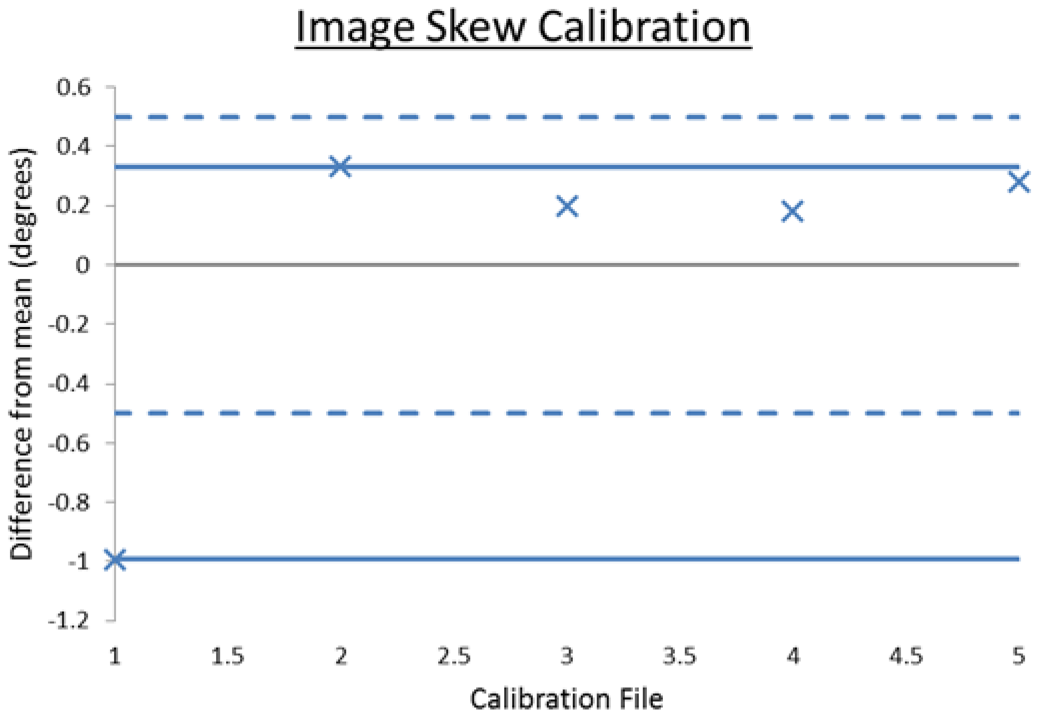


Figure 6.14: Image skew results for intrinsic parameter calibration. The range is shown by the solid lines while dashed lines show the standard deviation.

The five calibrations produced a range and standard deviation of 1.33 and 0.50, respectively. With the low values of skew this range translates to a percentage range of 128.4%. It may be seen from Figure 6.14 that the first calibration file produced a skew much larger than the other four calibrations, resulting in this large percentage range. While the range appears larger than those of the other parameters it was unlikely this result invalidated the calibrations.

As it was not practical to perform experiments with repeats for each of the five thecalibrations the median calibration was instead used. It was found that the fourth calibration provided the median focal lengths and image centres, in addition to the x-axis distortion centre. Therefore, calibration four was used as the median calibration.

6.4.3 Linear Precision Analysis

The first experiments investigated the ability of the system to determine the camera to marker positional relation. It was originally designed to directly determine the accuracy of the system. However, it was not possible to accurately measure the physical location of the origin of the system. Therefore a ground truth value for camera to marker position could not be accurately defined. As such, the experiment defined the accuracy as the ability of the system to correctly determine the change in position.

6.4.3.1 Methodology

The camera was mounted and aligned in the Instron as with the calibration procedure above. A 20 mm marker was placed on the planar test bed such that the optical axis of the camera intercepted the centre of the marker. The fine adjustment of the Intron was set to its minimum value of -30.0 mm. The coarse adjustment was then set such that the camera was 120 mm from the marker. The setup is illustrated by Figure 6.15 below.

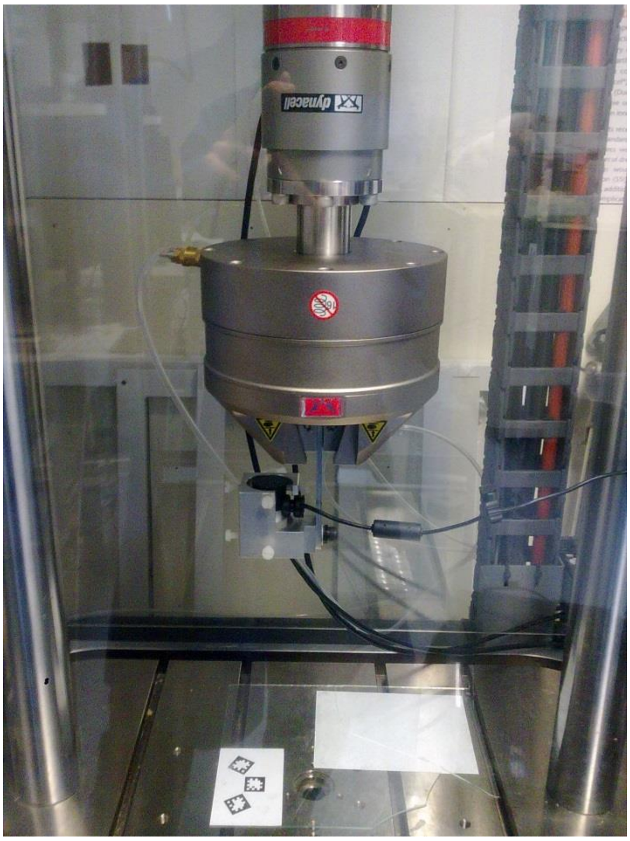


Figure 6.15: Experimental setup for linear precision analysis

Using the system, frames were captured and the pose of the marker calculated. For each frame the pose was written to a human-readable file using a basic file stream. This was repeated until one hundred poses had been successfully captured. Once complete the fine adjustment of the Instron was used to increase the separation by 1 mm, and another 100 poses were captured. This process was repeated until the fine adjustment of the Intron reached its maximum range of 30.0 mm. The separation between the camera and test bed was then measured. The fine adjustment was then set to its minimum value of -30.0 mm. After which the coarse adjustment was used to return the separation to the previously measured value. Again, 100 poses were captured for each 1 mm interval across the full range of the fine adjustment. This was once more repeated for a final coarse adjustment, providing a total range of 180 mm.

6.4.3.2 Results

For each separation the mean value and range of the three cardinal axes were calculated from the 100 samples. A linear regression was performed on each of the three data sections – resulting from course adjustment of the Instron. The resulting linear equations were used to align the data sets, and any overlapping points were removed.

The corresponding ground truth value was subtracted from each sample. The resulting errors are plotted against the Z-displacement in Figure 6.16 below. The standard deviation of each separation is shown via the colour scale. It is noted that the deviation and error are extremely large for some separations. These errors were likely the result of marker inversion and could be programmatically prevented, as discussed in Section 6.4.3.3 below. As these outliers skew the remaining results, Figure 6.17 shows the same data set with the axes and colour scale adjusted to exclude them.

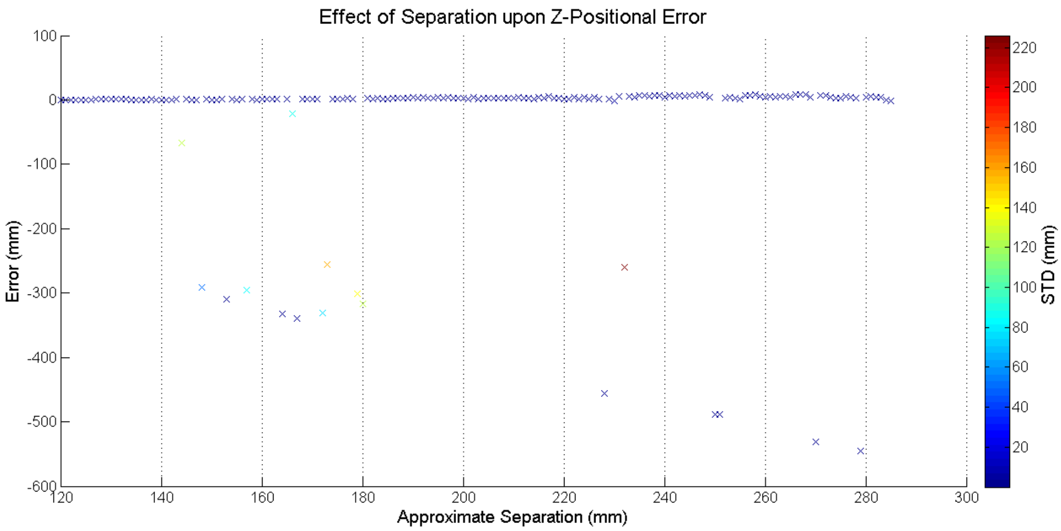


Figure 6.16: Effect of separation upon the error and standard deviation of Z-positional data.

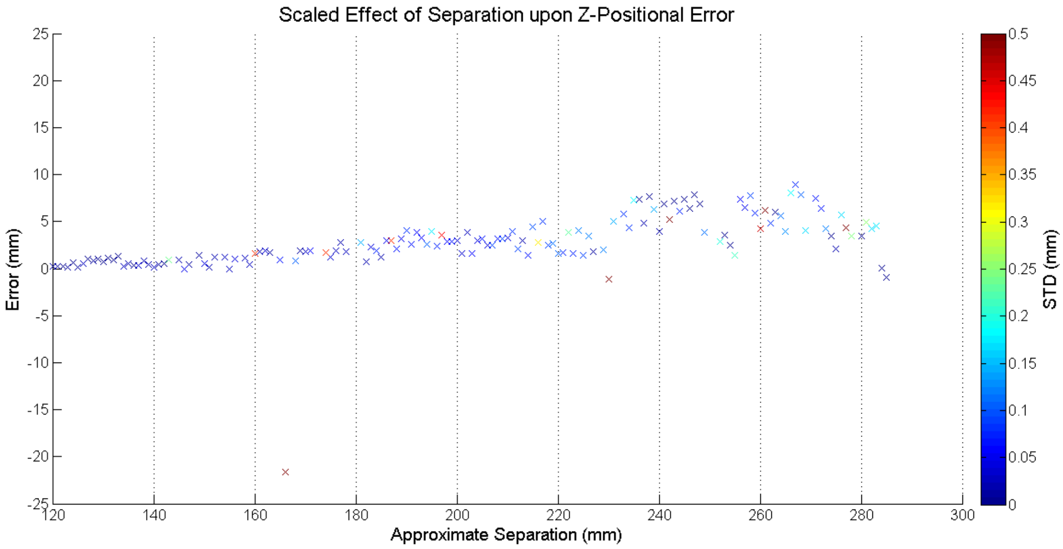


Figure 6.17: Effect of separation upon the error and standard deviation of Z-positional data scaled to remove disproportionality large errors.

Figure 6.18 below shows the results for the X and Y-positional data sets. Again the colour scale has been truncated to prevent the over dominance of the erroneous ranges.

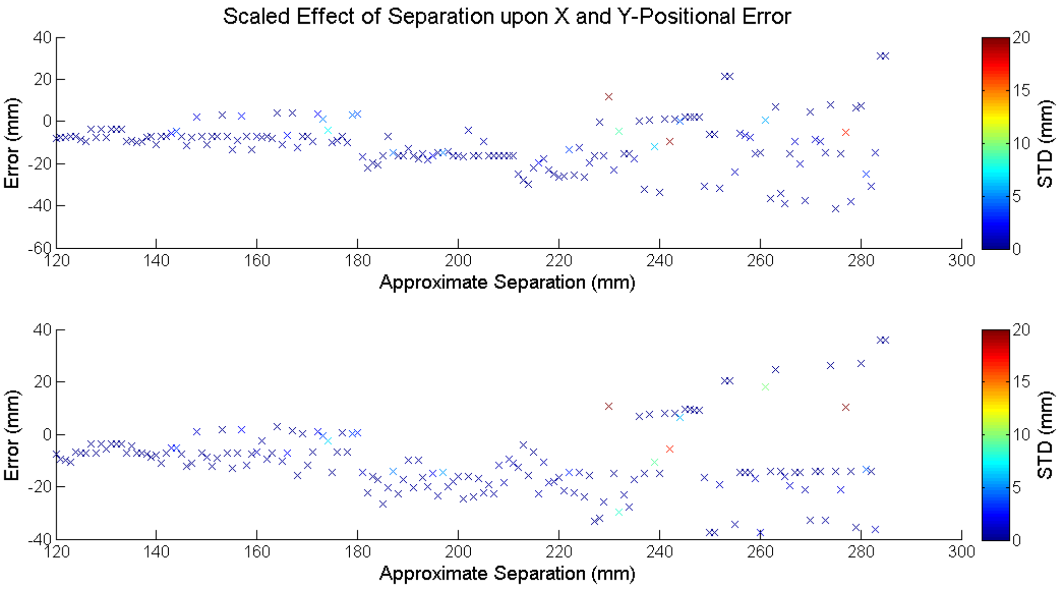


Figure 6.18: Effect of separation upon the error and standard deviation of X and Y-positional data scaled to remove disproportionality large errors.

As it was not possible to accurately measure the initial separation between the marker and camera system centre the accuracy of the system was measured based upon the incremental increase. A 1 mm step was performed each time. Therefore, the system should report a 1 mm increase, irrespective of any initial offset error. With the data set aligned the difference between successive measurements was calculated. As discussed above some signals produced an extremely large error. These errors could be characterised and in theory filtered from the system. Therefore, to prevent them from excessively skewing the data and concealing much of the detail they were removed at this stage. A running RMSE was calculated for the remaining data set. This is shown for Z-positional data in Figure 6.19 and X and Y-positional data in Figure 6.20 below.

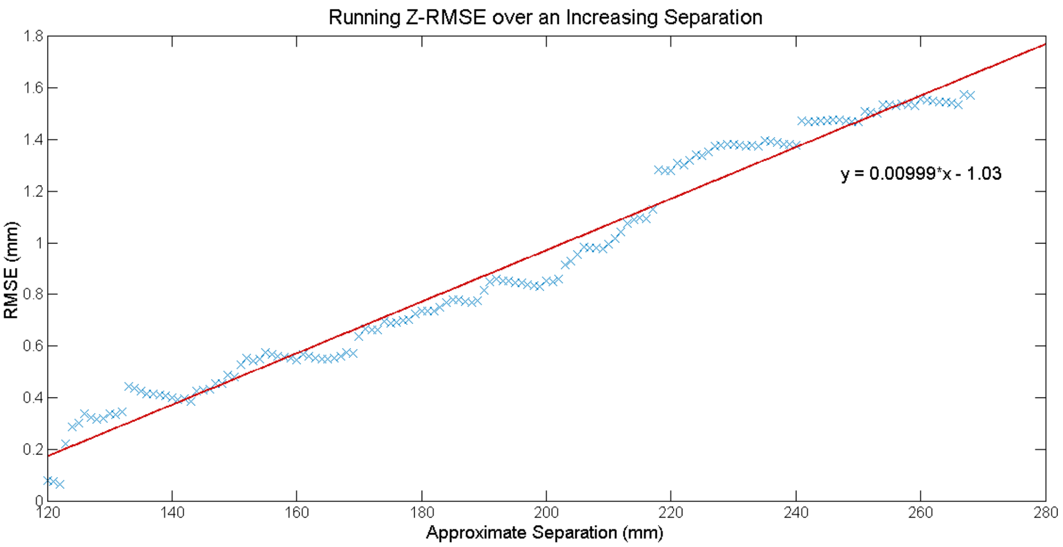


Figure 6.19: Running RMSE for Z-positional data over an increasing separation

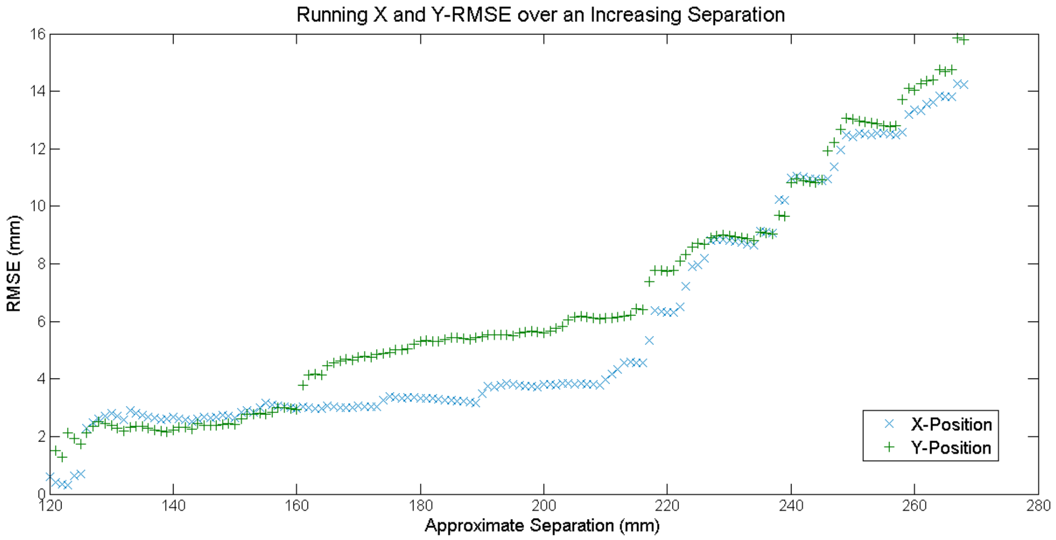


Figure 6.20: Running RMSE for X and Y-positional data over an increasing separation

To help characterise the distribution of the data a density plot was produced, Figure 6.21. This indicates, by way of colour scale, the density of samples that returned the same value of Z-position. Deep blue indicates a rarely reported value, while red indicates a common value. To reduce the effect of the large errors seen in Figure 6.16 the absolute value of the Z-position was taken.

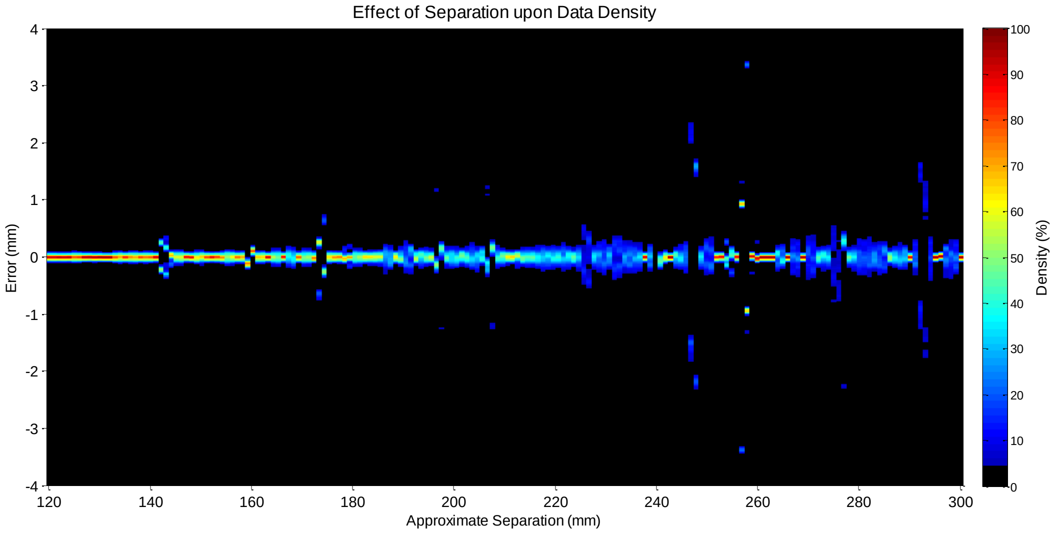


Figure 6.21: Data density plot of Z-positional data

It may be seen in Figure 6.21 that most separation sets showed the expected roughly bell shaped distribution, with a common central point and increasingly rare peripheries. Indeed, Figure 6.22 below plots several samples and confirms their normal distribution – Jarque-Bera test to 99% significance.

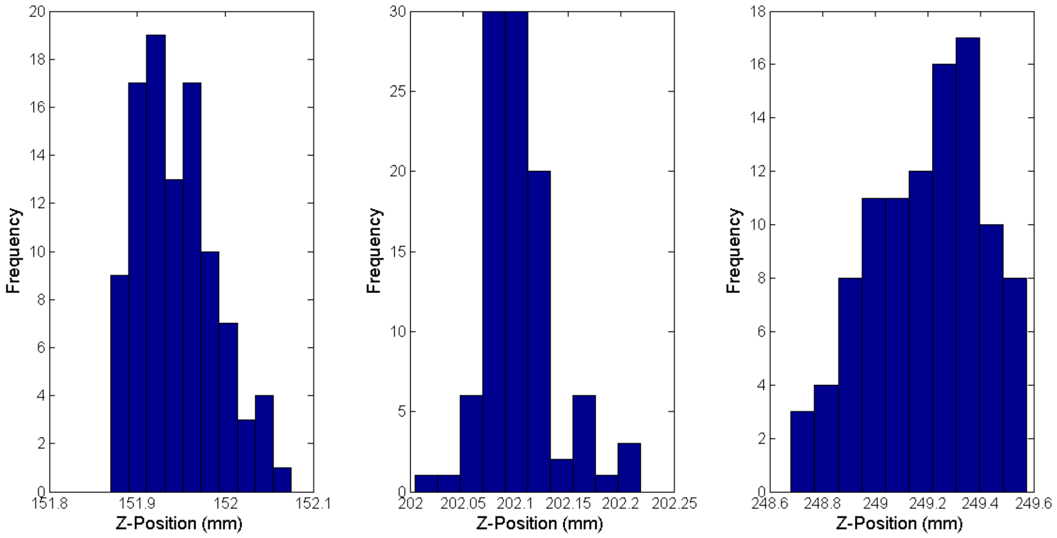


Figure 6.22: Standard single-peak Z-positional distribution examples

However, Figure 6.21 shows that not all points obey a normal distribution. Those that do not, tend to show two peak values, either separate or joined, often with one peak more densely populated that the other, as shown in Figure 6.23 below.

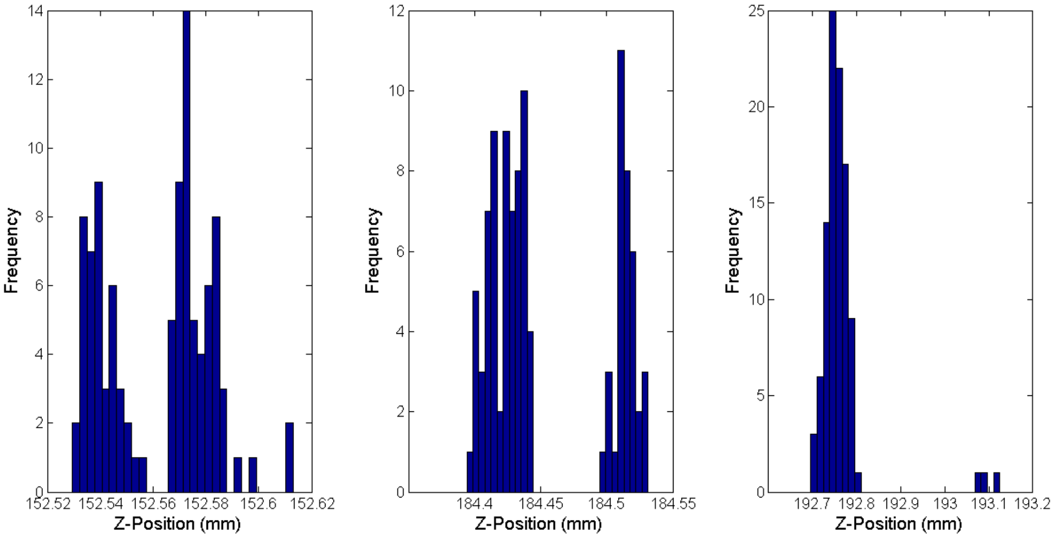


Figure 6.23: Dual-peak Z-positional distribution examples

A final distribution type is found for those separations returning disproportionally large errors. Several examples are shown below in Figure 6.24.

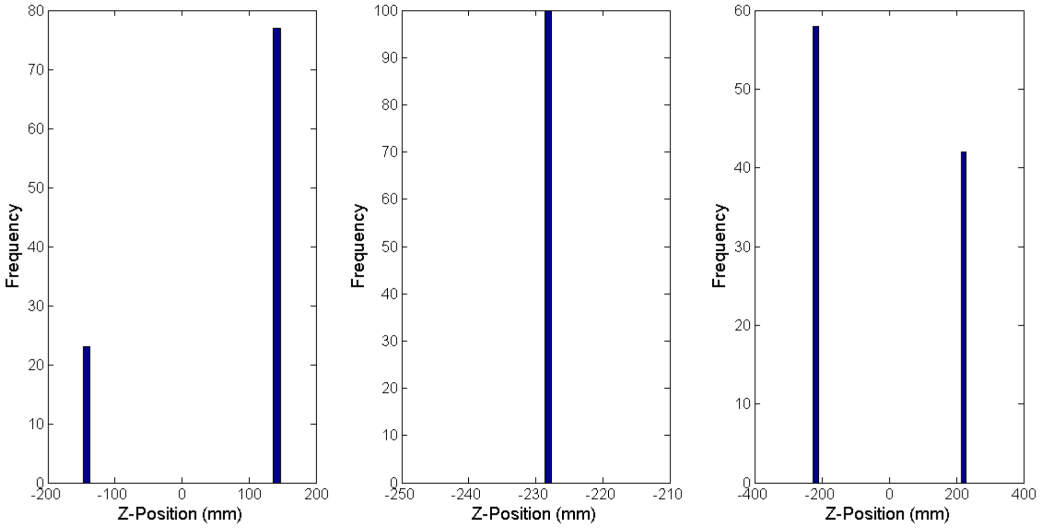


Figure 6.24: Erroneous Z-positional distribution examples

6.4.3.3 Discussion

Firstly, the disproportionately large errors seen in Figure 6.16 are discussed. These outliers were a hundred times the order of the standard system errors, and always negative. It is noted that all three data regions feature these outliers, and the high quantity within the first region suggest the cause is not related to the standard distance dependent errors.

The individual 100 samples of the erroneous separations were inspected. Examples of these are shown above in histogram form in Figure 6.24. Some of the erroneous samples, shown on the left and right of the figure, oscillated between two values. It was found that the positive value roughly agreed with the expected value and that the negative value was approximately the inverse of the positive. Therefore, the conclusion was made that the system occasionally inverted the pose of the marker, and assumed it was viewing the marker from behind. It is noted that the erroneous signals show very little noise. As such it is suspected that the geometric stage of pose estimation failed, by returning an inverted result. This poor estimate caused the iterative approach to fail to converge to a true answer.

Above it is mentioned that these outliers were removed from analysis due to the possibility of real-time filtering. As it is known that the camera cannot view the marker from behind any poses with negative Z values could simply be rejected. Alternatively, a more complex approach could invert the geometric solution before calling the iterative stage.

These errors aside, Figure 6.17 shows the expected relationship. As separation increases, so too does positional error and spread. Two factors affect the accuracy and precision of the system with range. Firstly, as discussed during camera selection in Section 5.2.2.1, the resolution of the camera dictates a physical resolution, which is how many pixels form per millimetre of the image. This physical resolution decreases with increasing separation as the FOV is increased. The ARToolKit library attempted to minimise this effect through subpixel corner detection. However, the resolution of the edges that define the lines used to find the corners is still decreased. Therefore, the accuracy of corner detection is still decreased with increasing separation. The second factor increasing the error with separation is the rotational error. The experiment reports the position of the camera relative to the marker. At 100 mm separation a 10° error in a single planar parallel axis of the rotational pose estimation would result in a Z-positional error of 1.5 mm, while the same rotational error at 200 mm separation would induce a 3.0 mm error. The presence of this factor is further supported by Figure 6.18 which shows much larger errors at larger separations for the X and Y-axes than the Z-axis. The same 10° rotational error would cause 17.4 and 34.7 mm errors in either the X or Y-axis at 100 and 200 mm separations, respectively.

To investigate the accuracy of the system, the change in system readings were investigated for each separation. Figure 6.19 shows the running Z-positional RMSE at each separation. As expected the RMSE error increases with separation. The error was found to follow a linear regression with a gradient of 0.01 and R-square value of 0.97. Therefore, increasing the separation from 120 mm to 240 mm sees an eight fold increase in error.

Chapter 3 concludes that an accuracy of 1 mm was required by the system. At separations below 200 mm Figure 6.19 indicates that the system could provide this accuracy in the Z-axis. This separation would be achievable during resection due to the proposed tool mounted design of the system. Unfortunately, Figure 6.20 confirms that the X and Y-axes are considerably less accurate. Both Figure 6.18 and Figure 6.20 show a sizeable error even at the minimum separation. It is suspected that at least some of this error is attributed to an initial error in the markers placement. The experimental setup ensured the marker was aligned with the optical axis of the camera. However, it was only assumed that the marker axis aligned with the camera. As the test bed of the Instron is perpendicular to the axis of motion and the marker was mounted to extremely flat float glass this was expected to be the case. However, a robust linear regression of the X and Y data sets indicate an initial offset of 2.1 and 7.1 mm respectively. This suggests the marker axes may not have been ideally aligned.

However, Figure 6.20 shows the X and Y-positional errors to increase considerably, while Figure 6.18 shows the errors to be both positive and negative at increased separations. These errors could only be partially accounted for by an initial offset.

In conclusion this experiment suggests the ARToolKit system would struggle to meet the accuracy and precision requirements of a guided UKA system. However, the proposed system makes use of a tool mounted design. These results showed that the system performs considerably better at the close ranges that this would facilitate.

Therefore, further testing was performed to test the performance of the system at a task more closely emulating its final function.

6.4.4 Probe Accuracy Analysis

The following series of tests were designed to test the probing accuracy of the system. These tests would determine the suitability of the system for guided UKA procedures, where many stages would utilise probe or probe-like tool tracking.

The experiments involved using the system to probe a known geometry that was tracked by the system. This allowed a comparison between the measurements produced by the system and the known ground truth data.

6.4.4.1 Materials

The design of the probe used for this series of experiments was discussed as part of the system design in Section 6.3.1. The design of the known geometry testing object follows.

To test the probing accuracy of the system the testing object had to meet the requirements listed by Table 6.1.

Testing Object Requirements
Provide a range of known points
Allow the probe to be firmly and consistently seated in these points
Allow a marker to be fixed relative to these points
Be manufactured to a high degree of accuracy

Table 6.1: Requirements of probing accuracy testing object

Any probe experiment would be performed using a probe of length between 100 and 175 mm. At these distances the camera provided a vertical FOV of approximately 80 and 140 mm respectively, and a horizontal FOV of 140 and 200 mm. To ensure the testing object could be tracked throughout the experiment the dimensions of the object were based upon the smaller of these measurements. With a central marker an object width and height of 100 and 50 mm respectively would help ensure the marker remained visible while probing all points. This size still provided a good range of points, exceeding the approximately 50x30x30 mm resection volume required to fit a large femoral component.

The final object, based on similar targets reported in litirature, was machined from 6082 T6 aluminum using computer aided manufacturing (CAM) (Ritter et al. 2007; Sinram et al. 2002). As shown in Figure 6.25 it provide 108 fixed probing points across a range of topologies, relative to a central 20 mm marker.

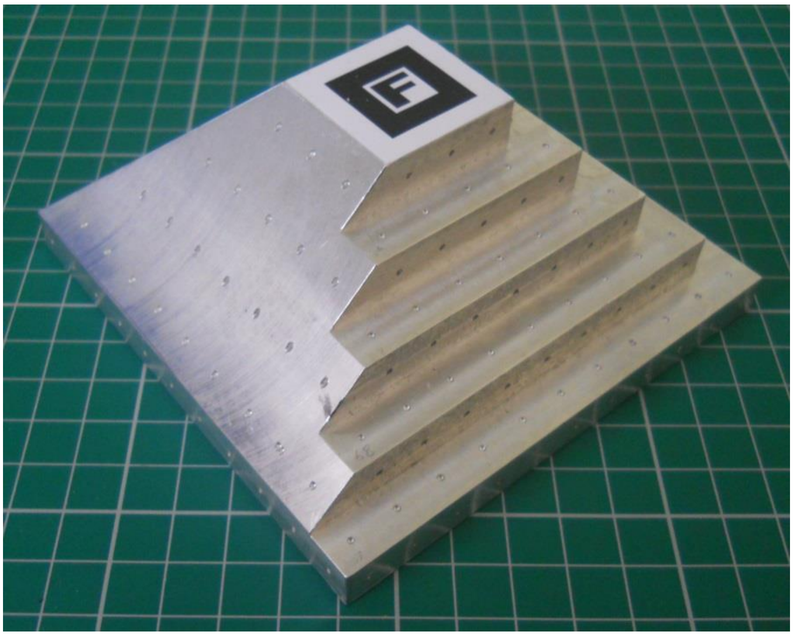


Figure 6.25: Final known geometry testing target.

When probing the object the system reported the position of the probe tip relative to the marker. Therefore, to accurately compare the measurements to the ground truth of the object it was fundamental to ensure the marker was correctly positioned and orientated upon the object. The marker was affixed using water soluble PVA glue, so that it could be cleanly removed if required. Several methods were explored to position the marker. Best results were achieved with the marker shown in Figure 6.26 below. The extra marks were aligned with several of the divots on the object. Once the glue had cured the excess sections were removed.

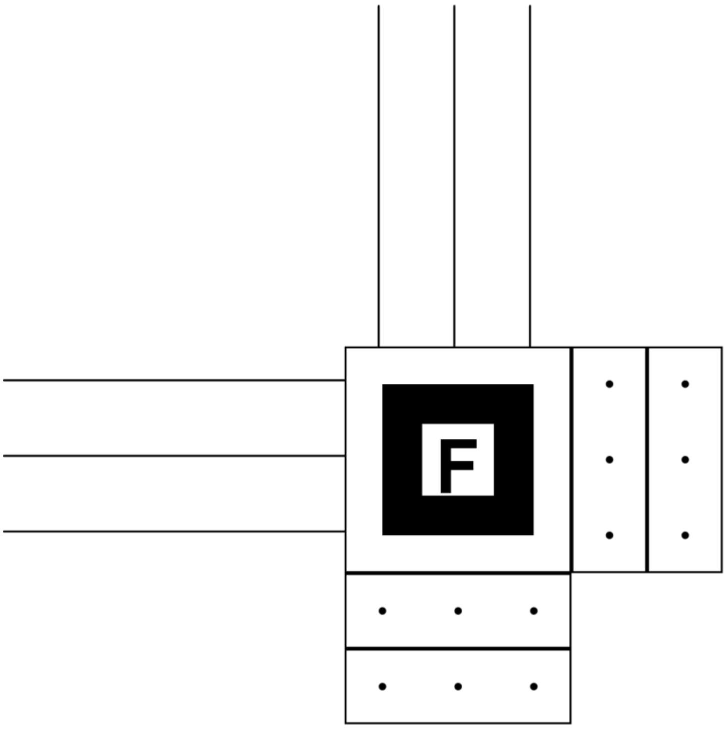


Figure 6.26: Probe test marker alignment guides

6.4.4.2 Methodology

Additional software was developed to capture the position of the probe tip, relative to the test marker, to file. To prevent errors during probe testing an augmented overlay was designed such that the measurement points were shown. Unmeasured points were coloured blue, while the current divot was coloured green. Points that had been previously measured were coloured red, as shown in Figure 6.27. This augmentation was found to be extremely useful during testing, and as discussed in Chapter 3, helped motivate the adoption of augmented reality as a key research focus.

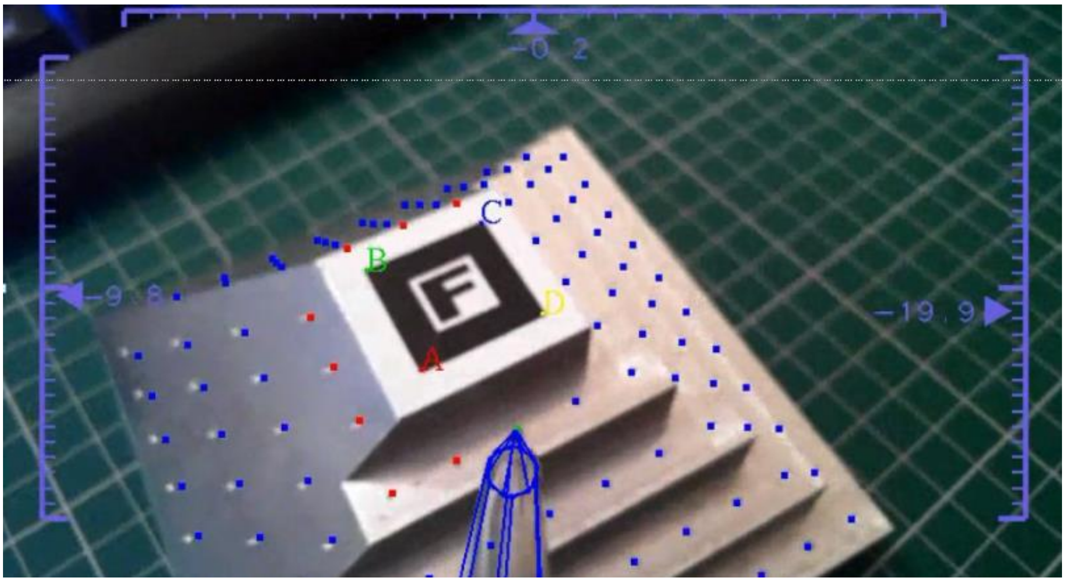


Figure 6.27: Augmented guidance during object probing tests.

The LifeCam software was used to set the focus of the camera to that used during calibration. The probe length was set to 120 mm and the median calibration file loaded. The position of the tip was calibrated using a planar target, as detailed in Appendix A1.1. Each vertical and sloped point was probed, starting from the sloped point at the top right of the marker and progressing in a counter clockwise direction. These captures were guided by the augmented overlay as discussed above. A total of 108 measurement points were probed, collecting 100 samples of each. The tool was orientated such that the marker was in the centre of the image, with minimal rotation about the optical axis. This procedure was repeated five times.

6.4.4.3 Results

Figure 6.28 shows the standard camera position and orientation during probing of the target.

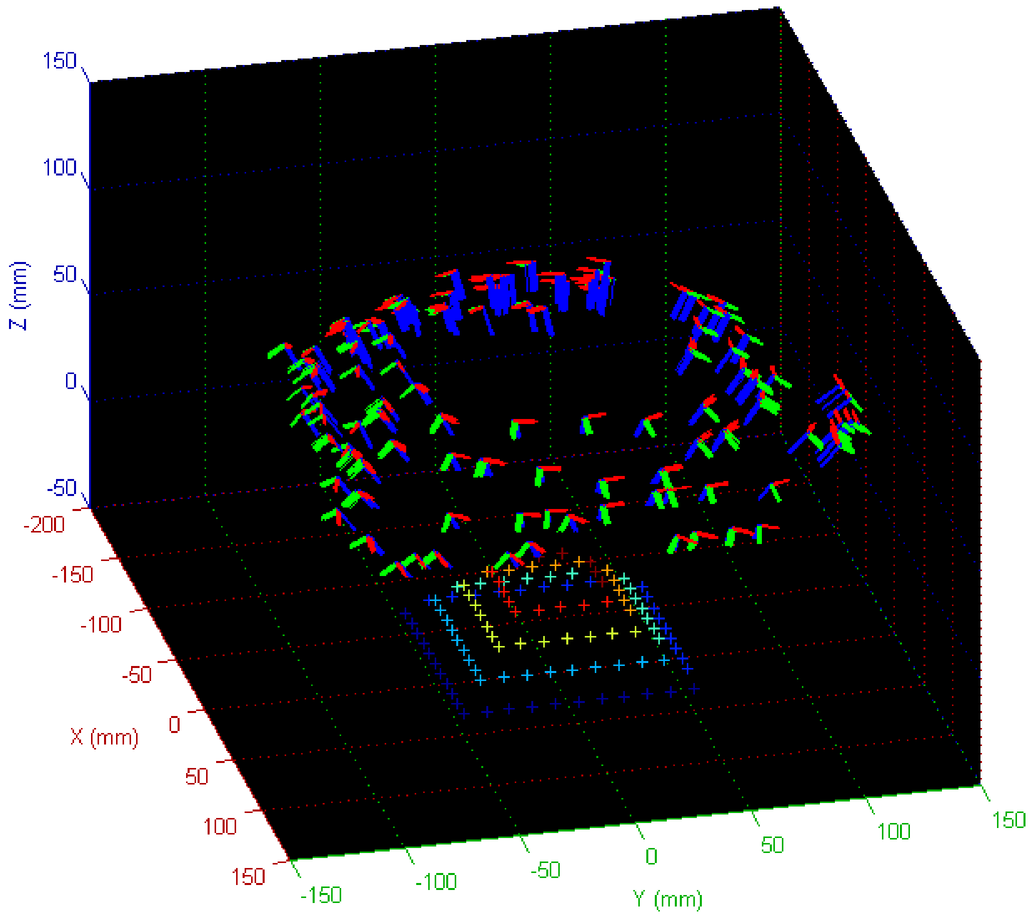


Figure 6.28: Tool positioning during standard probing experiment.

A Matlab script was developed to compare each measurement to its ground truth value. The mean error was calculated for each divot in millimetres. This was calculated in the three orthogonal axes separately, and for the total Euclidian difference. Mean errors were used over RMSE to preserve the direction of the error. These results are shown spatially in Figure 6.29. It is noted that the central 3x3 region and the diagonal towards the top right indicate object regions without measurement points.

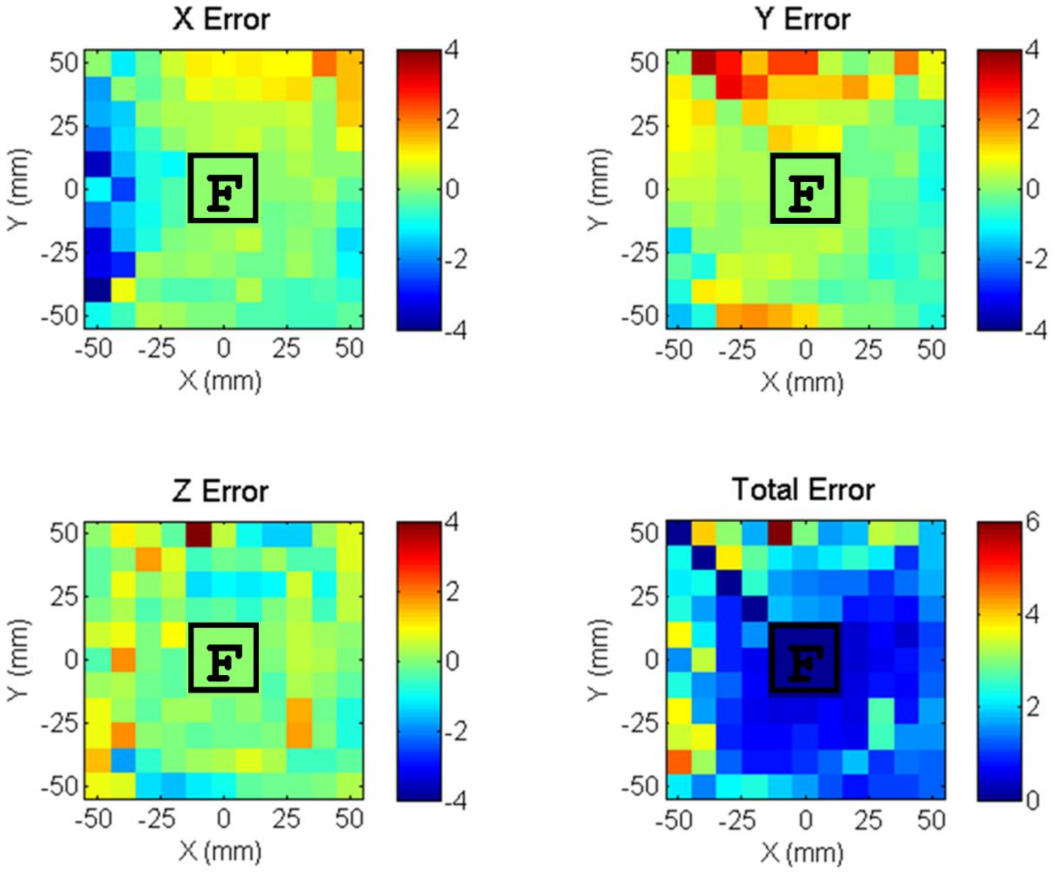


Figure 6.29: Mean error probing results.

RMSE were also calculated for each divot. These were averaged across the full data set to produce mean; x, y, z, and total RMSEs. The mean RMSEs for four of the five repeats were again averaged and the results are shown below in Table 6.2. One data set was excluded from analysis as it was found the probe had shifted, relative to the camera, during measurement.

X (mm)	Y (mm)	Z (mm)	Total (mm)
0.71±0.05	0.68±0.04	0.99±0.26	1.54±0.15

Table 6.2: Mean RMSE and standard deviation for probing tests

Similarly, range values were calculated for each divot in the three cardinal axes and in total. These are shown below in Figure 6.30 for the same data set as Figure 6.29 . Again these results were averaged across the full data set and four repeats in turn. The results are shown in Table 6.3. It is noted the standard deviations are much larger for precision than accuracy. Inspection of the data sets revealed that the range values pictured in Figure 6.30 were larger than the average.

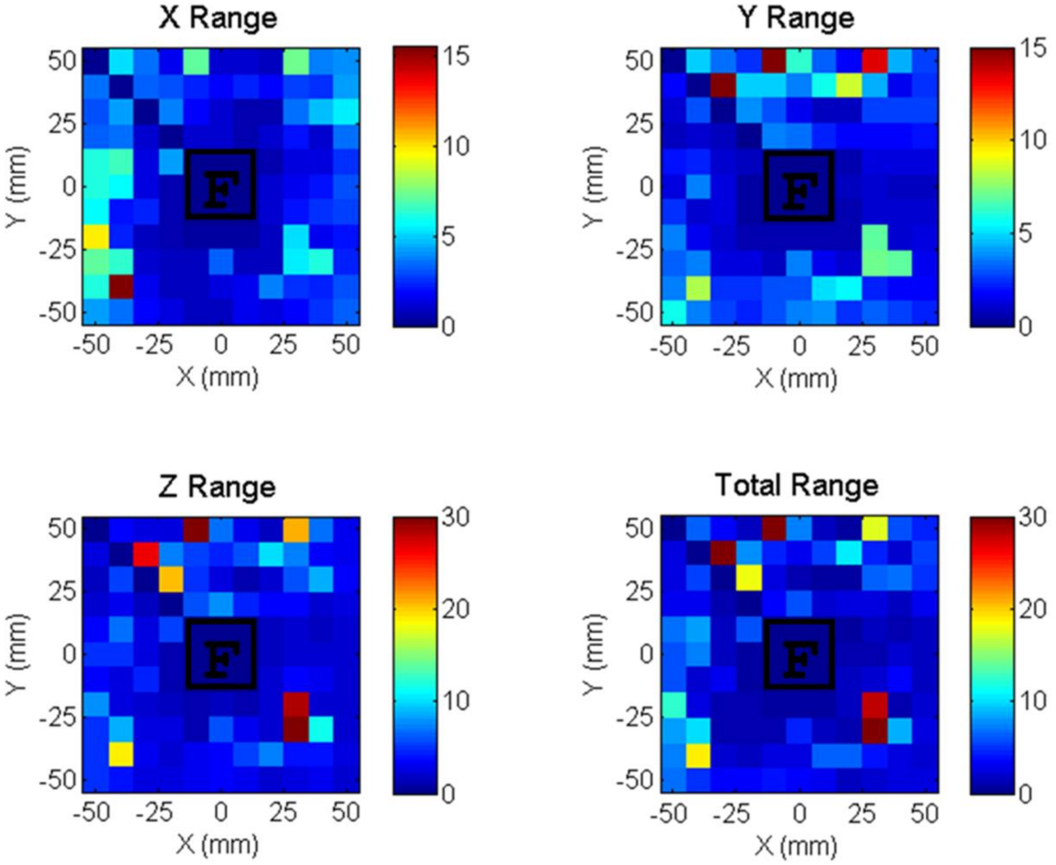


Figure 6.30: Range probing results

X (mm)	Y (mm)	Z (mm)	Total (mm)
1.20±0.73	1.29±0.90	2.28±1.63	2.07±1.64

Table 6.3: Mean range for probing test

6.4.4.4 Discussion

Initial testing showed that it was difficult to ensure the target marker was correctly aligned. Misalignment would result in large errors as it was the position of the marker that related the tool to the ground truth of the object. Therefore, the error patterns seen above in Figure 6.29, and the remaining error patterns, were investigated for evidence of marker misalignment.

Three types of misalignment or positioning were considered: translational, where the marker was not correctly centred; normal rotational, where the marker was rotated about its normal; and non-normal rotational, where the marker was rotated about an axis other than its normal, most likely resulting from inaccurate machining of the target. The effects of each of these misalignments are shown below in Figure 6.31.

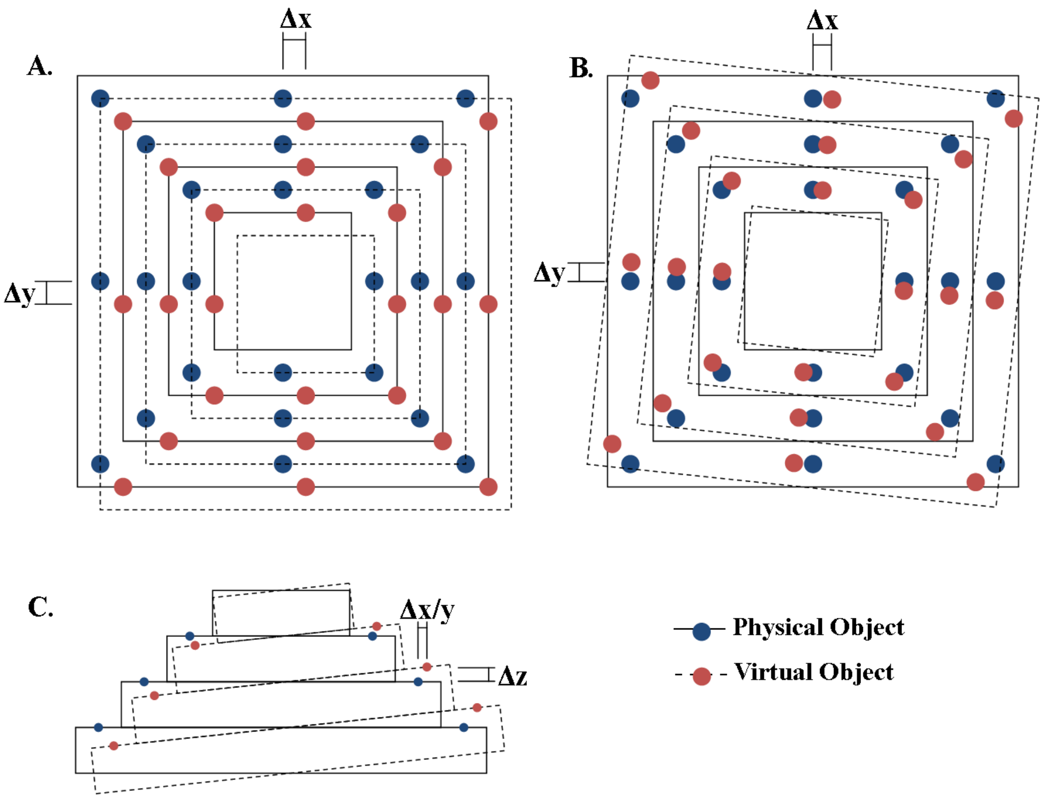


Figure 6.31: Possible effects of marker misalignment: A) Translation error, B) Normal rotational error, C) Non-normal rotational error.

No data sets indicated a translational error. As seen in Figure 6.31(A) this would have caused all divots to present an error in one direction. As seen in the example data of Figure 6.29 this was not the case. Normal and non-normal rotational errors would result in an equal and opposite error as shown in Figure 6.31. Evidence for non-normal rotational errors was not found as no data set produced this equal and opposite error pattern in the Z-axis. However, the data set shown in Figure 6.29 does present some evidence of normal rotational error. It is noted from the figure that the X-axis error figure shows roughly equal and opposite errors in the top and bottom regions. Likewise, the Y-axis error figure shows errors in the left and right regions. Looking at Figure 6.31 above it may be seen that this is the expected error pattern for a normal rotational error. Furthermore, the errors appear to increase away from the centre, again as predicted. This data set suggests a normal rotational error in marker alignment. However, all reported experiments were performed with the same marker placement, and only the presented data set in Figure 6.29 produced this pattern. It is unlikely that if such clear errors caused by marker misalignment, they would not be visible in the remaining data sets. Therefore, it was concluded that the marker was correctly aligned.

A clear pattern seen in all data sets was an increase in error and range with increasing separation from the target centre. Due to the camera being probe mounted and the design of the target it was noted that the camera-marker separation did not change substantially while probing different divots. Therefore, this increase in error was unlikely caused solely by the error relationship discussed in Section 6.4.3 above. The increase in error was likely the result of rotational pose estimation error. As may be seen in Figure 6.31, the effects of a rotational error increase away from the centre. Therefore, the same pose estimation error would induce a larger positional error while probing the distal points.

Several large errors may be seen for the most distal measurements, particularly the outer most corner divots. During the capture of these divots it was necessary to hold the probe at a very low angle to ensure the marker was within the camera image. It was proposed that low angles may induce increased error through one or both of two mechanisms. Firstly, the pose estimation provided by the ARToolKit may be less accurate at steep angles. Secondly, as the marker was imaged towards the edge of the image it is possible that it was more susceptible to image distortion, and therefore provided a less accurate pose estimate. Further experiments would be required to determine the cause of these increased errors.

Theoretically these low angles, and thus high errors, could be avoided by increasing the probe length and therefore effectively increasing the FOV of the camera. However, as shown by Section 6.4.3, increasing the camera-marker separation reduces the accuracy of the system. Figure 6.19 and Figure 6.20 of Section 6.4.3 suggest that camera-marker separation may be increased by 40 mm without causing significant increase to the error. The Z error was shown to increase with separation at 0.01 mm error per mm of separation. However, the X and Y-axes were shown to follow an exponential relation. Therefore, any further increase in probe length would rapidly increase the error beyond that caused by the low angles required due to limited FOV.

Looking now at the general errors a difference is noted between the error distributions of the X and Y-axes and the Z-axis. The X and Y-axes show clear grouping of consistent errors, as previously described during marker alignment discussed above. However, the Z-axis presents much weaker grouping. It may be seen from Figure 6.29 that positive and negative errors co-occupy the same target regions. This distribution is less evident in the remaining data sets, with the Z-error tending to be predominately negative. However the Z-axial data always presents less grouping than the remaining two axes. A pose estimation error may easily result in either positive or negative Z-axial error, as demonstrated by Figure 6.31(C). However, as shown by the figure this would also induce a smaller error in either the X or Y-axes. Correlations are observed between errors in the Z-axis and the X and Y-axes. For example the notable error of approximately -2 mm in the Z-axis at (-40, -40) is also present as large positive errors in both the X and Y-axes. Several examples are seen throughout the data sets. It is therefore suspected that the errors observed are a result of pose estimation error. However, it is likely a secondary source of error is masking the relationship by inducing larger errors in the X and Y-axes data sets.

While all data sets present error groupings in the X and Y-axes, the location of these groupings is not consistent. It is suspected that this inconsistence was the result of probe tip calibration. Appendix A1.1 describes how the system calibrates the offset of the tool tip relative to the origin of the camera system based upon the physical position of the tip and the estimated marker pose. Ideally, the tip position is calculated correctly and the errors observed in Figure 6.29 are the result of pose estimation errors at these divots. However, the calibration of the tip is just as susceptible to error as the divot measurements. Therefore, the errors observed in Figure 6.29 are the combination of the pose estimation error at that divot and the tip calibration error. It is proposed that this combination of errors accounts for the somewhat inconsistent distribution of error groupings across data sets.

In a process that should have been performed before experimentation, an attempt was made to validate the accuracy of the known geometry target. The geometry of the target and spacing of the divots was measured using digital callipers and confirmed to be accurate to 0.05 mm. However, the depth of the divots was found to be inconsistent. A sample of points were measured and the depths found to present a range of values of the order of 1 mm. An attempt was made to measure each depth and factor these values into the results above. However, measurement of the divots, particularly those of the sloped side proved inconsistent. As such it was decided instead to produce a new testing geometry, a process that shall be discussed in Section 7.3.3.1.

While the target was found to be inaccurate this did not entirely invalidate the presented experiments. As may be seen from Figure 6.29, several errors exceeded those measured from the target.

In conclusion the mean RMS accuracy reported by these experiments of 1.54 mm failed to meet the 1mm accuracy defined as a requirement in Chapters 3. However, as shall be discussed below, it was proposed that these results may be improved upon.

6.4.5 Hip Centre Analysis

A brief analysis was performed to determine the performance of the hip centre estimation method.

6.4.5.1 Materials and Methods

To simulate the ball and socket joint of the hip the 150 mm probe was used in conjunction with a planar printed target with 20 mm marker. The target represented the pelvis with the marker equivalent to the bed marker. An additional 20 mm was attached to the distal end of the probe, to similar the femur of the joint. The tip of the probe was placed in on the target at (40, 40, 0) mm displacement from the centre of the marker. The femur was rotated and 100 samples, over 1000 frames, were captured. These were then analysed using a Matlab equivalent of the hip centre estimation method.

6.4.5.2 Results

Figure 6.32 below shows the 100 samples of one of the ten randomly generated points transformed into the coordinate system of the bed marker. Every tenth point is jointed to the expected centre and estimated centre by blue and red lines respectively.

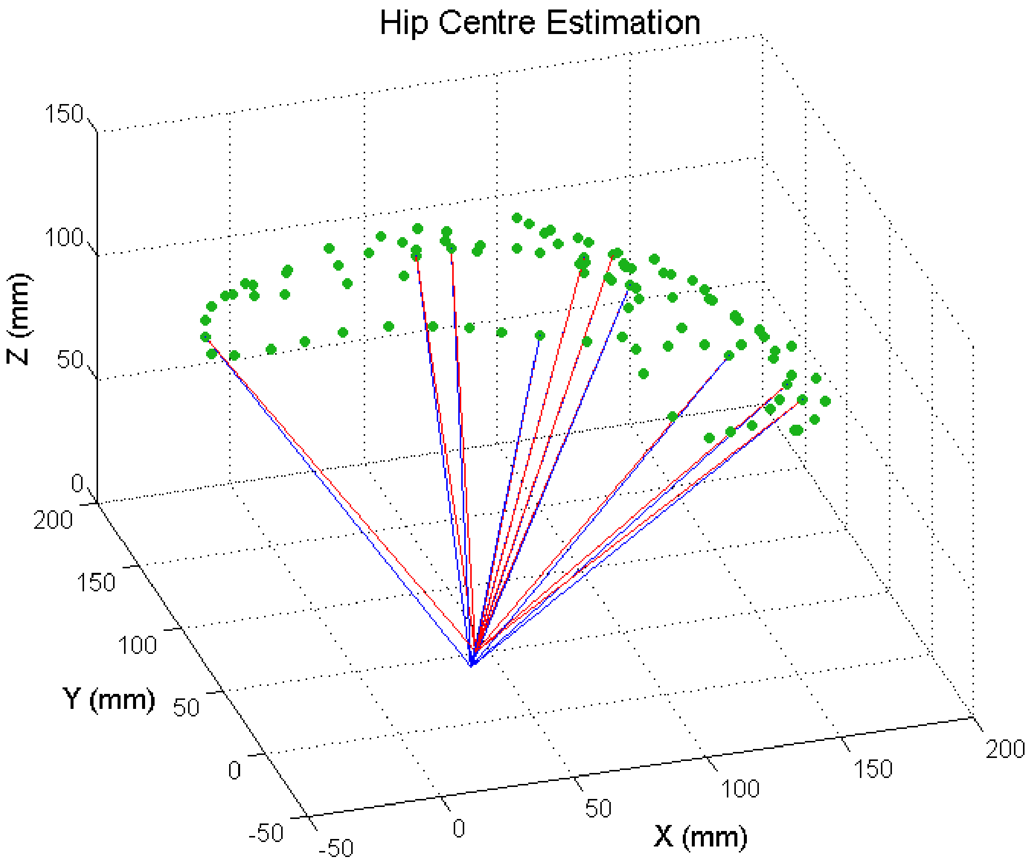


Figure 6.32: Hip centre estimation experimental results.

The differences between estimated and expected hip centres in the coordinate system of the bed marker are shown below in Table 6.4.

Axis	Expected (mm)	Estimated (mm)	Difference (mm)
X	40.0	41.5	1.5
Y	40.0	40.5	0.5
Z	0.0	6.2	6.2
Total	56.6	58.3	1.72

Table 6.4: Hip centre estimation accuracy

6.4.5.3 Discussion

Table 6.4 shows that the system was able to estimate the hip centre in the coordinate system of the bed to an accuracy of 1.72 mm. Given the average human femur length of 480 mm (Huang et al. 2012) a hip centre error of 1.72 mm would result in a HKA angle error of approximately 0.2°, less than 7% of the target accuracy of the procedure. However, both femoral and bed marker distances were considerably decreased compared to procedural use. Estimates based upon the scaling involved suggest the error could be approximately ten times that reported if used at full scale. This in turn would produce a 1.2° HKA angular error, equivalent to 40% of the target procedural accuracy.

While this does not immediately reject the use of this system for hip centre estimate, it puts extra demands upon the accuracy of the remaining elements of the system.

6.4.6 System Performance Analysis

This section analyses several core elements of the system. Thresholding, and its effect on corner detection and therefore pose estimation accuracy, is first investigated. Latency is then inspected, as a large latency is very detrimental to the performance of the system, both in terms of tracking and augmentation. Finally the last iteration of the resection model is analysed.

6.4.6.1 Thresholding

As explained in Section 6.2 the ARToolKit library uses a global approach to image thresholding. The process of thresholding converts the image into a binary representation to simplify later processing. Global thresholding applies the same thresholding test to each pixel. If the pixel value is greater than the thresholding value a high pixel is produced, if not a low pixel is produced. This provides a very computationally simple thresholding method. However, it is highly susceptible to error. Figure 6.33 below shows an image of a marker taken with the system and the corresponding binary image. It may be seen in the colour image that there is a notable light gradient across the marker – slightly exaggerated for the sake of illustration.

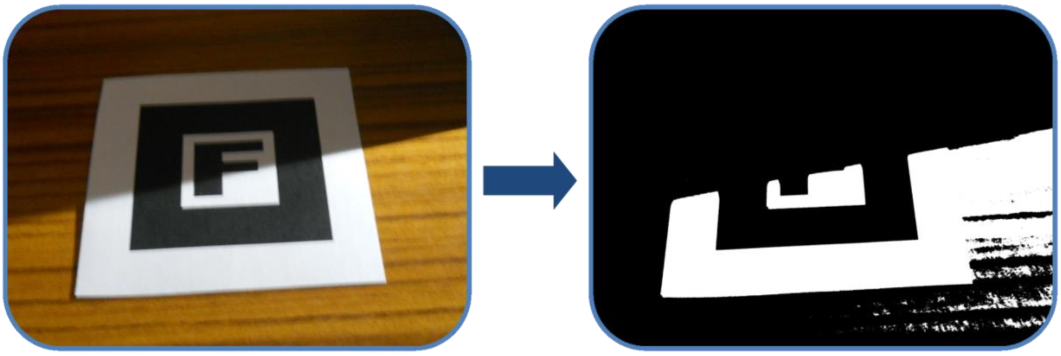


Figure 6.33: Effect of lighting gradient upon global thresholding algorithm.

It may be seen that a large section of the marker was not preserved by the binary image. Due to the varying lighting across the marker, some white sections were considered black. Analysis of the ARToolKit pipeline indicated that this thresholding error was the main cause of failed marker detection, as opposed to corner detection, marker identification, or pose estimation.

In addition to causing failed detections it is probable the primitive thresholding method also increased the corner detection error, therefore in turn increasing the pose error. Figure 6.34 below shows a more subtle example of Figure 6.33.

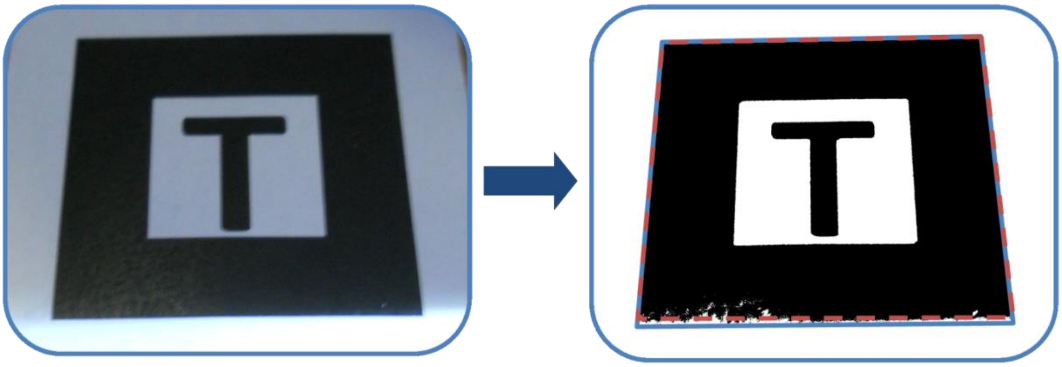


Figure 6.34: Lighting gradient leading to erroneous corner detection.

The blue overlay on the binary image of Figure 6.34 above shows the true marker edge, while the red overlay shows the marker edge the system extracted due to the lighting gradient. The system would then incorrectly identify the corner locations, leading to poor pose estimation.

6.4.6.2 Latency

Latency is extremely important to both tracking and augmented reality systems. If the system latency is too high the tracking data becomes too outdated to be of any use. In the case of guided UKA the surgeon would be able to resect bone outwith the planned region before the system registers the tool location. For augmented reality excessive latency may easily cause simulator sickness. When there is a notable delay between a user performing an action, such as turning the tool, and this having an effect on the system, the user may experience a sensation similar to seasickness, whereby different senses send conflicting signals to the brain. Therefore, for the sake of both accuracy and comfort it is extremely important to ensure a minimal latency.

The latency of the system as a whole was defined as the time between image capture and final augmented image display. This allowed measurement using physical measurement, without relying on system timers. However, individual elements of the system were timed using the high precision timer provided by the time.h standard C++ library. The relatively closed nature of the ARToolKit library made in-depth internal measurements difficult. As such several components were grouped.

To measure the whole system latency the system was run, at the standard 800x448 resolution, with a single marker visible to the camera. A large face digital timer with milliseconds was then placed in shot of the system camera. A standard CCD compact camera was then used to image the system running. The photo taken captured both the physical timer and its image in the systems augmented display shown on screen. Therefore, the total latency of the system was given by the difference between the two times. This procedure was repeated 20 times producing a mean total system latency of 128±15 ms. The same method was performed to measure the capture and display time without any image processing or 3D rendering. This was found to be 94±17 ms, representing 73% of the total system latency.

Table 6.5 below shows system section measurements performed using the high resolution clock within the C++ standard.

Stage	Image capture	Marker Detection	Marker Matching	Rendering	Total
Mean (ms)	9.0	9.0	0.4	3.0	**21.4**
SD (ms)	1.7	1.8	0.5	0.6	**2.9**

Table 6.5: ARToolKit system timings

It is seen that the majority of processing time is, as expected, dedicated to image capture, which involves moving a large amount of data onto memory, and marker detection, which constitutes most of the complexity of the ARToolKit library. The total of 21.4 ms is seen to roughly agree with the total system time of 128 ms captured physically once the camera time of 94 ms in included. Therefore, it is concluded that much of the system latency is out of the control of the system, and may not be readily optimised.

The total system time is similar to the 144 ms reported by Kang *et al.* for their augmented reality system based upon a commercial Polaris tracking system (Kang et al. 2014). Furthermore, Vercher and Gauthier determined that subjects could compensate for latencies between 250 and 300 ms in tasks requiring hand-to-eye coordination (Vercher & Gauthier 1992). Ruijter *et al.* consider 250 ms to be a critical threshold, but suggest risk may be minimised by maintaining latency below 140 ms (Ruijters et al. 2014). Therefore, the total system latency of 128 ms is considered acceptable for the intended application.

6.4.6.3 Resection model

Appendix A1.2 describes the several iterations of the resection model, changing from a simple monochromic point cloud to a morphing triangulated mesh. The following section discusses the final resection model.

The final implementation met several goals of the resection model. It could be morphed to match the topology of the bone surface, and then morphed in a restricted fashion to accommodate updating during resection. The model and rendered overlay were of sufficient resolution to accurately capture the probe’s position, despite the limited memory required by the model. The ability to adjust the colour of the surface helped to improve the intuitiveness of the system, by allowing unused sections of the model to be hidden.

The system performed well while generating tibial models. However, the femoral model was captured to a lower standard. The larger size of the volume required probing more distal from the marker, thus decreasing accuracy as shown by the above accuracy discussion. However, the major issue resulted from the shape of the femoral surface. Figure 6.35 below shows the challenge of modelling the femur surface using the current system.

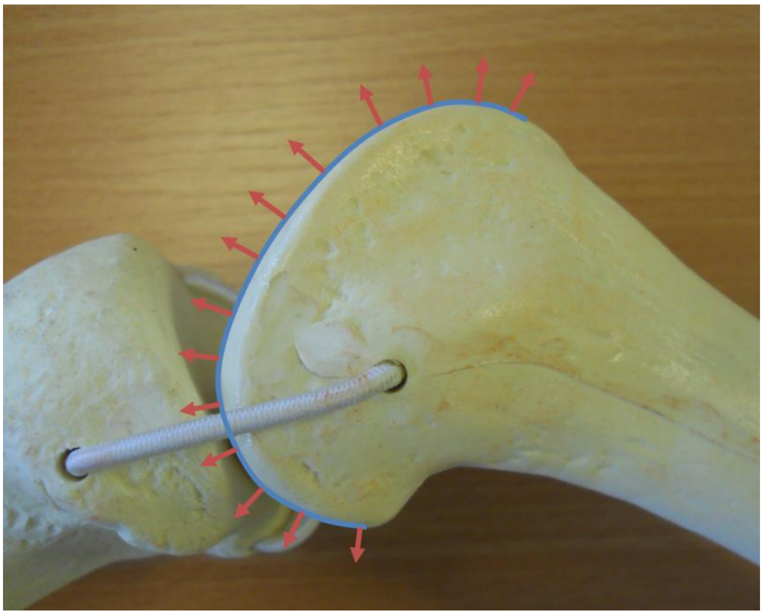


Figure 6.35: Normal angles of femoral surface.

Figure 6.35 shows a lateral view of a femoral saw bone. The outline of the surface has been marked in blue, and several surface normals have been marked in red. It is seen that the surface of the femur is highly convex in shape, thus the range of normal angles of the surface exceeds 180°. Appendix A1.2 describes the surface model as based upon a single plane of vertices, where each vertex may move normal to that plane. Therefore, with a surface range exceeding 180° there is no angle at which the original surface model plane could be placed without two sections of the physical surface lying upon the same normal. This is illustrated below by Figure 6.36 that shows several model plane angles and the resulting surface conflicts.

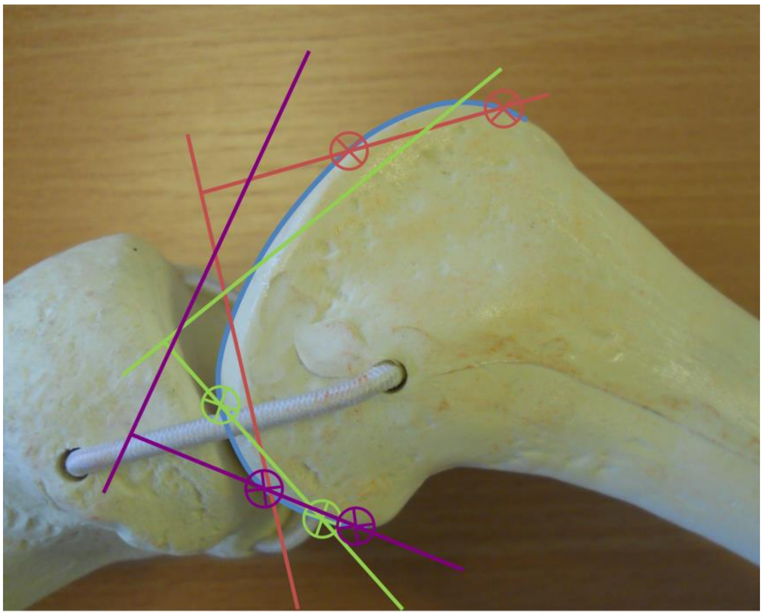


Figure 6.36: Possible femoral surface planes showing contradicting points

With the nature of the model system, the surface model would take the position of whichever of the two collinear points was last probed. Therefore, the model generation system was not able to properly capture the femoral topology. This same issue would limit the performance of the model during surface resection, particularly for implants featuring fixation pegs that require holes drilled into the bone, which would result in up to three collinear layers.

In addition to maintaining the model during resection the system is also required to control resection based upon the resection plan. This would require the implant plan to be imprinted into the model by some method. One proposal was to have a second model, of the same design, to represent the post-resection joint. During resection each time a vertex of the resection model was moved it would check against the post-resection model to ensure it remained within plan. The check would require the distance between each moving resection model vertex against each post-resection vertex. While this could be optimised using a similar approach to the tip-resection model interaction, it would be a relatively computationally intensive process and may considerably increase latency.

Due to these short comings alternative bone models were investigated as discussed in Section 7.2.

6.5 Conclusion

Section 6.4 above unfortunately concluded that the system, in its current state, was unlikely to deliver the required accuracy for a successful UKA guidance system. Analysis of the system and associated documentation revealed several elements associated with increased errors.

One of the main suspected sources of error was the system calibration, both from a method and algorithmic perspective. The first calibration stage repeatedly returned distortion values of zero. While the LifeCam used did produce a high quality image, it certainly contained some distortion. Figure 6.37 below shows a region of Figure 6.10 magnified. It may be seen that the superimposed lines do not align perfectly with the grid lines to the extremities of the image. This illustrates that the camera system was not, in fact, distortion free as calculated by the calibration.

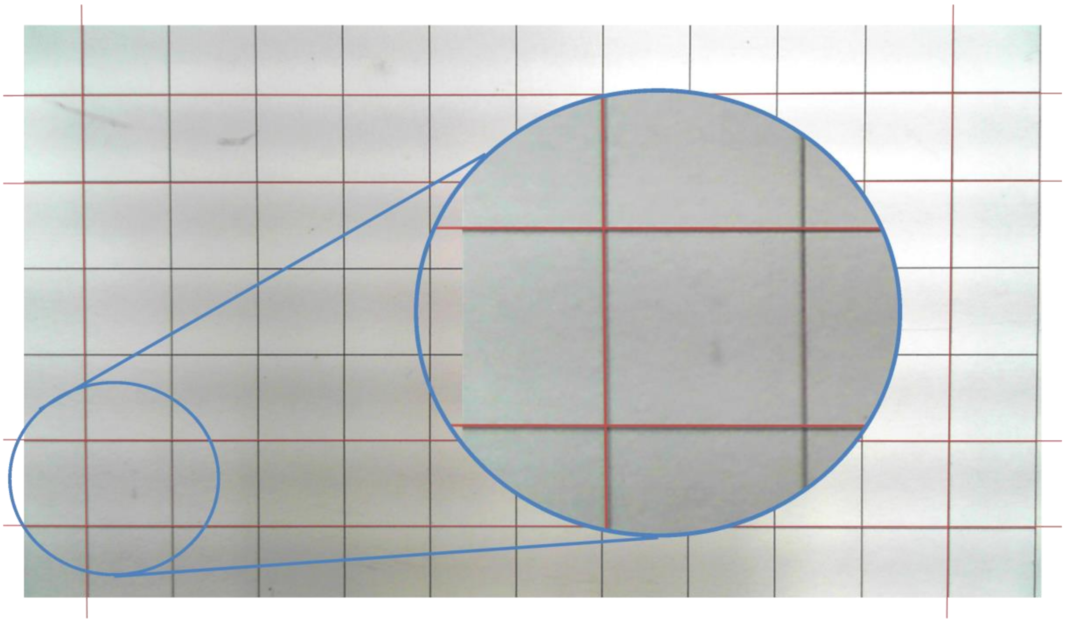


Figure 6.37: Grid image showing slight radial image distortion

Failure to correctly compensate for lens distortion fundamentally limits the accuracy of the system. This lack of distortion compensation potentially accounts for some of the error seen in the system, in particular, that seen when the marker was imaged by outer regions of the image.

The second calibration stage presented a very difficult requirement, which was well met by the Instron MTM. This was demonstrated by the consistent calibration results obtained. Despite the consistent nature of the calibration the range of results still represent an error within the system. As each calibration was performed under the same conditions, chiefly equal focal depth, an ideal system would have produced identical calibration results.

In addition to calibration errors, and their effects, the system itself was defined to contain two main sources of error: thresholding and pose estimation. The ARToolKit library utilises a global thresholding approach. As discussed in Section 6.4.6.2 above, this algorithm provides basic thresholding with minimal processing. However, it may be associated with failed marker detection and reduced corner detection accuracy.

In a similar fashion the pose estimation function likely favoured reduced computational intensity over accuracy. At the time of developing the ARToolKit library, computer systems were considerably less powerful than they are now. As such the real-time video-based pose estimation achieved by the library required heavy optimisation. This most likely accounts for the use of the global thresholding approach, already identified as a source of error. The pose estimation was also optimised to allow real-time results. As described in Section 6.2, an initial geometric estimate of the pose was produced. This was then improved through an iterative method. The geometric estimate significantly reduced the number of iterations required to obtain the pose. However, as described in Section 6.4.3 the geometric estimate occasionally produced inverted pose estimates. This would increase the error of the system, although it is suggested filtration methods may minimise this error.

Again, for the sake of real-time pose estimation, the number of iterations performed by the iterative stage is limited to five. Further iterations are expected to increase accuracy to a degree. However, these would increase the latency of the system.

Despite the limited accuracy of the ARToolKit system is was believed to offer great potential. In particular the augmented reality proved to provide a very natural and intuitive guidance. Therefore, it was decided to improve the ARToolKit library, resolving the issues above. Three core modifications were identified. Firstly, the calibration and distortion methods used were to be updated. An additional computer vision library, OpenCV, was to be used to provide more robust and accurate calibration. Thresholding was also to be updated, moving to an alternative thresholding approach that provided better resistance to lighting conditions. Again, this was to be facilitated via the OpenCV library. Finally, due to the potential of the augmented aspect of the system it was desirable to upgrade the current graphics engine. The system currently ran upon OpenGL 1.3. This is described as a fixed pipeline version of OpenGL, an outdated form compared to the programmable pipeline of modern OpenGL. Therefore, it was not possible to exploit much of the power offered by modern graphics hardware. It was probable that these outdated graphics may limit the full potential of the augmented reality guidance, and should therefore be upgraded to modern OpenGL.

The ARToolKit library was analysed to determine how these upgrades may have been implemented. After extensive analysis it was found that many of the elements to be upgraded listed above were heavily integrated into the system as a whole. Integrating modern calibration would require much of the underlying pose estimation code to be rewritten. Thresholding, while less so than calibration, would also require substantial restructure. Finally modernising the graphics would require a very substantial rewrite. The ARToolKit wrappers, used to interface with OpenGL, were used extensively throughout the program. Modern OpenGL represents a major change to the old fixed pipeline. Therefore, almost all of the old OpenGL code, including that strongly integrated with the ARToolKit, would require rewriting.

Due to the complexity and likelihood of problems presented by having to rewrite such large amounts of the ARToolKit it was decided to develop a new system, based upon similar principles, using modern tools and techniques. Therefore, work on a second generation system, built from the ground up, was started, as is discussed in the following chapter.

7

Second Generation – OCVCAOS

7.1 Introduction

7.1.1 Problem

As discussed in Section 6.5, the first generation system built upon the ARToolKit library had several short comings. Chiefly, the accuracy of the system was insufficient. This was predominantly attributed to relatively primitive camera calibration algorithms. The accuracy and marker detection of the system was also hampered by its use of global thresholding. Furthermore, it was suggested that the fixed graphics pipeline used by ARToolKit limited the potential of the system.

The ARToolKit library was closely examined, investigating all the core functions and algorithms. After an extensive review it was concluded that nearly fifty percent of the library would require reprogramming to rectify the highlighted problems. The calibration algorithm, particularly the distortion correction, was found to use a very different structure, with limited documentation explaining the variables used. Therefore, it would have been necessary to reprogram the calibration algorithms, along with all sections of the pose estimation code that utilised the calibration. The fixed pipeline OpenGL graphics were also found to be firmly embedded into the system. In addition to controlling the graphics, which required camera calibration data, it was also strongly linked to the control flow of the program through the closely related GLUT library. As with the calibration algorithms, updating the system graphics would have required substantial reprogramming of the library.

Another issue was the language of the ARToolKit library. It was programmed in C, while the first generation system used an object orientated C++ approach. While this mix of languages did not necessarily cause problems it may be considered undesirable. Clearly restructuring the ARToolKit library into C++ would require a major amount of reprogramming.

Due to the large amount of work required to resolve the problems highlighted above and in Section 6.5, it was concluded that a new system could be produced, without the ARToolKit library, for a similar amount of effort as would be required to improve the library. Furthermore, a ground up approach would allow full control over the system.

7.1.2 Solution

The new system was to be built upon two core libraries; OpenCV to provide computer vision functionality, and OpenGL. Several other non-standard libraries were also to be used to provide additional functionality. Together these would allow a modern and more robust implementation of the ARToolKit library.

7.2 Design

The following section discusses the design of the second generation system, describing first the hardware elements. The markers, which represent the core of the system, are first discussed in Section 7.2.1.1 before detailing the new probe design in Section 7.2.1.2. As shall be discussed in Section 7.3.4 resection tests were performed using the second generation system. Therefore, Section 7.2.1.3 follows the iterative design and development of the resection tool. The final hardware design section describes the external interface utilised by the system to communicate with external tools and users.

This design section then documents the software design of the second generation system, illustrating the general architecture and hierarchy of the system. As with the design discussion of the first generation system of Section 6.3.2 the following discussion omits much of the design detail to remain implementation independent. The implementation of the system is described in further detail in Appendix A2.

7.2.1 Hardware

7.2.1.1 Markers

As discussed in Section 6.2 above the ARToolKit system made use of square markers as they are easily detected within an image due to their well-defined corners, yet still provide sufficient points for pose estimation. Alternative fiducial marker systems, such as those discussed in Section 2.5 utilise an array of different marker shapes, including circular and blob-like. However, these often require more complex and computationally intensive algorithms to extract their features from the scene. As such it was decided to build the second generation system upon square markers.

In addition to providing object points, from which the pose of the marker may be estimated, markers were also required to provide identification information. In addition to identifying the marker this information was also used by the ARToolKit system to determine the orientation of the marker. As described in Section 6.2 the ARToolKit system used an image within the centre of the marker to both orientate and identify the marker. For the second generation system it was decided to separate these tasks into two distinct areas of the marker. With the addition of the region responsible for edge and corner detection and a whitespace buffer region to improve marker segmentation the markers consist of four distinct regions, as shown below in Figure 7.1.

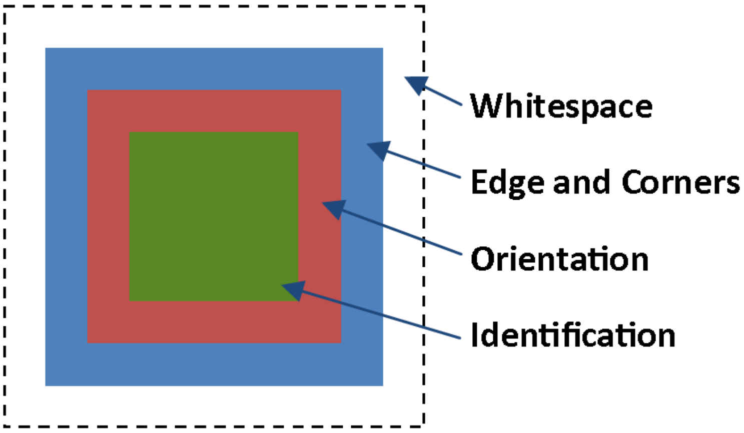


Figure 7.1: Second generation marker regions. The outer whitespace region used to improve marker segmentation from the background scene. The edge and corner region used to allow corner detection. The orientation region used to orientate the marker and determine the order of the corners. The identification region used to identify the marker.

Based upon the results of the first generation system it was considered prudent to marginally increase the marker size. An edge length of 32 mm was selected as an acceptable compromise between increased contour points and overall size. As discussed in Chapter 6, while a larger marker provides more edge points for pose estimation, this gain is weighted against complications that may result from the relatively limited FOV of the system resulting from the tool mounted approach.

The edge and corner region was designed to provide a strong contrast to the whitespace region to ensure consistent marker detection. Experimentation indicated that an edge thickness of only 4 mm was consistently detected at a range of angles and distances.

Orientation information would be provided by a barcode like pattern which possessed no rotationally symmetry, shown below in Figure 7.2. The lack of rotational symmetry ensures that the corners may be uniquely identified, independently of the physical orientation of the marker. The origin corner is defined as the top-corner.

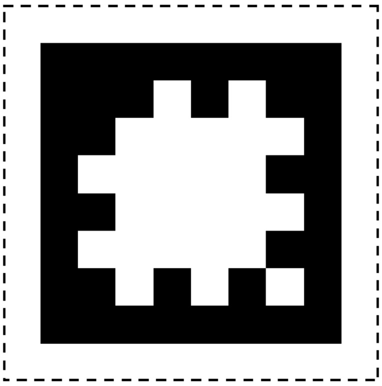


Figure 7.2: Second generation orientation barcode system

The orientation barcode contains 20 elements. For an imaged object to be classified as a potential marker, a process discussed in detail in Appendix A2.1.5, each of the 20 elements must be matched correctly. The odds of this occurring for a random object are approximately one in a million. In testament to this probability a non-marker item was never seen to be detected as a marker during the many hours of system testing. As with the edge and corner region, the width of the orientation region was 4 mm.

The final region of the marker pattern was the identification region. This consisted of a 16 unit binary barcode arranged in a 4x4 grid. The orientation region defined the order in which these units should be read to form the identification code of the marker. The identification region, and the value of each unit, is shown below in Figure 7.3.

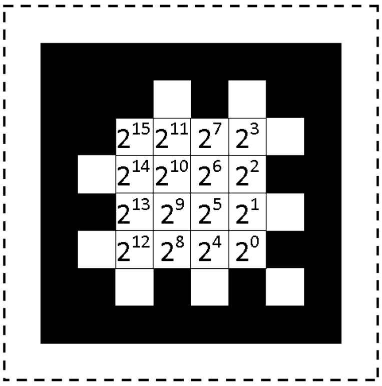


Figure 7.3: Second generation identification barcode system

The 16 unit barcode presented above allows for 65,536 markers, far more than any orthopaedic system could ever conceivably require.

For the second generation system the markers were originally printed onto standard A4 paper using an office laser printer at 600 dpi. The markers were then attached to 2 mm float glass using spray adhesive. As shall be discussed in Section 7.3, several experiments required the pattern of the marker to be well aligned with the external edge of the marker. This proved challenging to consistently achieve with the glass affixed markers. As such alternative marker production methods were sought.

Laser cutting combined with laser engraving allowed both the edge and pattern of the marker to be produced to a high accuracy and aligned relative to each other, as both the cutting and engraving were performed by the same tool in a single process. Laser cutting uses a high powered class 4 laser to burn neatly through the material, typically sheet acrylic of thicknesses below 10 mm. The same system is able to engrave, removing a thin upper layer of material, simply by reducing the power of the laser. To dramatically increase the contrast of the engraved pattern, laminate materials were used. A laminate material consists of a relatively thick layer of base material with a thin layer of contrasting material fused to the top surface. 75 µm white on 1.5 mm black laminate material was selected as the black surface exposed from laser engraving presented lost specular reflectivity.

To ensure high quality markers it was required to remove the full surface layer and as little of the base layer as possible. The effective power of the CTR TMX65 laser cutter (CTR Lasers, Northampton, UK) used was controlled by two variables; power, measured as a percentage between 0 (off) and 100% (full power), and speed, measured in millimetres per second. These were set using the LaserCut 5.3 software. To determine the optimal settings the test grid shown by Figure 7.4 below was prepared using a combination of speed and power settings.

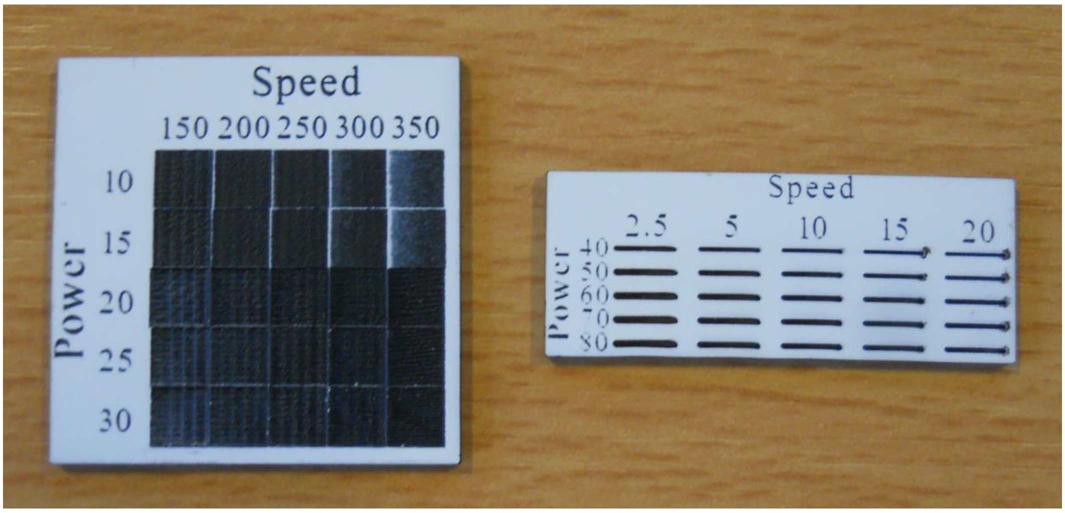


Figure 7.4: Laser engraving and cutting test patterns

The optimal settings were defined as those that produced the first completely black finish at the lowest power and speed settings, implying least base material had been removed. Preference was also given to a lower speed, as it was concluded a higher speed may negatively affect the accuracy of engraving. As such, a speed of 170 mms^{-1} and a power of 10% were chosen as the optimal engraving settings. Similar tests were performed to define the optimal cutting settings, also shown by Figure 7.4. This was defined as the lowest effective power that successfully cut fully through the material. The optimal cutting settings were found to be 10 mms^{-1} at 40% power. Additionally a corner power of 12% was used to compensate for the reduced speed of the laser while cutting corners.

Figure 7.5 below shows a selection of markers produced using the above settings.

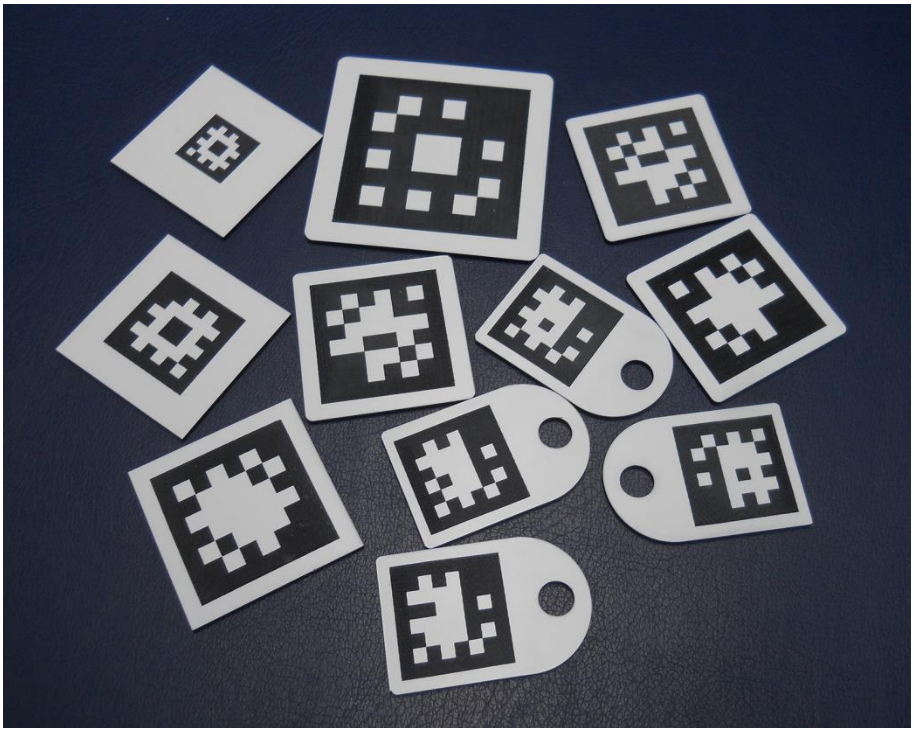


Figure 7.5: Selection of laser engraved markers

7.2.1.2 Probe Design

As discussed in Section 6.4.4 one data set was disregarded as it was found the probe had moved during experimentation. Therefore, a new probe and probe mount were designed to ensure consistent results. The probe was designed under the requirements outlined by Table 7.1 below.

Probe Requirements
Maximum separation from camera front of 200 mm
Maximum flexion below 0.1 mm under 1 N lateral force
High accuracy spherical probing tip

Table 7.1: Second generation probe design requirements

The general design of the probe was a cylindrical rod with a spherical tip. The probe would be mounted along the full length of the camera, as such would have a full maximum length of 260 mm. Simulations were performed to determine the appropriate material type and diameter of the probe. The probe was rigidly supported along its initial 60 mm length. A force was then applied to the distal end of the probe and the maximum displacement measured. A force of 1.0 N was used based upon force measurements with a similar dimensioned rod. Figure 7.6 shows three examples of these simulations using 6, 8, and 10 mm mild steel rods respectively.

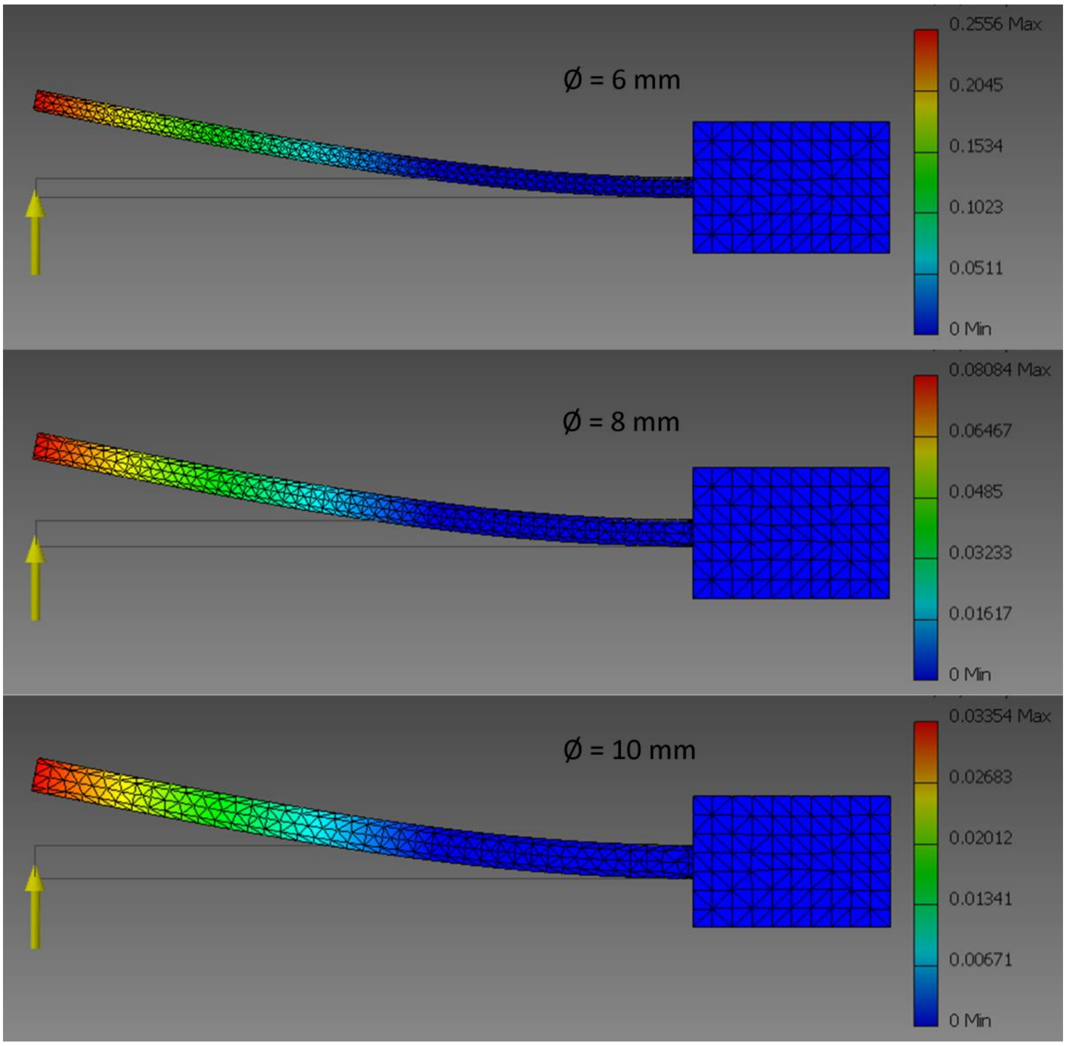


Figure 7.6: Probe flexion simulation sample set.

0.8 mm mild steel produced a flexion below the 0.1 mm threshold defined above, presenting a maximum flexion of 0.08 mm, and as such was selected for the probe. As shown by Figure 7.6 above, a larger diameter substantially decreases the flexion of the probe. However, as discussed below the existing camera holder was to be used to attach the probe to the camera, and therefore a probe diameter above 8 mm would leave insufficient material to securely attach the probe.

To provide an accurate spherical probe a 3 mm grade 10 chrome steel ball bearing was attached to the probe. To allow maximum range of rotation about the sphere the probe was tapered over a length of 20 mm to a diameter of 1.8 mm. A 1.5 mm spherical recess was cut to secure the ball bearing to the probe. A further simulation confirmed this tapering would not increase the maximum flexion of the probe above

0.1 mm. A tip diameter of 3 mm was selected as it would allow probing of 5 mm regions with 1 mm of material surrounding the probe for support.

The camera holder described in Section 6.3.1 was modified to allow probe attachment. Two 8 mm holes were machined in the top-left and bottom-right corners of the camera holder, at a radial distance of 27.5 mm from the camera centre. 3 mm thumb screws were used to secure the probe into either hole. This design would allow the length of the probe to be adjusted. The final assembly is shown below in Figure 7.7.

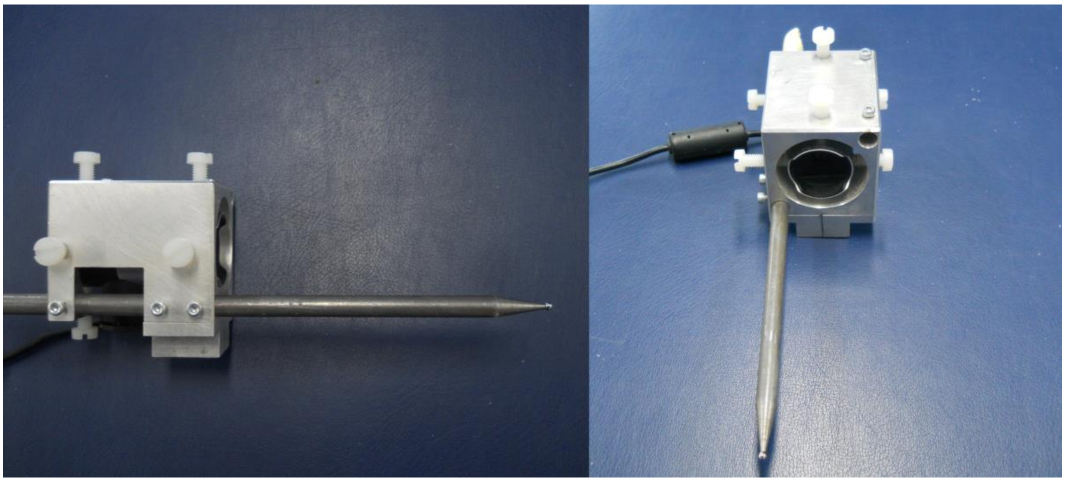


Figure 7.7: Assembled second generation probe and probe mount.

7.2.1.3 Resection Tool

As discussed in Chapter 5, it was originally intended to facilitate resection via an Anspach eMax 2 (Synthes Inc. PA, USA), as done by both the Blue Belt and MAKO systems. However, this proved cost prohibitive as a full Anspach system, comprising base unit compressor, hand piece and burrs, and foot pedal, was quoted in excess of US$10,000. Therefore, an alternative resection tool was sought.

All medical grade tools would likely prove too expensive, therefore commercial tools were investigated. Rotary tools, such as those by Dermal provided similar functionality to the Anspach system at a fraction of the cost. Tests showed that the reduced rotary speed of these tools, approximately half that of the Anspach, still allowed resection at an acceptable rate.

A 170W variable speed rotary tool with maximum speed of 35,000 RPM was selected. While relatively slow, this resected neatly without chipping of the material or snagging of the tool. To provide a lightweight design a 1 m flexible drive handpiece was used. A wooden handpiece was designed to allow control switches and the tracking camera to be securely attached to the burr, as shown by Figure 7.8 below.

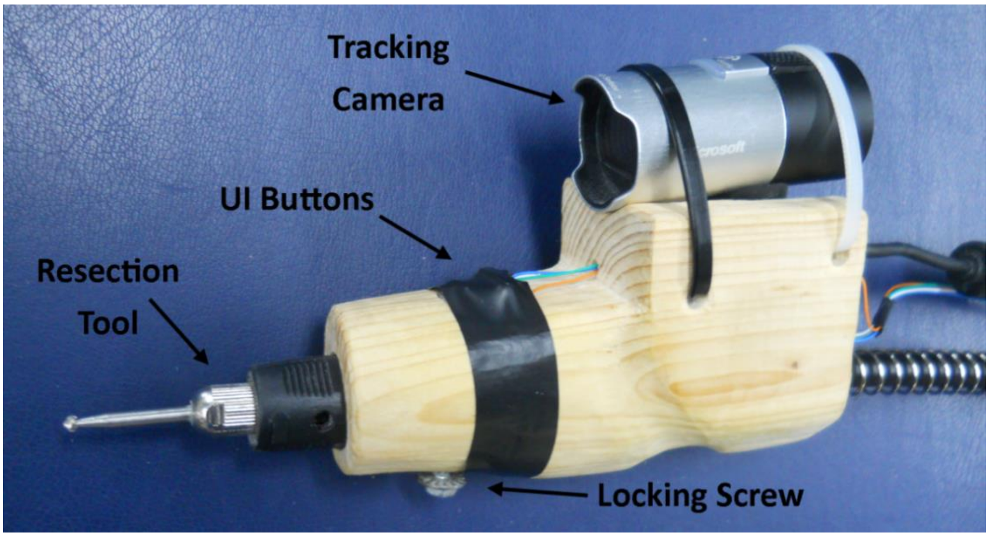


Figure 7.8: Rotatary tool resection handpiece

Figure 7.9 below shows the control circuit developed for the tool. Based upon a HK4100F-DC5V-SHC relay, this circuit allowed the control board, discussed in Section 7.2.1.4 to activate and deactivate the burr.

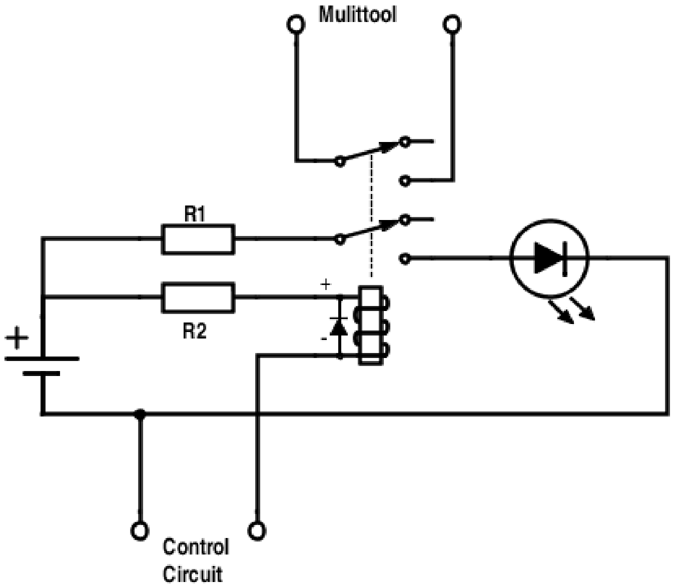


Figure 7.9: Control circuit for rotary tool resection system

Additional testing with the rotary tool revealed that while the lower rotation speed did not cause major problems the tool took a substantial time to reach speed once activated, and to stop once deactivated. To remedy this, the electromagnetic breaking clutch shown in Figure 7.10 was designed.

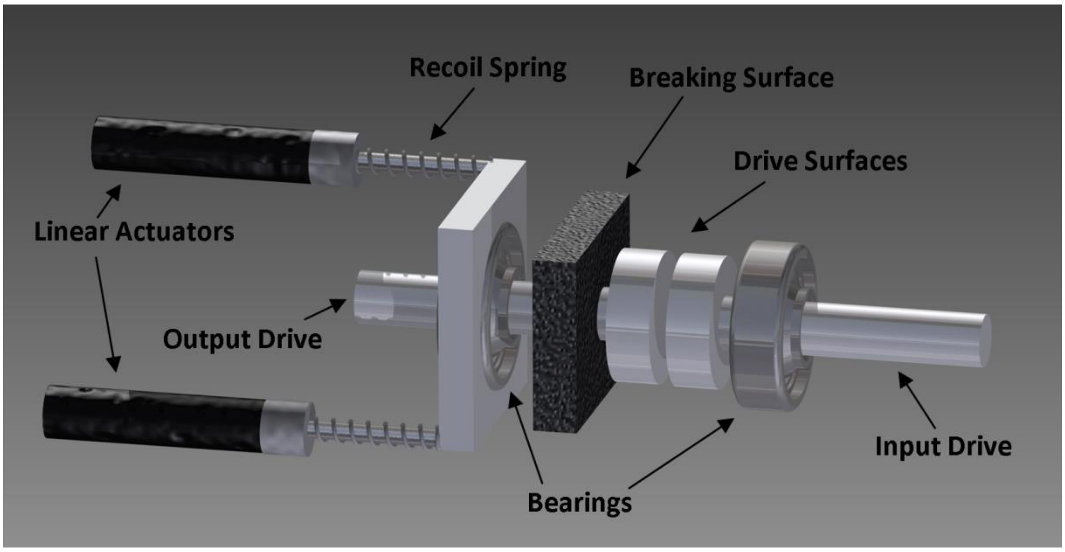


Figure 7.10: Final breaking clutch design

Electromagnetic linear actuators were to be used to control the breaking clutch. When active the drive contact plates would be brought into contact, thus transferring the drive of the tool to the hand piece. When not active, or in the case of power failure, recoil springs would passively disengage the drive plates and engage the breaking contact plates, stopping the cutting burr. However, before implemented the clutch an Anspach eMax 2 system became available within the department.

The Anspach system offered several advantages over the current system, providing higher rotational speed, much shorter spin up and down times, due to an internal breaking mechanism, and greatly reduced vibration – the significance of which shall be discussed in Section 7.3.4 below.

To accommodate the Anspach system an alternative camera-tool mounting system was designed. Based upon testing with the initial headpiece several design requirements were highlighted, as listed below in Table 7.2.

Headpiece Requirements
Camera position adjustable relative to burr tip
Camera angle adjustable relative to burr tip
Light weight and compact
Ridgely attach camera and burr
Dampen vibration of burr

Table 7.2: Design requirements of Anspach headpiece

To meet the first two requirements a handpiece was designed such that the camera and burr were joined by four arms, two fore and two aft. These would allow the camera to be position and orientated relative to the burr tip. The arms were friction locked in places ensuring a ridged coupling. To allow a light weight design the handpiece was 3D printed in ABS using an Ultimaker 2 (Ultimaker B.W., The Netherlands). This allowed internal geometry which would be difficult to machine traditionally. The arms were in part expected to dampen the burr vibration reaching the camera. However, further dampening was applied by lining all mounting contact surfaces with 2 mm thick shock absorbing foam rubber. Figure 7.11 below illustrates the handpiece design.

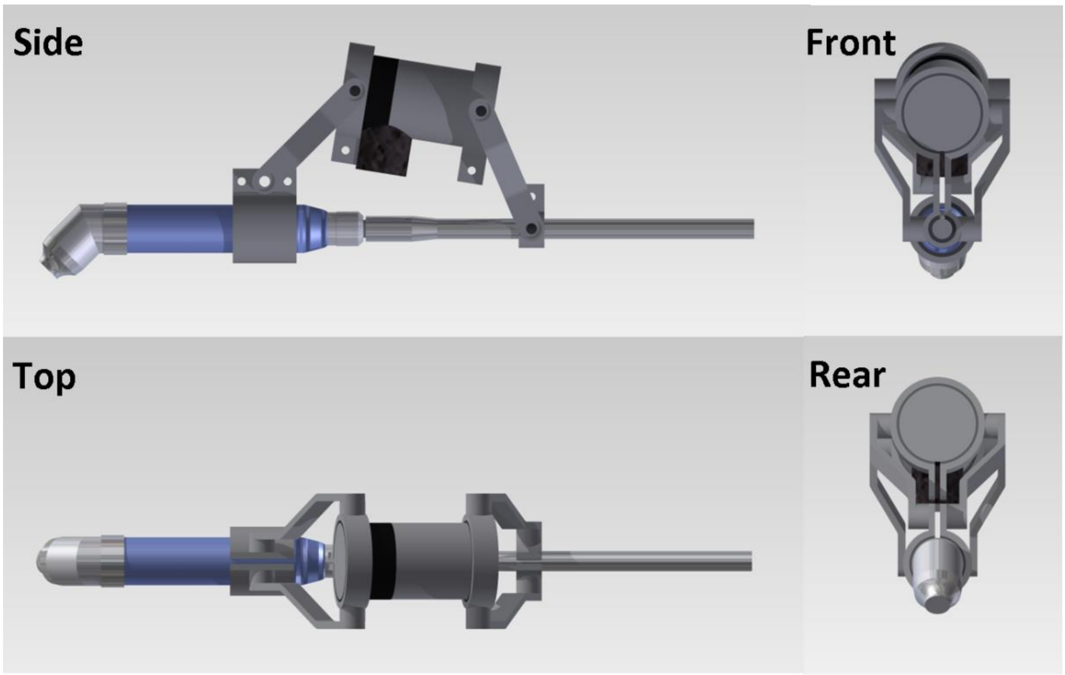


Figure 7.11: Design for Anspach handpiece

After clean-up and attachment of rubber padding and metal hardware the handpiece was seen to perform as desired. The mount sections held both camera and burr firmly and required considerable force to induce displacement or rotation. The arm sections also locked well and did not loosen with use. The final handpiece is shown below in Figure 7.12.

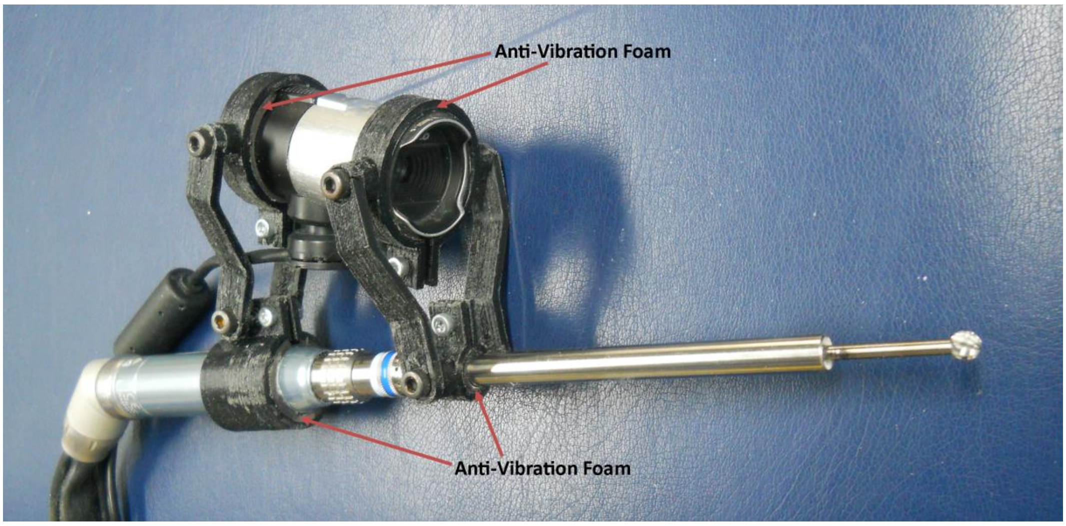


Figure 7.12: Final Anspach handpiece

To facilitate speed resection control with the Anspach system a new control circuit was produced. The Anspach is typically controlled via a foot pedal switch, which is connected to the compressor unit via an 8-pin Lemo connector. Analysis of the pedal revealed a basic single-pole double-throw switch. This connected a common control pin to either ground or high, to control the burr. It was decided to introduce a system control circuit between the foot pedal and compressor unit. This circuit would either pass or block the pedal signal dependent upon the resection control state of the system. Additionally, the control circuit was designed to disregard the pedal input and utilise only the control signal. This would allow an alternative hardware trigger connected to the interface system discussed below to operate the burr. Figure 7.13 below shows the Anspach control circuit. An Axicom DPDT non-latching relay provided signal control. This induced a maximum switching latency of 6 ms.

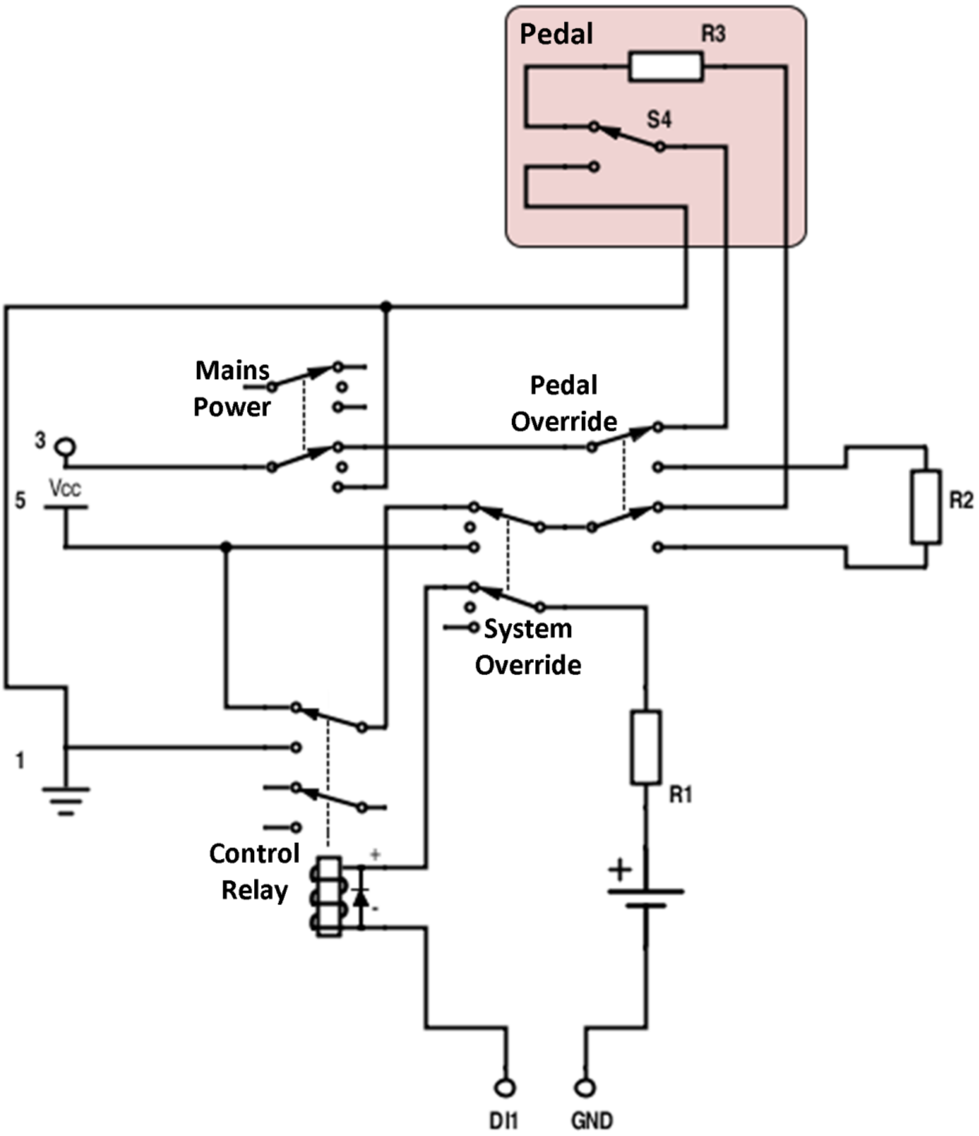


Figure 7.13: Anspach system control circuit

To further reduce the size of the handpiece the three UI buttons were replaced with a single 10 mm five-direction momentary switch. This allowed five digital inputs for the space of a single standard switch. Each of the five switch signals were connected to the external interface system discussed below.

The complete second generation Anspach based system is shown below in Figure 7.14.

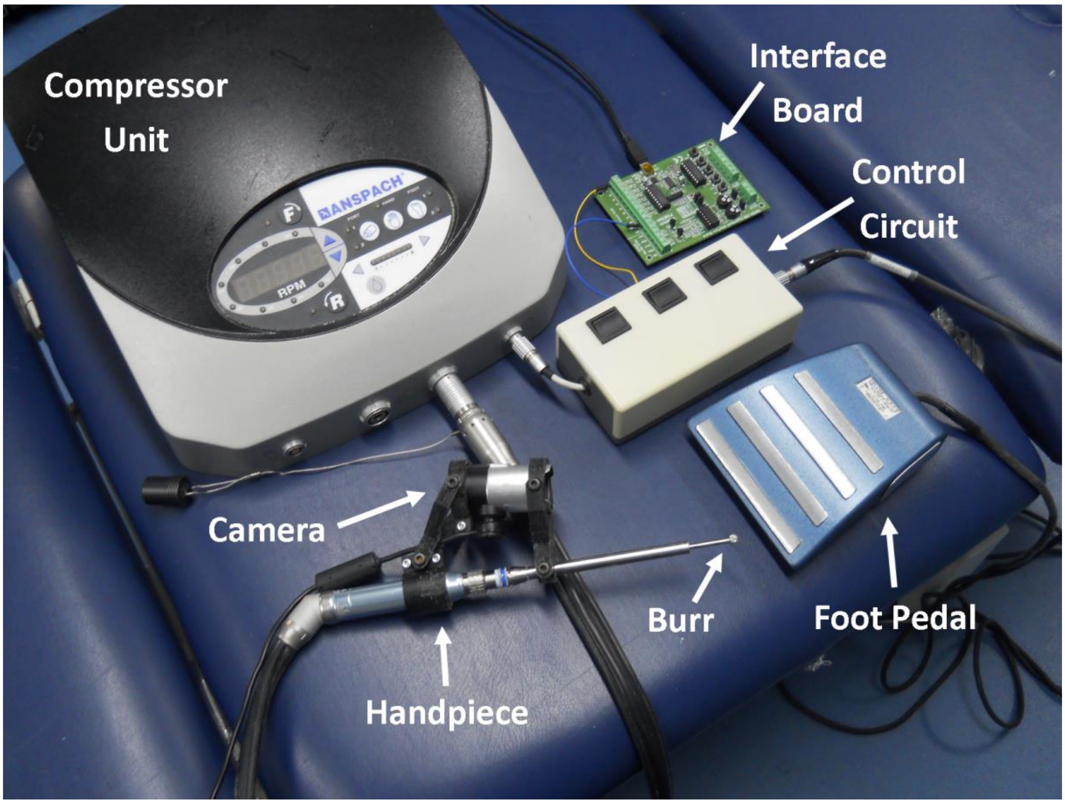


Figure 7.14: Second generation Anspach based system

7.2.1.4 External Interface

As discussed above the proposed system required several external input and output signals. Due to the lack of interface ports or expansion slots available on the development PC, as described during camera selection in Section 5.2.2.1, a USB interface was required. Three potential interface systems were investigated, each based upon microcontroller devices. The first system was based upon a PIC 16F628A microcontroller (Microchip Technology Inc., AZ, USA). The second utilised an Arduino nano (Arduino, MA, USA). The third system would use a Velleman VM110N experimental board (Velleman, Gavere, Belgium). Table 7.3 below compares several important aspect of each of these interface options.

Feature	PIC	Arduino	Velleman
Physical Connection	Custom USB	Built in USB	Built in USB
Connection	Serial	Serial	
Software Interface	C library	C++ libraries	X84 and x64 DLL
Output	Up to 16 Digital PWM TTL Serial	Up to 14 Digital Up to 6 PWM TTL Serial	8 Digital (with LEDs) 2 Analogue/PWM
Input	Up to 16 Digital Up to 4 Analogue	Up to 14 Digital Up to 8 Analogue I^{2}C	5 Digital (with momentary switches) 2 Analogue (with trim pots)
Latency	~2 ms	~ 4 ms	~ 2 ms

Table 7.3: External interface device comparison

As seen from Table 7.3 the PIC and Arduino offer similar solutions with the PIC presenting a smaller package size and the Arduino providing the easier interface. The Velleman experimental board meanwhile offers much more basic input and output options. However this is countered by the provided interface libraries. Due to these interface libraries, and the rapid testing afforded by the built in physical controls, the Velleman VM110N experimental board was selected to provide the system with external interface. If further functionality were required the Arduino nano would be selected, again due to the simplified interface in comparison to the PIC microcontroller.

As discussed above in Section 7.2.1.3 the initial resection tool design required three digital inputs and one digital output from the system. These requirements were met by the Velleman board. Digital input was achieved by pulling one of the five digital pins to grounds via a momentary switch. Digital output pins controlled current flow to the ground pin. Therefore, digital output was achieved by treating the connection between each digital output pin and the ground pin as a momentary switch controlled by the system. Figure 7.15 below shows the Velleman experimental board.

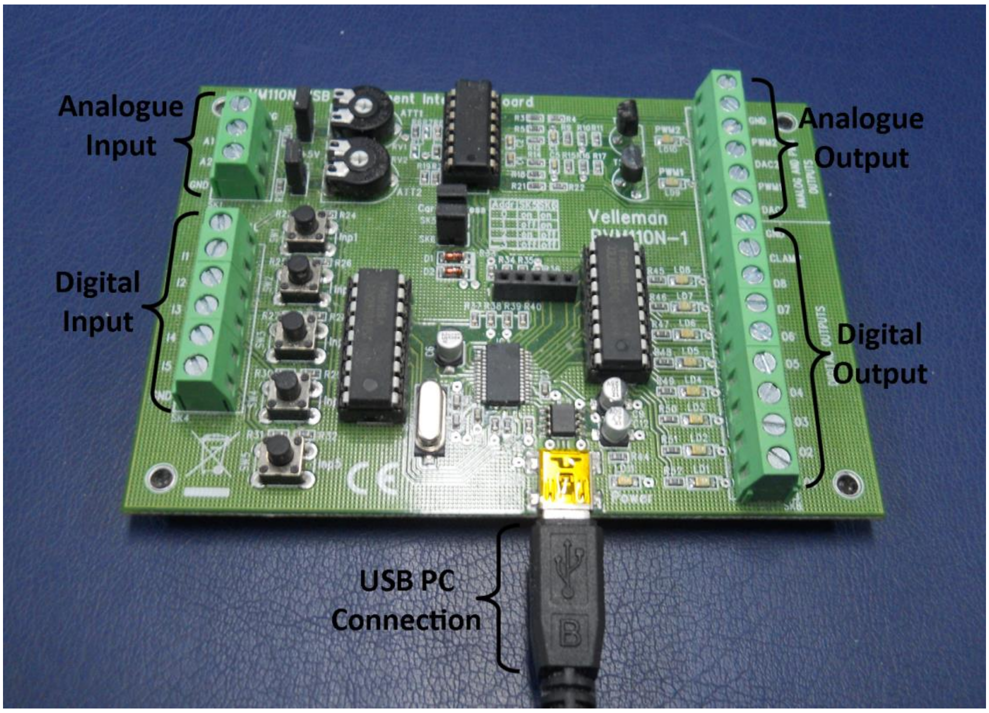


Figure 7.15: Velleman USB experiment board

7.2.2 Software

The second generation system was based upon a hierarchical structure, whereby the different sections of the procedure, i.e. model generation and resection, were divided into stages. Each stage consisted of a world, which contained the physical and virtual elements used by the system, such as models and the resection tool and camera. This hierarchy is shown by Figure 7.16.

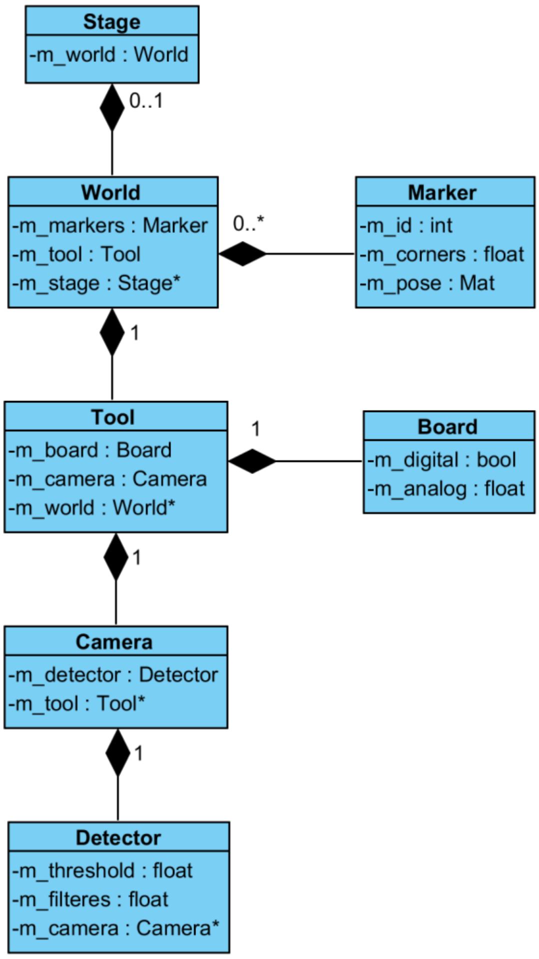


Figure 7.16: UML class diagram overview of the second generation hierarchy.

The Detector class seen at the bottom of Figure 7.16 was responsible for marker detection and corner extraction from the 2D image, a process that is summarised by Figure 7.17 below.

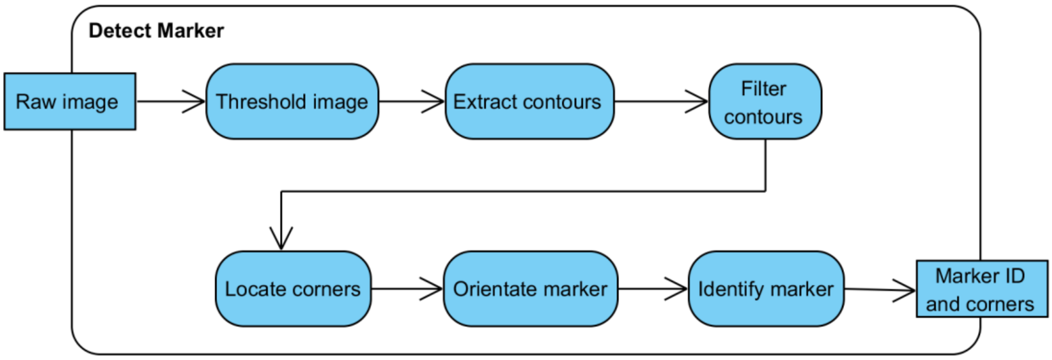


Figure 7.17: Core marker detection activities of the Detector class.

The detected marker features are then used to perform pose estimation. Therefore, the system is able to determine the relative position of the tool to other physical objects, and corrected overlay virtual guides. The overall control flow of the system is illustrated by Figure 7.18. Further details of thesecond generation system may be found in Appendix A2.

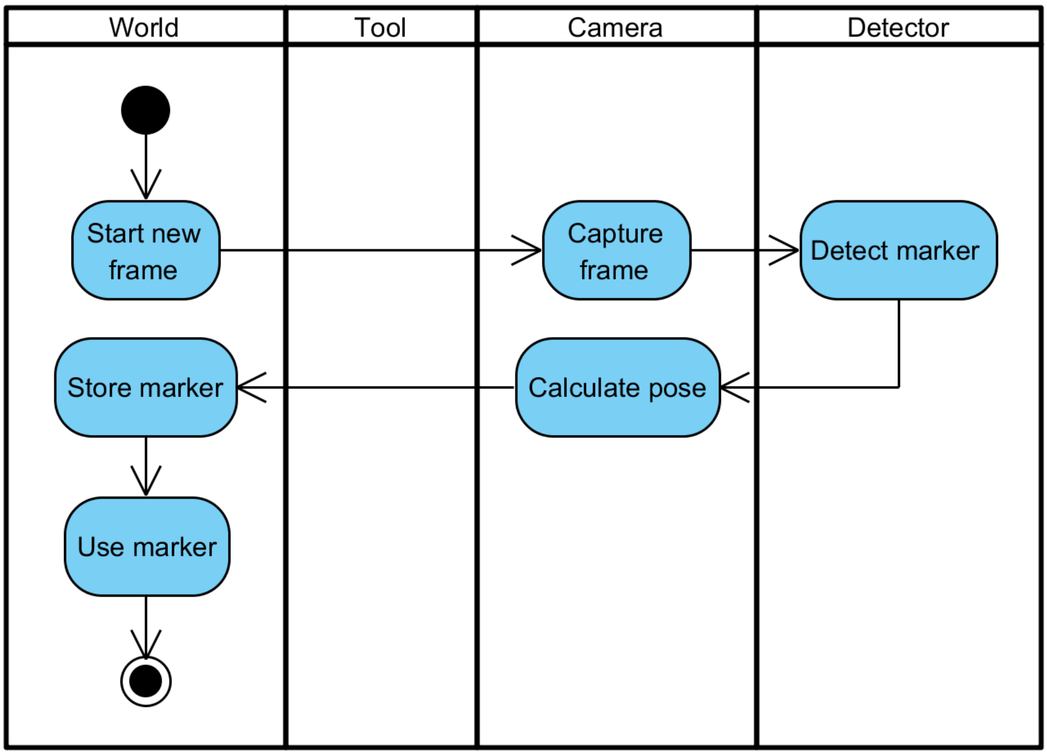


Figure 7.18: Marker generation process

7.3 System Analysis

The following sections discuss the evaluation of the second generation system. Initially the system calibration, which was highlighted as a probable cause of error for the first generation system in Section 6.5, is detailed. After this, the effect of marker orientation relative to the camera shall be investigated, as Section 6.4.4.4 noted that the first generation system may present an orientation accuracy relation. As with the first generation system a series of probing experiments were performed. However, due to the probe target errors of Section 6.4.4 a new target shall be presented. Finally, this section concludes by presenting several controlled resections performed with the system.

7.3.1 System Calibration

As with the first generation system a series of calibrations were performed and the standard distribution of their resulting parameters used to gauge the consistency and therefore reliability of the calibration procedure.

The presented system was initially calibrated and run at a resolution of 800x448, as this allowed the maximum camera framerate of 30 fps to be achieved. However, in the interest of accuracy the system was recalibrated to operate at 1280x720. This permitted a maximum framerate of approximately 20 fps. Only the latter shall be presented here.

7.3.1.1 Materials and Methods

Calibration of the second generation system was a single stage process, performed by taking several images of a known calibration grid. By matching image points to their known relative physical location the system is able to iteratively calculate the camera parameters.

In addition to the intrinsic parameters, OpenCV also determines several parameters which characterise the distortion induced by the lens system. Two distortion models are utilised by OpenCV. Firstly, radial distortion is compensated for by a three term model. The combination of the three factors allows OpenCV to better compensate for complex radial distortion. The radial model is shown by Equation 7.1 below.

$$\begin{aligned} x_c &= x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y_c &= y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{aligned} \tag{7.1}$$

It is observed that the optical axis of the lens system is unlikely to align perfected with the normal of the image sensor plane. Therefore, a second tangential distortion model is applied, as shown by Equation 7.2 below.

$$\begin{aligned} x_c &= x + \left(2p_1 xy + p_2(r^2 + 2x^2)\right) \\ y_c &= y + (p_1(r^2 + 2y^2) + 2p_2 xy) \end{aligned} \tag{7.2}$$

A calibration suite was developed to implement the five stages of camera calibration listed by Table 7.4 below.

Calibration Stages
Camera configuration
Image capture
Feature extraction
Parameter calibration
Result storage

Table 7.4: Five stages of second generation camera calibration

The suite was built fully upon the OpenCV library, utilising its built in display and 2D render functions, as opposed to the OpenGL used by the main system.

Camera configuration was performed as with the main system. A connection was established with the camera and settings were applied. As images were captured from the video stream the user was able to adjust the focus and exposure of the camera to obtain the optimal image quality.

A frame was captured for processing by a key press. Originally the frame was immediately processed. However, feature extraction could take several seconds and substantially increased the capture time. Therefore the calibration suite was modified to perform offline processing. Each captured image was stored. Once sufficient captures had been made all stored images were batch processed.

Image features, the centres of a large series of circles, were extracted using the $findCirclesGrid$ function. In addition to the image frame this function was passed the grid size, 4x11, and type, asymmetric circles. The asymmetric circle grid was used as it was reported to offer improved results over the standard chess pattern.

To allow erroneous feature detections to be manually screened the feature centres were drawn onto the image frame before being saved for later inspection. An example frame with detected features is shown below in Figure 7.19.

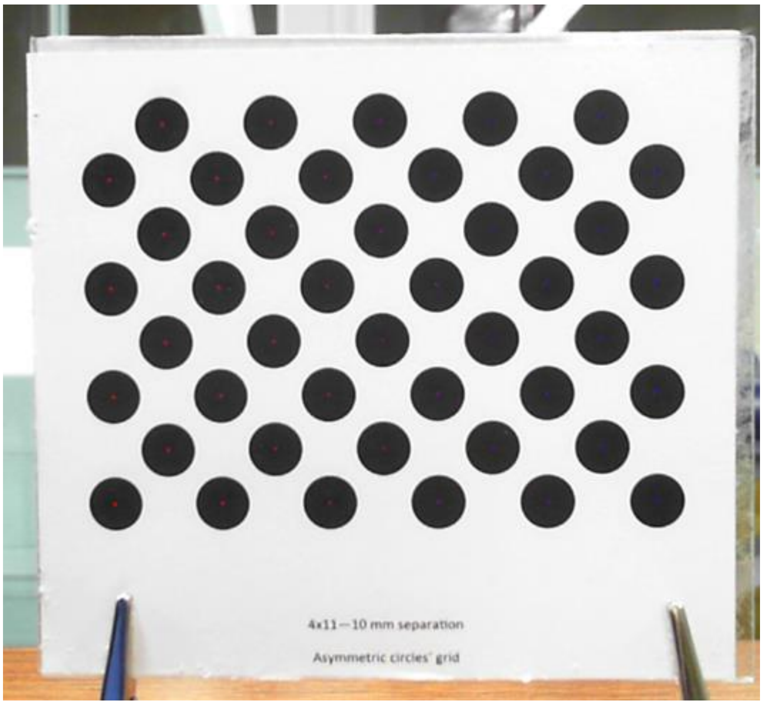


Figure 7.19: Asymmetric calibration grid with detected centre features marked

The detected image features of each frame were stored within a vector array, which were in turn stored within an array for all frames.

As discussed, camera calibration compares these extracted image features to the know geometry of the calibration target. Therefore, the known geometry of the calibration grid was required. The grid, as with the final iteration of the first generation system, was printed using a 600 dpi office laser printer on standard A4 paper before being affixed to 2 mm float glass via spray adhesive. Figure 7.19 above shows the 4x11 asymmetric circular grid with 10 mm separation used by the calibration suite. To maximise calibration accuracy the grid points were measured using a Zeiss Imager Z1 optical microscope (repeatability 0.3 μm). The mean printing error across the full target was found to be 0.047±0.039 mm.

The measured centre points and extracted image points were passed to the OpenCV $CalibrateCamera$ function. This returned the intrinsic matrix and distortion parameters that minimised the projection between the two data sets. The RMSE of the minimisation in pixel units was also returned. A successful calibration threshold of 0.25 pixels was applied. At a working distance of 120 mm with a vertical resolution and FOV of 720 and 36°, respectively, a physical resolution of approximately 0.1 mm/pixel is obtained. Therefore an RMSE error of 0.25 pixels corresponded to an error of 0.025 mm.

Once calibrated the intrinsic matrix and distortion parameters, in addition to the camera settings at which they were obtained, were stored to a calibration file in XML format for use by the main system.

An initial set of five calibrations were performed with the calibration pattern held by hand. However, as discussed in Section 6.4.2, this incurred error due to intra-capture movement. The five initial calibrations produced a mean RMSE of 0.47±0.19 pixel compared to 0.15±0.01 pixel, of a second set of five calibrations performed with both camera and calibration pattern fixed, a significant improvement ($P = 0.03$).

Unlike the first generation system, the second generation calibration procedure was not limited to a set number of image samples. Therefore, a very large sample ($n = 788 \pm 84$) was taken for each calibration. This ensured that the full range of target and camera orientations could be imaged across the full range of the image. This large dataset reduced the effect of anomalies and ensure a high quality calibration.

7.3.1.2 Results

As described above each calibration produced four intrinsic and five distortion parameters. The mean intrinsic parameters across the five calibrations are listed by Table 7.5. The distributions of the parameters are shown below in Figure 7.20.

F_X	F_Y	C_X	C_Y
983.1±2.2	983.5±2.3	635.8±0.2	369.3±0.2

Table 7.5: Mean values of intrinsic parameters - all units in pixels

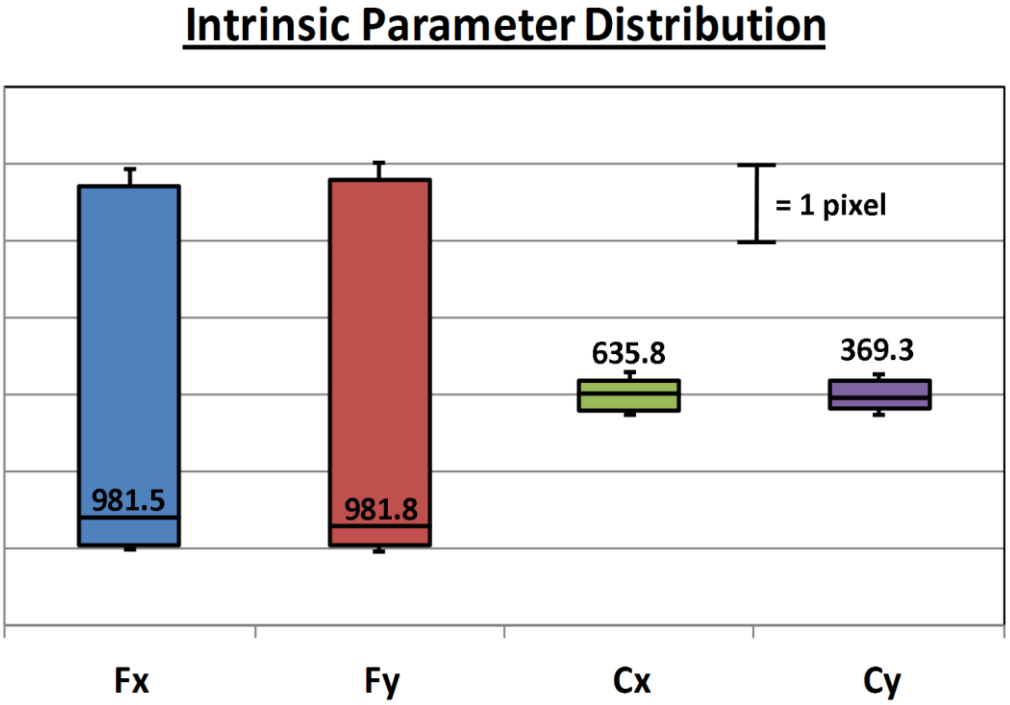


Figure 7.20: Distributions of intrinsic parameters

The parameters above have been centred about their mean to allow plotting upon a single figure. The median value of each parameter is labelled.

The mean values of the five distortion parameters are listed below in Table 7.6, their distribution is then presented by Figure 7.21. The values are scaled as a fraction of their mean to allow plotting upon a single figure.

k_1 (e-3)	k_2 (e-1)	p_1 (e-4)	p_2 (e-4)	k_2 (e-1)
3.13±1.57	1.01±0.07	-8.16±0.41	-9.08±0.56	-2.07±0.14

Table 7.6: Mean values of distortion parameters

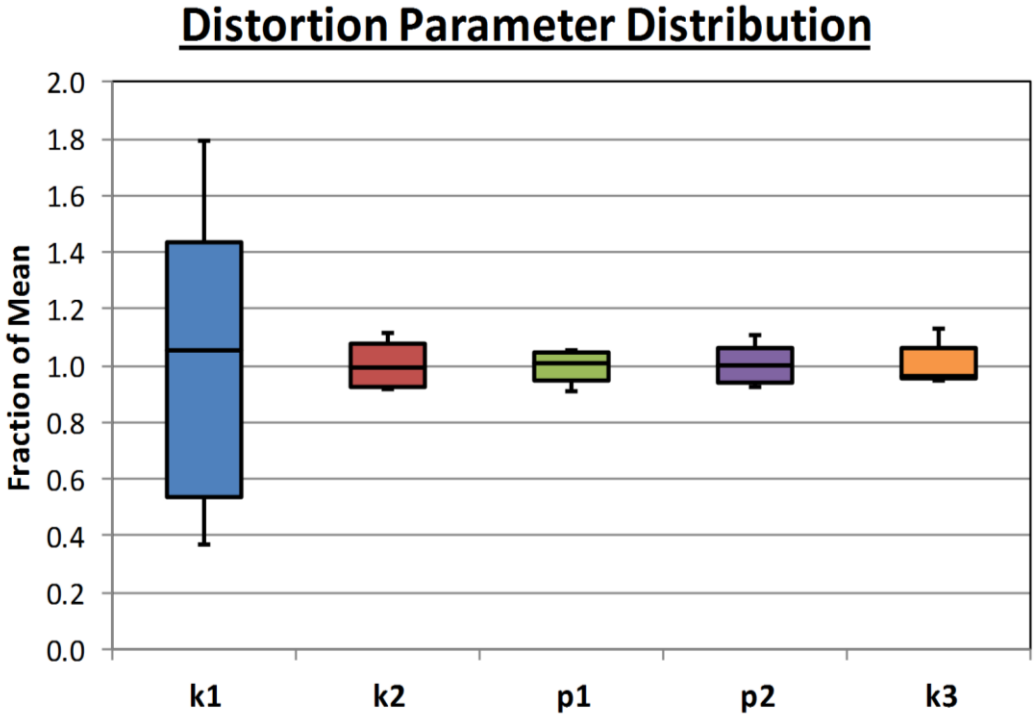


Figure 7.21: Distributions of distortion parameters

7.3.1.3 Discussion

As with the first generation system the intrinsic parameters are compared to their expected values. It is found that both focal parameters agreed with the predicted value of 957 (CV(RMSE) = 0.027 and 0.028, respectively). The X and Y focal parameters presented standard deviations of 2.23 and 2.29 pixels, respectively. These were found to be smaller (P = 0.05) than the first generation system. Likewise the image centres were found to agree with their predicted values of 640 and 360 (CV(RMSE) = 0.007 and 0.026). The standard deviations 0.19 and 0.19 were both found to be significantly smaller than those obtained by the first generation system (P = 0.02 and P < 0.01). The spread of these data are very low, with a range of 0.57 and 0.55 pixel respectively

Based upon the consistence of calibration produced by the two systems it is concluded that the second generation system most likely offers improved intrinsic calibration to the first generation system. Further to this it is recalled from Section 6.4.2.1 that the first generation failed to provide meaningful distortion characterisation, returning zero distortion parameters for each calibration. The second generation system successfully generated values for the five distortion parameters discussed by Section 7.3.1.1 above. The distributions of these five parameters are plotted by Figure 7.21 above. Due to the range of magnitudes returned for the parameters, as shown by their mean values listed in Table 7.6, the distribution of the parameters is shown as the fraction of their mean result.

It may be observed from Figure 7.21 that the latter four distortion parameters produce relatively low distributions. However, the first parameter, k_1, produces a significantly larger distribution ($P < 0.01$). Based upon Equation 7.1 it is seen that k_1 is related to the r-squared radial distortion term. Variance of this term therefore has the least effect on distortion compensation. The variance observed for this term is found to produce a significantly smaller variance to the final distortion correction result than the remaining two radial terms ($P < 0.01$).

While limited variance of the calibration parameters is an indicator of consistent calibration it does not directly indicate good calibration. The following sections analyse the accuracy of the system. The mean values of the five calibrations, shown by Table 7.5 and Table 7.6 above were used for all experiments. The small distribution of the results justified combining them to obtain optimal values.

7.3.2 Angular Analysis

As discussed in Section 6.4.4 and based upon preliminary experimentation with the second generation system, there was suspected to be an angular dependence of the accuracy of marker pose estimation. The following series of experiments were designed to investigate this angular dependence by imaging markers over a wide range of known angles.

Angular accuracy is an extremely important feature of tracking systems, and one occasionally omitted from literature. For objects distal from the marker origin a small rotational error may be readily amplified to a significant positional error.

7.3.2.1 Materials and Methods

A rotation may be expressed as three components, rotation about the X, Y, and Z-axis. It was decided to investigate the effect of each of the components upon the accuracy of the system, both separately and combined. Therefore, an experimental rig was designed such that a marker could be rotated fully through these three axes. The design requirements of the rotational rig are listed below by Table 7.7.

Angular Rig Requirements
180° rotation about X and Z-axis of marker
360° rotation about Z-axis of marker
All rotational origins equal to marker origin (centre)
Precise readout for each axis
Orientation lockable
Multiple markers (optional)

Table 7.7: Requirements of angular analysis experimental rig

Based upon the requirements listed by Table 7.7 above the experimental rig shown by Figure 7.22 was developed.

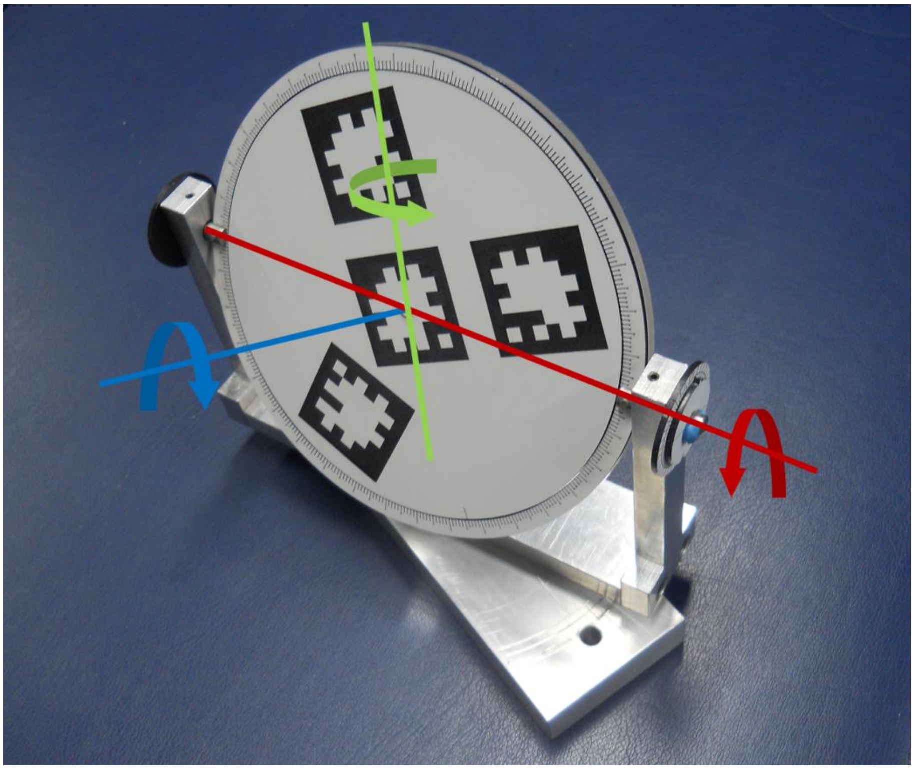


Figure 7.22: Angular analysis experimental rig

The X and Y-axes could be rotated fully and could be set accurately using alignment marks are 5° intervals. The orientation of both axes could be locked. The Z-axis could also be rotated fully and set accurately using a laser engraved 1° scale. All the axes of rotation were centred upon the origin of the central marker. The split rod of the X-axis ensured the correct centring. The ability to correctly set the angle was validated against known geometry and a digital inclinometer.

The optical axis of the camera and Z-axis of the central marker were carefully aligned with each other, using a process similar to that discussed in Section 6.4.2. The standard experimental assembly is shown by Figure 7.23 below.

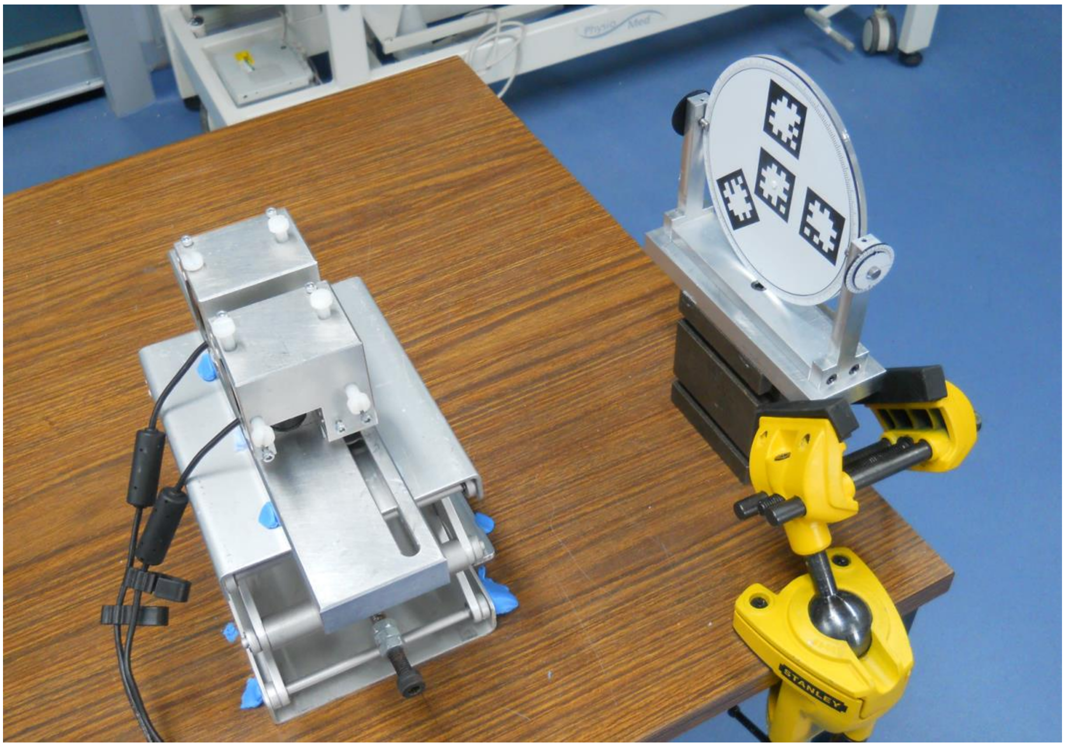


Figure 7.23: Standard configuration from angular analysis

In addition to the angular dependency investigated Sections 6.4.3.3 and 6.4.4.4 suggested separation and image region dependencies, respectively. To analyses these relations the camera-marker separation was varied over a 70 mm range, and the camera-marker alignment was vertically shifted over a 100 mm range.

The range of configurations utilised are summarised by Table 7.8 below.

Separation (mm)	X-Axis Displacement (mm)	Y-Axis Displacement (mm)
200	0	0
175	0	0
150	0	0
130	0	0
175	0	+25
175	0	+50
175	0	-25
175	0	-50

Table 7.8: Summary of experimental positions

Three versions of measurements were performed, full, mini, and spin. A full measurement varied the X and Y-axes between -90 and 90° in 5° increments, and the Z-axis between 0 and 360° in 10° increments. Combined angles were then performed in 30° increments for the X and Y-axes between -60 and 60° and 120° increments for the Z-axis. A mini measurement doubled these increments using 10° for the X and Y-axes and 20° for the Z-axis. Combined angles were instead performed at 45° and 120° for the Z-axis. Finally, for spin measurements the Z-axis was rotated between 0 and 360° for each incline about the X-axis in 15° intervals between 60 and 0°.

Full tests were designed to give a detailed characterisation of the angular dependence of the system, while mini tests were implemented when less detailed characterisation was sufficient. Finally, spin tests were designed to mimic the probing tests, described in Section 7.3.3 below, allowing separation of the system and probing accuracy.

7.3.2.2 Results

Figure 7.24 summarises a single repeat of a centred full measurement performed at 175 mm separation. Each of the 184 angular positions is plotted as a scaled sphere. The spheres are colour coded to indicate the RMSE of the orientation, as shown by the colour bar. White spheres illustrate orientations at which the system failed to detect the central marker. The spheres are then scaled in the three orthogonal axes based upon the standard deviation of the 100 samples taken for each orientation.

Figure 7.25, Figure 7.26, and Figure 7.27 express the same data set with the RMSE divided into its three orthogonal X, Y, and Z components, respectively.

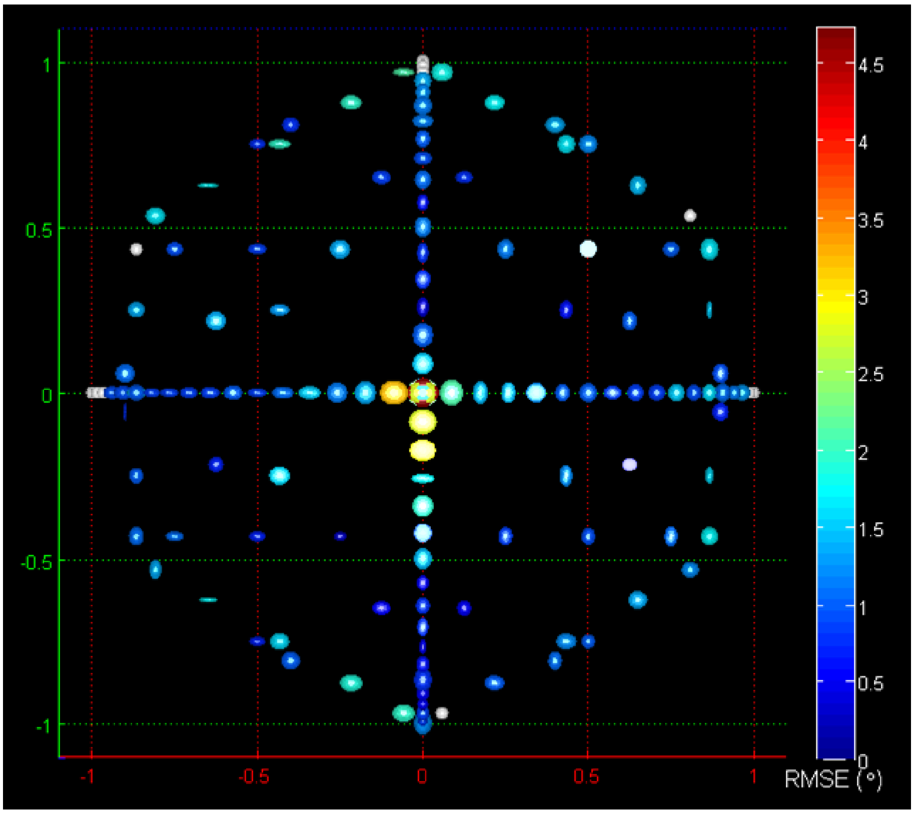


Figure 7.24: Summary figure of rotational RMSE performed at 175 mm separation and zero lateral displacement.

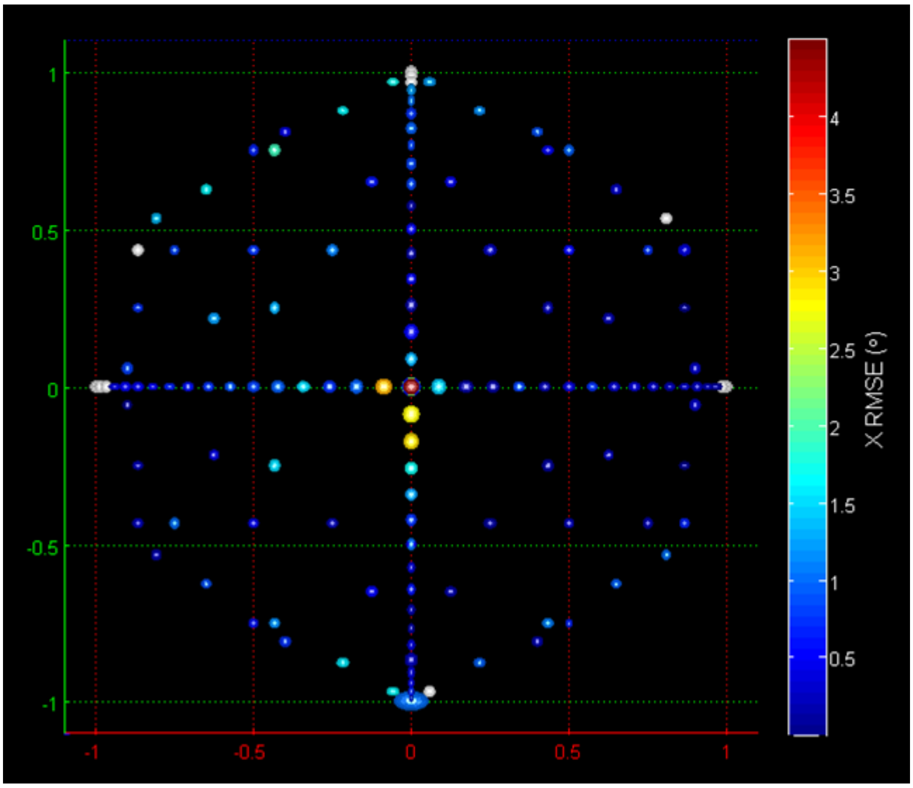


Figure 7.25: Summary figure of rotational X RMSE performed at 175 mm separation and zero lateral displacement.

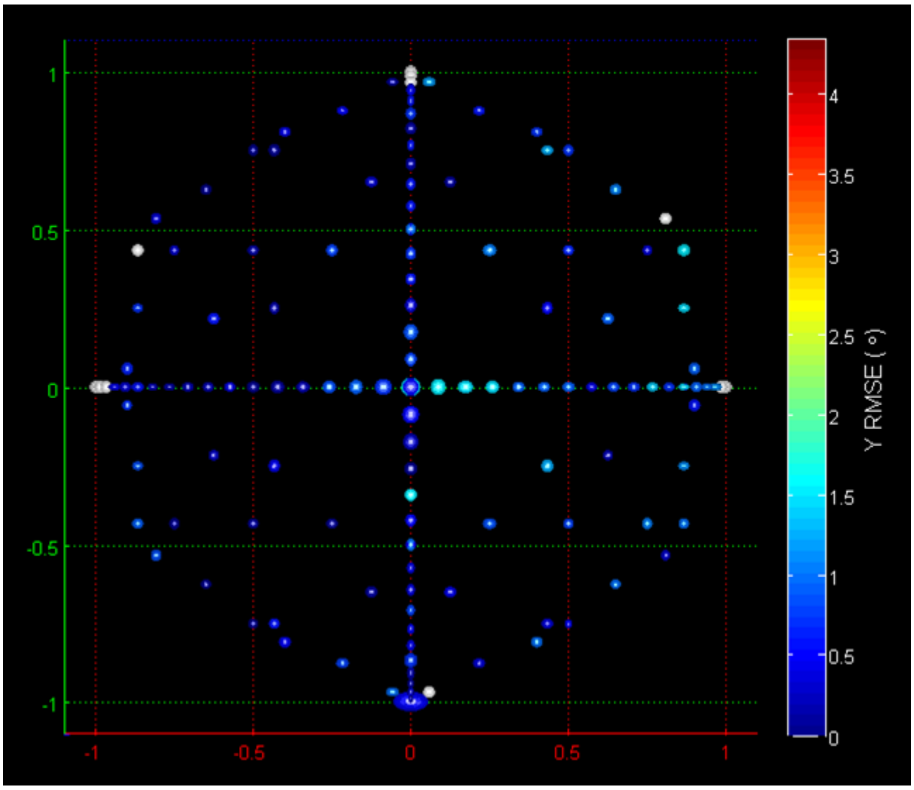


Figure 7.26: Summary figure of rotational Y RMSE performed at 175 mm separation and zero lateral displacement.

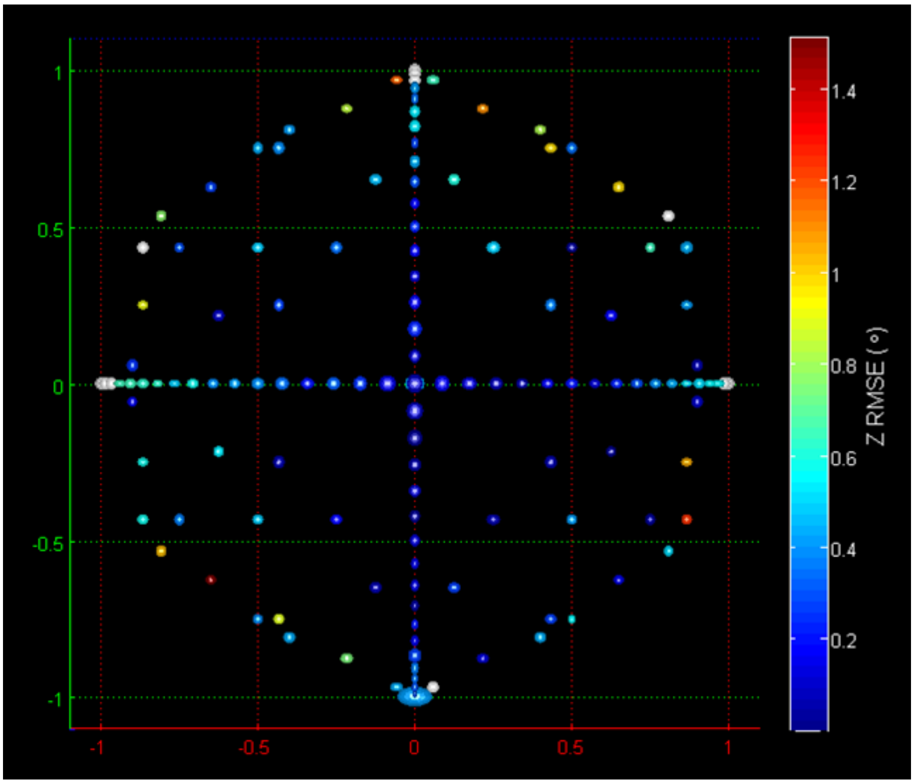


Figure 7.27: Summary figure of rotational Z RMSE performed at 175 mm separation and zero lateral displacement.

Figure 7.24 strongly suggest an RMSE dependency upon marker orientation as noted by the increased error towards the centre of the figure. Furthermore, it is noted by Figures Figure 7.25, Figure 7.26, and Figure 7.27 that the three angular components are affected to differing degrees. Whereby, the X and Y components present the largest errors at near-parallel angles, while the Z component indicates the opposite behaviour, with errors increasing away from the centre.

To highlight these dependencies Figures Figure 7.28, Figure 7.29, and Figure 7.30 plot the RMSE of the three components and their RMS total during rotation about the X, Y, and Z-axes respectively.

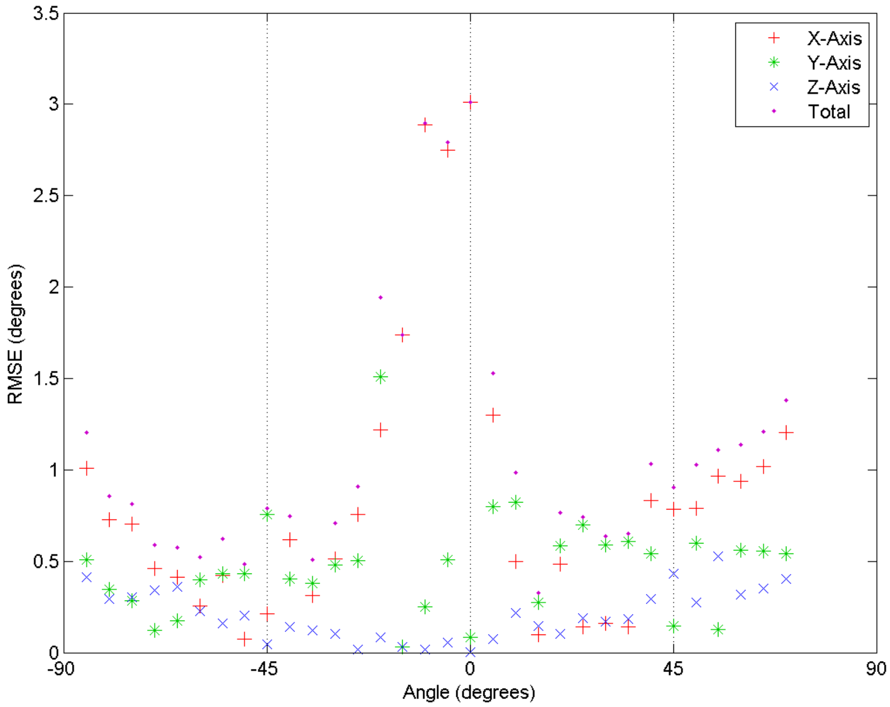


Figure 7.28: Effect of rotation about the X-axis upon component and total RMSE

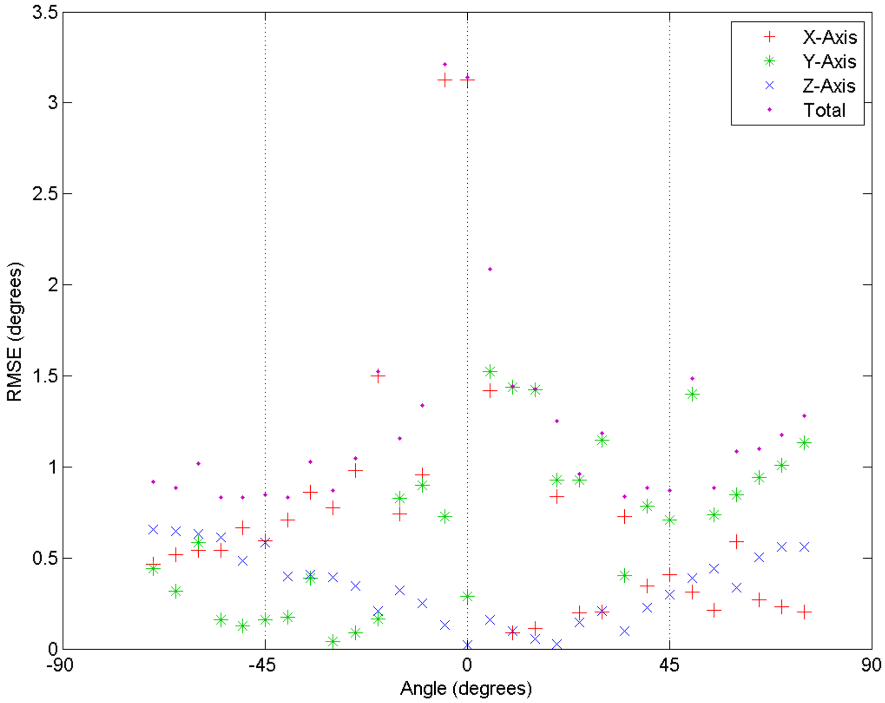


Figure 7.29: Effect of rotation about the Y-axis upon component and total RMSE

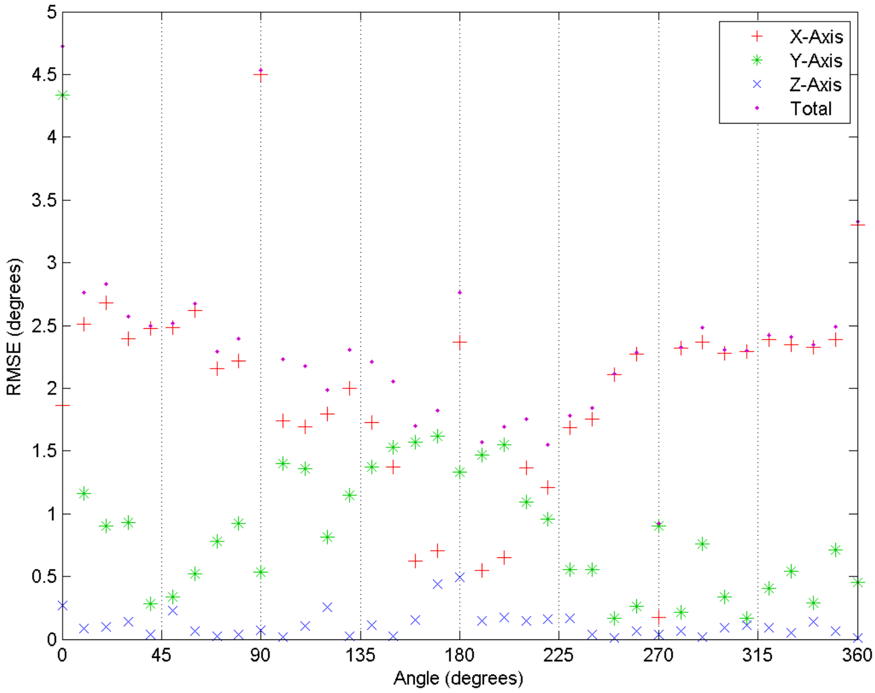


Figure 7.30: Effect of rotation about the Z-axis upon component and total RMSE

Figures Figure 7.28 and Figure 7.29 confirm the increase in error at near parallel angles for the X and Y components, in addition to the inverse behaviour of the Z component. Errors for near-parallel orientations, defined as begin between -30 and

30°, were found to be significantly larger than the remaining samples for the X component during rotation about the X-axis (P = 0.04) and Y-axis (P = 0.03). However, over this range the Y component was not shown to be significantly larger during either X rotation (P = 0.13) or Y rotation (P = 0.13). Upon reducing the definition of the near parallel range to -15 to 15° the Y component was found to be significantly larger during Y rotation (P = 0.03) but not during X rotation (P = 0.25). Increases in both X and Y components during X and Y rotations, respectively, are observed at far-parallel orientations, defined as outside of the -60 to 60° region. However, these failed to reach significance (P = 0.07 and P = 0.08). The Z component, however, was shown to be significantly larger in this region during both X and Y rotation (P = 0.04 and P = 0.01). During the analysis of these regions the near-parallel region was omitted from the base data.

Under the -30 to 30° near-parallel definition the near-parallel Z components was found to be significantly smaller than the remaining samples, during both X rotation (P < 0.01) and Y rotation (P < 0.01).

During rotation about the Z-axis the X component was found to be significantly less accurate (P < 0.01) when upright, between the angles of 90 and 270°, than when inverted. The Y component reversed this relation, being significantly more accurate (P < 0.01) when upright. The Z component showed no significant difference over this range (P = 0.18).

It is noted from Figure 7.30 that samples taken at multiples of 90° present as possible outliers to the general trend. These samples were found to have a significantly larger variance than the remaining data (P < 0.01).

Figure 7.24 also indicates an effect of marker angle upon the precision of the system, measured as the standard deviation. This is illustrated for each rotation more clearly by Figures Figure 7.31, Figure 7.32, and Figure 7.33 below.

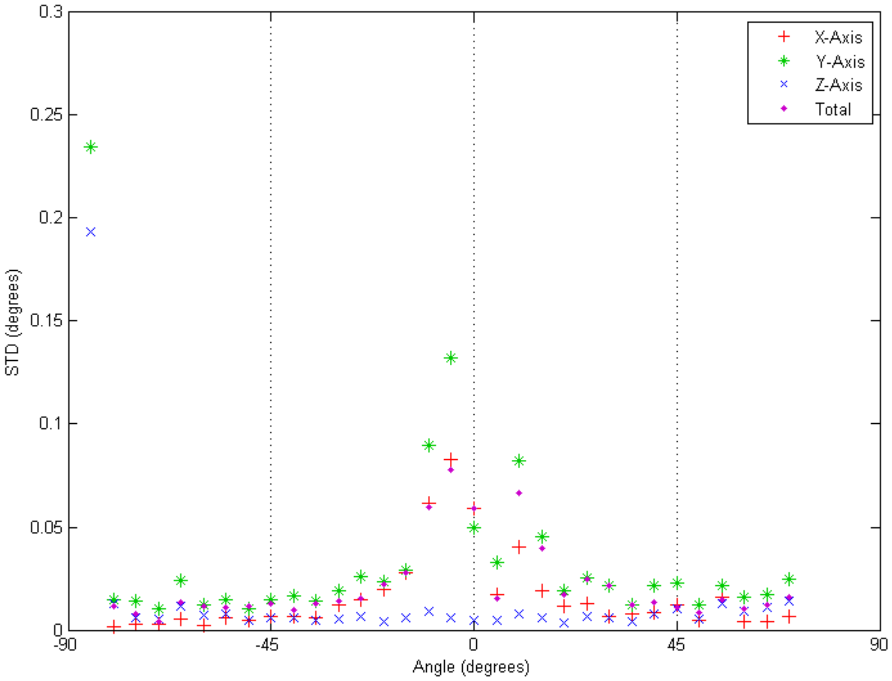


Figure 7.31: Effect of rotation about the X-axis upon component and total standard deviation

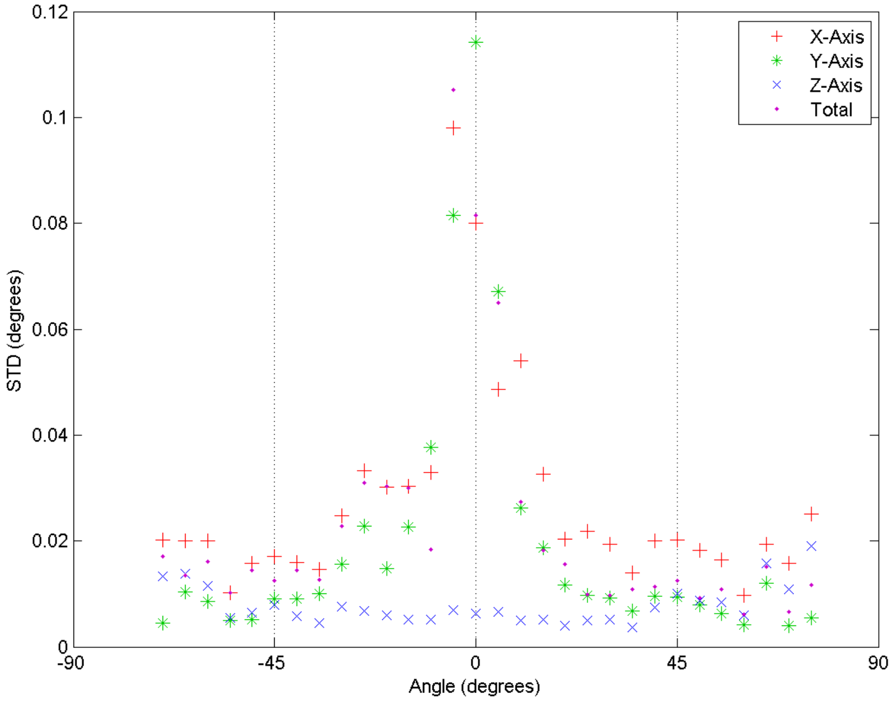


Figure 7.32: Effect of rotation about the Y-axis upon component and total standard deviation

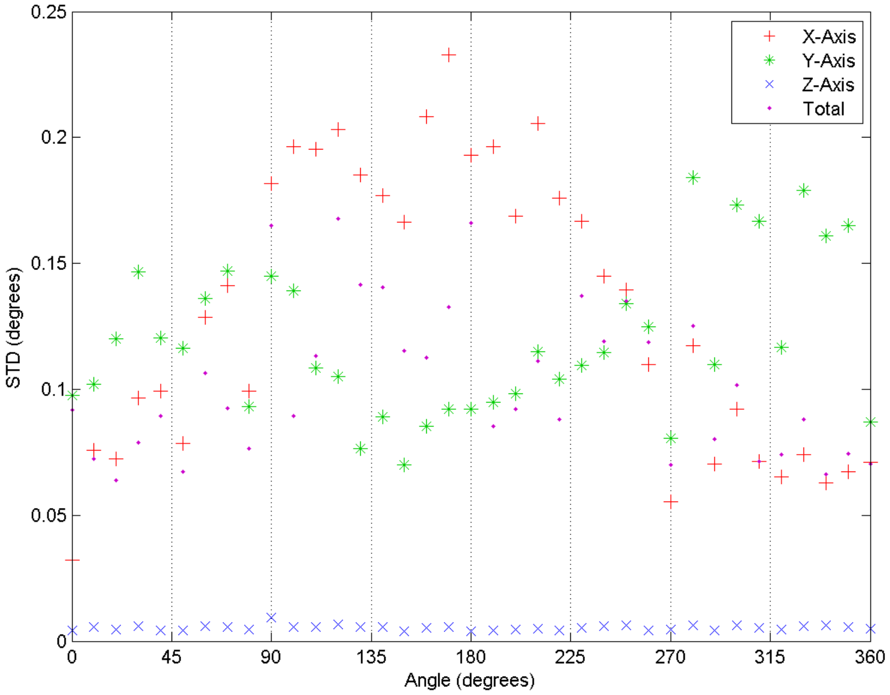


Figure 7.33: Effect of rotation about the Z-axis upon component and total standard deviation

Statistical analysis upon the standard deviation showed that both the X and Y components were significantly larger in the near-parallel range during both X and Y rotation (All P < 0.01). The inverse relation of the Z component was also found to be significant over both rotations (Both P < 0.01). During rotation about the Z-axis the X component was found to be significantly more precise when upright (P < 0.01), while the Y component was found to be more precise when inverted (P < 0.01). Neither the Z component (P = 0.42) nor the 90° multiple samples (P = 0.13) presented significant relations.

Both RMSE and standard deviation statistical analyses were repeated for the remaining two repeats of the standard measurement configuration. The same results were found bar the Y component RMSE value, which was found to be significantly larger for three of the four rotations over the larger near parallel range of -30 to 30°. The fourth rotation again only reached significance over the reduced -15 to 15° range.

Figures Figure 7.34 and Figure 7.35, respectively, show the RMSE and standard deviation distribution across the three axial components. The data is divided into rotation modes. Firstly, the pure rotations about the X, Y, and Z-axes are shown separately, as illustrated above. Secondly, these results are combined into single mode rotation. Thirdly, rotations involving two or more axial components are grouped. Finally, all samples are grouped into total results.

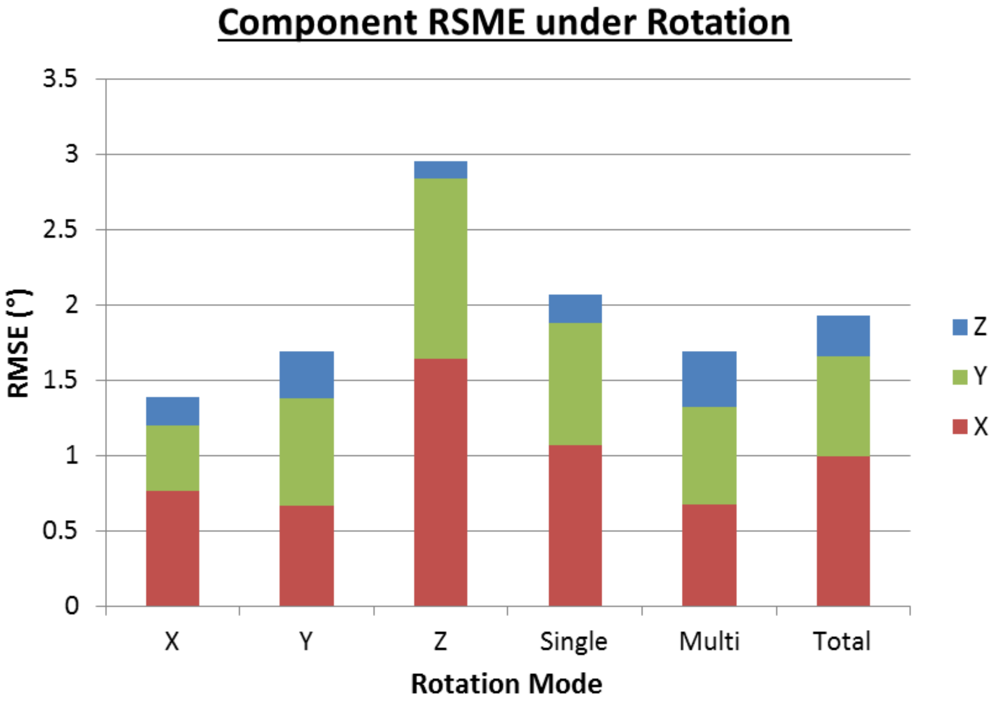


Figure 7.34: Component mean RMSE during different rotation modes

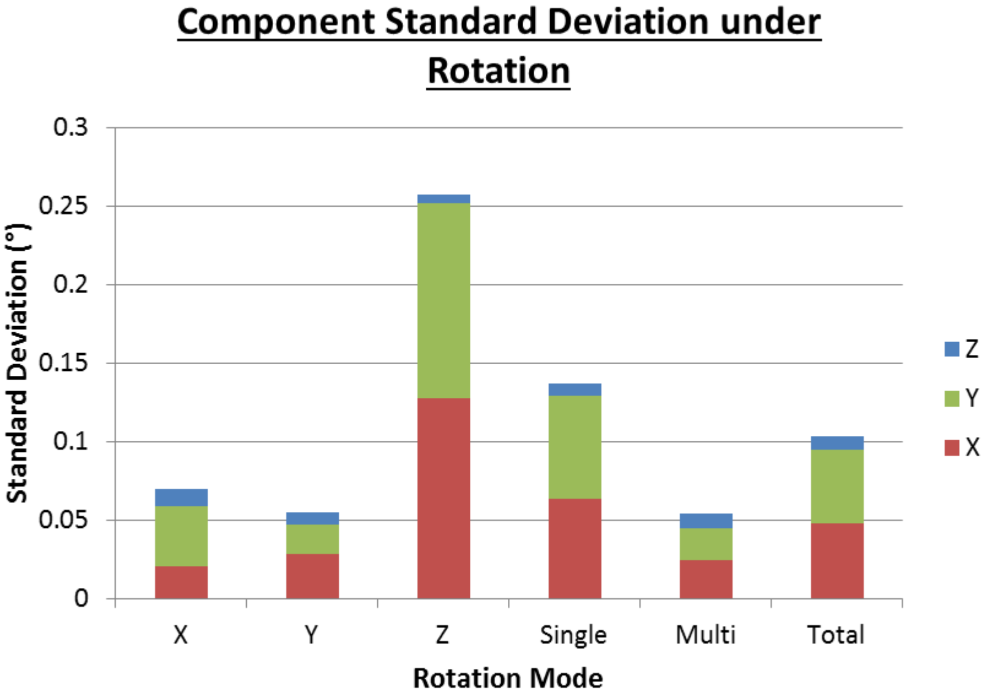


Figure 7.35: Component mean standard deviation during different rotation modes

The RMSE and standard deviation results of the spin data set are shown below by Figures Figure 7.36 and Figure 7.37 respectively.

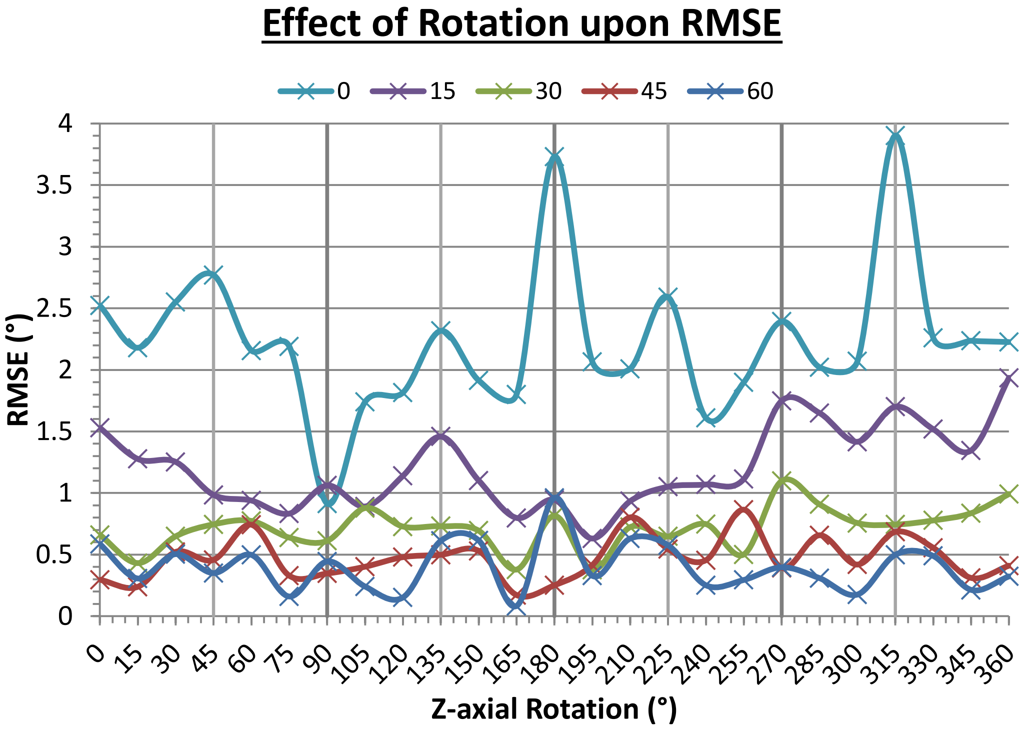


Figure 7.36: Spin data RMSE results illustrating the effect of Z-axial rotation upon a range of X-axial rotations

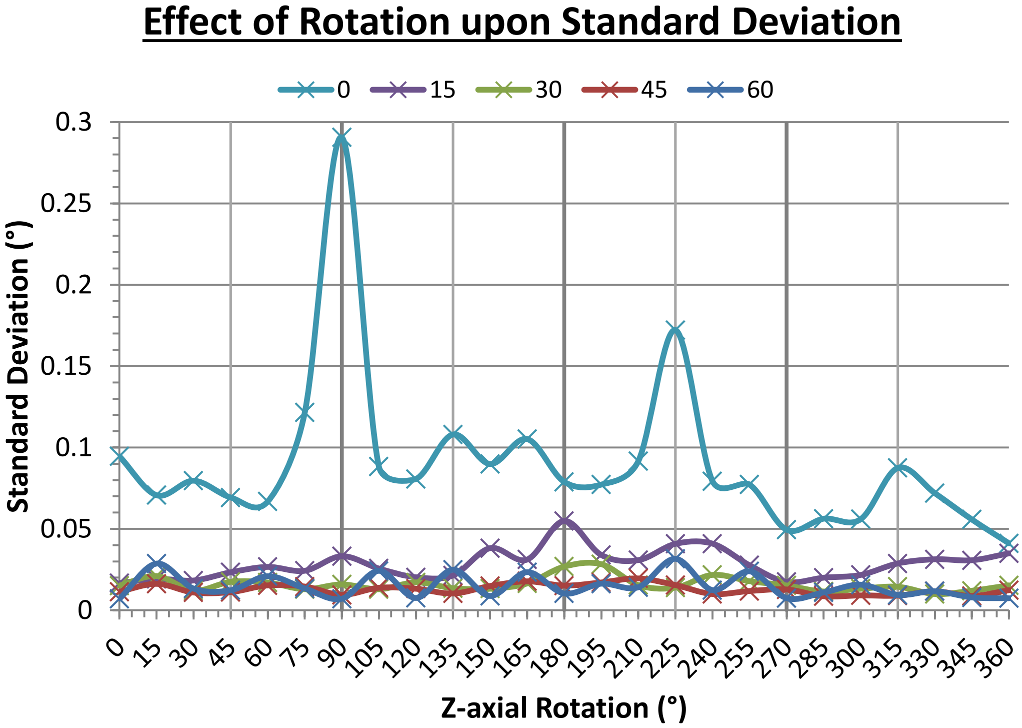


Figure 7.37: Spin data standard deviation results illustrating the effect of Z-axial rotation upon a range of X-axial rotations

Figures Figure 7.36 and Figure 7.37 both indicate an increase of error as the X-axial angle tends to zero. Statistically, each increment from zero resulted in a significant decrease in both RMSE and standard deviation, excluding the final increment from 45 to 60° (All P < 0.01, excluding P = 0.09 and 0.11 respectively).

Several error peaks are noted within both figures. These are observed to occur more frequently at multiples of 45°. The significance of these peaks were calculated at multiples of 45 and 90° with an initial offset of zero and 45°, for both RMSE and standard deviation. Of the 30 values calculated, six reached significances at a 95% confidence interval. The significance values failed to present a pattern.

The effect of camera-marker separation upon angular RMSE and standard deviation is shown by Figure 7.38 below.

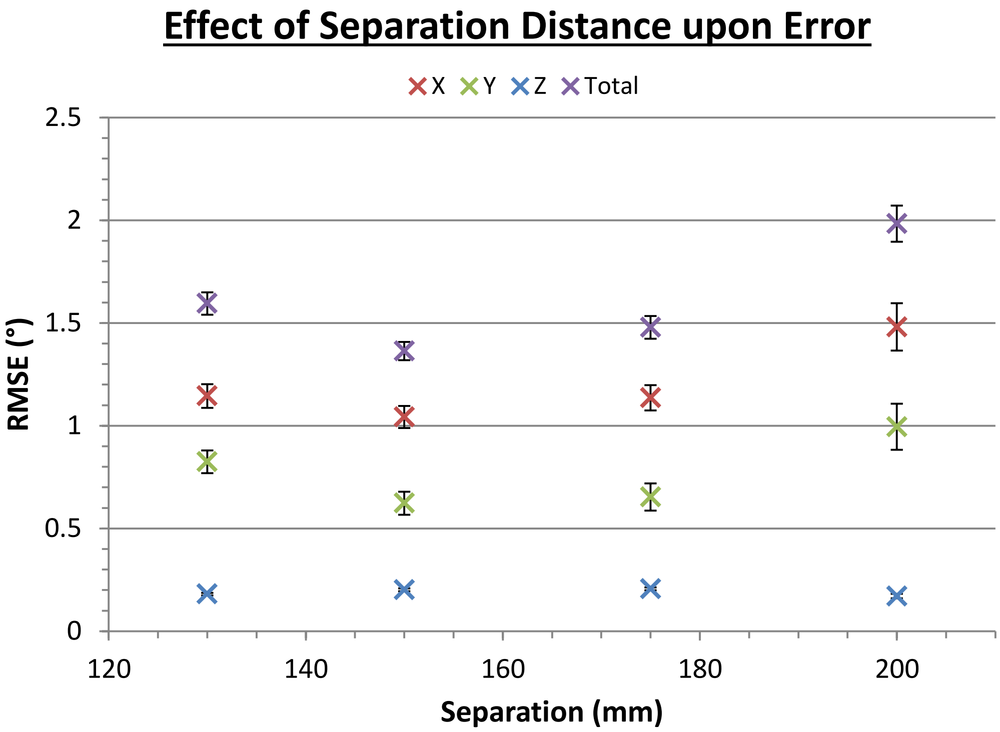


Figure 7.38: Effect of camera-marker separation upon angular accuracy and precision

Figure 7.38 presents a decrease in both accuracy and precision between 150 and 200 mm separation. However, the measurement performed at 130 mm separation contradicts this trend.

Figure 7.39 shows the effect of vertical offset, and therefore the effect of marker image location, upon accuracy and precision.

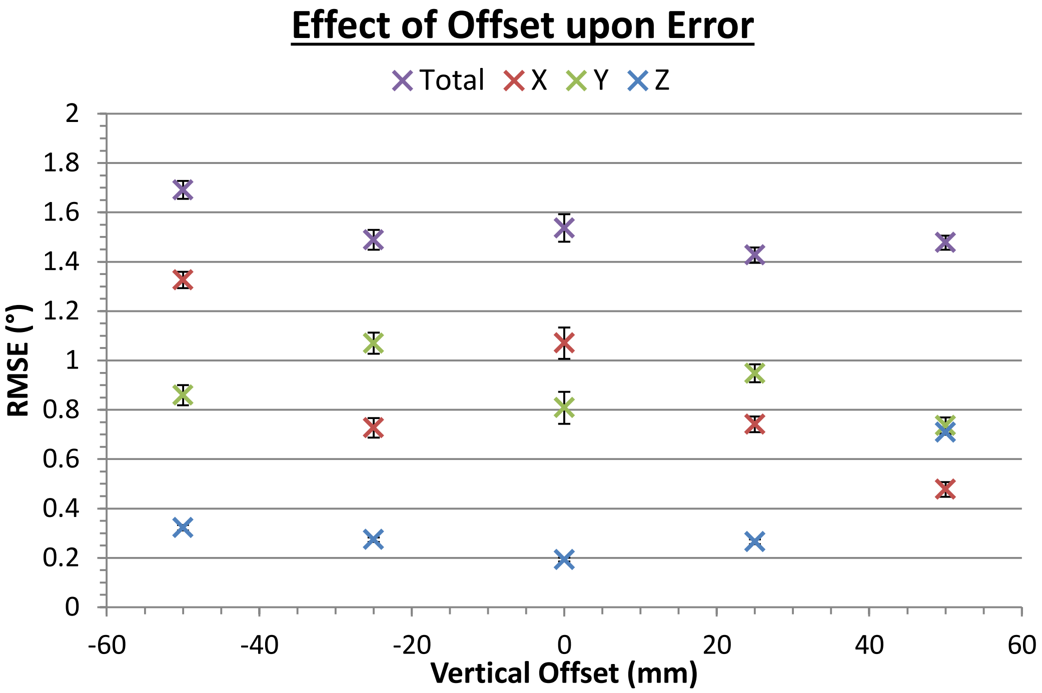


Figure 7.39: The effect of vertical offset, or image position, upon angular RMSE

No accuracy trend is readily visible from Figure 7.39. However, it is observed that the zero offset samples produced a significantly larger variance in all three components than the offset samples (All P < 0.01). Furthermore, in contrast to zero offset samples, the Y component presents a larger variance than the X component, although the difference failed to reach significance (P = 0.33).

7.3.2.3 Discussion

As first noted by Figure 7.24, and expanded upon by Figures Figure 7.25 to Figure 7.33, the system presented a clear angular dependence, whereby near-parallel alignments produced both the largest inaccuracies and variance. The same observation was reported by Kato and Billinghurst and Abawi *et al.* during evaluation of the ARToolKit (Kato & Billinghurst 1999; Abawi et al. 2004). It is difficult to directly compare results as both previous studies used larger markers, 80 and 55 mm respectively, and different working distances. However, both studies report maximum angular errors in excess of 15°, far larger than the maximum 5°

error observed during these experiments. Additionally, Abawi *et al.* reported upon standard deviations in excess of 4° for near parallel angles. Again, this is far larger than the 0.5° maximum observed during the 200 mm separation experiment. To further complicate comparison the reported systems used lower camera resolutions. Abawi *et al.* made use of a 640x480 pixel camera in comparison to the 1280x720 used here. While previous discussion has concluded that the accuracy to resolution relation is not directly proportional, due to the application of sub-pixel algorithms, the accuracy of the system is still affected.

The apparent improvements of the presented system over the ARToolKit system is likely three fold. Firstly, the increased camera resolution as discussed. Secondly, the improved camera calibration method utilised by the presented system. Thirdly, the extended iterative pose estimation used by the new system should improve the consistency of results, likely having a major effect on the large variance difference seen.

Although highlighted as a probable cause of system error by Section 6.5 , it is not suspected that the improvements made to image thresholding had a significant effect upon the result. It is likely that both reported studies were performed under controlled lighting. Therefore, gradient lighting, which produces poor results under global thresholding, was unlikely to be present.

The increase in both error and standard deviation observed at near-parallel sampling is explained by Figure 7.40 below.

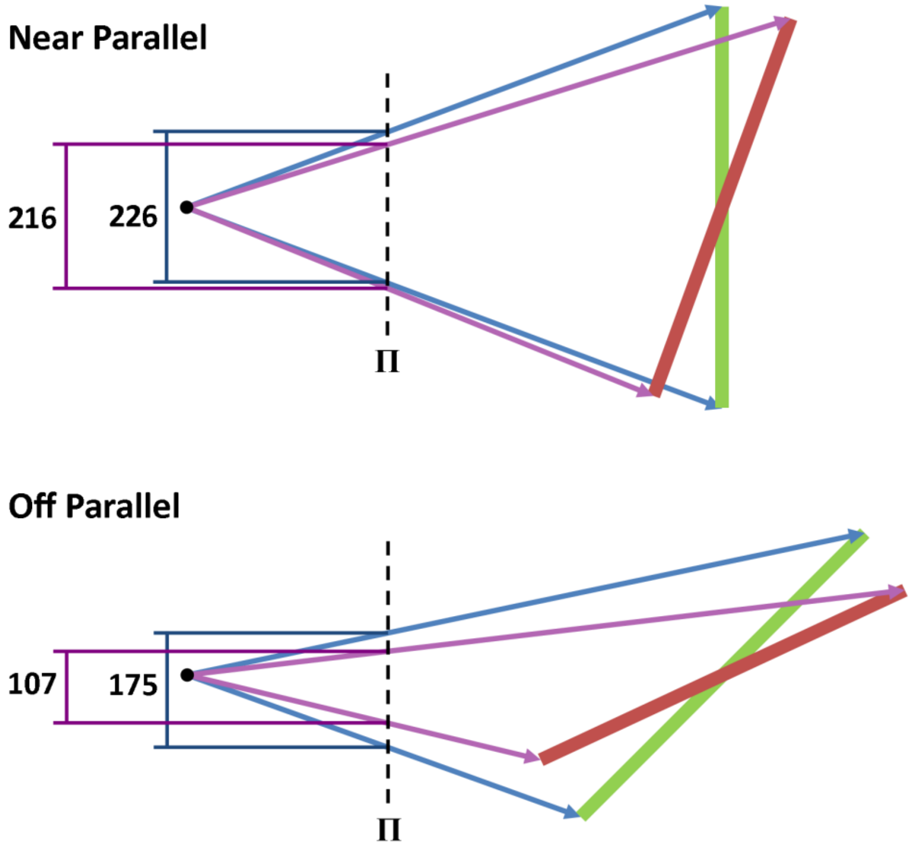


Figure 7.40: Effect of marker orientation upon marker image

Figure 7.40 illustrates the effect of two different marker rotations. The first, in the near-parallel region, shows a marker rotated from parallel (green) to 20° (red). The second shows the same 20° rotation, only from an initial off parallel angle of 45°. The height of the image produced by each marker before and after rotation is shown. It is seen that the near-parallel rotation produced an image height difference of 10 pixels, while the off parallel rotation produced a difference of 68 pixels. The system is based upon the reversed pinhole model. Therefore, the reverse of Figure 7.40 indicates that a small corner detection error would result in a larger error in the near-parallel region.

Figure 7.39 presents the expected over-unity increase in error with distance. However, the measurement performed at 130 mm does not agree with the trend. It is observed that the standard deviation of this measurement does match the trend. This suggests a potential initial alignment error. The alignment procedure used features on the side of the angular rig to centre the marker. The reduced range of the 130 mm measurement caused these side markings to be considerably blurred. Therefore, slight misalignment may have occurred.

The general increase in error and standard deviation is also a result of image error. As the camera-marker separation is increased the marker occupies less image pixels. Therefore, a rotation results in a reduced change in pixels. As such, at greater separation an image error results in a larger rotational error.

Regressions applied to Figure 7.39 failed to produce high R-squared values, indicating that vertical displacement did not substantially affect the accuracy of the system. Therefore, it is concluded that the system was well calibrated, with regards to lens distortion and image centre. The considerably larger zero offset error supports the discussion of Figure 7.40. This is further supported by the observed relative increase of the Y component error. The vertical displacement reduces the incidence of near-parallel orientations in the X-axis, but not the Y-axis. Therefore, the Y-axis still presented the larger errors associated with near-parallel measurements, unlike the X-axis.

The preceding experiment demonstrated the strong dependency of the accuracy of the second generation system upon the angle of the imaged marker, in addition to the dependence upon camera-marker separation. Finally, it indicated that the system was not considerably affected by vertical displacement. This information will allow better analysis of the probing experimentation below.

7.3.3 Probing Analysis

As discussed in Section 6.4.4 for the first generation system, probing or probe-like actions formed the core functionality of the system. Therefore experiments were designed to determine the probing accuracy of the system.

As with the first generation system these experiments were performed using a single marker attached to a target of known geometry.

7.3.3.1 Materials and Methods

Section 6.4.4.4 concluded that the target had not been machined to the desired accuracy. Therefore, for the second generation system a new probing target was designed. Based upon use of the first generation system new target design criteria were defined, these are listed by Table 7.9.

Probing Target Requirements
Manufactured to an accuracy of 0.1 mm
Hemispherical probing points
Minimum separation of 50 mm in each dimension
Rotational symmetry
Accurate marker placement

Table 7.9: Design requirements for second generation known geometry probing target

Hemispherical probing points were selected over the tapered points of the original target as it was observed that the probe tip could move during measurement, particularly during dynamic measurements where the camera was moved.

It was decided to move to a rotationally symmetric target as it was postulated that the complex shape of the original target masked information. For example as errors increased towards corners of the target it could not be determined if this was a result of the increased marker separation, or the change in viewing angle.

Figure 7.41 below shows the design of the second generation probing target that meets these requirements. To ensure the level of accuracy required the target was to be 3D printed using a Stratasys Object Eden 350 (Stratasys Ltd., MN, USA) printer with a nominal accuracy of 20-85 μm. The target was hollowed to reduce material costs but maintained a wall thickness of 6 mm to prevent deformation.

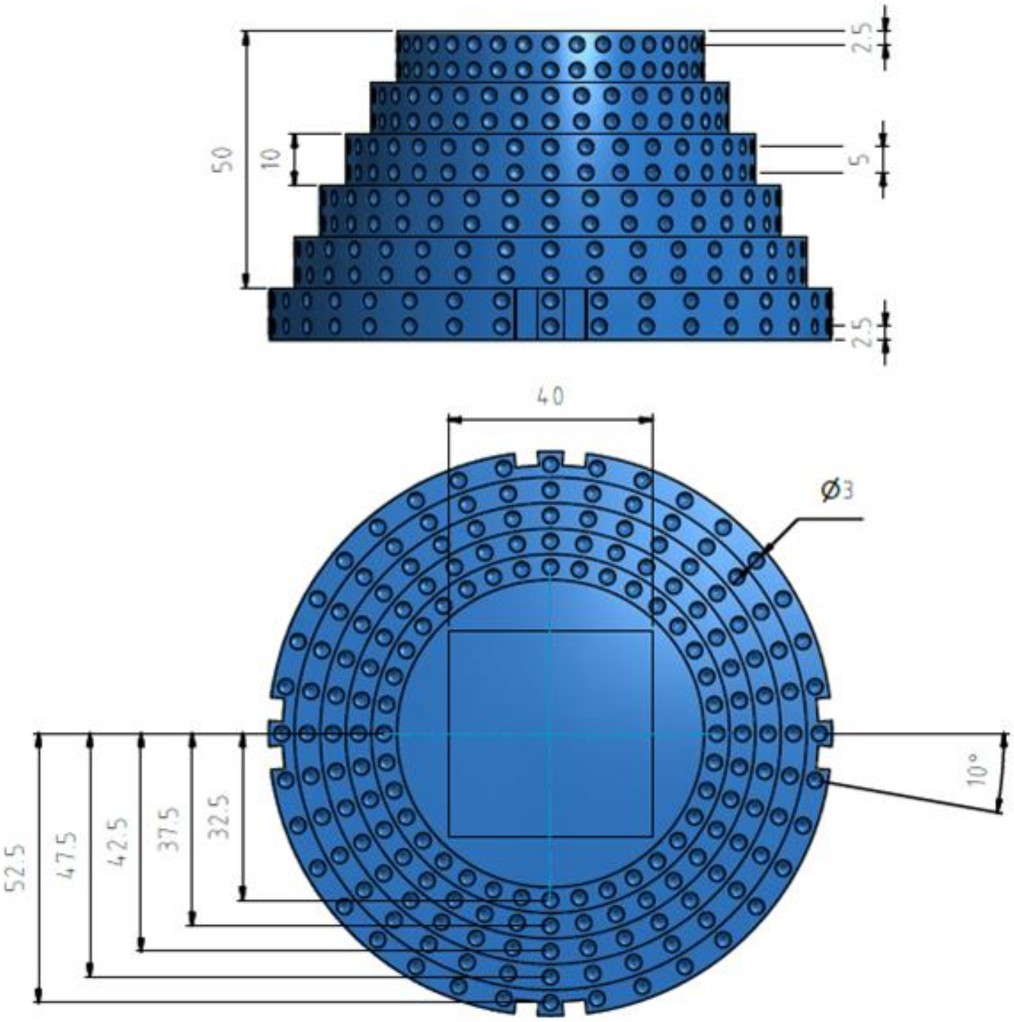


Figure 7.41: Second generation probing target design.

To ensure accurate marker placement a 40x40x1.6 mm square recess was placed within the target. This allowed a laser cut marker to be secured flush with the surface of the target. The accuracy of the marker pattern relative to its edges was measured using the Zeiss microscope. The marker was iteratively scaled and reproduced to obtain a sub-0.05 mm error for each corner.

The printed target is shown below by Figure 7.42. Calliper measurements supported the accuracy of the target, including the depth of the probe points and marker recess.

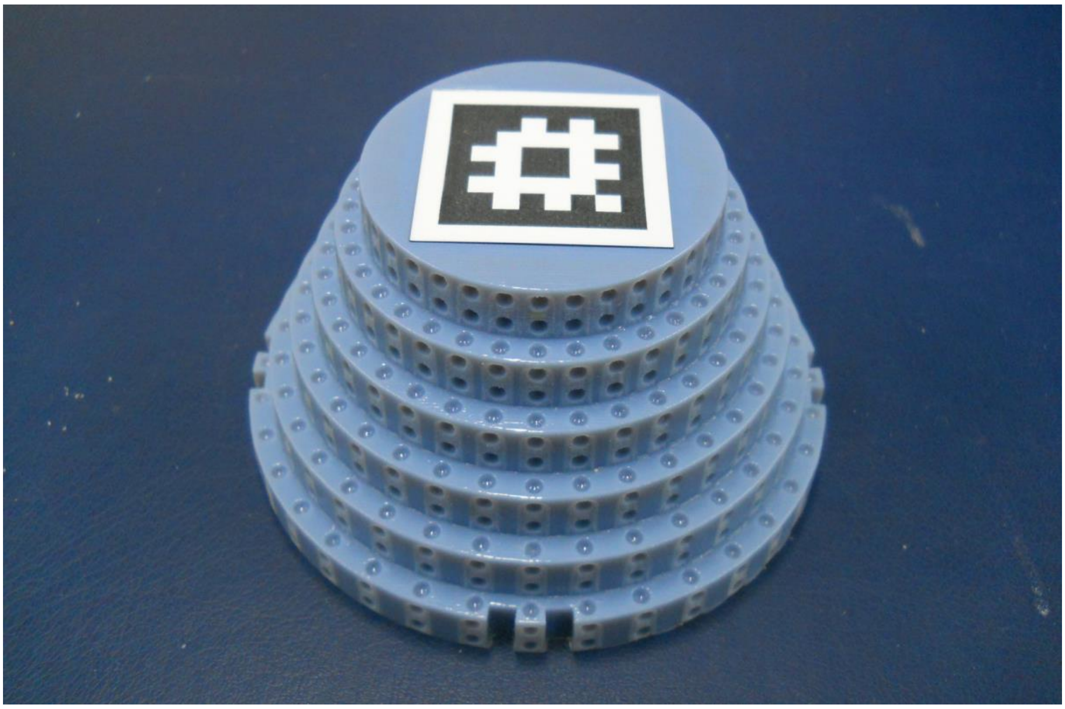


Figure 7.42: Second generation probing target and target marker

Probing tests were categorised by two criteria, point count and tool motion. Point count could be either full or mini. Full measurements used all 180 upper surface measurement points, while mini used every third point, providing 60 measurements. Tool motion was mounted, held, or moved. During mounted tests the tool was clamped in place. For held tests the tool was held by hand with minimum motion. Finally, for moved tests the tool was held by hand and actively moved in a pseudo-random pattern at a relatively constant rate and intensity. These dynamic measurements were included as Nafis *et al.* highlighted literature often reported exclusively upon static performed, which does not match the practical use of such systems (Nafis et al. 2006).

In theory due to the characteristics of the tracking system the peak accuracy should be found under static conditions, with increasing motion resulting in increased error. Therefore, the range of motion types allowed the testing of this theory.

The probe length was set to 120 mm and calibrated for each measurement. The tool was mounted for all calibrations. 200 sample calibrations were performed maintaining the central 90% of values for the final calibration. All calibrations were performed by placing the probe tip in the first target point at (0.0, -32.5, -10.0) mm, relative to the origin of the target marker. As with the first generation system, an augmented overlay, shown by Figure 7.43 below, was used to guide measurements.

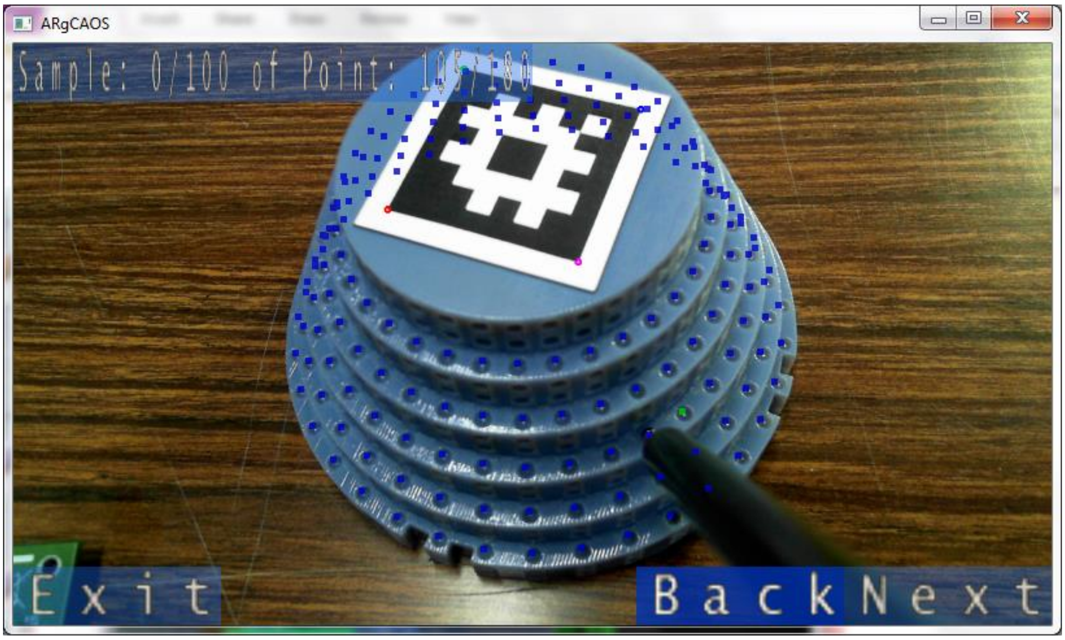


Figure 7.43: Second generation probing measurement augmented guidance.

100 samples were taken for each measurement point, with analysis performed offline in Matlab.

Three experiments were performed for each of the three motion modes. The three mounted experiments were performed as full measurements, while the remaining experiments were performed as mini measurements. The predominant use of mini tests was justified as they still produced a broad data set and would likely highlight any patterns which could then be further explored using full measurements.

After initial testing a potential system optimisation was implemented. The effectiveness of this was gauged by three further mounted mini experiments.

7.3.3.2 Results

Figure 7.44 below summaries the results of a representative result of the mounted samples. The RMSE of the measurement is indicated by colour while the dimensions of the sphere indicate the standard deviation of the measurement. As the standard deviations were found to be several orders of magnitude smaller than the dimensions of the target these was scaled by the value given in the heading of the figure for clarity.

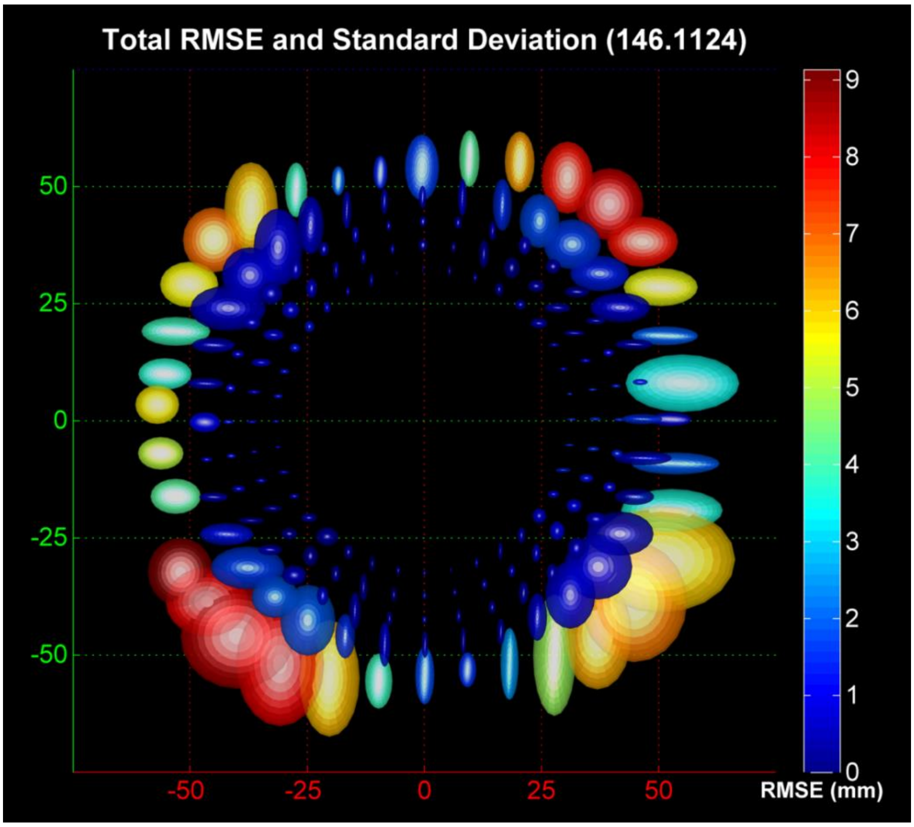


Figure 7.44: Total RMSE and standard deviation of mounted probing experiment. The three components of standard deviation have been scaled by a factor of 146.1

Inspection of Figure 7.44 presents two potential error patterns. Firstly, the error, both RMSE and standard deviation, is seen to increase at lower levels, or greater tip-marker separation. Secondly, the error pattern suggests an angular dependence. These potential patterns are expanded upon by Figures Figure 7.45, Figure 7.46, and Figure 7.47 below which show the X, Y, and Z signed axial error components respectively.

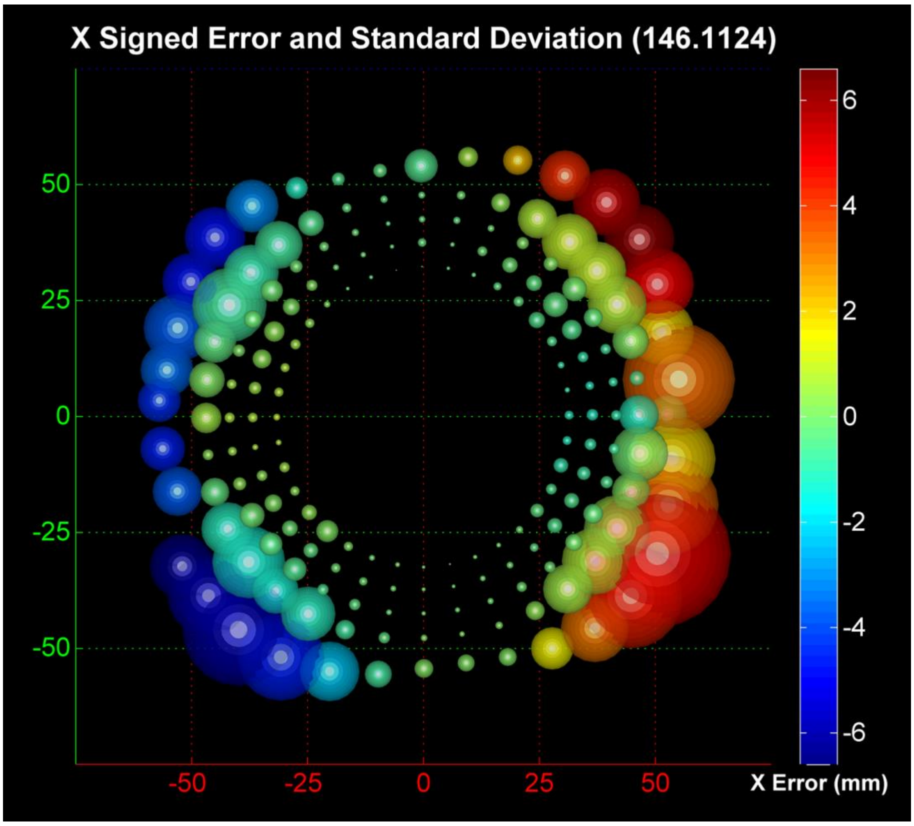


Figure 7.45: Signed X component of probing error

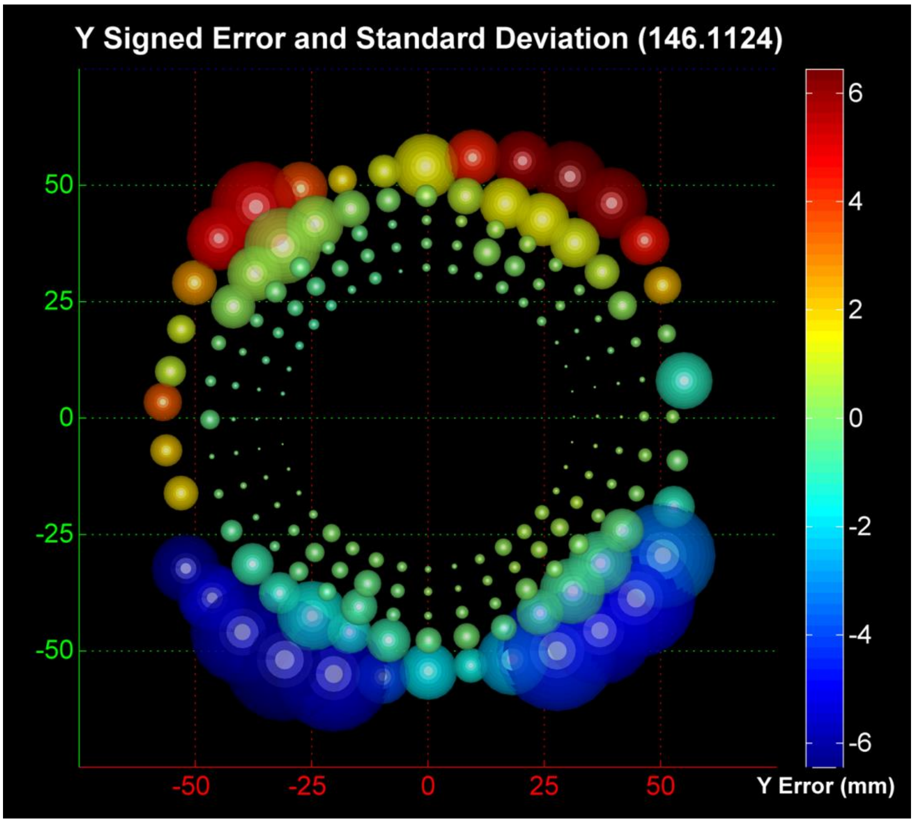


Figure 7.46: Signed Y component of probing error

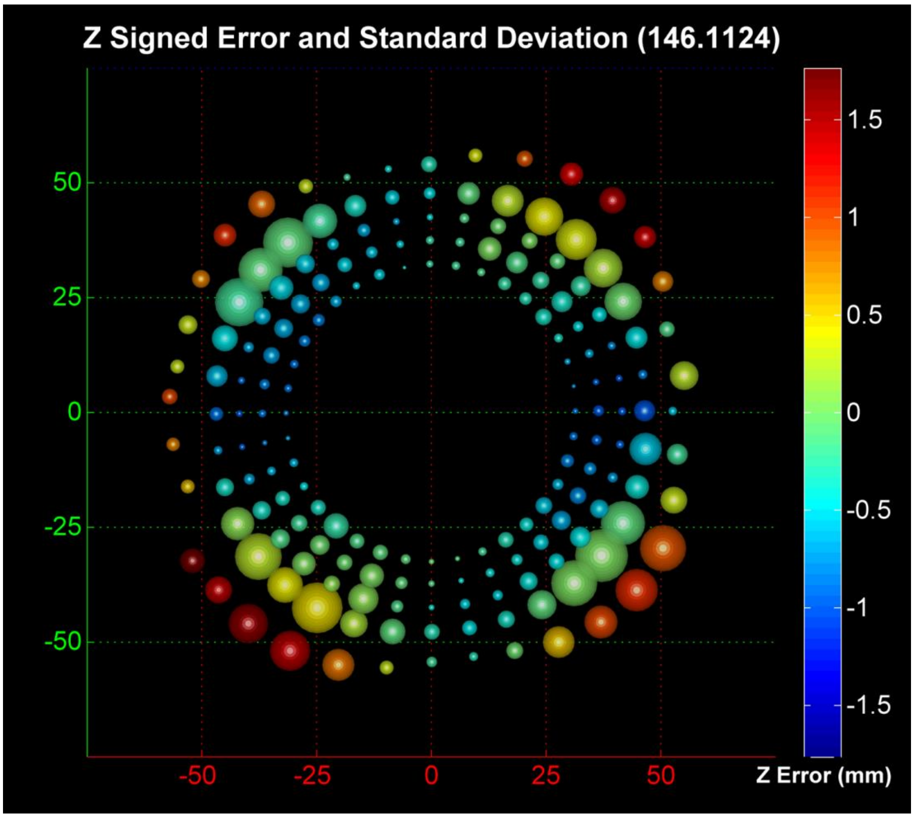


Figure 7.47: Signed Z component of probing error

The two error patterns are again seen in the figures above. The X and Y components present a symmetrical RMSE distribution. Figure 7.45 shows negative X points as being more negative (more to the left) than expected, while positive X points were reported as being more positive than expected. Figure 7.46 also shows this relation for the Y component in the Y-axis. The Z component shows central points as being either on target or more negative than expected. However, the most peripheral points are seen to be more positive (higher) than expected.

To investigate marker-tip separation or level relationship in the RMSE and standard deviation, data was averaged across levels, defined by their separation from the origin of the marker. Figure 7.48 below shows the error of each level.

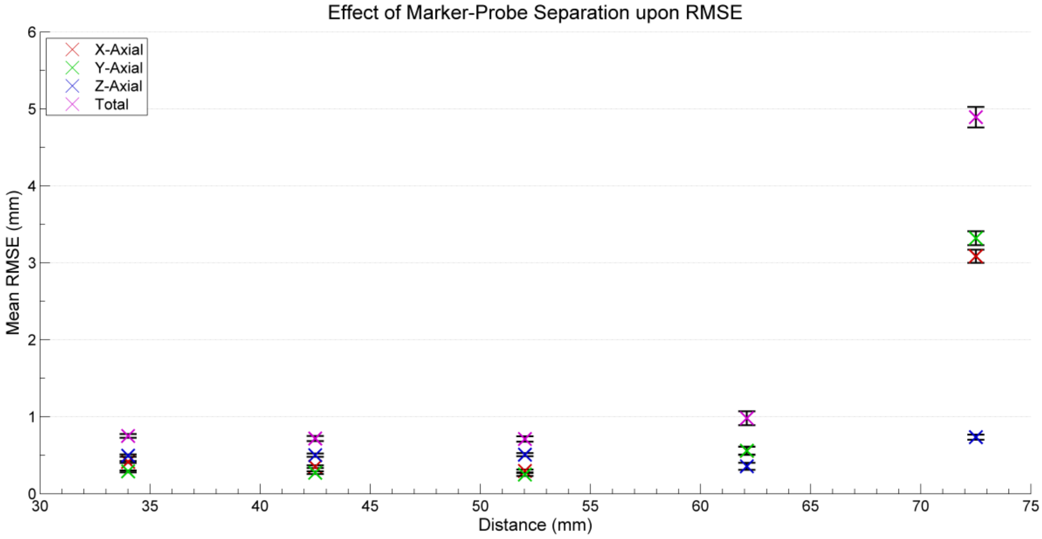


Figure 7.48: Effect of marker-probe separation upon RMSE and standard deviation

The error, both RMSE and standard deviation, are seen to increase with distance. The data of the three mounted repeats were combined and statistical analysis performed. No significant increase in RMSE was found until transitioning from the third to fourth level, at which point all increases across all components were found to be significant (All $P < 0.01$), except the third to fourth Z component ($P = 0.08$). The majority of level increases were found to result in a significant increase in standard deviation ($P < 0.01$), excluding all components of the second to third transition (All $P > 0.10$) and the Z component of the fourth to fifth ($P = 0.46$).

To investigate the angular relation the data sets were grouped and averaged by angular rotation about the Z-axis. The mean RMSE and standard deviation of the grouped data is shown by Figure 7.49 below.

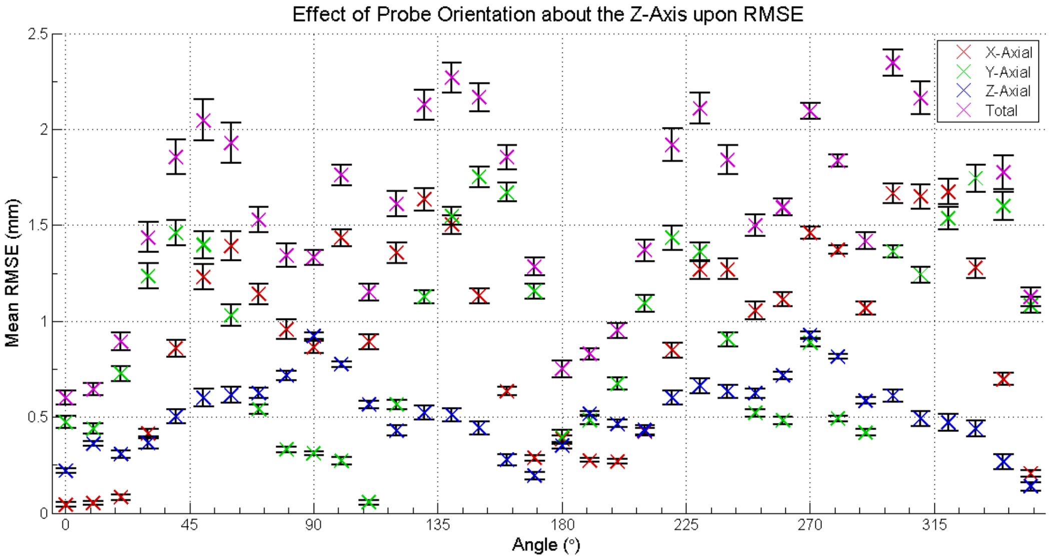


Figure 7.49: Effect of probe orientation about the Z-Axis upon mean RMSE and standard deviation

The three mounted repeats were combined and statistical analysis performed. The results are summarised by Figure 7.50 below, whereby angles that returned RMSEs significantly ($P < 0.05$) larger than the total data set are shown above the base line in the orange region, while angles that produced significantly smaller RMSEs are shown below in the blue region. It is noted that the difference of height within regions does not indicate variable significances, but is instead only to separate the components for clarity. This procedure was repeated for the standard deviation data, as shown by Figure 7.51.

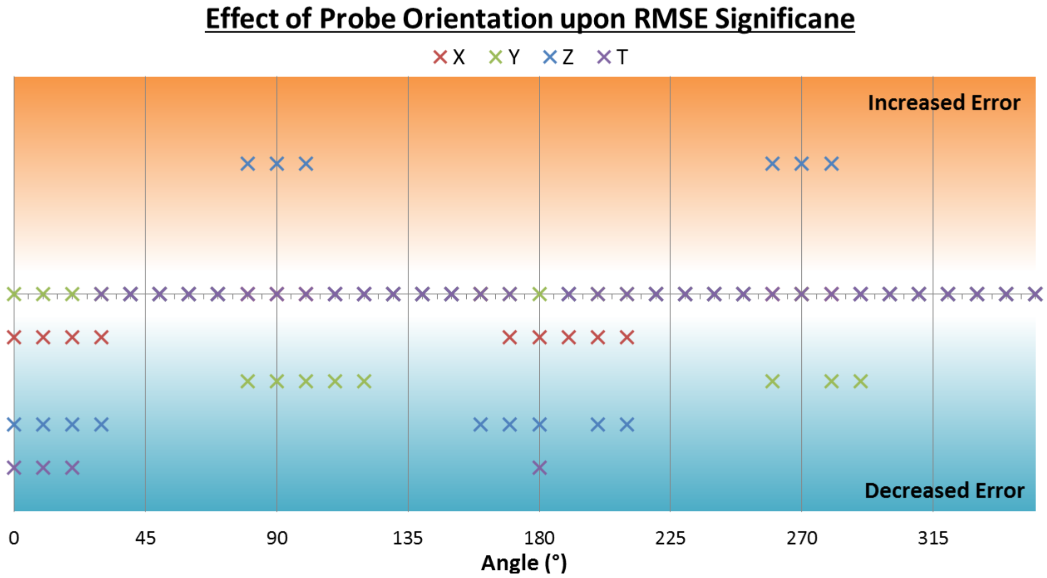


Figure 7.50: Significance of effect of Probe Orientation about the Z-axis upon RMSE

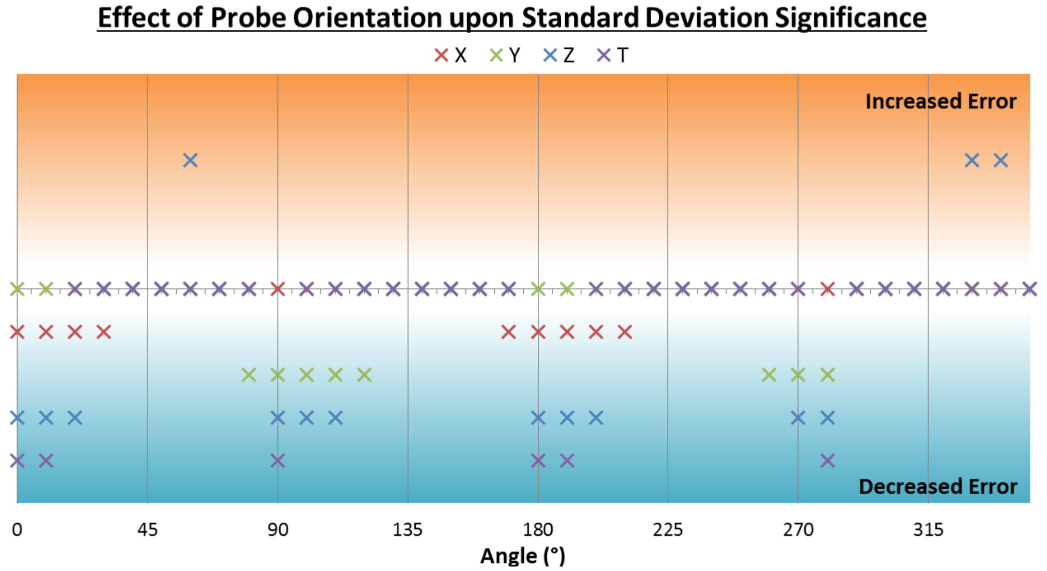


Figure 7.51: Significance of effect of probe orientation about the Z-axis upon standard deviation

Only the Z component was shown to produce significantly larger errors as a function of angle. Figure 7.50 shows these to group neatly about 90 and 270° for RMSE, and less consistently at 60 and 335° for standard deviation. The significantly larger RMSEs are mirrors at 0 and 180° by significantly smaller results. Z component standard deviations are seen to be significantly smaller at 0, 90, 180, and 270°.

Significantly smaller errors were much more commonly reported for both RMSE and standard deviation. As predicted by Figures Figure 7.44 to Figure 7.47, the X component is significantly more accurate and precise about both 0 and 180°, while the Y component mirrors this, providing optimal results around 90 and 270°.

The Euclidian total of the three components is seen to be significantly smaller at 0 and 90° for RMSE and at 0, 90, 180, and 270° for standard deviation.

To assess the effect of tool mounting, the mean RMSE and standard deviation was calculated and averaged across the three repeats for each of the mounted, held, and moving data sets. These are shown by Figures Figure 7.52 and Figure 7.53 below.

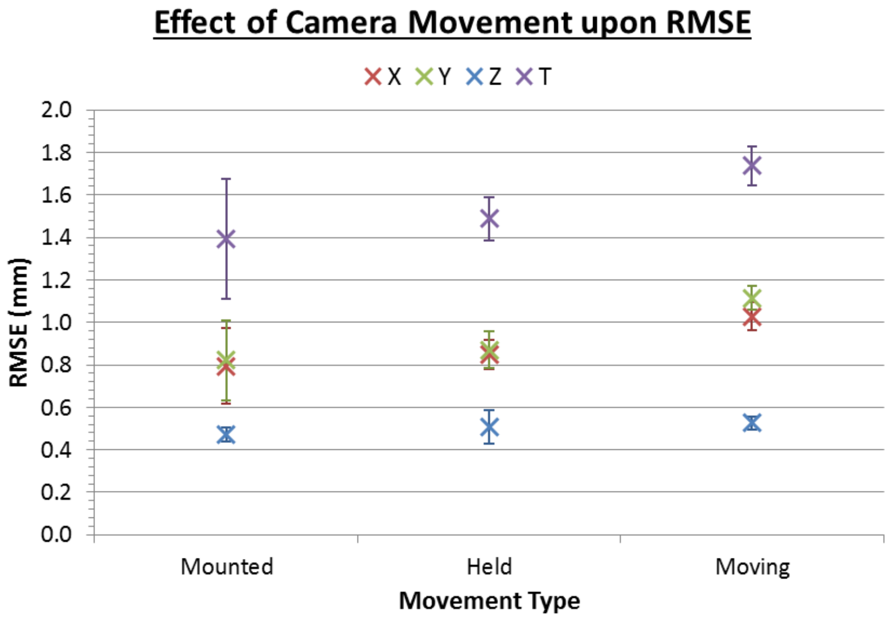


Figure 7.52: Effect of camera mounting movement upon RMSE

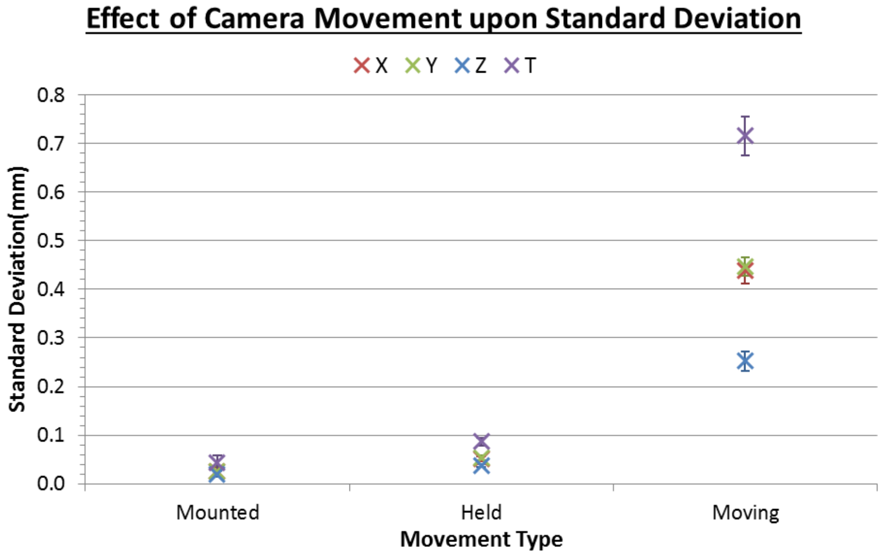


Figure 7.53: Effect of camera mounting movement upon standard deviation

Figure 7.52 shows an increase in RMSE as the dynamics of the camera motion increased, producing a total error increase of 0.34 mm, or 25%. Transitioning from mounted to held failed to reach significant for all components (All P > 0.3). However, held to moving produced a significant increase in error for both the X (P = 0.03) and Y (P = 0.02) component but not the Z component (P = 0.40). The total RMSE was found to increase significantly (P = 0.03) only between held and moving.

It is noted that one mounted sample produced a significantly smaller error, and removal of this sample from analysis caused both transitions to reach significances. However, no justification could be presented for the samples lower error, and as such it could not be neglected.

Standard deviation, as shown by Figure 7.53, is also seen to increase with increased camera dynamics, resulting in a total increase of 0.67 mm, or 1,576%. All increases were found to be significant (ALL P < 0.03).

The effect upon RMSE and standard deviation of system optimisation are shown by Figures Figure 7.54 and Figure 7.55 respectively.

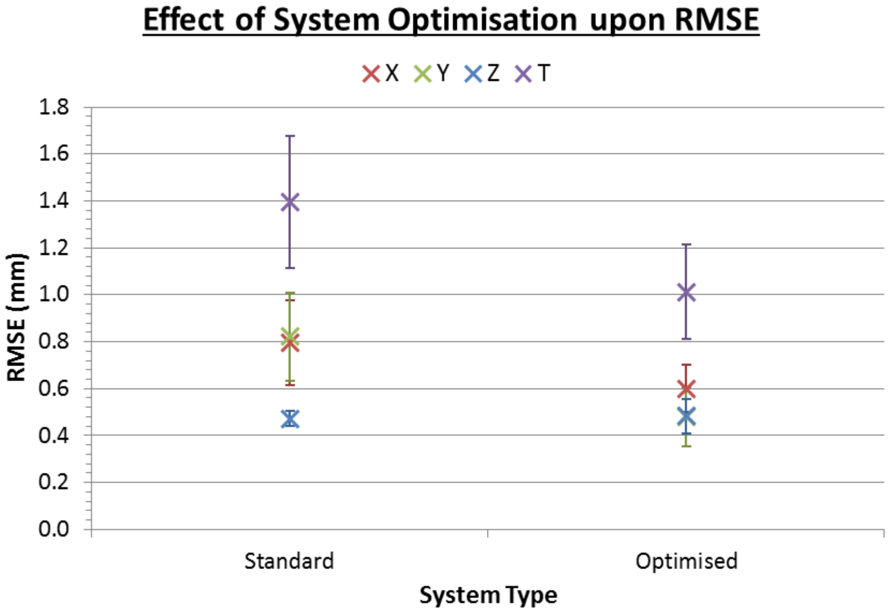


Figure 7.54: Effect of pose estimation system optimisation upon RMSE

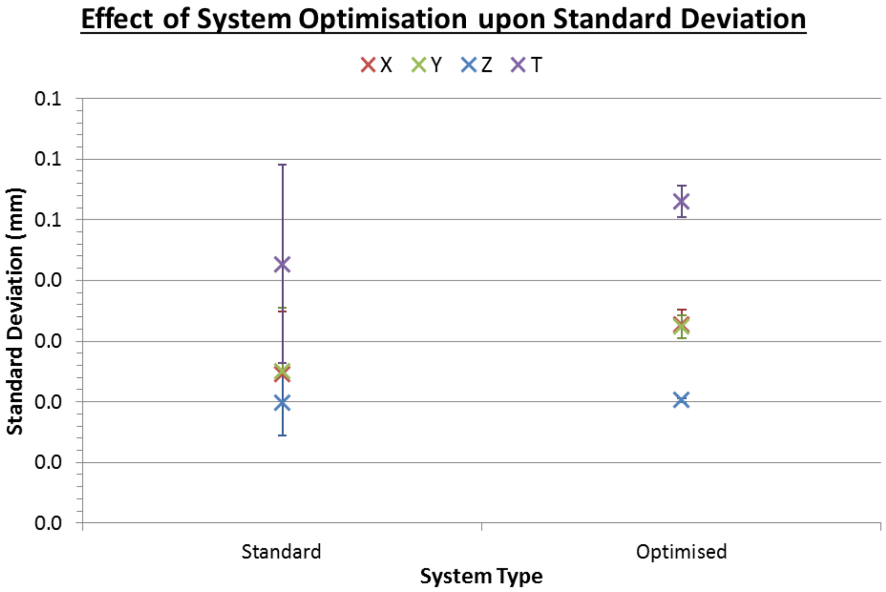


Figure 7.55: Effect of pose estimation system optimisation upon standard deviation

Figure 7.54 illustrates that the optimisation reduced the errors of the X and Y components, resulting in a mean total RMSE of 1.01±0.20 and 0.05±0.00 mm standard deviation. However, analysis showed only the Y reduction to be significant (P = 0.05). The X reduction and Z increase were found to be insignificant (P = 0.13 and P = 0.43). Again, removal of the same most accurate standard sample allowed these results to reach significance.

Figure 7.55 indicates an increase in standard deviation of all components. However, analysis shows these increases failed to reach significances (All $P > 0.2$). Omission of the lower standard sampled did not affect significances for the standard deviation.

7.3.3.3 Discussion

During mounted experiments the system was seen to produce a mean total RMSE of 1.4±0.28 mm and standard deviation of 0.04±0.02 mm, before optimisation, and 1.01±0.20 and 0.05±0.00 mm after. However, both RMSE and standard deviation were found to vary across the target. Figure 7.48 and the accompanying statistical analysis showed that the marker-probe separation affected the accuracy of the system, and this became significant at the larger separations. Furthermore, a significant reduction in precision occurred at greater separations.

As with Section 6.4.4.4, initial analysis was performed to confirm the marker was unlikely to be misplaced. No equal and opposite relation was found in any of the data sets. Therefore, it was concluded that the marker was correctly positioned and orientated.

The increase in error was most likely the result of the amplification of angular error at increased separation. The camera-marker separation did not change considerably while probing different levels. Therefore, the camera-marker separation error relationship highlighted by Section 7.3.2 was unlikely to be the major contributing factor to the error pattern seen here. However, as the tip probed at greater separations from the marker an angular error would be multiplied by this increased length, to produce a larger positional error. This would contribute to the increase observed in both RMSE and standard deviation. However, as seen by Figure 7.48, the error increase exceeds unity with probe-marker separation. Therefore, it is concluded an additional factor must contribute to the observed error.

An angular dependence of the error was also noted. Figures Figure 7.50 and Figure 7.51 demonstrated the significance for both RMSE and standard deviation respectively. The RMSE patterns indicate that the system was most accurate at multiples of 90°. Furthermore, Figures Figure 7.45 and Figure 7.46 indicate that the X component of the error was at a minimum when the X component of the tip position was also at a minimum. This was also seen for the Y component error and position.

An initial account for the observed error is an incorrect scaling of the system model. If the system were to under scale the model it would consider the tip to be placed further from the centre than expected. This would explain the error seen in Figures Figure 7.45 and Figure 7.46 at more distal points. For example the system would consider a point measured on the left of the target to be more left than expected. Therefore, the X component of the error would be shown as red-orange, as seen by Figure 7.45. The Y component of the error would be minimal and thus coloured turquoise, as shown by Figure 7.46. Under-scaling of the model was observed during measurements, as may be seen in Figure 7.56 below.

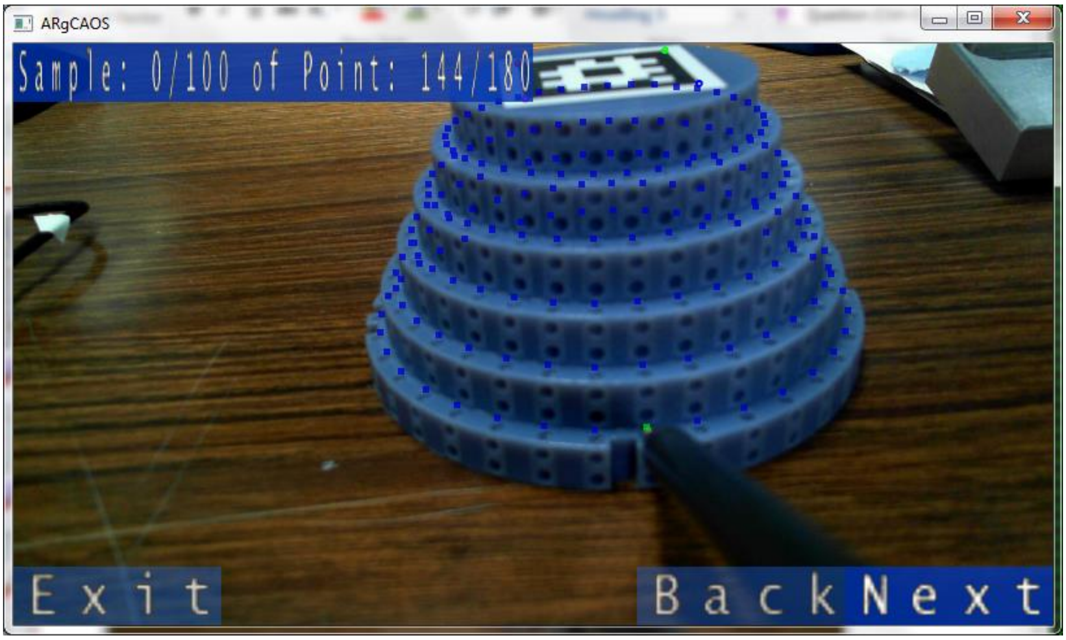


Figure 7.56: Model under-scaling shown by augmented probing points not aligning correctly

However, it is noted that the component errors are largest at multiples of 45°. The above theory would indicate the largest errors at multiples of 90°, whereby the positional components are largest. Building upon this it is considered that the errors observed may be in some way inverted.

It is recalled that the tip position relative to the camera was calibrated for each experiment. This was performed by placing the tip at (0.0, -32.5, -10.0). If the system over-scaled the model during calibration, the model tip would be placed more proximal to the camera than expected. If then used to probe a correctly scaled model this calibration error would result in an equivalent error pattern to that of an under-scaled model, as discussed above. This would account for the increase standard deviation seen at high RMSE points, as the additional offset causing the increased RMSE would also amplify any rotational jitter resulting in a higher standard deviation.

However, again, this theory fails to fully account for the increased errors observed at multiples of 45°. These increased errors are indicative of a variable model scale. It is noted from Figure 7.56 that the vertical angle of the probe is above the standard 45°. This was to allow imaging of the marker. However, as discussed in Section 7.3.2, this increased angle may increase the pose estimation error. Furthermore, to measure the points at 45° the probe angle was further increased, to accommodate the corners of the marker. This would further increase the error, both RMSE and standard deviation. The large change in probe angle is likely responsible for the significant error increase seen in Figure 7.48, when transitioning to the fifth level.

Therefore, it is concluded that incorrect model scaling combined with the increased error resulting from increased vertical probe angle may account for both the signed nature of the error components and the increased error observed at rotational multiples of 45°.

Wiles *et al.* reported upon the accuracy of the Polaris tracking system, which is commonly used in surgical guidance and navigation applications (Wiles et al. 2004). A rigid body of three passive markers, arranged in a 50 mm isosceles right triangle, was observed to provide a positional and rotational accuracy of 0.231±0.137 mm and 0.383±0.211°, respectively. Based upon these results it may be calculated that a probe attached to the rigid body, protruding 120 mm, as used in analysis of the second generation system, from the centre of rotation of the rigid body would present a positional error of 1.03±0.58 mm. This is seen to be approximately equal to the 1.01±0.05 mm error observed for the second generation with regard to RMSE. However, the precision of the Polaris system is seen to be an order of magnitude larger. This suggests that the second generation system may provide sufficient accuracy for guided resection. However, it is noted that the Polaris provided this accuracy over a much larger volume, with a maximum separation of 2400 mm.

Palmer *et al.* present a similar monoscopic augmented reality system for use with square planar fiducial markers, designed to externally augment ultrasound results (Palmer et al. 2015). The system was built upon the Vuforia framework (Qualcomm, CA, USA) and run upon an iPhone 5S. Accuracy measurements were performed upon a single marker over a separation distance of 150 to 300 mm at -30, 0, and 30°. The marker size was not stated, although is estimated as approximately 50 mm. The system was observed to provide a mean positional and rotational error of 6.2±5.2 mm and -1.60±2.85°. It is noted error was reported as mean error and as such may be considerably lower than RMSE, depending upon the bias of the system. As above these results were used to estimate the expected error of a 120 mm probe. The mean error was calculated as 9.5±11.2 mm, an order of magnitude larger than the second generation system. This large value likely arose from the use of non-camera specific calibration. However, as the tracking was solely to provide visual overlay this accuracy was sufficient for their application.

Augmented reality systems presented in literature typically present the accuracy of navigation. This value is not readily comparable to the tracking accuracy presented here as navigation accuracy includes, and may potentially be dominated by, human error, which may increase or decrease the effect of tracking error. However, related AR navigation systems, using a range of tracking technologies, within the medical field typical report accuracies on the order of one to a few millimetres (Kang et al. 2014; Cheung et al. 2010; Khamene et al. 2003; Banerjee et al. 2007; Shin et al. 2014). Abe *et al.* reported upon the single-camera planar marker virtual protractor with augmented reality (VIPAR) system used for percutaneous vertebroplasty (Abe et al. 2013). The accuracy of the system was reported as angular deviation from planned insertion error, and was found to be 0.96±0.61 and 0.61±0.70° in the axial and sagittal planes respectively. Unrealistically assuming no pure positional error, this angular error would result in a tip accuracy of 2.38±1.94 mm for a 120 mm probe. Therefore, accounting for user error, the accuracy results reported here appear similar to those of other systems.

Increased movement of the camera during sampling was found to increase both RMSE and standard deviation. However, only full motion was found to significantly increase the RMSE. Comparing mounted to moving the RMSE was found to increase by 25% while the standard deviation increased over 1,500%. This dramatic increase was predominately caused by the initial standard deviation being only 0.04 mm.

Two factors are expected to increase error under dynamic tracking, standard calibration error and rolling shutter error. As previously discussed if the camera is not correctly calibrated, either with regards to intrinsic or distortion parameters, the pose estimation will be altered depending upon where within the image frame the marker is imaged. Therefore, increased movement will cause an increased range of image frame positions and therefore an increased range of tip positions. As discussed in Section 6.4.2 the camera utilises a rolling shutter. This causes different parts of the image to be taken over the period of a few milliseconds. This allows the image to become distorted during motion, which in turn increases the error of pose estimation. Therefore, increased motion results in increased error as illustrated by Figures Figure 7.52 and Figure 7.53. Additionally, during full moving the camera repeatedly entered the near-parallel range of the marker. This likely accounted for the significantly larger standard deviation seen for the final mounting mode.

As described by Section 7.3.3 above, the target marker was iteratively produced until a sub-0.05 mm error was obtained for each corner. The system optimisation sought to correct these small errors. Pose estimation originally used a marker model derived from the planned geometry of the marker. This was updated to reflect the marker corner positions as measured using the Zeiss microscope.

The optimisation was found to produce a non-significant reduction in RMSE and increase in standard deviation. It is probable the optimisation failed to reach significance as the physical marker dimensions were very accurate, with a mean error of 0.025 mm. Therefore, the marker optimisation was minimal. However, it was an extremely time consuming process to iteratively produce the marker to this accuracy, requiring seven markers to be individually engraved and measured. Therefore, this decrease in error, although not significant, was sufficient to justify the use of custom model markers, removing the need for such extensive iterative marker production. As such, custom markers were more fully incorporated into the third generation system discussed in Section 8.

7.3.4 Resection Analysis

Chapters 3 and 4 define controlled resection as the primary aim of the system. It was originally intended to perform resection upon synthetic saw bones. However, this introduced a complexity in determining the accuracy of the system. To determine the accuracy of the resection it would be necessary to extract the intended resection region from the system. Given the dynamic nature of marker placement, this extraction would likely incur an error. Therefore, it was decided to perform a known resection upon a known geometry with a fixed spatial relation to a marker. This would allow the results to be reliably compared to the plan.

7.3.4.1 Materials and Methods

Table 7.10 lists the requirements of a resection target.

Resection Target Requirements
Permanently fixated 32 mm marker
Resection volume greater than 50x50x20 mm
Reusable
Accurately relate planned resection

Table 7.10: Requirements of resection target

The marker was to be permanently fixed for all experiments to remove errors associated with marker placement. A resection volume of 50x50x20 mm was selected as this would provide sufficient volume for potential implantation of a tibia component. The presented system was designed for near marker operation. However, the maximum marker-probe separation provided by the probe target discussed above would likely be exceeded during a real procedure. Therefore, the resection target was designed to have a maximum separation of 90 mm.

One part 617H32=2.300 Pedilen Rigid Foam 300 was combined with one part 617P21=4.600 hardener (Otto Bock GmbH, Germany), to simulate bone for the resection volume. This was cast into a 70x70x30 mm mould. The resulting blocks were sanded flush with the upper surface of the resection target.

The target body and upper plate were produced by laser cutting. As with the probing marker discussed above, the laser cutting of the resection target was iteratively performed until an accuracy of 0.1 mm was achieved across all measurements. The layers of the resection target were aligned using guide holes and 2 mm pins. The completed resection model is shown below by Figure 7.57.

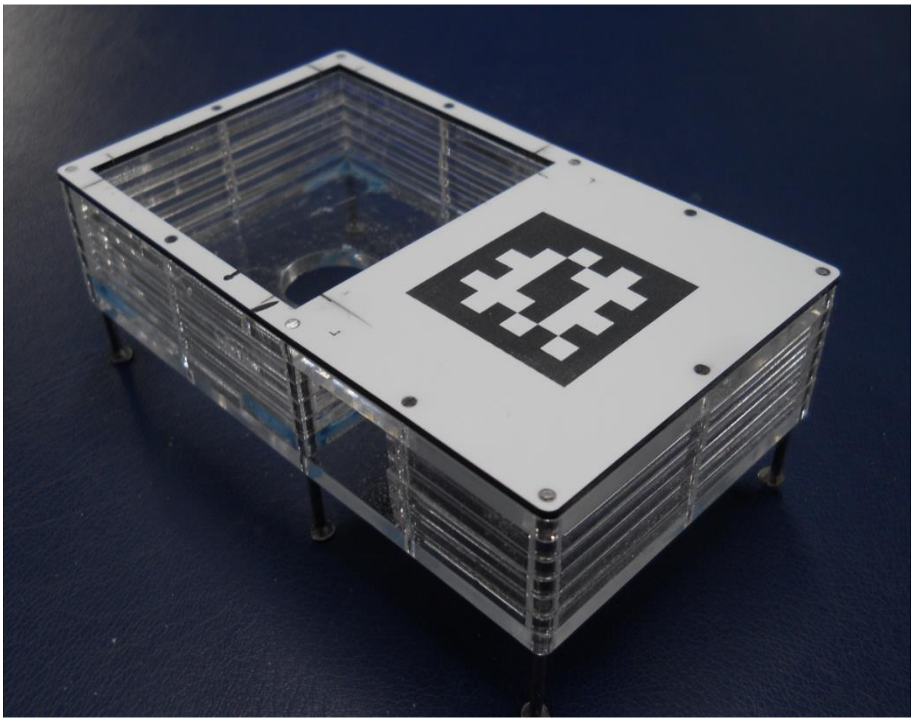


Figure 7.57: Resection target

The resection plan was designed to simulate two forms of resection commonly required by modern UKA implants, standard surface resection and peg hole resection. Figure 7.58 below shows the resection plan. It is seen to contain three trenches of radial profile and four peg holes of 12 mm depth.

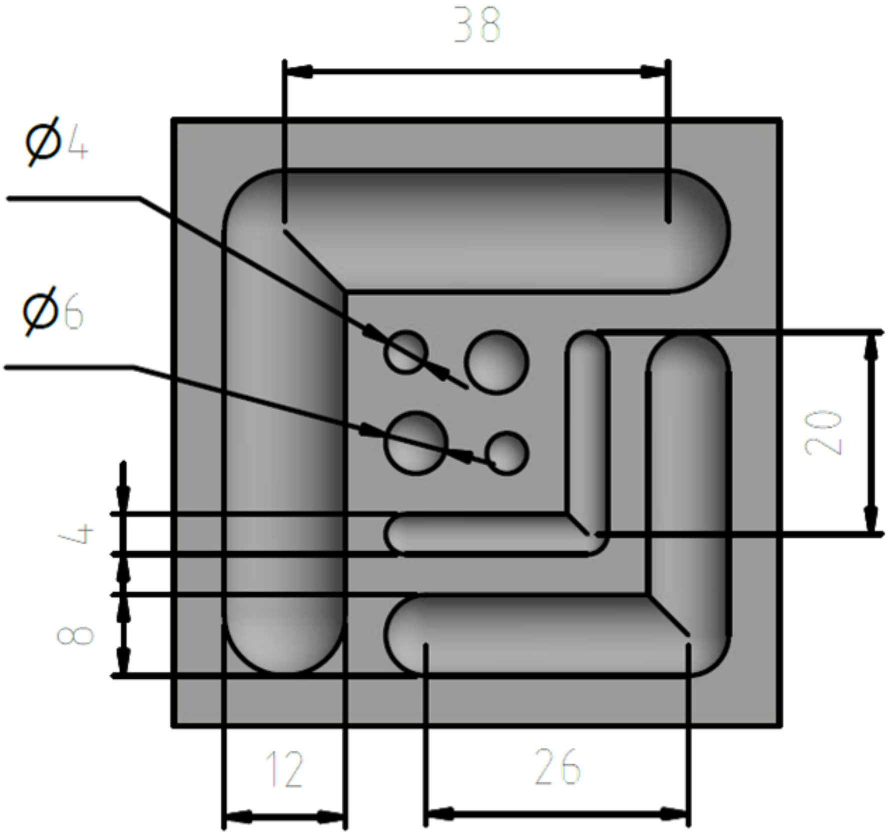


Figure 7.58: Resection plan

Resection control was implemented by binary speed control of the resection volume, as described in Appendix A2.5, whereby the burr was set to full speed or depowered. Therefore, the resection plan was coloured using green for region to be resected, and red for regions that would depower the burr. Figure 7.59 shows the augmented volume model upon the resection target, invisible virtual walls have been rendered to obscure sections of the resection volume, making it realistically appear within the physical equipment.

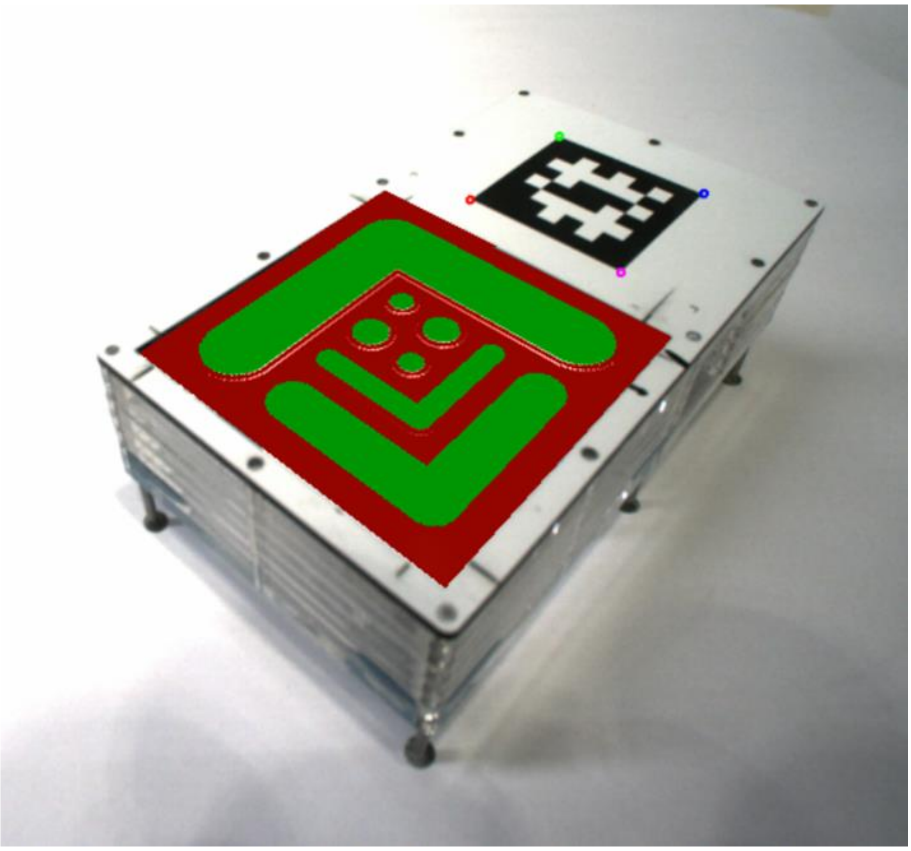


Figure 7.59: Simulated obstruction resection volume

One resection was performed using the rotary tool based system discussed in 7.2.1.3. Due to the poor results and availability of the Anspach system repeats were not performed.

Using the Anspach based resection tool three independent resections were performed. A 6 mm diameter spherical burr was used. The camera-tip separation was set to 120 mm. The angular difference between the optical axis of the camera and axis of rotation of the burr was set to 30°. The Anspach was set to a maximum speed of 80,000 RPM. The tool controller was set to peddle-system mode. Finally, the system resection control was set to binary mode with a threshold minimum distance of 0.1 mm and a search radius of +0.5 mm.

Calibration was performed using the 3D printed calibration target shown in Figure 7.60 below. The burr was placed within the 6 mm hemispherical divot to the left of the marker. The tool was angled at 45° to the normal of the marker and 200 samples were acquired. During sampling the tool was slowly rotated in a pseudo random fashion to a maximum angle of ±15°. The central 90% of samples were used for calibration.

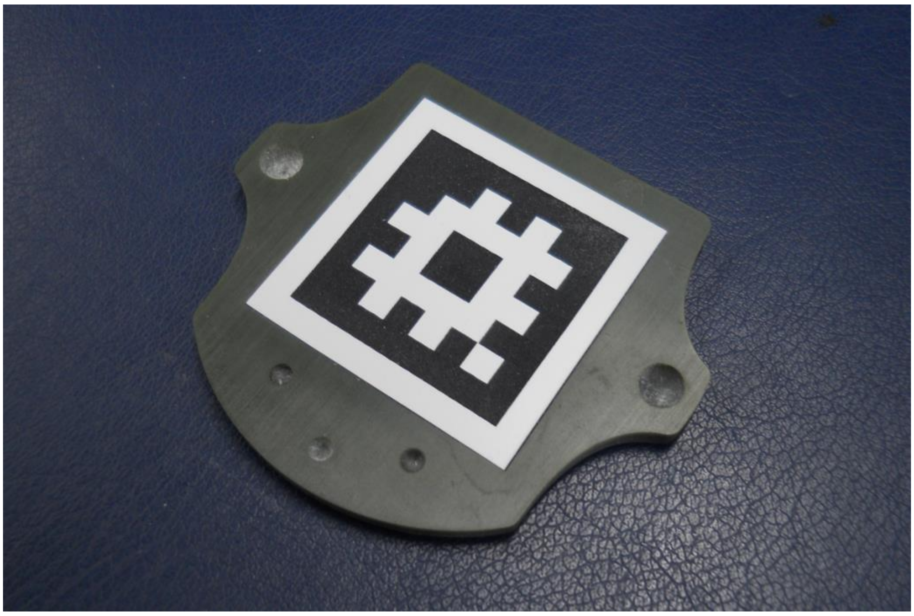


Figure 7.60: Burr calibration target

The volumes were resected following the augmented guide. Resection was continued until the majority of the green resection regions were removed. This defined the resection time. After which the burr was slowly traced across the resection surface with the peddle depressed, to ensure resection had been completed to the limit of the system. A finished resection is shown by Figure 7.61 below.

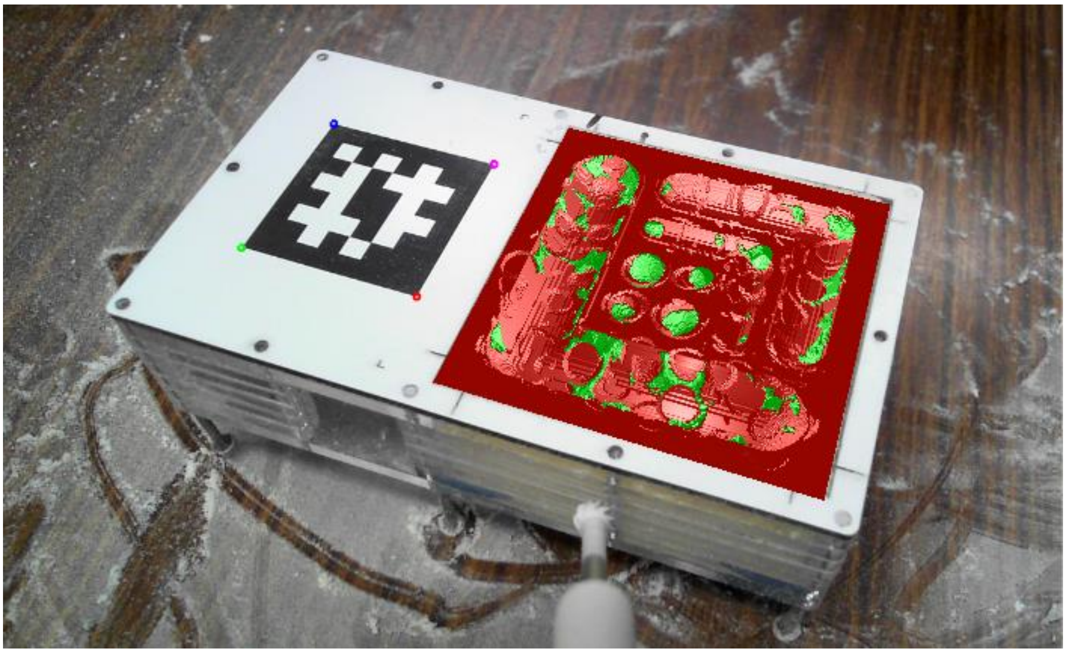


Figure 7.61: Completed resection

Once resected the tick-marks of the resection target were used to label the four corners of the resection volume. These labels were later used to align the foam target for analysis.

7.3.4.2 Results

Originally the resection volumes were to be laser scanned for analysis. However, a laser scanner was not available due to licencing issues. Therefore, 123D Capture (Autodesk Inc., CA, USA) was used. 123D Capture takes a series of images of an object to generate a 3D model. Based upon validation work performed upon larger scan regions, 123D Capture was expected to reach near 0.1 mm accuracy of the present volume (Chandler & Fryer 2013; Erickson et al. 2013). A Nikon D40X (Nikon Corporation, Tokyo, Japan) camera with 28 mm manual focus lens with F-stop of f/4.2 was used to image each volume. An average of 37±5 images were taken of each volume. The model quality was set to maximum vertex density. To scale the model a steal rule was included in the imaging and model generation. This was used to provide a known measurement that 123D used to scale the model.

The models were then cropped to include only the resection volume and exported into Wavefront Object format. These were imported into CloudCompare (CloudCompare V2.6.1) where they were further cropped to retain only their upper surfaces. The resection model was also imported into CloudCompare to provide the ground truth for mesh comparison. The resection surface was manually aligned with the resection model using the corner points as guides. The non-resected surfaces were aligned such that they presented maximum agreement, before absolute and signed matching was performed.

Figure 7.62 below shows the absolute and signed difference between the resected volume and plan for the rotary tool system.

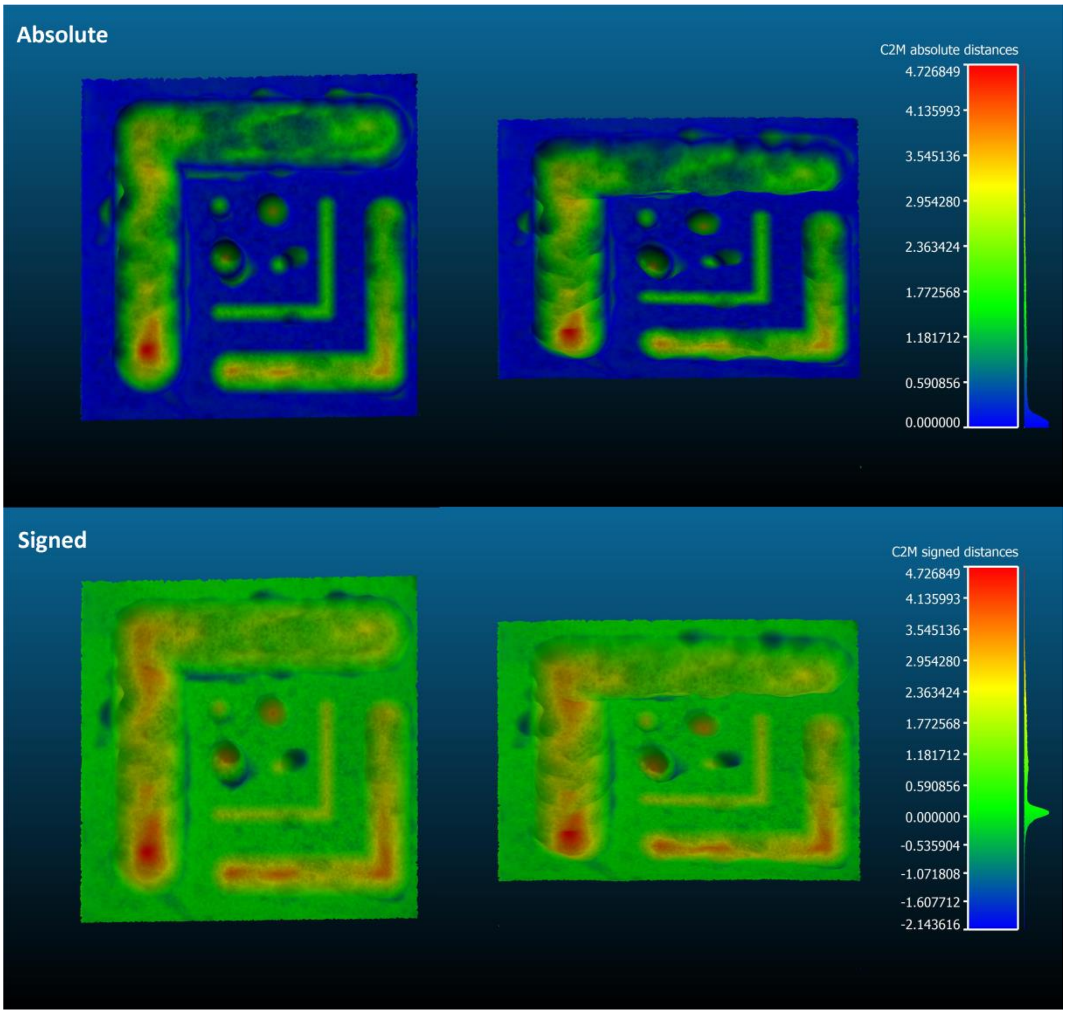


Figure 7.62: Rotary tool based system signed and absolute resection errors

The mean absolute and signed errors across the resection volume for the rotary tool system resection were 0.85±0.91 and 0.77±0.97 mm, respectively.

The same procedure was repeated for the three resections performed with the Anspach system. The absolute and signed errors of these resections are shown by Figure 7.63 and Figure 7.64, respectively.

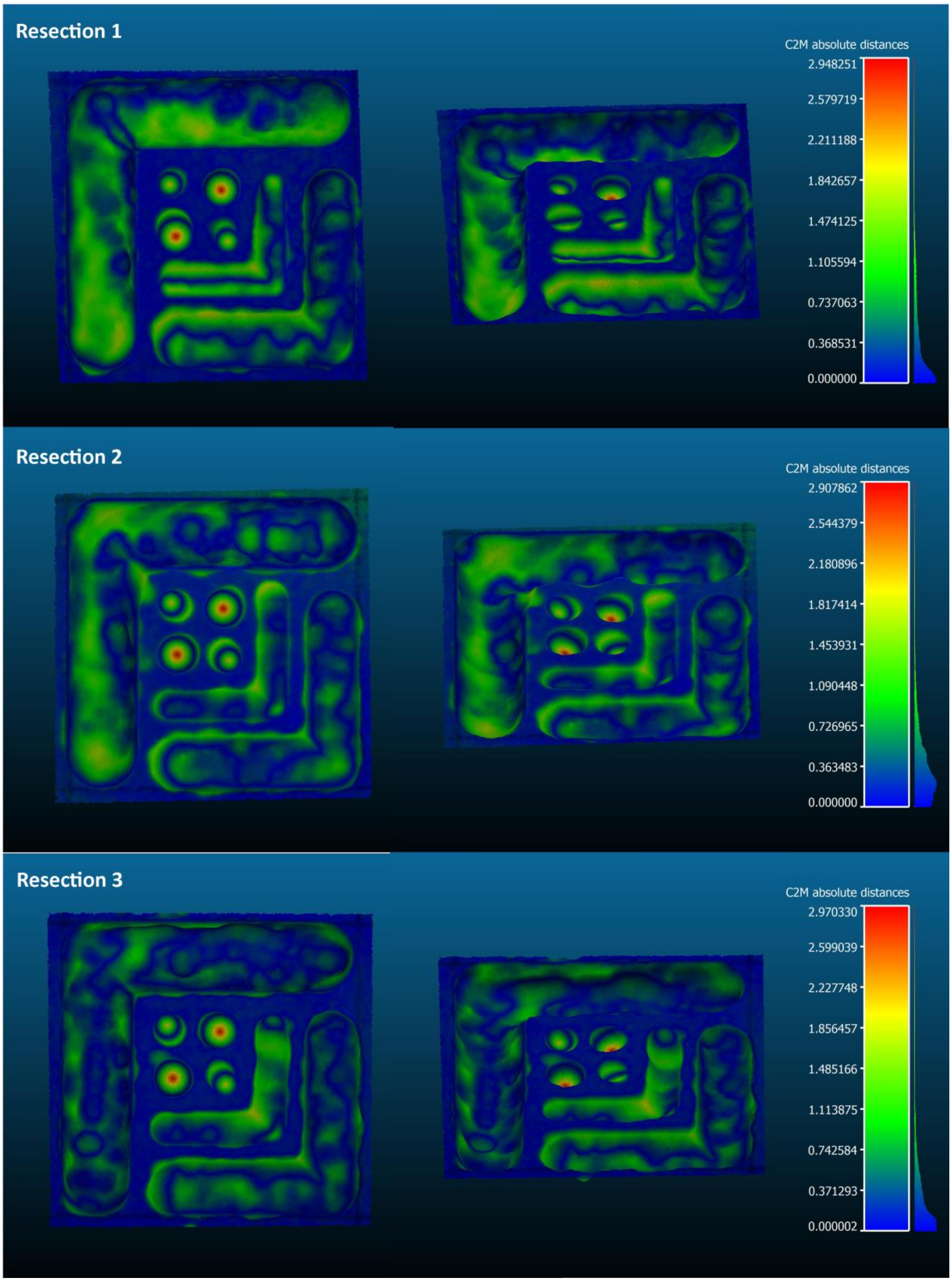


Figure 7.63: Absolute resection error of Anspach system

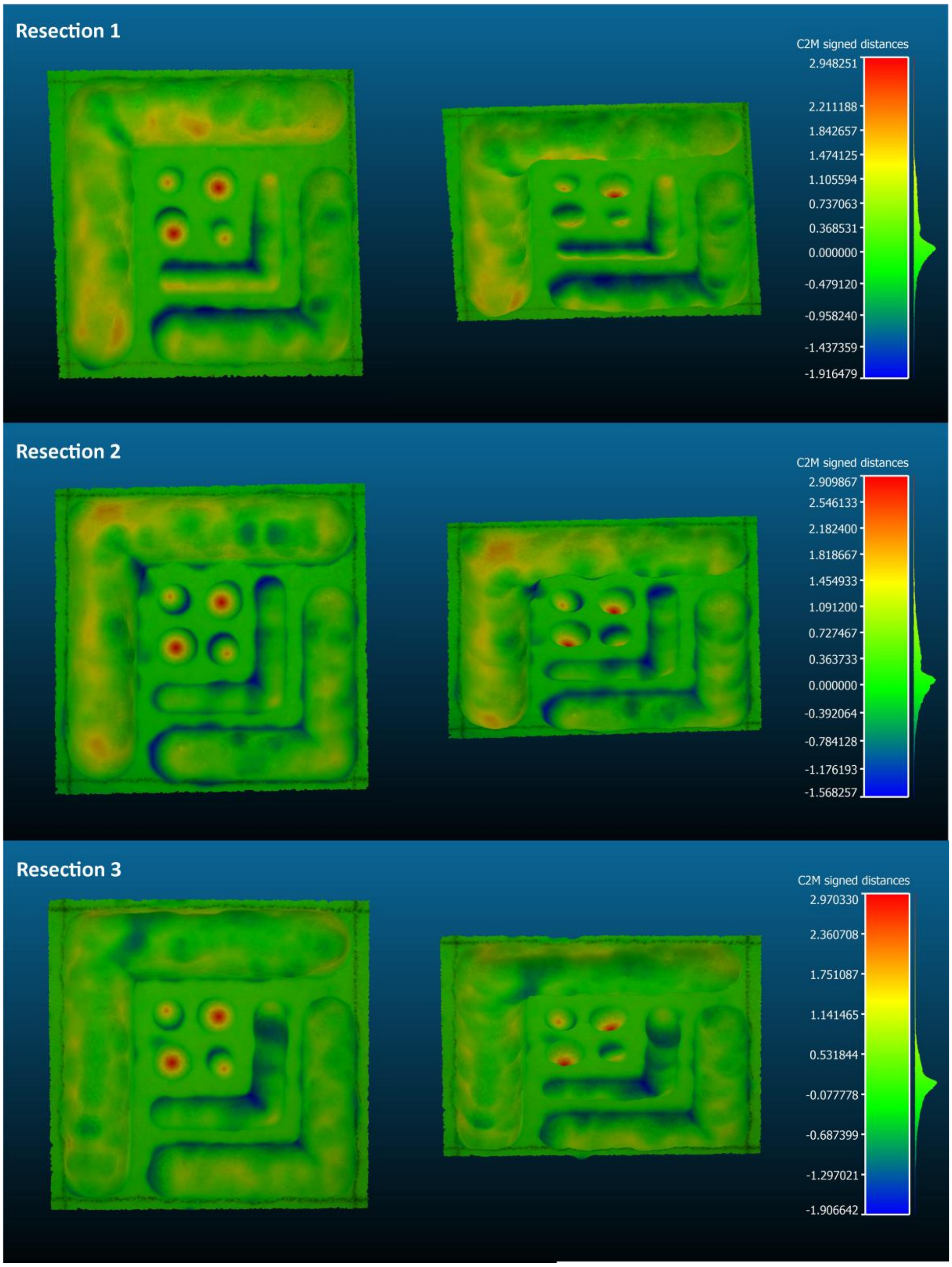


Figure 7.64: Signed resection error of Anspach system

The mean signed and absolute values of the three resections are shown below by Table 7.11.

Resection	Absolute	Signed
1	0.40±0.40	0.15±0.54
2	0.38±0.31	0.14±0.49
3	0.30±0.30	0.02±0.42
Mean	**0.36±0.34**	**0.11±0.48**

Table 7.11: Resection error of Anspach system

On average resections were completed within 7.7±0.3 minutes.

7.3.4.3 Discussion

It may be seen that the rotary tool based system produced significantly poorer results than the later Anspach system. As stated in Section 7.2.1.3 this increased error could be attributed to the substantial spin up and down times of the rotary tool in combination with the high vibration of the tool. During resection as the tool reached resection speed the vibration of the tool appeared to reach resonance with the camera. This caused a sudden increase in the distortion of the image, such that the system failed to detect the marker. This triggered the system to deactivate the resection tool. Therefore, as the speed decreased, resonance was lost and the system was able to detect the marker, therefore repowering the burr. This oscillation of activation, combined with the spin latency of the tool, caused resection to progress erratically. This resulted in the considerable undercutting seen in the trench resections, whereby the two smaller trenches were relatively unresected, in addition to the sporadic overcutting seen around the edges of the largest trench and peg holes.

The Anspach system considerably rectified these two issues, producing an RMSE of 0.36±0.34 mm. Both spin up and down times were substantially shorter and tool vibration, across all burr speeds, appeared to induce minimal image distortion, insufficient to induce marker loss. The rapid activation of the burr caused the tool to kick during switching. However, this was readily compensated for by a firmer grip.

Figure 7.63 indicates the Anspach system resected to plan much more consistently than the rotary tool system. This is further confirmed by the mean absolute resection errors that present a fractional improvement of 2.47±2.70. Analyses of Figure 7.63 indicates that 96.1±0.8% of the resected surface presents an absolute error below 1.0 mm, while 77.2±2.5% were below 0.5 mm absolute error. However, errors above 1.0 mm are consistently seen in two regions across the three resections. Figure

7.64 indicates the base of the large trench is often under-resected, presenting a mean maximum error of 1.45±0.25 mm. Conversely, the medial edges of the two smaller trenches typically indicate over-resection, 1.64±0.06 mm. The second trench also indicated under-resection, although over a smaller percentage area and to a lesser degree producing a mean maximum error of 0.98±0.07 mm.

Lin *et al.* presented an AR navigation system combined with a physical surgical template to guide dental implantation (Lin et al. 2015). Although passively guided through visual feedback only, the system is similar in its use of single-camera tracking of a square planar fiducial marker, measuring 60 mm. The mean absolute error of the system, found by averaging entry point, apex, depth, and laterals errors, was 0.74±0.34 mm. This is approximately double that of the reported system. However, as the system was passive and no tool tracking or semi-active constraint was applied the system was likely more susceptible to user-error. Additionally, the use of a physical guide would affect results.

Finally it is noted that the four peg holes show substantial under-resection. Unfortunately, the resection volume was designed for the 3 mm burr of the original rotary tool based system, and was not updated for the 6 mm burr of the Anspach system. Therefore, the burr was able to produce only a small indent before deactivating. Extreme caution may be expected to successfully resect the larger 6 mm peg hole. However, it is recalled that the burr has an additional 0.5 mm search radius applied, effectively producing a 7 mm diameter burr.

The under-resection seen in both the large and medium trenches may also be partially accounted for by the 0.5 mm search radius. Regions resected below this 0.5 mm limit would therefore be the result of system latency. A clear example of system latency is given by the second resection of Figure 7.64. The medial corner of the largest trench is seen to be considerably over-resected by up to 1.54 mm. During resection of this region the burr snagged, due in part to a loss grip, moving quickly into the trench corner. As a result of the high speed the over-resection occurred before the system could depower the burr.

The over-resection of the two smaller trenches did not occur at speed, and therefore was unlikely the result of system latency. The relative consistency of the location of over-resection also implies latency was not a core factor. Instead, as with Section 7.3.3, erroneous model scaling error is suspected. Firstly, as discussed in Section 7.3.3 above, errors in marker detection or pose estimation may result in the virtual model being incorrectly scaled. A model under-scaling with respect to the length of the resection target would allow the burr to over-resect the volume before depowering. Similarly, under-scaling with respect to the width of the resection target could account for the over-resection observed on the vertical edges. However, an error symmetry would be predicted for incorrect modelling scaling. For example, an under-resection of the external horizontal edges would be expected to mirror the over-resection observed upon the medial horizontal edges of the smaller trenches. As this was not observed it may be assumed that model scaling was not responsible for the over-resection. However, during resection the orientation range of the tool, relative to the Z-axis of the target marker, was in excess of 270°. As shown by Section 7.3.3 above, model scaling is not consistent with probe orientation. An over--resection is clearly permanent. However, an under-resection that occurs due to model under-scaling at one tool orientation may be removed when the model is correctly scaled at an alternative orientation. Therefore, upon complete resection over-resections would be preserved while the predicted symmetrical under-resections removed.

The resection tests were shown to produce a significantly smaller ($P = 0.02$) and consistent ($P = 0.04$) error than the probing experiments of Section 7.3.3. This was likely the result of model scaling averaging, as each region of the resection volume was resected at a wide range of tool orientations. This is in contrast to the single orientation used for each measurement during probing experiments.

7.3.5 Conclusion

The angular experiments highlighted an important caveat of the system, in that near-parallel and near-perpendicular orientations increase the error of the system, to a significant degree for the former. Therefore, during use it would be important to avoid these regions to maintain accuracy. The probe experiments demonstrated both the significances and difficulty of this task. Due to the proximal nature of the camera to the tool the FOV was relatively limited. Therefore, near-perpendicular orientations were required to image the marker while probing the more distal points. The angular experiments demonstrated that the probe length could not be considerably increased, to broaden the FOV, without increasing the pose estimation error.

Both the static and significantly larger errors of the probing experiment under dynamic conditions, 1.40±0.04 mm a 1.73±0.72 mm respectively, indicated that the system lacked the accuracy required, failing to reach the 1.0 mm target defined in Section 4.3. Much of this error was attributed to inconsistent model scaling, which was reduced by system optimisation that accounted for marker manufacturing errors. This resulted in an error of 1.01±0.05 mm, only marginally above the 1.0 mm target. This indicates the system may provide sufficient accuracy to provide the desired functionality.

The angular dependency of accuracy appeared to average out during resection, reducing the effect on the system accuracy. This was shown by the three resections performed with the Anspach system, which mirrored the resection plan extremely well, where possible, resulting in a mean absolute error of 0.36±0.34 mm. Therefore, the system appeared to offer sufficient accuracy. However, as shall be discussed below the system presented several complications to the total procedure.

7.4 Conclusion

Section 7.3 above demonstrated the ability of the system to meet the accuracy requirements for constrained resection. However, a complete UKA system, as described by Chapter 4, requires several additional functions, which are expected to exceed the presented system, namely location of the ankle and hip centres.

As discussed by Section 4.2, the ankle centre was to be found by probing the medial and lateral malleoli of the joint. This presents a number of challenges to the current system. Firstly, to allow accurate resection the markers were to be placed proximal to the knee, to view these while probing the ankle the camera would be positioned distal of the ankle. Therefore, the camera-marker separation would be approximately 400 to 500 mm. The angular experiments of Section 7.3.2 indicate the orientation error increases significantly with separation. Furthermore, this error is twofold. The increased separation firstly increases orientation error, which is then further amplified by the increased separation into a larger positional error. An approximate regression performed upon the data of Section 7.3.2 indicate a positional error of approximately 30 mm at a separation of 450 mm, far exceeding the desired accuracy. Furthermore, during ankle probing the marker would likely be imaged at a near-perpendicular angle, further reducing the accuracy of the system.

As with the first generation system, the hip centre would be located by tracking two markers simultaneously. As discussed this intrinsically increases the error of the system. Furthermore, the camera-marker separation would again be increased to image both markers. Additionally, it is not possible to both operate the tracking system and manipulate the limb.

The system appeared to provide intuitive guidance, most notably the resection guide. Although not quantified, the learning curve associated with watching the augmented display as opposed to the physical target was relatively minimal. However, this approach reduced spatial awareness, due to the use of a two dimensional display. Furthermore, the coupling of the camera and tool limited the intuitiveness of the system. As a result of the coupling it was not possible to resect from one orientation while viewing from another. This proved to be inconvenient during the simple experiments presented above, and would be expected to be more so during a practical procedure.

The total system latency was measured to be 188±48 ms, as described in Appendix A2.1.6. This is approximately 50 ms higher than the optimal safety latency defined by Ruijters *et al.* (Ruijters et al. 2014). However, it is still below the intuitive hand-to-eye guidance threshold of 250-300 ms observed by Vercher and Gauthier (Vercher & Gauthier 1992). Therefore, while nonideal, the latency is considered acceptable for the proposed application.

In conclusion this system further demonstrated the potential of augmented reality guidance for UKA. The accuracy of the first generation system was improved upon, reaching the 1.0 mm probing accuracy target, and demonstrating sub-millimetre mean absolute resection accuracy.

However, the accuracy was strongly affected by camera-marker separation, and the tool mounted approach would potentially prove inhibitive to several of the peripheral functions of a guided UKA system. Furthermore, the augmented guidance, while intuitive, could be argued as lacking due to the use of a two dimensional display.

Therefore, it was decided to develop a third generation system, based upon an alternative pose estimation methodology. The new system, providing hands free tracking and 3D display is discussed below in Chapter 8.

8
Third Generation – ARgCAOS

8.1 Introduction

8.1.1 Problem

The second generation system was able to provide the accuracy required for controlled resection. However, this accuracy was limited to a relatively small tracking volume. It was concluded that this limited volume would inhibit the peripheral functions required of a guided UKA system, such as joint centre location. Furthermore, despite providing intuitive guidance, it was proposed that this intuitiveness could be optimised by the inclusion of the third dimension within the augmented display.

8.1.2 Solution

To rectify these factors a stereoscopic approach was proposed, for both marker pose estimation and display.

As illustrated by Section 2.4.3, a stereoscopic tracking system would provide improved accuracy over the monoscopic approaches presented by Chapters 6 and 7 above. Furthermore, by utilising both camera images with a HMD a 3D augmented environment could be displayed to the user, providing additional intuitiveness to the guidance.

8.2 Design

As with the second generation system, the design of the system is divided into its hardware and software constituents. The third generation system utilises much of the core hardware and software elements developed as part of the second generation system, such as the physical markers and their detection within the image frame.

Hardware design first discusses, in Section 8.2.1.1, the HMD selected to provide a stereoscopic 3D display to the system. After which, the design and development of the three stereoscopic camera assemblies, utilised by the third generation system, is described in Section 8.2.1.2. This incorporates an extension of Section 5.2.2.1, whereby further camera selection is performed.

The hardware design section is then concluded with a brief discussion of the third generation probe and resection tool.

Section 8.2.2 then discusses the integration of these hardware elements into the system, describing the stereoscopic pose estimation and control flow required to render to the HMD.

8.2.1 Hardware

8.2.1.1 HMD

Section 2.5.2 of the literature review summarised the two main modalities of HMD, namely OST and VST. A total system latency of 188±48 ms was determined, as detailed in Appendix A2.1.6, with the majority of this time being attributed to the camera hardware. It was concluded that while higher than desired this latency did not substantially hamper the use of the system. However, as described within the literature review, OST devices are unable to delay the physical world. Therefore, the augmented overlay would be approximately 200 ms behind the physical imagery. It is suspected this would greatly reduce the immersion and intuitiveness of the system. As such a VST HMD was selected.

To produce a VST HMD a standard HMD, used for virtual reality, was to be combined with a stereoscopic camera system to stream the imagery of the environment onto the display.

After investigating the range of commercial HMDs available the Oculus Rift DK2 (Oculus VR, LCC., CA, USA) was selected. The Oculus was selected as it provided a high image resolution and low latency. Additionally, it was a cost effective solution with SDKs provided for integration into C++ systems. Finally, one of the core advantages of the Oculus over alternative HMDs, such as the HMZ-T2 by Sony, was the large FOV. This would allow the user to have a more natural view, and greatly improve the immersion of the system.

The Oculus uses two lenses to focus opposite halves of a single display into the separate eyes, simulating a 3D scene. A 1920x1080 AMOLED screen is used, providing a per eye resolution of 960x1080. Each lens provides a horizontal FOV of 84°, resulting in a large total display FOV 100°. The display is connected via a standard HDMI interface.

The Oculus also provides 6 DOF head tracking via an active IR monoscopic tracking system. However, as this could only provide one-to-one tracking it was not utilised. Instead the stereoscopic tracking system discussed below would be attached to the Oculus providing multi-marker tracking relative to the user.

8.2.1.2 Stereo Camera

Initial development was performed using the LifeCam Studio webcams selected in Section 5.2.2.1. An initial mounting assembly was developed to allow precise camera positioning, for use during system analysis. Additionally, a light weight mounting assembly was design to allow the cameras to be mounted to the HMD.

In an effort to improve the performance of the system two research cameras were acquired. Section 8.2.1.2.2 discusses the selection process of these cameras and the design of their mounting assembly.

8.2.1.2.1 Web Cameras

To obtain optimal results the two cameras of the stereoscopic system must be well aligned, presenting minimal relative rotation in all three axes. Due to the use of commercial cameras the internal alignment with the outer casing could not be guaranteed. Therefore, the camera assembly was required to allow the precise adjustment of each of these axes.

The existing monoscopic camera assembly presented in Section 6.3.1 provided these functionalities. Therefore, the stereoscopic assembly utilised two monoscopic camera mounted to a 200 mm base plate. The assembly is shown below in Figure 8.1.

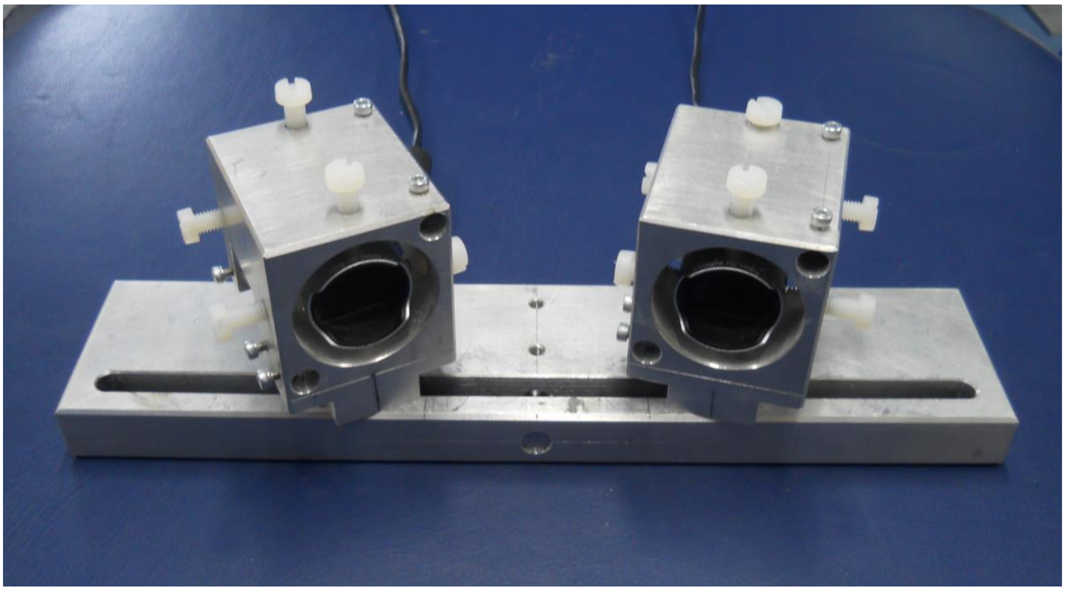


Figure 8.1: Stereo camera assembly

The slot of the base plate was designed to allow the base line separation of the cameras to be adjusted. However, a base line of 64 mm was used for all experiments as this is the mean *interpupillary distance* (IPD) of human eyes (Dodgson 2004).

Evidently the design of Figure 8.1 was not practical for head mounting. Therefore, an additional, light weight, camera assembly was designed. As the assembly was not to be used for accuracy analysis, a light-weight design was favoured over the ability to precisely align the cameras. Figure 8.2 shows the camera assembly used to mount the stereo camera system to the Oculus.

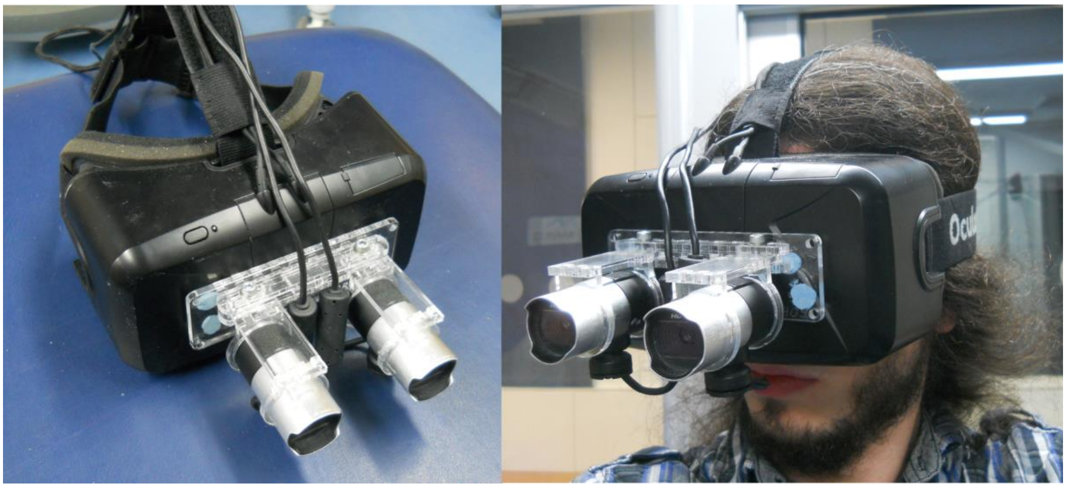


Figure 8.2: Third generation HMD camera assembly

The assembly was manufactured with 6 mm laser cut acrylic, designed with nine tongue and grove press fit components. Table 8.1 below summaries the mass of the total HMD assembly.

Component	Mass (g)
Oculus Rift	440
2X LifeCam	110
Mounting Assembly	44
Total	**594**

Table 8.1: HMD assembly mass

While heavier than desired for a viable commercial product the assembly performed well and did not cause notable discomfort, even during prolonged use of several hours.

8.2.1.2.2 Research Cameras

The commercial cameras used to date presented several disadvantages. Primarily, their resolution and framerate were notably limited by the 480 Mbps bandwidth of the USB2.0 connection used. Additionally, as noted previously, the digital focus of the camera was inconsistent and measures had to be taken to ensure equal focus between experiments. Finally, the FOV and aspect ratio of the LifeCam did not well match those of the Oculus HMD. This resulted in large image distortion and the disorientating effect of viewing the world through binoculars.

As a result of these observations three additional camera requirements were appended to the list of Table 5.1. The camera was required to provide a similar aspect ratio and FOV to that of the Oculus. Furthermore the focus should be manually adjustable and lockable.

Modern research cameras offer several connection modalities. USB3.0 was selected, as this provided good compatibility with portable systems. The maximum bandwidth of USB3.0 is reported as 5 Gbps. This is sufficient to allow streaming of uncompressed full HD video in excess of 60 fps.

Camera selection was initially based upon five criteria, resolution, framerate, encoding, sensor size, and shutter type. Sensor size indicates the size of the pixel elements, which in turn dictates the amount of light a sensor is able to capture per unit time. Larger sensors are therefore desirable. However, larger sensors are typically cost prohibitive. As discussed in previous chapters, errors may be induced under dynamic conditions due to the rolling shutter used by the LifeCam. For the research camera this was to be eliminated by the use of a global shutter. Unlike a rolling shutter, a global shutter captures all pixels across the image sensor simultaneous, reducing motion artefacts.

After analysis of various image sensors and camera manufactures the Point Grey Flea3 2.0MP Colour USB3.0 Vision (Point Grey Research, Inc., Canada) camera was selected. The camera was built upon the e2v EV76C5706F image sensor (e2v technologies plc., England). This provided a maximum resolution of 1600x1200 using a 1/1.8” sensor, with 4.5 µm pixel elements. The resolution produced an aspect ratio of 1.3, similar to the 1.2 of the Oculus. The camera could stream colour images encoded in either RAW or YUY444 format at 59 fps, using a global shutter with minimum exposure of 0.014 ms. A shorter exposure would help minimise blur when observing fast moving objects.

Physically the camera was very small, measuring 29x29x30 mm and weighting only 35 g. However, this does not include the lens array, the selection of which shall now be discussed.

To produce an immersive and undistorted display it was important to match the FOV of the lens to that of the Oculus. Measurements performed upon the Oculus indicated a horizontal and vertical FOV of approximately 84° and 94°, respectively. Pose estimation accounted for image distortion through camera calibration. However, marker detection did not account for distortion and required relatively straight lines to detect the marker edges. Therefore, a lens with low distortion was also desirable. This would also improve the quality of image displayed via the HMD.

Wide FOV lenses, such as that required, typically induce considerable barrel, or fish-eye, distortion. Complex multi-element lenses may minimise this distortion through internal correction. However, this typically results in a large, heavy, and expensive lens.

The Kowa LM3NCM (Kowa Company, Ltd., Japan) lens selected provided a large horizontal and vertical FOV of 89.0° and 73.0°, with a distortion of only 0.46%. The wide FOV was achieved by a focal length of 3.5 mm. The lens was designed for a 1/1.8” sensor with 4.5 μm pixels, providing a central image resolution of 120.0 LP/mm. The lens offers manual focus and iris control, facilitating a minimum working distance of 100 mm, and control over light intensity.

Physically the lens measured 38.2 mm with a diameter of 42 mm, and a mass of 75 g. This again would likely result in an impractical commercial HMD system. However, it would provide an acceptable proof of concept system.

An assembly was developed to allow analysis of the research cameras. The research cameras were ensured to be a well aligned within their casing, therefore a ridged assembly was developed. The cameras featured several mounting holes on both their top and bottom surfaces. These were used to secure the cameras to the laser cut 6 mm acrylic assembly, shown in Figure 8.3. The cameras were aligned in parallel with a base line of 64 mm.

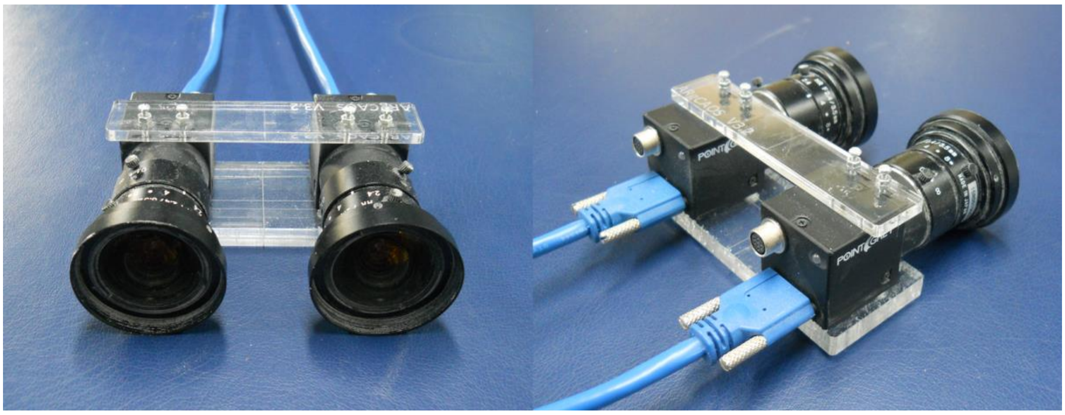


Figure 8.3: Research camera experimental assembly

To facilitate the connections required by the HMD and research cameras an alternative development system was acquired. The system featured an i5 5020 CPU with 8 GB of ram and a dedicated GeForce 840M GPU.

8.2.1.2.3 Probe and Resection Tool

The third generation system does not use a tool mounted tracking approach. Therefore, both the probe and resection tool were redeveloped to allow independent tracking.

In both incidences the camera mounting was replaced with a single 32 mm laser engraved marker. Both the probe and sheath of the Anspach system were 8 mm in diameter. Therefore, the same 3D printed marker mount was used, as shown by Figure 8.4 below. This attachment method allowed the tool-tip to marker separation to be adjusted. The separation for both tools was typically set to 80 mm.

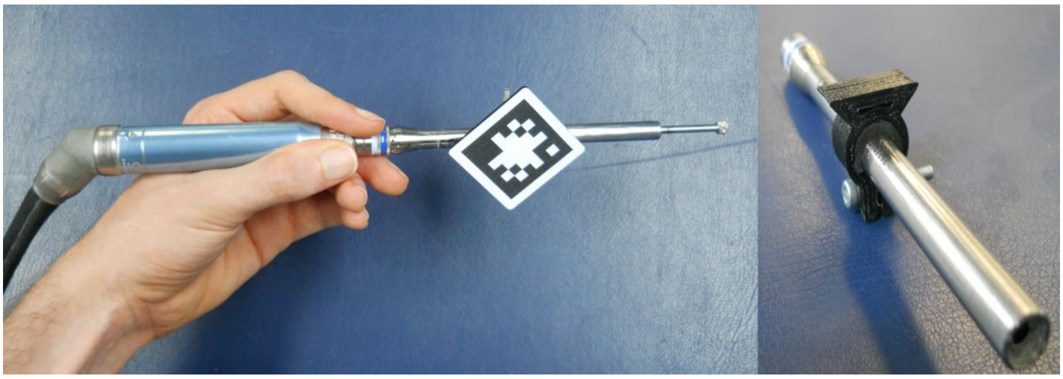


Figure 8.4: Probe and resection tool marker mounting

As with the second generation resection tool, vibration dampening foam rubber was used to line the mount. The remaining hardware of the resection tool, such as the control circuity, remained unchanged from the second generation system, as did large sections of the control software, as shall be discussed below.

8.2.2 Software

The third generation system followed many of the core design principles of the second generation system. One notable change to the system hierarchy was the removal of the $Stage$ classes. With the HMD it was desirable to move away from the traditional mouse and keyboard interface system. With this change, soft buttons - one of the core functionalities of the $Stage$ class - was removed. Therefore, the system was simplified by removing the $Stage$ class and placing the residual functions, such as call-back routing, within the $World$ class.

The second major modification was the incorporation of the HMD. Due to incomplete support by the Oculus SDK of the GLUT library, originally used for window management, the system was converted to the *Simple DirectMedia Layer library* (SDL2). SDL2, like GLUT, managed window creating and event forwarding. However, SDL2 was able to correctly generate handles to the window required to pass the render to the HMD.

The wide angle lenses within the Oculus induce considerable distortion. To resolve this, the associated software pre-distorts the image with inverted aberrations to cancel the effect of the lens, much like a compound lens. This is implemented via an intermedial framebuffer. All rendering is performed onto this framebuffer. The Oculus SDK then applies aberrations, both spatial and chromatic, before rendering the result to the default back buffer for display.

Furthermore, all rendering is performed twice, to produce the stereoscopic image required. Each camera image is first rendered to respective sides of the main framebuffer. To produce a convincing augmented experience it is required to transform and render each model such that it aligns with the physical view in both images. The marker pose is calculated relative to the left camera, as such for the left image objects are rendered as with the first generation system. The calculated pose, which represents the view matrix of the marker, is multiplied by the camera-to-camera transformation matrix, obtained through calibration. This transforms the pose of the marker such that it is viewed from the right camera, allowing aligned rendering to be performed as standard.

To obtain the original marker pose, stereoscopic pose estimation is performed. As previously discussed in Section 5.2, stereoscopic pose estimation is based upon epipolar geometry. Markers are located within the image frame, as described for the second generation system. The image corners correspond to the physical corners of the marker. Therefore, by repeating the marker detection process for the second camera it is immediately possible to match the corner image points between views, as illustrated by Figure 8.12 below.

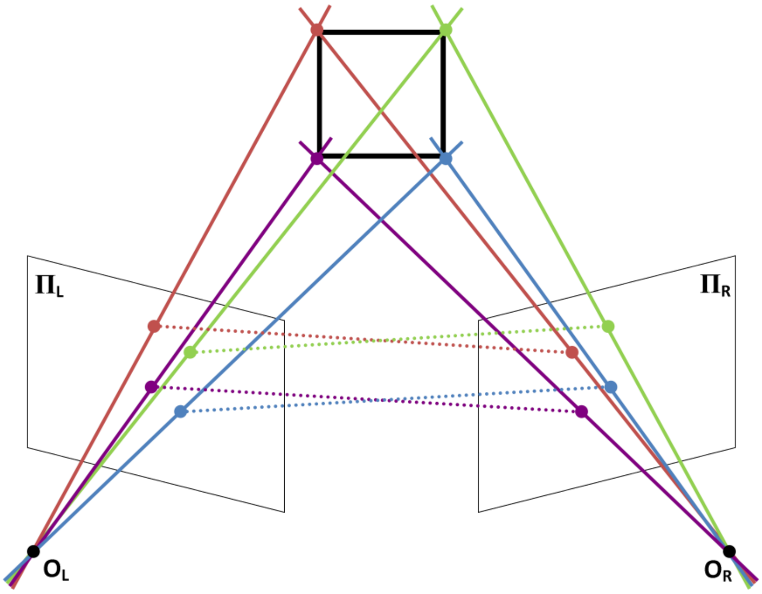


Figure 8.5: Stereoscopic matching of the four marker corners

With the four marker corners matched, the disparity coordinates may be defined. The X and Y components are taken from the left image coordinates, while the disparity value (d) is calculated as the left X component minus the right.

Once the four disparity values are calculated they may be converted to physical positions in the world coordinate system via the reprojection matrix (Q), as shown by Equation 8.1.

$$\begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -c_{L_x} \\ 0 & 1 & 0 & -c_{L_y} \\ 0 & 0 & 0 & f \\ 0 & 0 & -\dfrac{1}{T_X} & \dfrac{c_{L_x} - c_{R_x}}{T_X} \end{bmatrix} \begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix} \tag{8.1}$$

The reprojection matrix, similar to the intrinsic matrix used by the second generation system, defines a vector between the origin of the left camera and the image of the corner, using the principle points (c_{L_x} and c_{L_y}) of the camera. The known relative geometry of the two cameras (T_X), obtained during calibration, is then used to define the point along this vector as which the corner lies. Therefore, the corner locations are calculated in world space in three dimensions.

Equation 8.1 applies to a system where both image planes are coplanar and frontal-parallel – this results in both cameras having matching principles points reducing the bottom right element of Q to zero. As this is extremely challenging physically it is instead achieved through stereo-calibration, where images are adjusted to simulate a coplanar system, and stereo-rectification, where epipolar lines are aligned. These adjustments are applied on a per-corner image bases before application of Equation 8.1.

With the positions of the four corners of the marker defined in world space it is possible to calculate the pose of the marker. This is performed by calculating the transformation required to move a model of the marker corners (M_i) from the system origin onto the observed corners (O_i) calculated above, as shown by Equation 8.2 below.

$$O_i = [R|T]M_i \quad where\ i = 1..4 \tag{8.2}$$

Assuming an ideal system the transformation could be calculated geometrically using only three corners. However, the nonidealities of the system necessitate the use of a regressive approach, where the fourth corner may be used to increase the robustness of the solution. Here, least squares regression was performed via *singular value decomposition* (SVD).

The marker pose may be defined as the optimal 4x4 homogeneous matrix which minimises the total distance between the four corresponding corners of the observed and model marker. Equation 8.2 is rearranged to give the least squares Equation 8.3.

$$\Sigma^2 = \sum_{i=1}^{4} \left\| O_i - \hat{R}M_i - \hat{T} \right\|^2 \tag{8.3}$$

$\hat{T}$ is seen to be independent of either of the corner positions, and as such may be treated separately. Therefore, the rotation may be solved by centring both corner sets about the origin. This is performed by subtracting the centroids of each set from each corner, as shown by Equations 8.4.

$$O_{c_i} = O_i - \bar{O} \qquad M_{c_i} = M_i - \bar{M} \tag{8.4}$$

With the omission of $\hat{T}$ Equations 8.3 and 8.4 are combined to define the new least squares Equation .

$$\Sigma^2 = \sum_{i=1}^{4} \left\| O_{c_i} - \hat{R} M_{c_i} \right\|^2 \tag{8.5}$$

To minimise, Equation 8.5 is rearranged to Equation 8.6 below, whereby the squared rotation term is dropped as a rotation matrix multiplied by its transpose is equal to the identity matrix.

$$\Sigma^2 = \sum_{i=1}^{4} (\left\| O_{c_i} \right\|^2 + \left\| M_{c_i} \right\|^2 - 2 O_{c_i}^T \hat{R} M_{c_i} \tag{8.6}$$

It may be seen from Equation 8.6 that $\hat{R}$ may only effect the final term, as such this term must be maximised to minimise the summation. The final term may be expressed as a trace of product which may be rearranged by the cyclical permutability of trace functions to give Equation 8.7.

$$\Sigma^2 = \sum_{i=1}^{4} O_{c_i}^T \hat{R} M_{c_i} = Tr(O_{c_i}^T \hat{R} M_{c_i}) = Tr(\hat{R} M_{c_i} O_{c_i}^T) \tag{8.7}$$

From Equation 8.7 we define the correlation matrix at H by Equation 8.8 below.

$$\Sigma^2 = \sum_{i=1}^{4} O_{c_i}^T \hat{R} M_{c_i} \Rightarrow Tr\left(\hat{R} H\right) \quad where\ H = \sum_{i=1}^{4} M_{c_i} O_{c_i}^T \tag{8.8}$$

To maximise the function were take the SVD of H and substitute it into Equation 8.8, so shown by Equation 8.9 below.

$$\Sigma^2 = Tr\left(\hat{R} U \Sigma V^T\right) \quad where\ SVD(H) = U \Sigma V^T \tag{8.9}$$

Again, using the cyclical permutability of the trace function Equation 8.9 may be rearranged to give Equation.

$$\Sigma^2 = Tr(\Sigma V^T \hat{R} U) \tag{8.10}$$

It is noted that V^T, $\hat{R}$, and U are all orthogonal matrices. As such their product, denoted by M, is also an orthogonal matrix. Therefore, each element of M must be less than or equal to one. As Σ is a non-negative diagonal matrix, the trace function is maximised when the diagonal elements of M are one. To be an orthogonal matrix this defines M as the identity matrix. Therefore the trace function may be maximised by solving M as shown by Equation 8.11 below.

$$I = M = V^T \hat{R} U \Longrightarrow V = \hat{R} U \Longrightarrow \hat{R} = V U^T \tag{8.11}$$

With the optimal rotation calculated by Equation 8.11 above the optimal translation may be trivially calculated by Equation 8.12 below.

$$\hat{T} = \bar{O} - \hat{R}\bar{M} \tag{8.12}$$

To allow stereoscopic pose estimate the $Marker$ class was modified to store two sets of corner image coordinates. Furthermore, as information was required from both cameras pose estimation functions were moved to the new Hmd class. The class structure of the third generation system is illustrated by Figure 8.6 below.

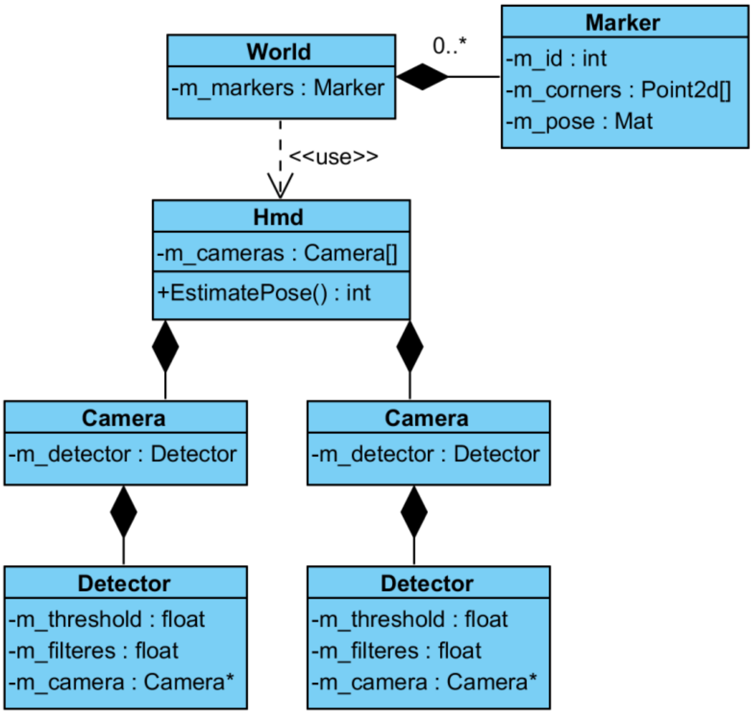


Figure 8.6: General class structure of third generation system

Each $Camera$ and associated $Dectector$ class may run idenpendently. Therefore, to optimise the system the two $Camera$ classes should be run simutaniously upon two separate processor threads. This considerably reduces the overall latency of the system, and allows frames to be better synchronised. Figure 8.7 illustrates the multi-threaded approach to frame acquisition and pose estimation.

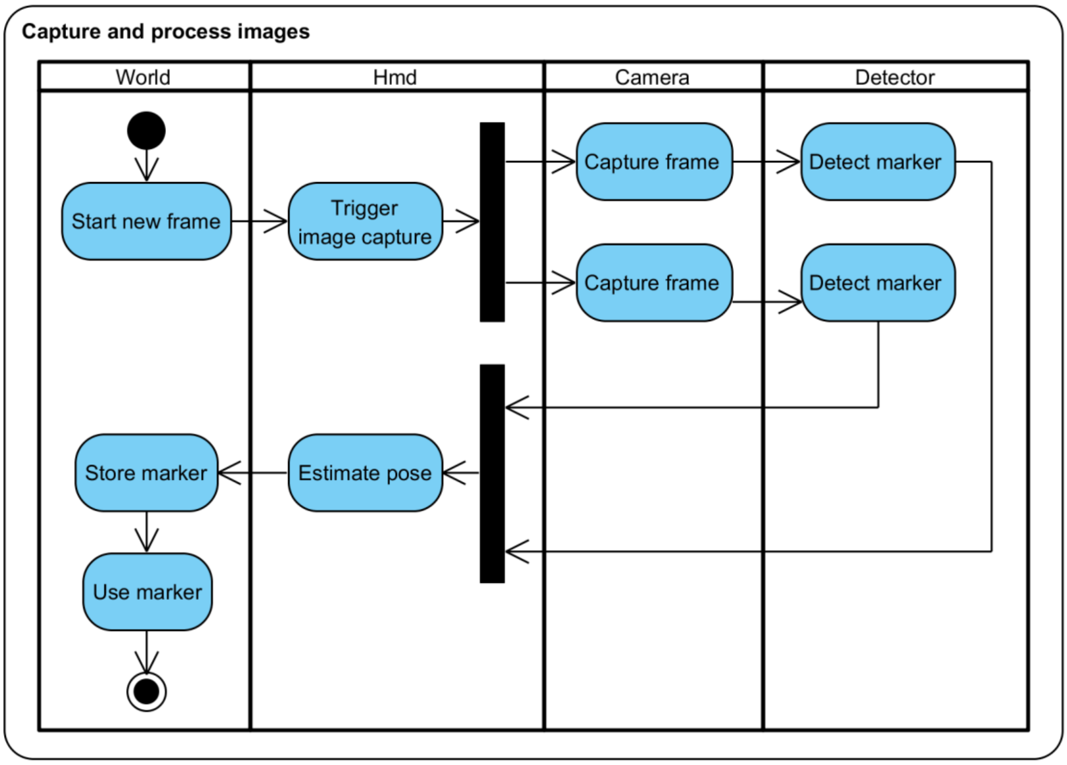


Figure 8.7: Third generation multithreaded image capture and processing

The implementation of the third generation system is briefly discussed is Appendix A3.

8.3 System Analysis

The third generation system underwent the same analysis procedure as the second generation system, namely angular, probe, and resection. An additional analysis was performed on two hip centre algorithms developed as part of the demonstration system.

8.3.1 System Calibration

Stereo calibration was performed upon the LifeCam system with both cameras set to a resolution of 1280x720, producing a maximum framerate of 20 fps. The research cameras were calibrated at their maximum resolution of 1600x1200, which was found to produce a maximum framerate of 35 fps.

8.3.1.1 Materials and Methods

Prior to calibration the cameras were aligned parallel in all axes. The baseline of the stereo pairs was set to 64 mm. A target with two crosses with a 64 mm separation was placed 4 m from the cameras. The left camera was aligned with the left cross using the standard alignment method. The right camera was then adjusted such that it was aligned with the right cross, thus aligning the two cameras. The camera alignment within the calibration suite is shown by Figure 8.8 below.

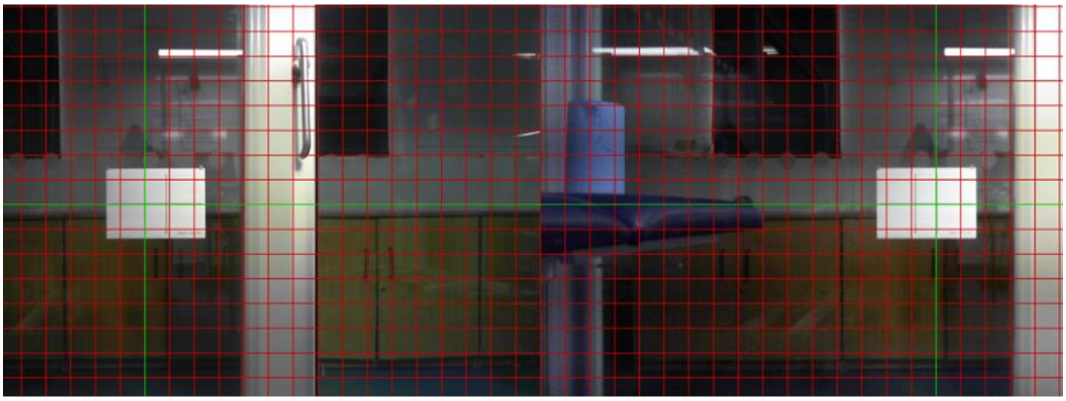


Figure 8.8: LifeCam optical alignment

The LifeCam system was calibrated at a nominal working distance of 250 mm, using the same asymmetric circle calibration grid as the second generation system. However, due to the wide FOV of the research cameras the original calibration grid proved too small, potentially reducing the quality of calibration. Therefore, a larger 13x5 grid was produced with 12 mm circle separated by 24 mm. The new grid was too large to measure fully under the microscope. Therefore, a series of measurements were taken and the mean printing error of 7.4 µm applied to the grid. The research cameras were calibrated at a nominal working distance of 300 mm.

The calibration suite was modified to perform stereo calibration. Following the procedure of the original system, the focus of both cameras was set. A large series of images were then captured and stored by both cameras with the calibration grid at various positions and orientations. Once all images were captured the system proceeded to calibrate the left and right cameras separately. Once calibrated, the individual intrinsic and distortion parameters were passed into the $StereoCalibratate$ function. This two stage approach was documented as producing superior results. Once calibrated the results were used to perform stereo rectification, as discussed in Section 5.2. Stereo rectification provided the disparity-to-depth mapping (Q) matrix fundamental to the system. The resulting rectification transformation was applied to the captured images as a sanity check for good calibration.

The above approach performed well for the LifeCam system. However, initial calibration of the research camera system produced poor and inconsistent results. Several modifications were explored including alternative distortion models. Consistent calibration results were achieved by adjusting the sensitivity and robustness to projective distortion of the blob extraction algorithm. This was not typically used as the system became more sensitive to background noise, considerably increasing erroneous detections. To compensate for this sensitivity calibrations were performed in front of a white screen.

Final stereo calibration was performed three times for each system, using an average of 515±69 and 911±192 images, for the LifeCam and research camera system respectively.

8.3.1.2 Results

The four intrinsic and five distortion monoscopic parameters were returned for each camera. Several additional parameters, which characterise the stereoscopic arrangement of the cameras, were also returned. The focus of this section shall be the reprojection Q matrix and the rotation and transformation matrices which are combined to produce the camera-to-camera transformation matrix. Further calibration parameters were utilised by the system, however, these two matrices best define the consistency of stereo calibration.

The LifeCam system produced a mean monoscopic RMSE of 0.11 pixels, significantly smaller than the original monoscopic calibration ($P < 0.01$). Conversely the research camera system produced a mean of 0.17 pixels, insignificantly larger than the original ($P = 0.1$). The RMSE of stereo calibration was also calculated as 0.14±0.02 and 0.18±0.01 pixels for the LifeCam and research camera systems, respectively. This was found to be a significant difference ($P = 0.01$).

The monoscopic calibration of the LifeCam system was found to produce significantly different focal parameters to the second generation system ($P = 0.02$ and $P = 0.01$). The image centres were not found to be significantly different ($P = 0.07$ and $P = 0.37$). All but the k_2 term ($P = 0.07$) of the distortion parameters were found to be significantly different. The variance of each of the parameters were not significantly different, excluding C_x ($P = 0.01$), k_2 ($P = 0.02$), and p_2 ($P < 0.01$). In these incidences the original calibration produced the smaller variance.

The research camera system was found to produce an image centre of 773.1±0.6 by 592.6±0.2 pixels for the left camera. F_x and F_y were found to be 810.9±0.1 and 811.2±0.1 pixels respectively. Distortion parameters k_1 and p_2 were found to be significantly larger than the LifeCam ($P < 0.01$ and $P = 0.04$). Parameters k_2, k_3 and p_1 were smaller, with the former two demonstrating significance ($P = 0.02$, $P = 0.01$, and $P = 0.24$). The variance of all monoscopic parameters, excluding p_1, were shown to be significantly smaller for the research camera system. With regard to stereo calibration, all parameters showed no significant different in variance between the two stereo systems (All $P > 0.05$).

The distributions of the rotation matrix and translation vector are shown below by Figure 8.9 and Figure 8.10 respectively. The mean results of these parameters are then summaries by Table 8.2.

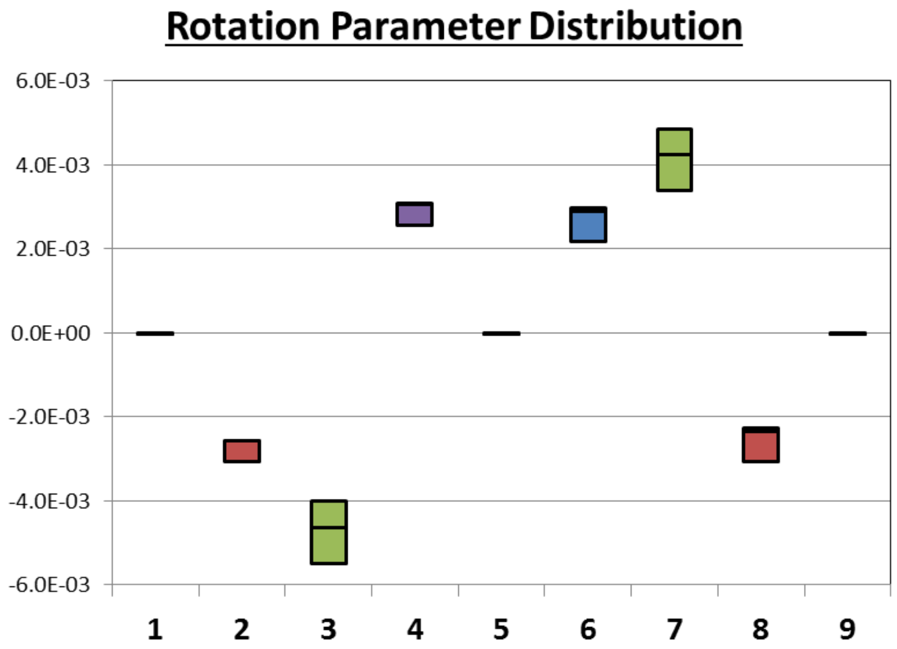


Figure 8.9: Rotation parameter distribution of third generation LifeCam system

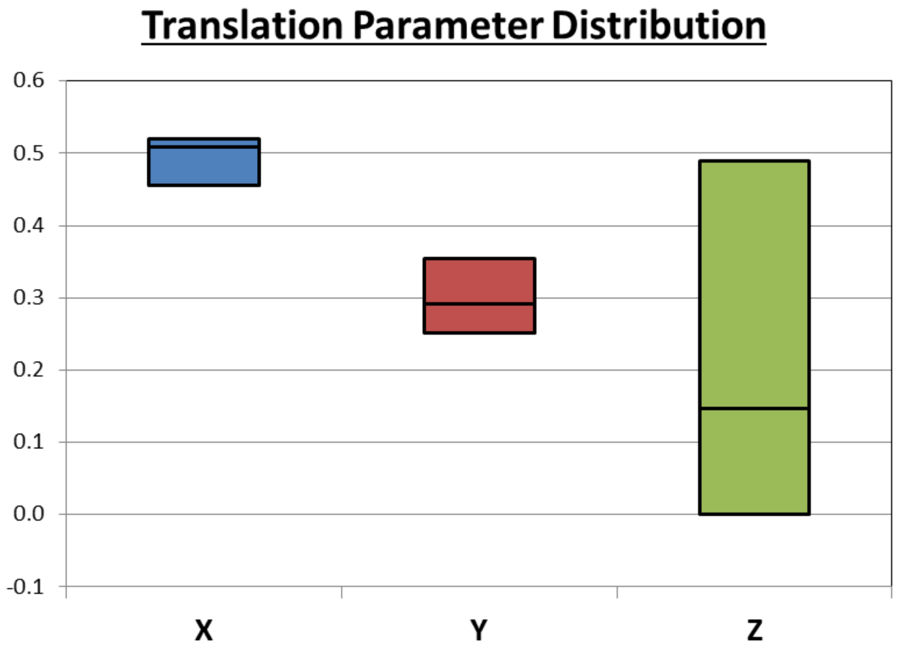


Figure 8.10: Translation parameter distribution of third generation LifeCam system

	Rotation		Translation (mm)
1.00±(1.97E-6)	-2.89±0.24E-3	-4.17±0.60E-3	-63.51±(2.79E-2)
2.90±0.24E-3	1.00±(1.57E-6)	2.67±0.36E-3	2.99±0.42E-1
4.16±0.61E-3	-2.68±0.36E-3	1.00±(1.72E-6)	1.70±0.21E-1

Table 8.2: Mean rotation and translation parameters of third generation LifeCam system

The same results are summarised for the third generation research camera system by Figures Figure 8.11 and Figure 8.12 and Table 8.3 below.

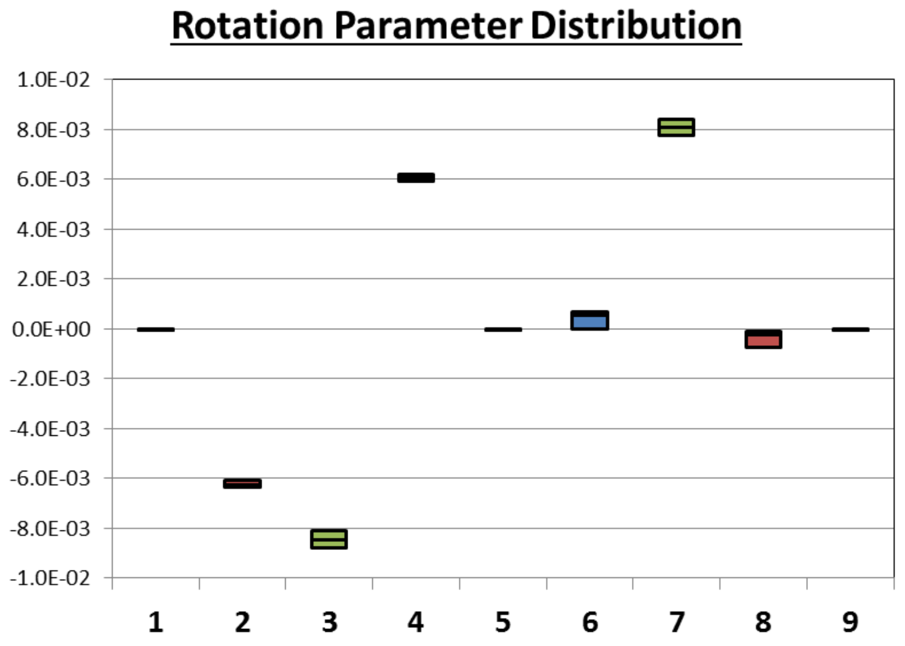


Figure 8.11: Rotation parameter distribution of third generation research system

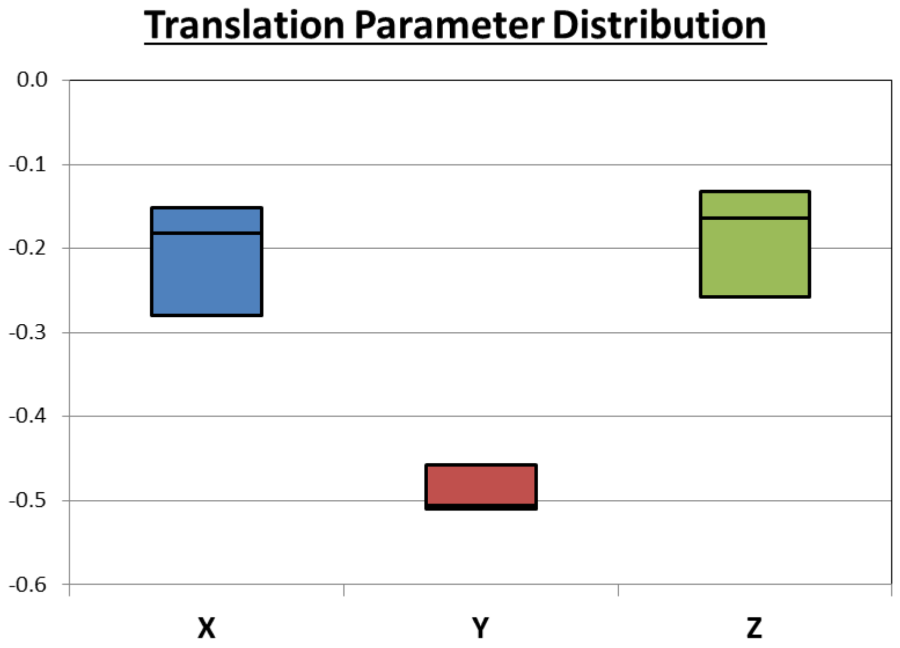


Figure 8.12: Translation parameter distribution of third generation research system

	Rotation		Translation (mm)
1.00±(2.53E-6)	-6.06±0.12E-3	-8.09±0.27E-3	-64.20±(5.47E-2)
6.06±0.11E-3	1.00±(6.51E-7)	2.89±2.94E-4	-4.28±0.23E-1

8.08±0.27E-3	-3.38±2.94E-4	1.00±(2.22E-6)	-1.73±0.53E-1

Table 8.3: Mean rotation and translation parameters of third generation research system

8.3.1.3 Discussion

Both stereo systems produced monoscopic RMSE values below the success threshold of 0.25 pixels, and therefore were considered to accurately describe the characteristics of the cameras. It was noted that the research camera system produced a larger RMSE for both monoscopic and stereoscopic calibration. It is recalled from Section 7.3.1 that the calibration RMSE is defined in pixels and may be approximately converted to millimetres using the FOV and resolution of the camera. Due to the very wide FOV provided by the research cameras the system is found to present a physical resolution of 0.3 mm/pixel, approximately three times that of the LifeCam system. The RMSE errors convert to 0.015±0.002 and 0.056±0.003 mm for the LifeCam and research camera system respectively. Therefore, calibrated systems provide a maximum theoretical accuracy required to meet the target of the system.

In contrast to the increased RMSE the research camera system was shown to produce the least variable results for the majority of calibration parameters, and as such may be considered more reliable. Additionally, the intrinsic monoscopic parameters were found to be consistent with expectations.

With cameras aligned both systems were expected to produce an identity matrix for the rotational matrix and 64 mm translation in the X-axis only. Both Tables 8.2 and 8.3 confirm the systems to produce the expected results. Both rotation matrices present a maximum error of 8.1E-3 with maximum standard deviation 0.3E-3. The LifeCam system produced the maximum error of 0.49 mm in the X-axis and maximum standard deviation of 0.2 mm in the Z-axis.

Both systems were shown to agree well with expected values and present minimal inter-calibration variance. Therefore, both systems were considered to be well calibrated and the mean results of the three calibrations were used for the remaining evaluation of each system.

8.3.2 Angular Analysis

As with the second generation system angular analysis was performed to segment marker pose errors from probe errors.

8.3.2.1 Materials and Methods

Angular analysis was performed as with Section 7.3.2.1. The axes of the central marker and left camera were aligned, as this defined the origin of the system. The base separation of the webcam system was set to 250 mm. Separation experiments were performed between 200 and 350 mm at 50 mm intervals. Vertical displacement varied between 50 and -25 mm in 25 mm intervals. Additionally, horizontal displacement was performed at 100 and 50 mm left and 25 mm right.

The research camera system used a base separation of 300 mm. Separation was varied between 200 and 350 mm. Vertical displacement was performed between 100 and -50 mm in 50 mm intervals. Horizontal displacement was performed between 200 mm left and 100 mm right, at 100 mm intervals.

8.3.2.2 Results

Figures Figure 8.13 and Figure 8.14 summarise the results of the aligned baseline measurements for the LifeCam and research camera systems respectively.

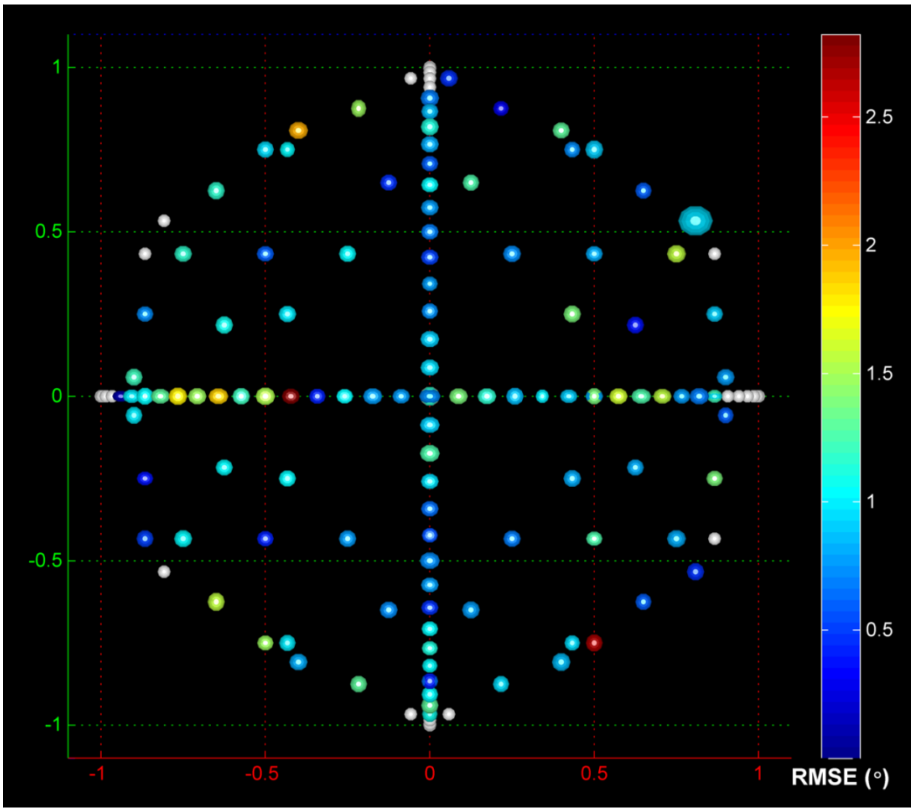


Figure 8.13: Summary figure of rotational RMSE of third generation LifeCam system, performed at 250 mm separation and zero lateral displacement

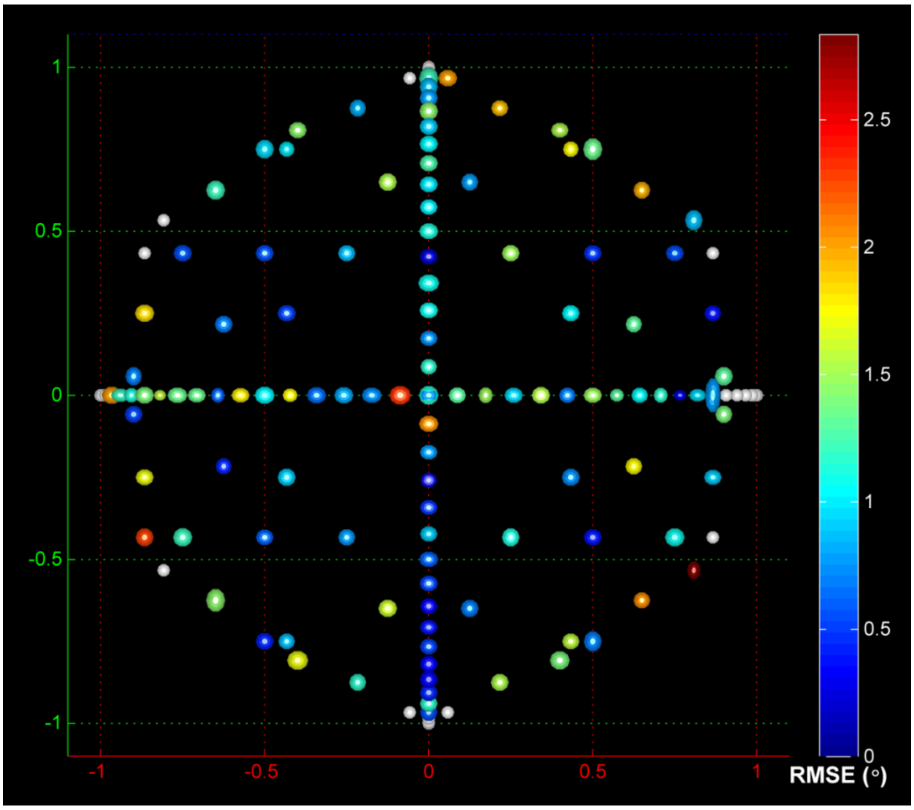


Figure 8.14: Summary figure of rotational RMSE of third generation research system, performed at 300 mm separation and zero lateral displacement

The statistical analysis of 7.3.2.2 was repeated for both systems. The X RMSE component was found to be significantly larger in the near-parallel range during Y-rotation (P = 0.01) for the LifeCam system. However, no other significant differences were found. Furthermore, several near-parallel ranges were found to present lower mean RMSE. The research camera system reported only a significantly smaller RMSE in the near-parallel region for the Z component during Y rotation (P < 0.01).

During Z rotation both systems reported significant differences when the marker was inverted. The LifeCam system found all three components to present larger errors when inverted, with the Y and Z components reaching significance (P = 0.1, P < 0.01, and P < 0.01). The research camera system found all three components to be significantly different (All P < 0.01). However, the Y component was significantly smaller.

The LifeCam system reported no significant differences at near-perpendicular angles. The research camera system found the Y component to be significantly larger in the near-perpendicular region during both X and Y rotation (P = 0.03 and P = 0.02).

Several significant differences were found in the standard deviations of both systems. However, as no trend was noted they shall not be discussed.

Figures Figure 8.15 and Figure 8.16 summarise the mean error, RMSE and standard deviation respectively, of the third generation LifeCam system during different modes of rotation. Figures Figure 8.19 and Figure 8.20 repeat this summary for the research camera system.

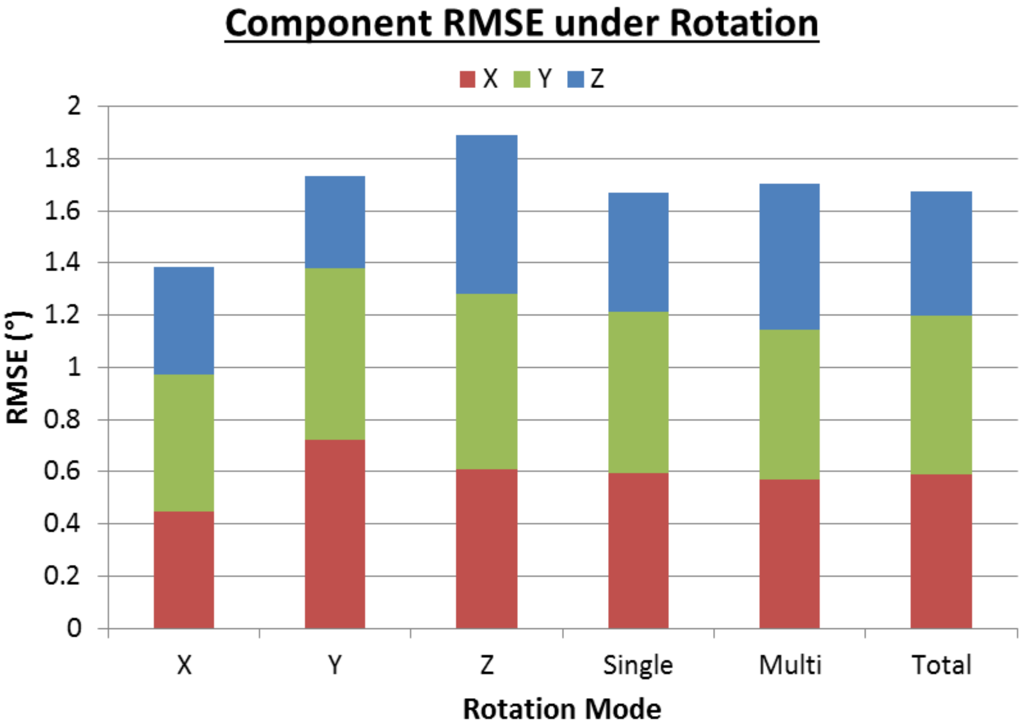


Figure 8.15: Effect of rotation modes upon mean component RMSE of third generation LifeCam system

All three components were found to be significantly smaller during X rotation (P < 0.01, P = 0.03, and P < 0.01). Y and single mode rotation also presented significantly smaller Z components (All P < 0.01). Conversely, Z and multi-mode rotation produced significantly marker Z components (All P < 0.01).

The Y component was significantly larger than the remaining two components during X, Y, and single mode rotation. The Z component was significantly smaller during Y and single mode rotation.

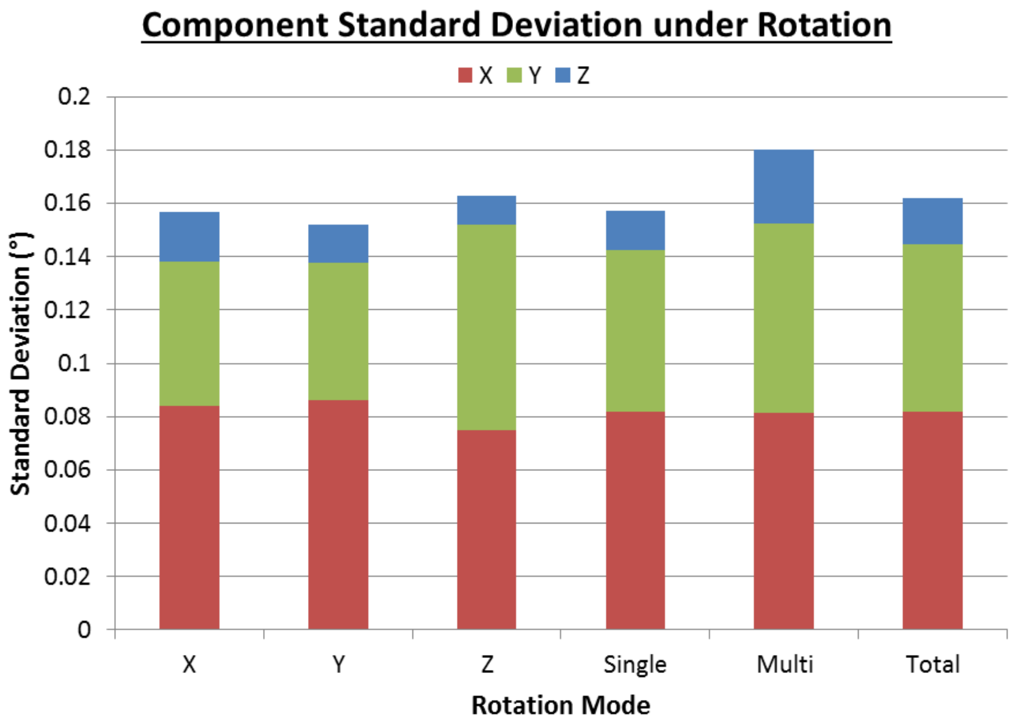


Figure 8.16: Effect of rotation modes upon mean component standard deviation of third generation LifeCam system

The standard deviations are seen to be very consistent across all modes of rotation. For all modes excluding Z rotation the Y component was significantly smaller than the X component (All P < 0.01 excluding P = 0.26). The Z component was significantly smaller than the remaining components for all rotation modes (All P < 0.01).

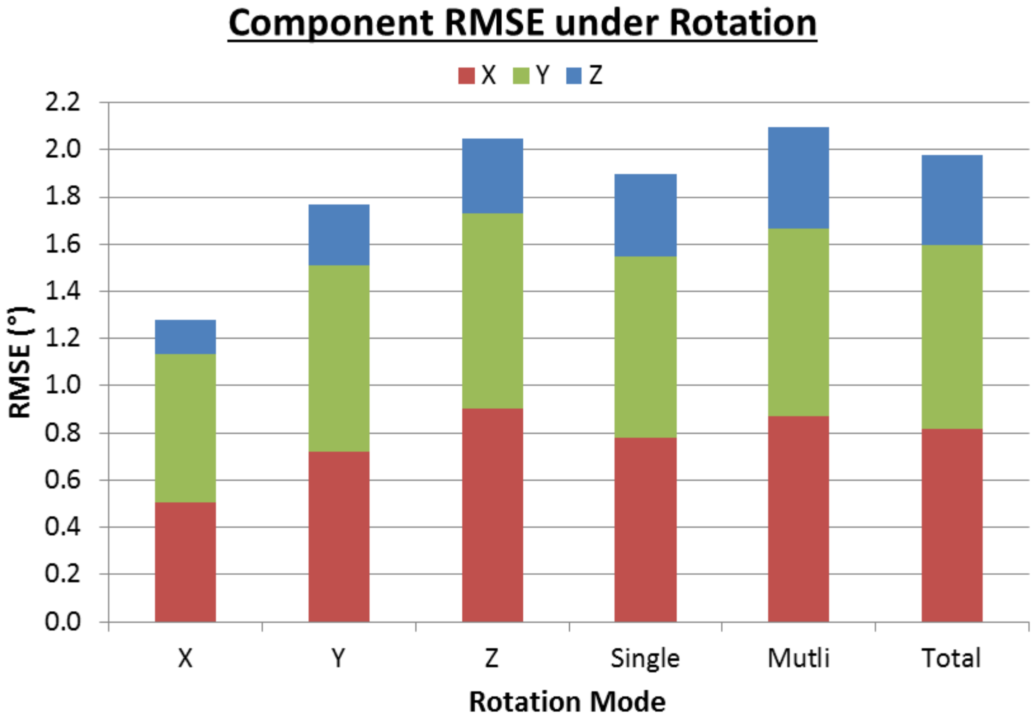


Figure 8.17: Effect of rotation modes upon mean component RMSE of third generation research camera system

The mean RMSE of the research camera system shows a similar distribution to the LifeCam system shown by Figure 8.15. Again all three components were found to be significantly smaller during X rotation (All P < 0.01).

The Z component was found to be significant small than both the X and Y components during all modes of rotation (All P < 0.01). No significant differences were found between the X and Y components.

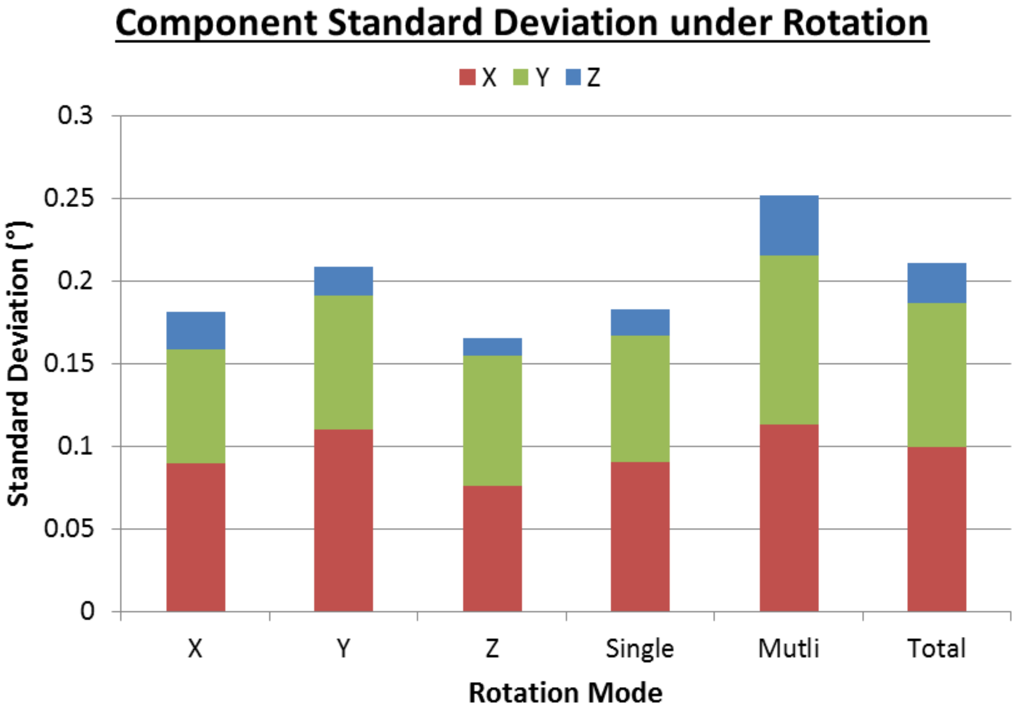


Figure 8.18: Effect of rotation modes upon mean component standard deviation of third generation research camera system

The X, Y, and single mode rotations were found to produce significantly smaller values for seven of nine components (All P < 0.01). The Z component was significantly smaller for each mode (All P < 0.01). The X component was significantly smaller than the Y component for each mode excluding Z and multi-mode (All P < 0.02 excluding P = 0.26 and P = 0.08).

Figures Figure 8.19 and Figure 8.20 illustrate the effect of camera-marker separation upon the angular error of the system.

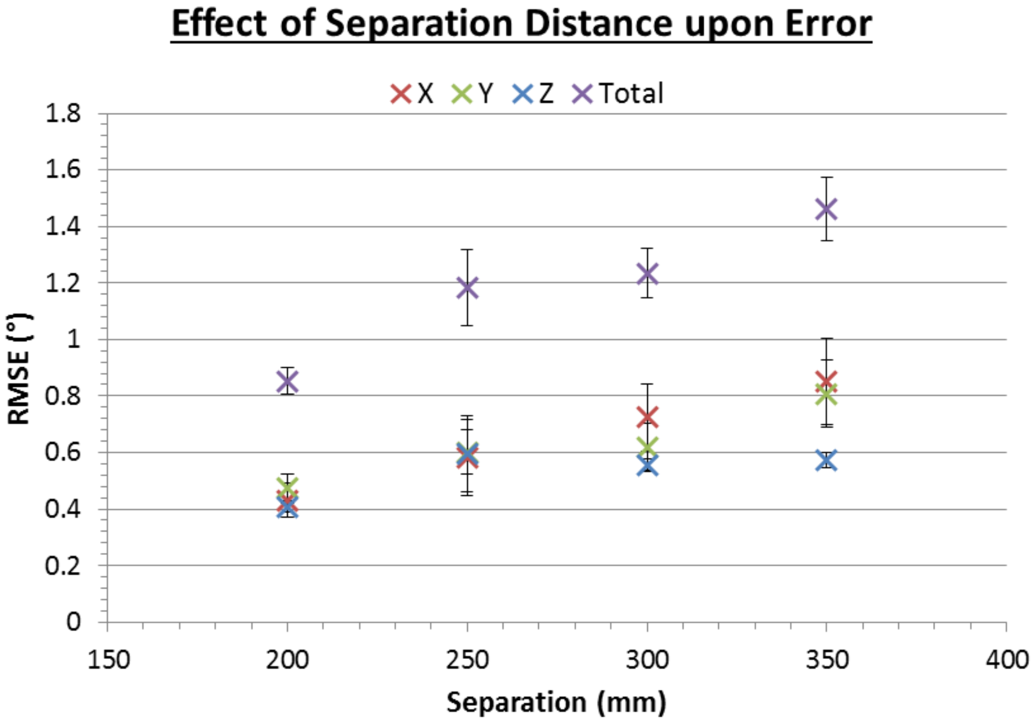


Figure 8.19: Effect of camera-marker separation upon the error of the third generation LifeCam system

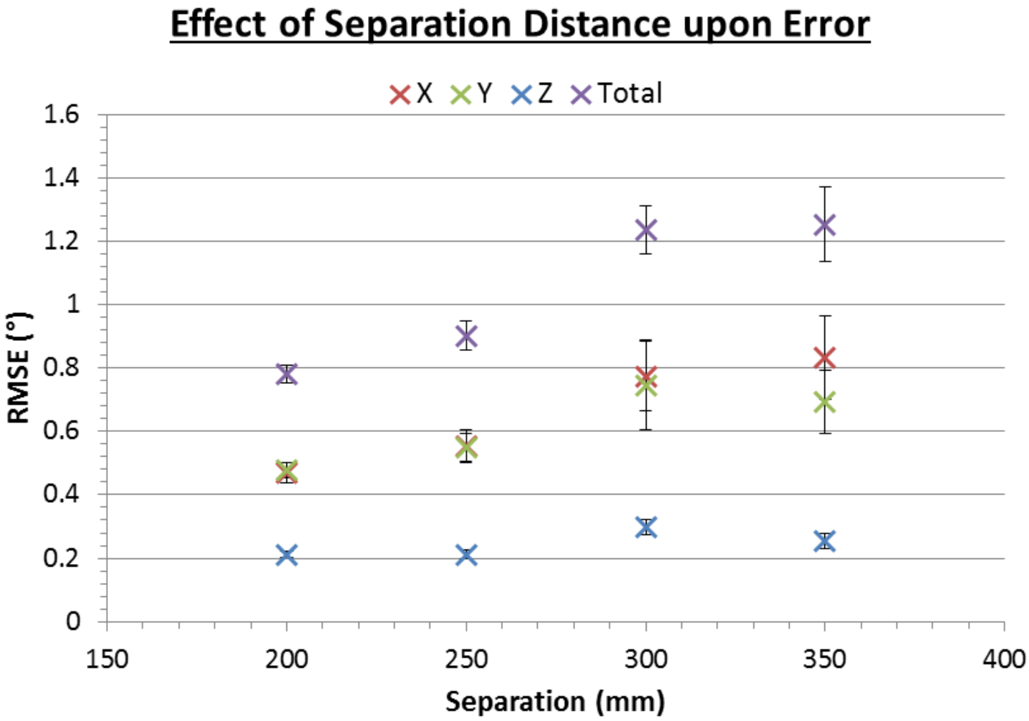


Figure 8.20: Effect of camera-marker separation upon the error of the third generation research system

Both systems presented the expected decrease in both accuracy and precision with increased separation. The mean total errors of both third generation systems are compared with those of the second generation system by Figure 8.21 below.

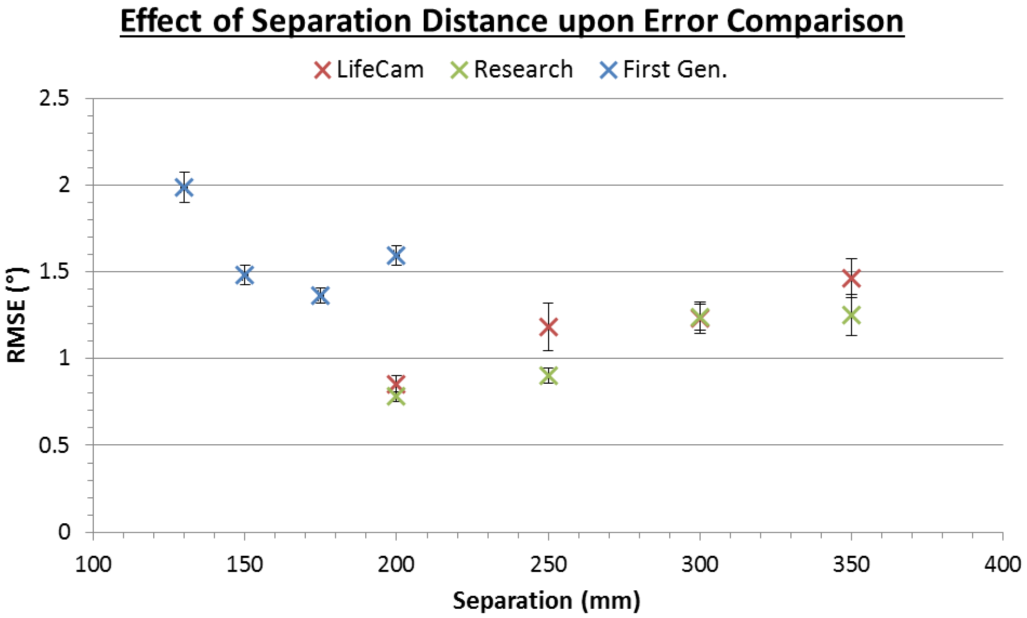


Figure 8.21: Comparison of separation error of both LifeCam and Research third generation systems with second generation system

Despite the lower separation utilised during analysis of the second generation system is it observed from Figure 8.21 that it produced larger RMSE error than both third generation systems. At a separation of 200 mm the research camera system was found to produce a significantly lower RMSE and standard deviation than both the second generation system ($P < 0.01$ and $P < 0.01$) and the LifeCam third generation system ($P < 0.01$ and $P = 0.05$). The LifeCam system was also found to be significantly better than the second generation system ($P < 0.01$ and $P < 0.01$). The research camera system was found to produce significantly better results that the LifeCam system at all separations excluding 300 mm ($P = 0.19$).

Figures Figure 8.22 and Figure 8.23 show the effect of vertical displacement upon the error of the LifeCam and research camera systems, respectively.

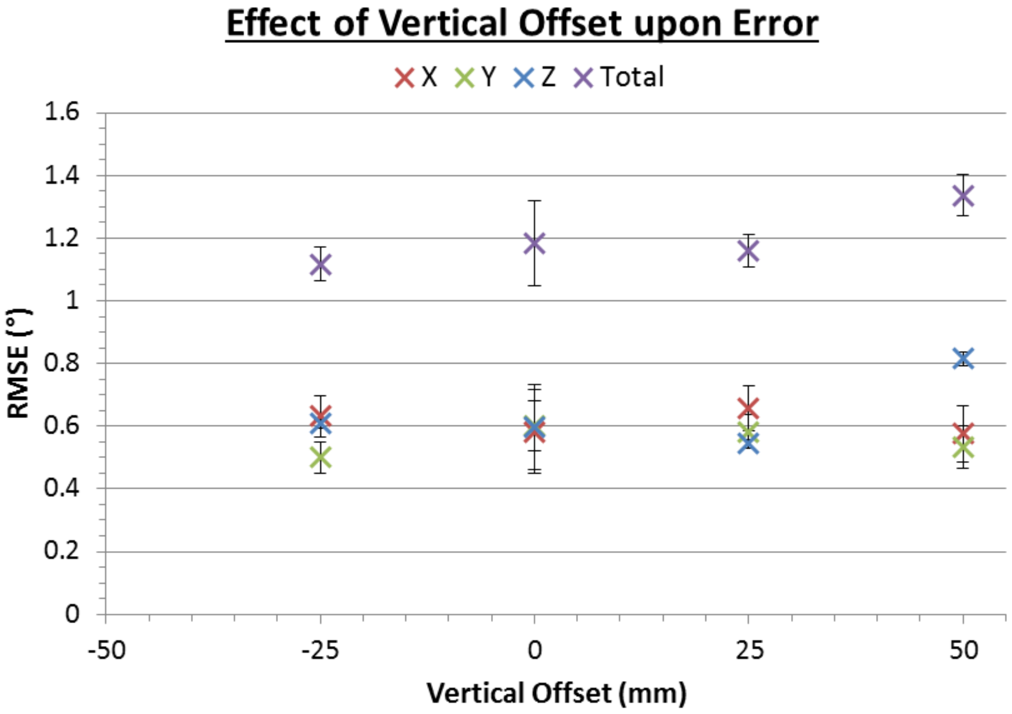


Figure 8.22: Effect of vertical offset upon error of third generation LifeCam system

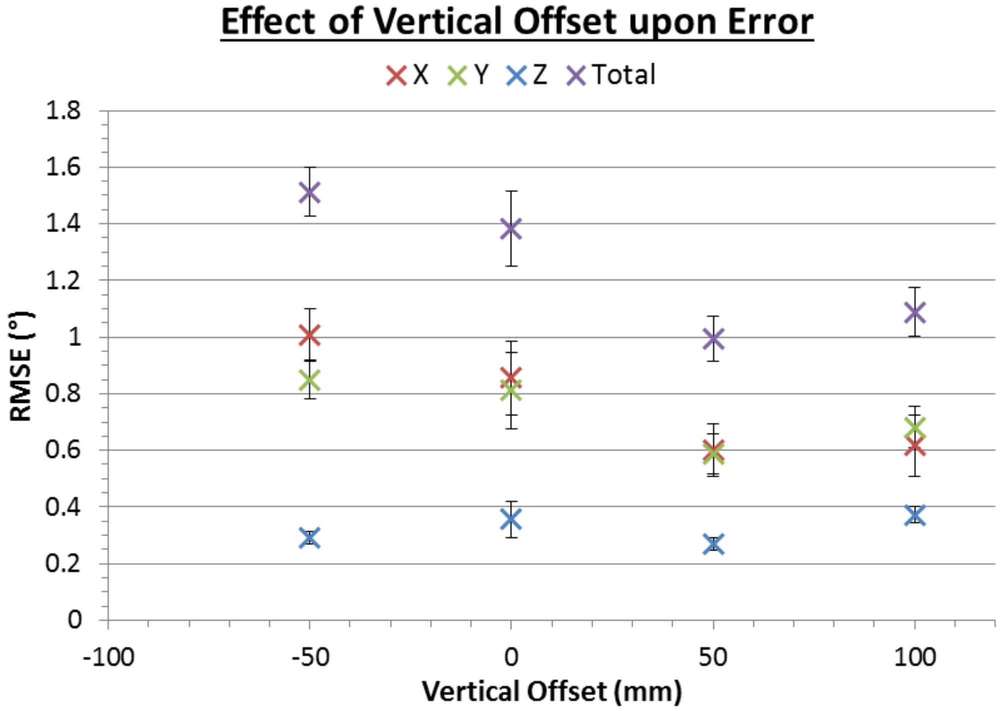


Figure 8.23: Effect of vertical offset upon error of third generation research camera system

As may be seen in Figure 8.22 the LifeCam system produced relatively consistent results across the range of vertical offsets. Only the Z component at 50 mm offset was found to be significantly larger (P < 0.01). The research camera system, as shown by Figure 8.23, presents an improved mean error under positive vertical offset. However, all components of the 100 mm offset error were shown to be greater than those of the 50 mm offset, with the Y and Z components reaching significance (P = 0.05 and P < 0.01). All error components of the 50 mm offset were significantly reduced in comparison to the aligned data (All P < 0.01). Only the X component of the -50 mm offset was shown to be significantly worse than the aligned data (P < 0.01).

The effect of horizontal offset upon the LifeCam and research camera systems are shown by Figures Figure 8.24 and Figure 8.25 respectively.

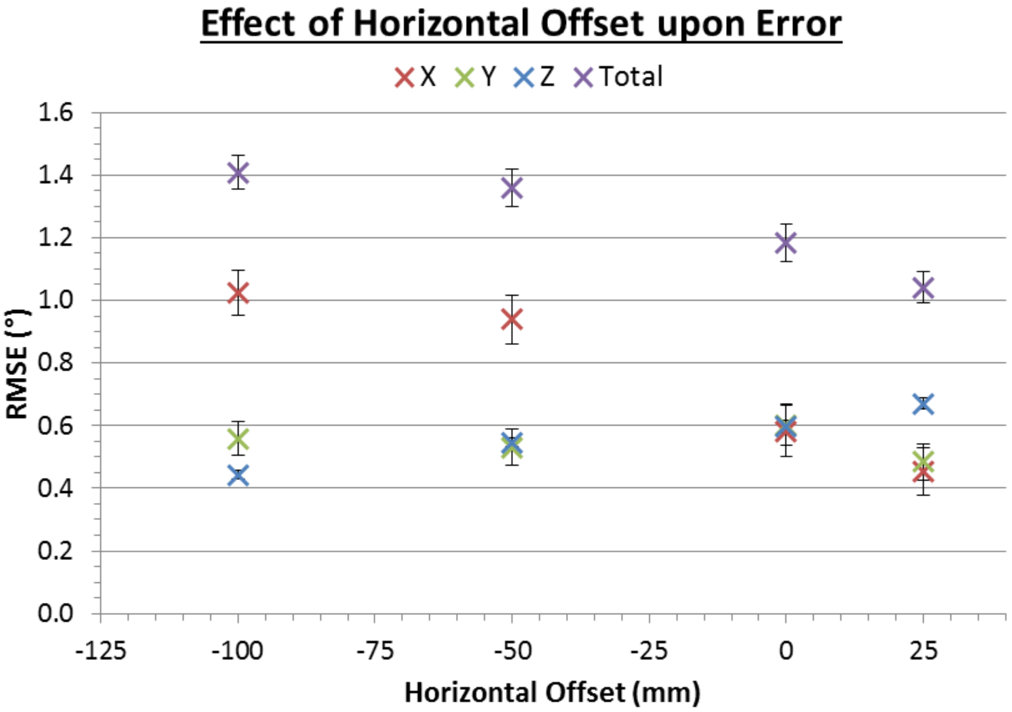


Figure 8.24: Effect of horizontal offset upon error of third generation LifeCam system

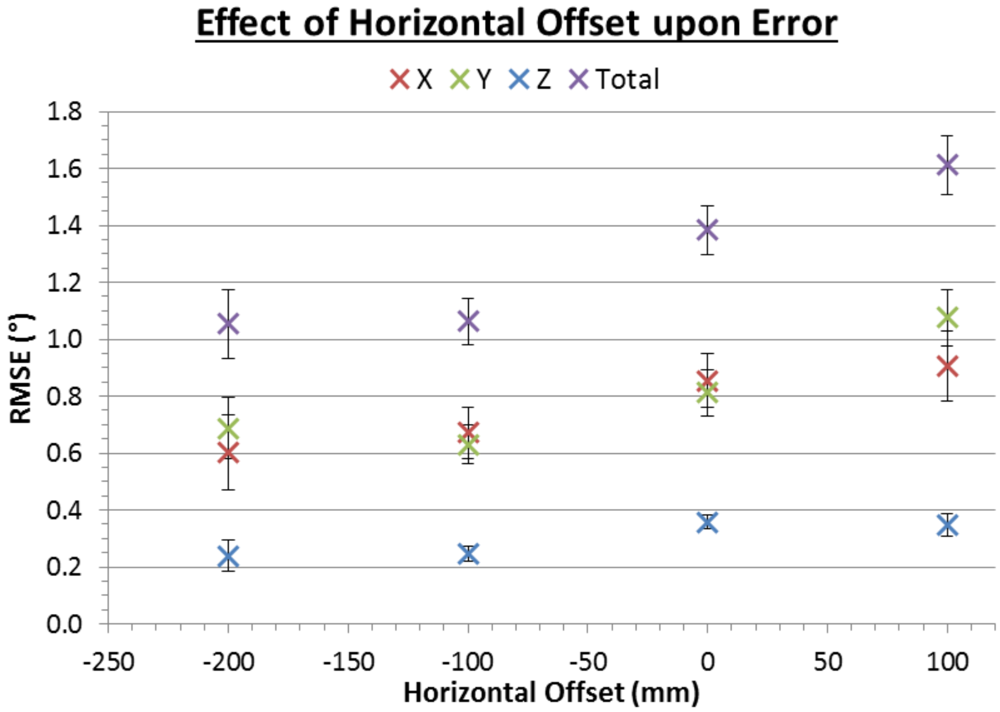


Figure 8.25: Effect of horizontal offset upon error of third generation research camera system

The relatively limited FOV of the LifeCam system considerably reduced the horizontal range over which an offset could be applied. The X and Y components of the error were found to increase significantly between 0 and -50 mm ($P < 0.01$ and $P < 0.05$). The Z component decreased over this range. However, this did not reach significance ($P = 0.13$). This resulted in a significant total error increase of a factor of 1.24 ($P < 0.01$). No significant differences were observed between the -50 and -100 mm measurements (All $P > 0.11$). The transition between 0 and 25 mm significantly reduced the X and Y components but increased the Z component (All $P < 0.01$). This resulted in a non-significant total decrease factor of 1.05 ($P = 0.14$). With regards to standard deviation the Y and Z components of the -50 mm offset were significantly reduced (Both $P < 0.01$). The X component produced a non-significant decrease ($P = 0.17$). This resulted in a significant total decrease of factor 0.92 ($P = 0.03$). No components of the -100 mm offset researched significance (All $P > 0.14$), presenting both increases and decreases. The Y component of the 25 mm offset was significantly smaller ($P = 0.02$), which resulted in a significant total decrease factor of 0.8 ($P = 0.01$).

The increased FOV of the research cameras permitted a broader offset range. The -100 mm offset was found to be significantly reduced for all components (All $P < 0.01$), resulting in a significant reduction factor of 0.8 ($P < 0.01$). Increment to an offset of -200 mm produced no significant differences (All $P > 0.13$) with a total factor of 0.99. The 100 mm offset was found to increase all components. However, only the Y component reached significance ($P = 0.16$, $P < 0.01$, $P = 0.08$). This resulted in a total increment factor of 1.2 ($P < 0.01$). The standard deviations of the zero and -100 mm offset were found to be statistically similar, barring the Y component ($P = 0.15$, $P < 0.01$, $P = 0.20$, and $P = 0.09$). All components of the remaining -200 and 100 mm offsets were found to be significantly larger (All $P < 0.01$), producing increase factors of 1.39 and 1.21, respectively.

8.3.2.3 Discussion

In contrast to the second generation system, neither of the third generation systems presented a significant increase in error at near-parallel orientation. This is indicative of the direct extraction of 3D information possible via the stereoscopic arrangement. As the position of each corner was determined in 3D world space the third generation systems are considerably less sensitive to the image induced orientation errors discussed in Section 7.3.2.3. The cause of increased error during marker inversion is not immediately clear. The error could not be readily attributed to misalignment, as this would be expected to produce a symmetrical error. Furthermore, the standard deviations, which were not directly affected by misalignment, presented a similar error distribution, although did not show significance.

The research camera system presented a significantly larger Y component in the near-perpendicular regions, not seen with the LifeCam system. This is potentially a result of the decreased spatial resolution of the research camera system. At near-perpendicular orientation the system has fewer edge pixels to define the corner locations and as such pose estimation is more susceptible to errors. Furthermore, as shall be discussed below, stereoscopic systems present a non-linear error with separation along the optical axis. At near-perpendicular angles two of four of the corners are at greater separation, which may contribute to the increased error.

As indicated by Figures Figure 8.15 and Figure 8.17 both systems produced similar RMSE distributions. The LifeCam system was seen to produce marginally superior results. This is indicated for by the increase spatial resolution provided by the reduced FOV and separation of the LifeCam system.

Despite presenting no significant increase in error in the near-parallel range, both systems indicate an increased error during Z rotation. This potentially indicates a weak near-parallel error dependence whereby the increased samples obtained during Z rotation allow the error to reach significances. As illustrated by Figures Figure 8.16 and Figure 8.18, neither system presented a significant increase in standard deviation during Z rotation. Therefore, the near-parallel error mechanism implicated for the second generation system in Section 7.3.2.3 is unlikely to contribute significantly to the error observed.

In an effort to optimise the system and as a result of the relative lack of visual distortion, distortion correction was not applied to the camera images. Instead, the compensation was applied to the induvial extracted corner points, greatly reducing the computational intensity. However, accounting for the distortion at this late stage has a potentially detrimental effect on the accuracy of the subpixel corner detection algorithm used, as a linear regression was performed upon the contour edges. Such errors would be most pronounced at near-parallel angles where the four edges of the marker would experience most distortion. It is noted, however, that preliminary experiments performed using image un-distortion produced significantly larger errors.

The increase in both RMSE and standard deviation observed during multiple axes rotation resulted from the highly angled and particularly inverted measurements. Due to the reduced lighting and resulting edge pixel count the edges of the marker were less well defined, resulting in increased pose errors.

Figure 8.26 below depicts a frontal-parallel coplanar system, which the third generation system represents after rectification.

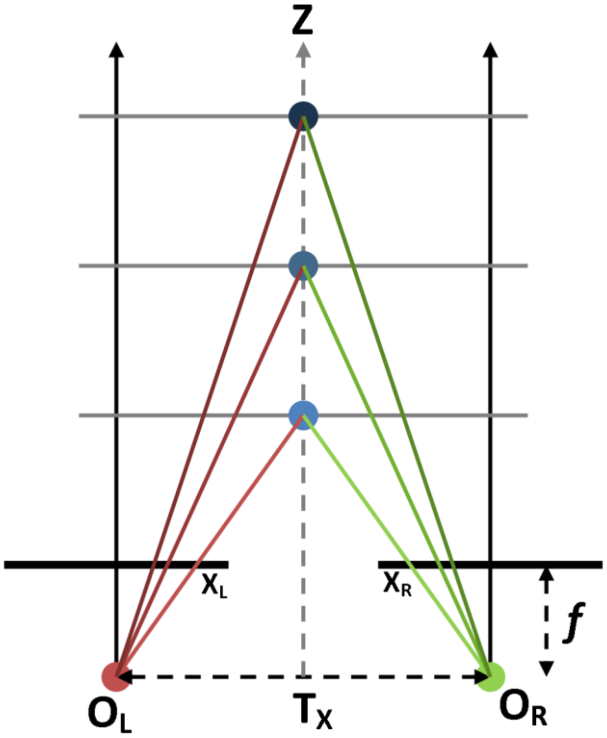


Figure 8.26: Relation between depth and disparity in a frontal-parallel system

Similar triangles are used to define the depth in terms of disparity as shown by Equation 8.13.

$$Z = \frac{fT_X}{d} \quad where\ d = x_L - x_R \tag{8.13}$$

The error of Z may be derived by taking the partial derivative with respect to d. The remaining two elements may be omitted as they are expected to produce relatively little error for a well calibrated system. Therefore, the depth error may be defined by Equation 8.14 below (Bradski & Kaehler 2008).

$$\partial Z = \partial d \frac{Z^2}{fT_X} \tag{8.14}$$

It is found the depth error is proportional to the square of the physical depth. Furthermore, disparity error is expected to increase with distance due to reduced spatial resolution. A depth error would induce a rotational error as shown by Equation 8.15, where L is the edge length of the marker.

$$\partial\theta = \tan^{-1}\left(\frac{2\partial Z}{L}\right) \qquad (8.15)$$

Over the range measured this produces a near linear increase, as observed for both third generation systems ($R^2 = 0.93$ and 0.90). Figure 8.21 indicates that the research camera produced superior results at all separation. Initial consideration suggests that the LifeCam should provide greater accuracy due to the increased spatial resolution resulting from the small FOV resulting in a larger spatial resolution. This is supported by Equation 8.14 where f is inversely proportional to FOV. It was demonstrated by Section 8.3.1.3 that the research camera system produced the lower standard deviation of calibration parameters, and thus presented a more consistent calibration. This is indicative of the lower quality, and potential inter-calibration movement, of the LifeCam lens and likely accounts for the improved accuracy observed for the research camera system.

Both systems were seen by Figure 8.21 to outperform the second generation, irrespective of camera-marker separation. This demonstrates the advantage of extracting the third dimension from the image disparity, reducing the dependence on pose estimation.

As shown by Figure 8.22 the LifeCam system presented no significant effect of vertical displacement upon orientation error. Under a well calibrated system this is the expected result as shown by Equation 8.14. However, Figure 8.23 demonstrates a significant variance in the error of the research camera system under vertical offset. It is postulated that this variance results from differing lighting conditions. Variant lighting is expected to have a more impact on the research camera system for two reasons. Firstly, the vertical offset of the research cameras was varied by a greater degree, and as such greater variance in lighting was observed – the marker was considerably darker while downward facing. Secondly, the research cameras utilised a much shorter exposure time, and as such were more sensitive to lighting changes.

Under horizontal offset the two systems produced contrasting results. The LifeCam system showed an increase in error as the camera was moved from right to left, while the research camera predominately indicated the opposite. However, only the initial offsets showed significance. Again, under ideal calibration, error is expected to remain relatively constant as all measurements are performed upon a single plane of constant disparity. However, increased horizontal offset results in increased camera marker separation. Therefore, the spatial resolution is reduced and the disparity error increased. This would be expected to have greater effect under positive offset, as the separation from both cameras is increased. This was observed for the research camera system. However, a 100 mm positive offset would produce an equivalent total camera separation as a -164 mm offset. Figure 8.25 shows the -200 mm offset to produce an RMSE significantly less than that of 100 mm. However, the standard deviation, which again is less susceptible to misalignment errors, does show both -200 and 100 mm to be significantly larger than zero and -100 mm. Therefore, the research camera system is potentially shown to present the expected response. However, the LifeCam system, as shown by Figure 8.24, produced the opposite response.

As shown by Section 8.3.1, the LifeCam produced a significantly less consistent calibration than the research camera system, particularly with respect to the rotation matrix. It may be shown that rotational alignment error may induce position errors. Furthermore, these errors may be shown to be dependent upon the 3D position of the imaged object (Dwarakanath et al. 2014; Santoro et al. 2012; Zhao & Nandhakumar 1996). Therefore, it is possible the reduced calibration quality is responsible for the error pattern observed for the LifeCam system.

8.3.3 Probing Analysis

As with previous chapters the probing analysis was performed as it simulated many of the core functionalities of the system.

8.3.3.1 Materials

Probing analysis was performed upon the same known geometry described by Section 7.3.3.1. A 32 mm marker was affixed to the 8 mm probe. The marker-tip separation was set to 80 mm for both the LifeCam and Research camera systems.

The sampling procedure of Section 7.3.3.1 was repeated for the LifeCam system including marker error adjustment measurements. Fixed, held, and moving samples were performed for the research camera system. The effect of marker size was also investigated by replacing the 32 mm target marker with either a 24 or 16 mm marker. Additionally, the camera-target separation was varied between 200 and 500 mm at 100 mm increments. Finally, an alternative marker was investigated. The 46 mm marker, shown in Figure 8.27, was designed to increase the edge lengths, thus providing greater contour points, potentially increasing accuracy, without considerably increasing the footprint of the marker. A marker perimeter-area ratio of 0.14 was obtained, compared to 0.09 for a square marker of equal edge length. All experimental modifications were tested using a fixed probe.

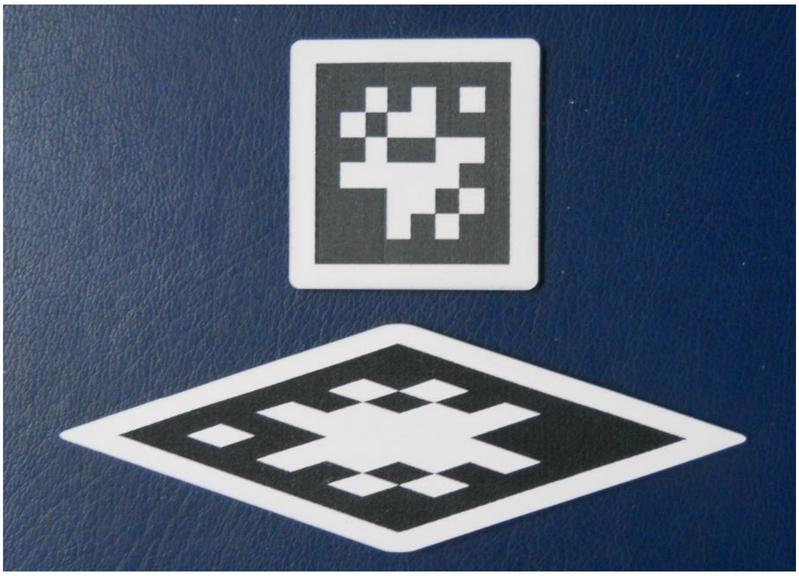


Figure 8.27: Alternative 46 mm marker design shown below standard 32 mm marker

8.3.3.2 Results

Figures Figure 8.28 and Figure 8.32 summarise the probing accuracies and precisions of the LifeCam and research camera systems respectively.

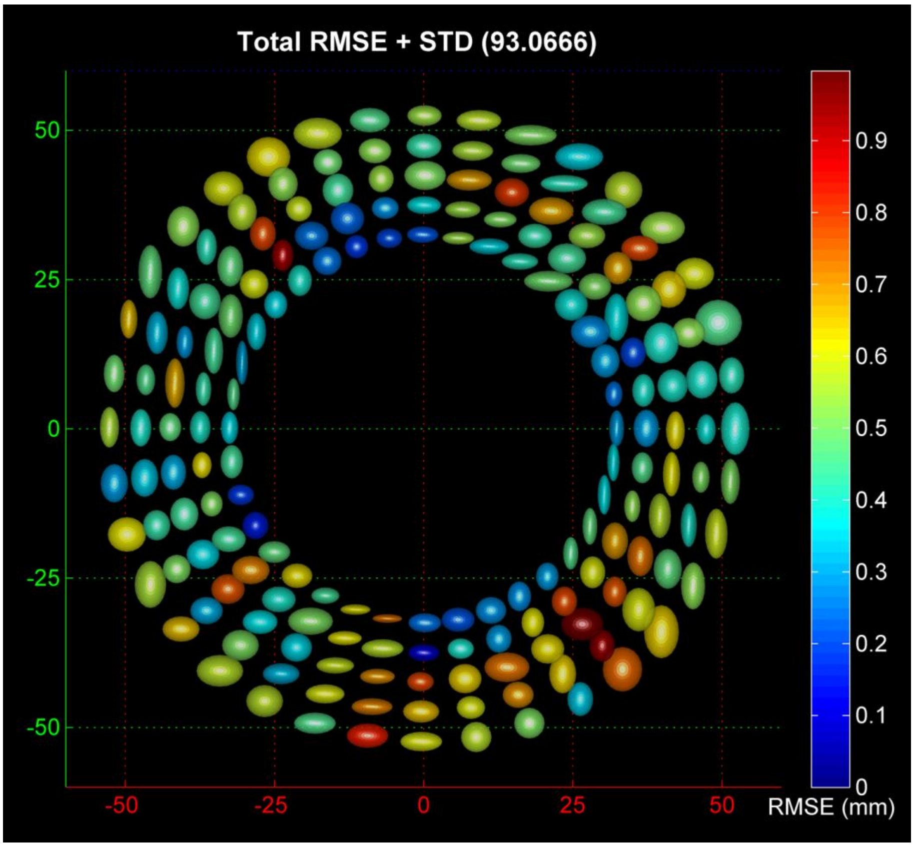


Figure 8.28: Example of third generation LifeCam probing experiment

The LifeCam system presented a mean RMSE of 0.54±0.03 mm and standard deviation of 0.08±0.01 mm over the fixed probe samples. This was adjusted to 0.52±0.07 mm (P = 0.42) and 0.12±0.02 mm (P = 0.02), respectively, upon compensation of marker errors. The effect of increased marker-probe separation, resulting from different levels, is shown by Figure 8.29 below.

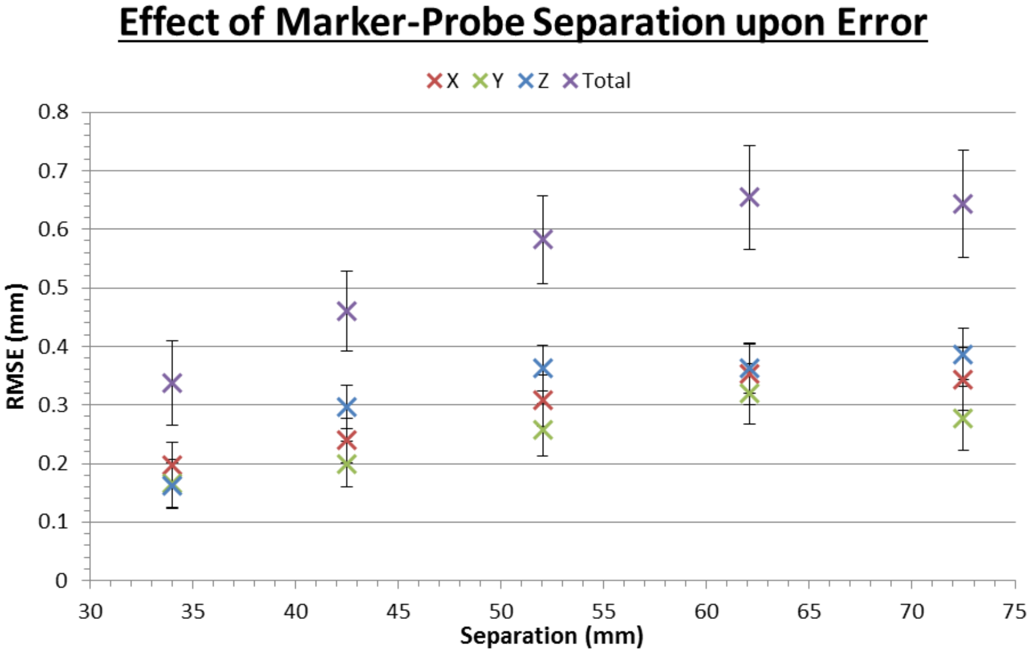


Figure 8.29: Effect of marker-probe separation upon third generation LifeCam system

Analysis indicated a significant RMSE increase between the initial three levels across all axes (All P < 0.01). The third increment reached significants only for the Y and total component (P = 0.08, P = 0.13, P = 0.49, and P = 0.02). For the final increment all components excluding the Z-axis produced a non-significant decrease in error (P = 0.39, P = 0.08, P = 0.09, P = 0.36).

The standard deviation also reported significant increases for the initial three increments. However, the first increment produced an insignificant decrease in the X and Y components (P = 0.27 and P = 0.12).

Figures Figure 8.30 and Figure 8.31 plot the significance of RMSE and standard deviations, respectively, as a product of target orientation about the Z-axis of the target marker.

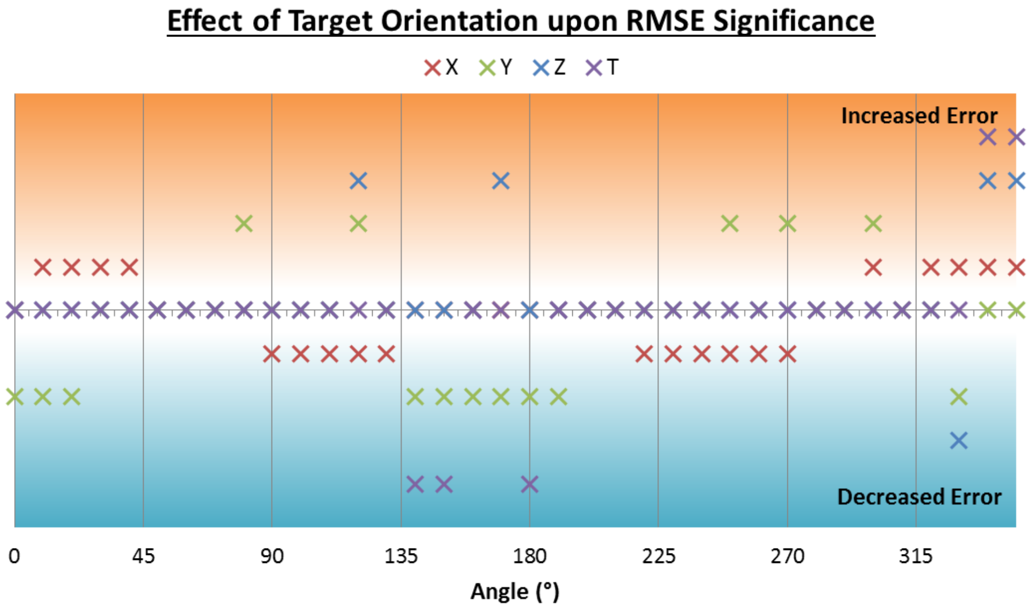


Figure 8.30: Effect of rotation of the target about the Z-axis relative to the camera upon the RMSE of the third generation LifeCam system

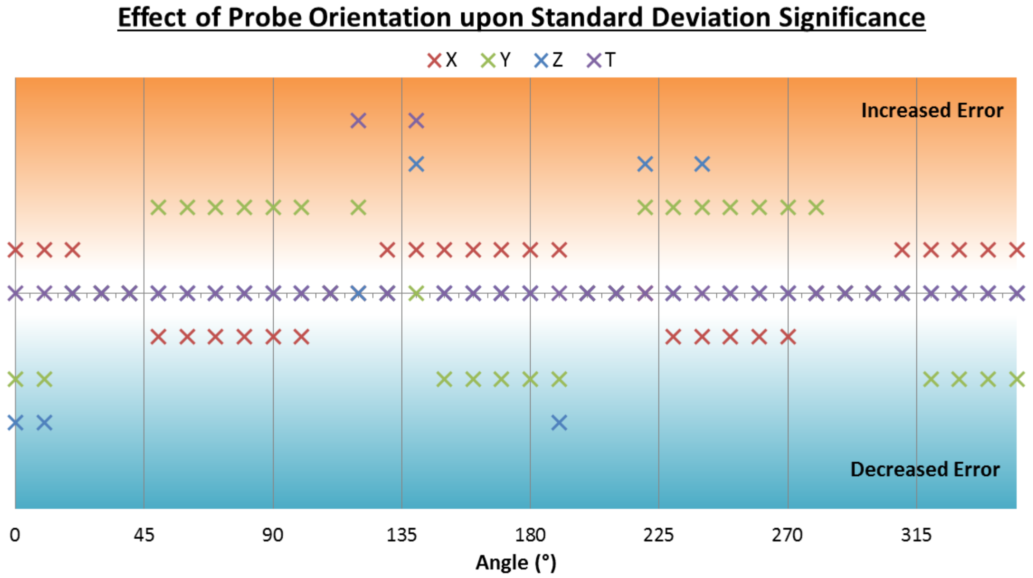


Figure 8.31: Effect of rotation of the target about the Z-axis relative to the camera upon the standard deviation of the third generation LifeCam system

Neither Figure 8.30 or Figure 8.31 present a notable pattern of significance resulting from rotation, this is in stark contrast to the results of the second generation system shown by Figures Figure 7.50 and Figure 7.51.

Figure 8.32 below illustrates an example error distribution for the research camera system.

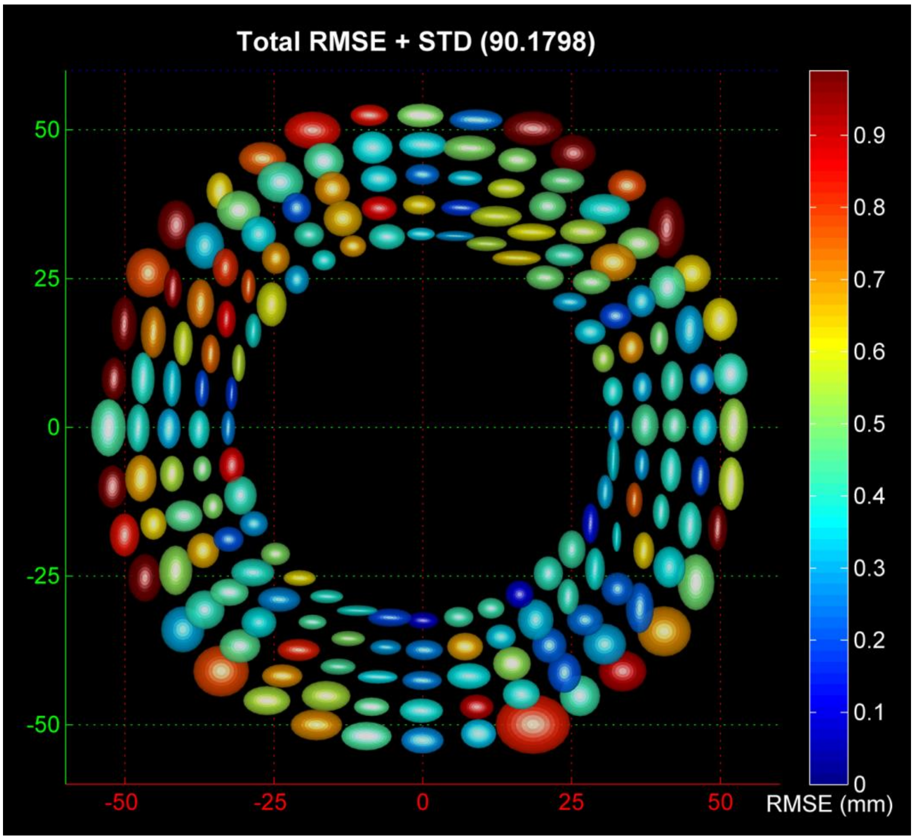


Figure 8.32: Example of third generation research camera system

A mean RMSE of 0.55±0.04 mm and standard deviation of 0.10±0.01 mm were obtained by fixed probe measurements. The RMSE was found to be insignificantly larger than the LifeCam system by a factor of 1.02 (P = 0.29). The standard deviation was larger by a factor of 1.25, although again this did not reach significance (P = 0.05).

The effect of marker-probe separation is shown by Figure 8.33 below.

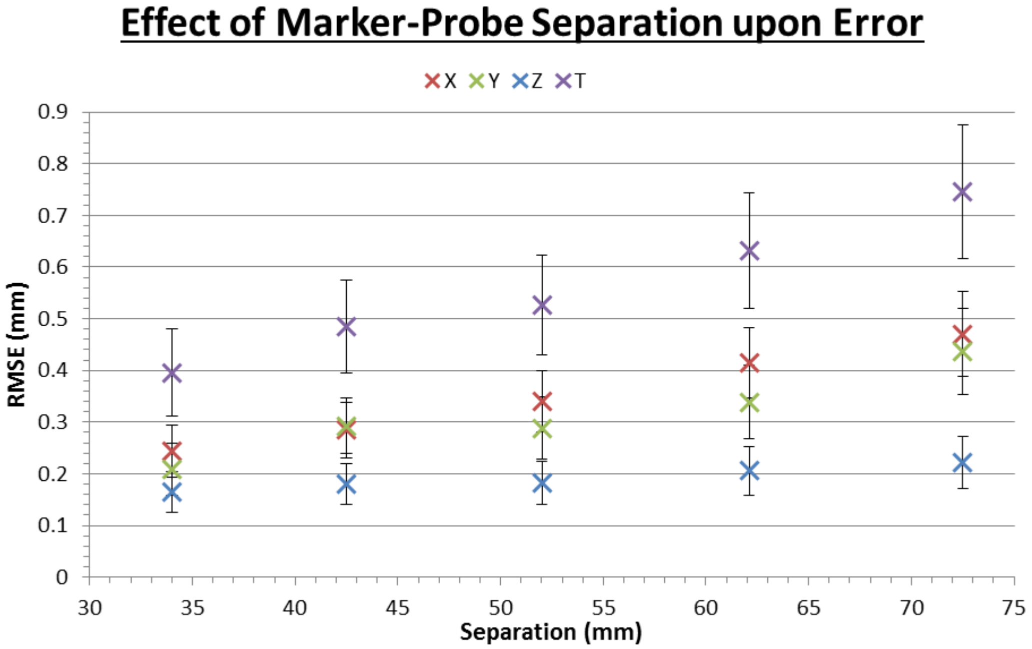


Figure 8.33: Effect of marker-probe separation upon third generation research camera system

The significance of the error increases, shown by Figure 8.33 above, presented less consistency than those of the LifeCam system. The X component was found to be significant for the second and third increment (P = 0.03 and P = 0.02). The Y component reached significance for the first, third, and fourth increment (P < 0.01, P = 0.03, and P < 0.01). The Z component did not reach significance at any increment (All P > 0.07). Finally, the total component reached significance for the first, third, and fourth increment (All P < 0.01).

The effect of target rotation about the Z-axis upon RMSE and standard deviation is shown by Figures Figure 8.34 and Figure 8.35, respectively, below.

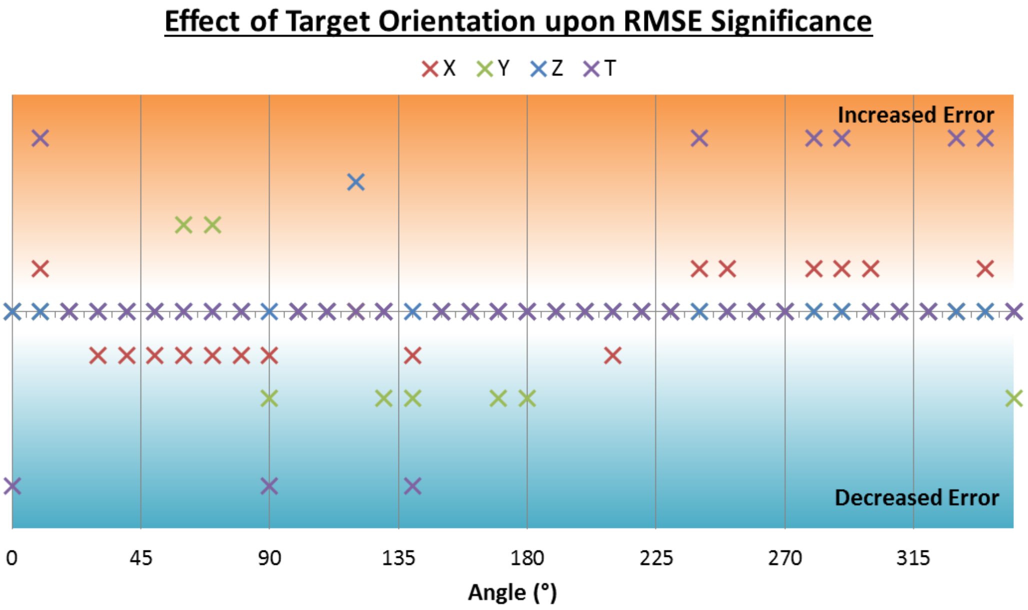


Figure 8.34: Effect of rotation of the target about the Z-axis relative to the camera upon the RMSE of the third generation research camera system

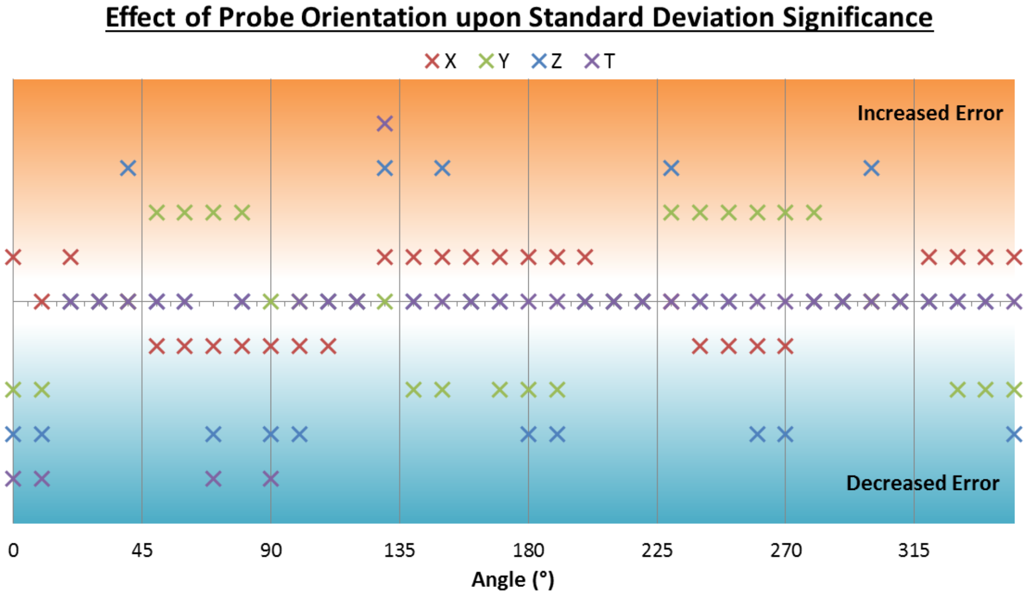


Figure 8.35: Effect of rotation of the target about the Z-axis relative to the camera upon the standard deviation of the third generation research camera system

The effect of probe motion upon the mean RMSE and standard deviation of the LifeCam system is illustrated by Figures Figure 8.36 and Figure 8.37 below respectively.

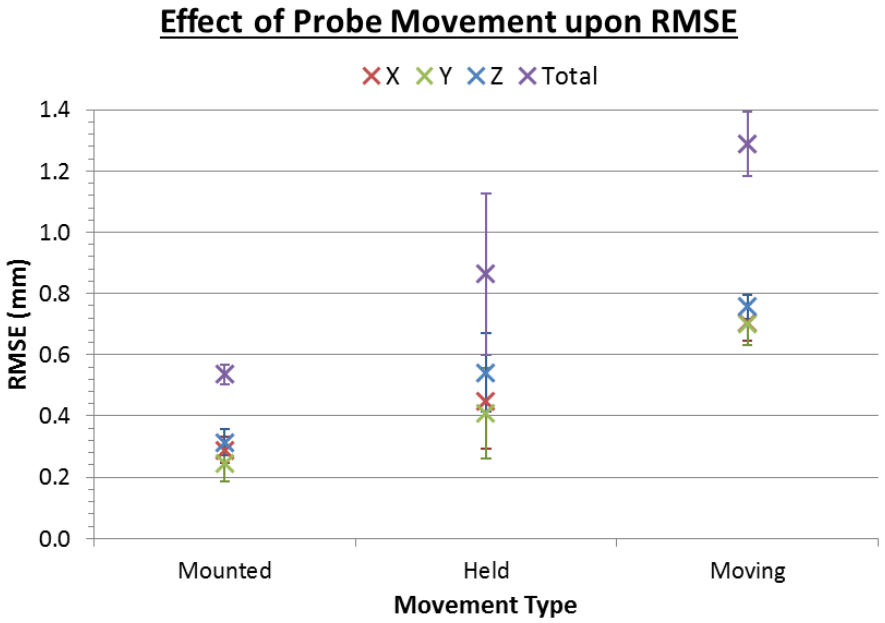


Figure 8.36: Effect of probe movement upon RMSE of third generation LifeCam system

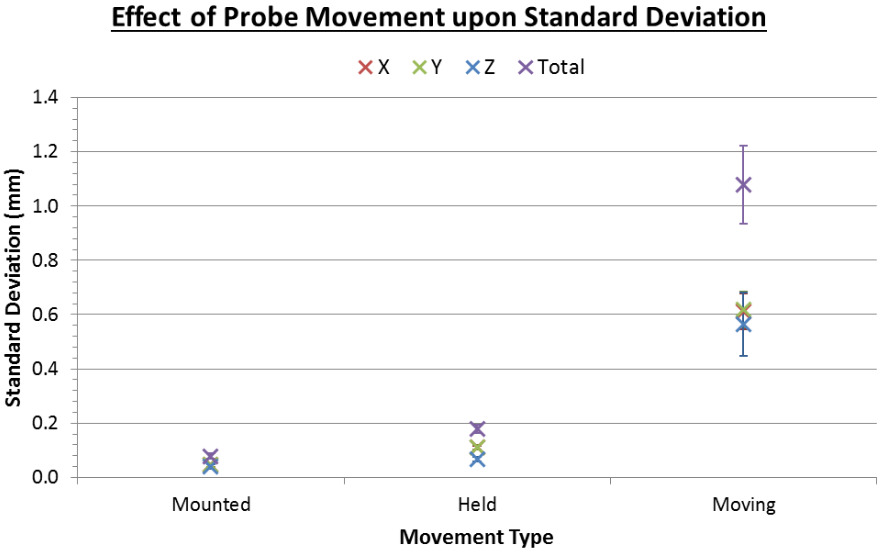


Figure 8.37: Effect of probe movement upon standard deviation of third generation LifeCam system

The error of the LifeCam system is shown by Figures Figure 8.36 and Figure 8.37 to increase with increased movement, as expected. The transition from mounted to held movement produced a significant RMSE increase only in the Z component (P = 0.04, All others P > 0.1). Held to moving resulted in a significant increase to all individual components (P = 0.04, P = 0.03, and P = 0.04). However, the total component did not reach significance (P = 0.05). All components significantly increased between mounted and moving (All P < 0.01). Furthermore, all transitions produced significant increases in all components of the standard deviation (All P < 0.01).

Figures Figure 8.38 and Figure 8.39 repeat the above analysis for the research camera system.

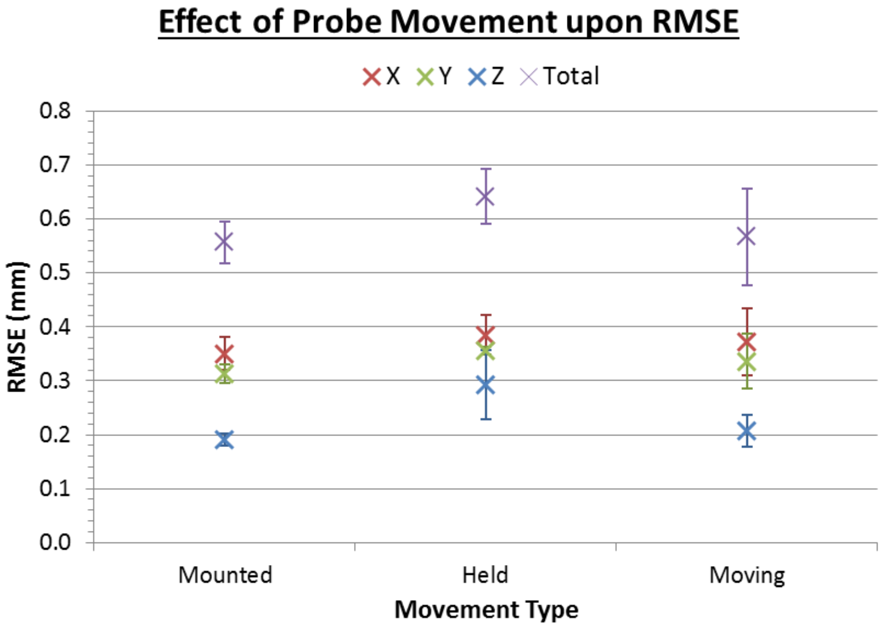


Figure 8.38: Effect of probe movement upon RMSE of third generation research camera system

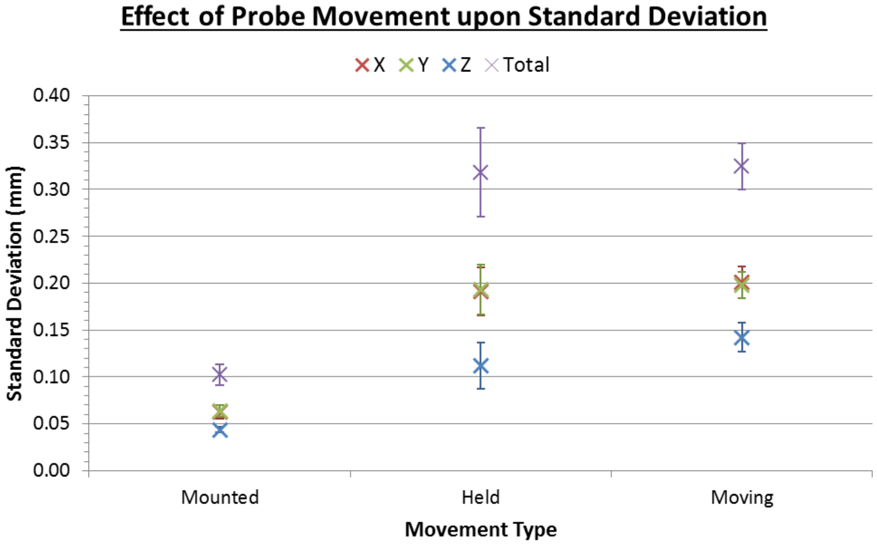


Figure 8.39: Effect of probe movement upon standard deviation of third generation research camera system

The research camera system does not present the increase of RMSE with movement, in Figure 8.38, seen by the other system. All components presented an increase between mounted and held methods, with only the Y and Z components reaching significance (P = 0.02 and P = 0.04). The held to moving transition produced a decrease of all components, with the total RMSE reducing by a factor of 0.88. However, no components reached significance (All P > 0.08). There was no significant difference between the components of the mounted and moving experiments (All P > 0.25).

The standard deviation, as shown by Figure 8.39, presented a significant increase of all components between mounted and held and mounted and moving (All P < 0.01, excluding Z component of mounted to held P = 0.03). No significant difference was found between held and moving standard deviations (All P > 0.1).

The effect of probe movement upon the second generation and two third generation systems is compared by Figures Figure 8.40 and Figure 8.41 below.

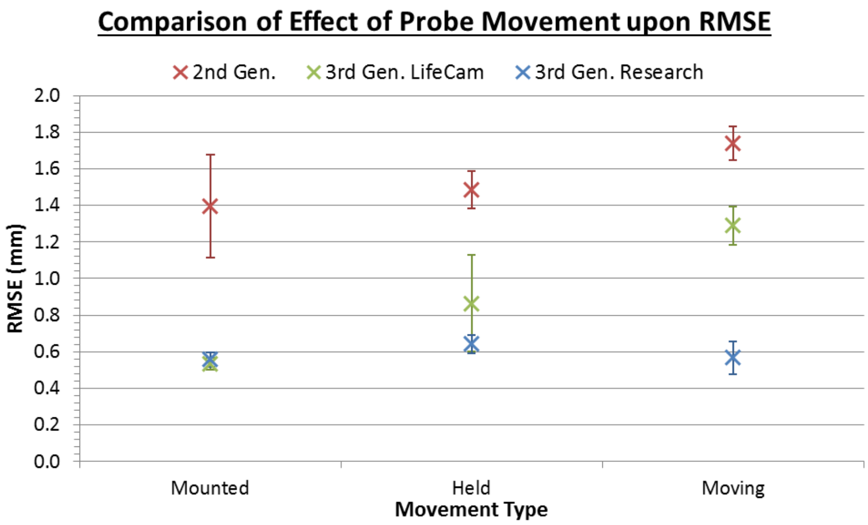


Figure 8.40: Comparison of the effect of probe movement upon RMSE of the second generation and two third generation systems

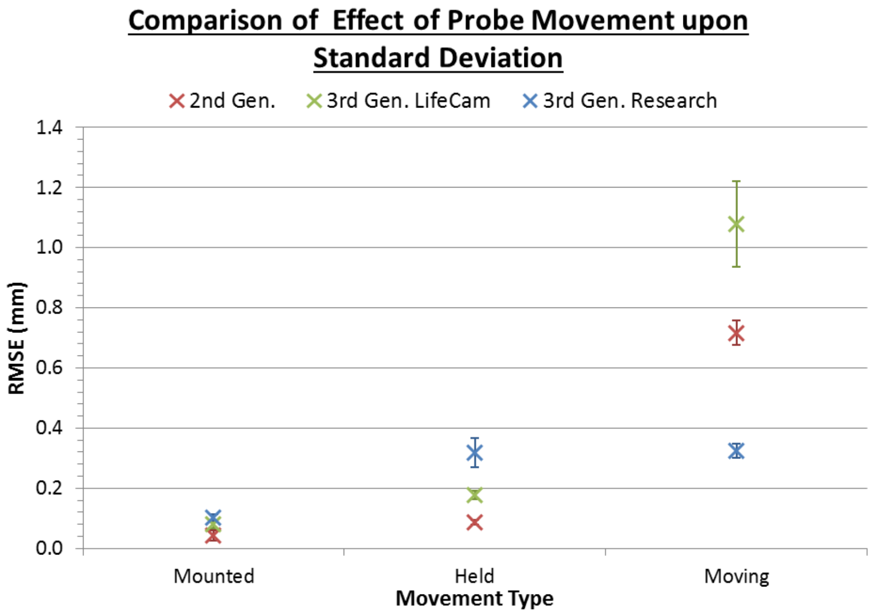


Figure 8.41: Comparison of the effect of probe movement upon standard deviation of the second generation and two third generation systems

As shown by Figure 8.40, the second generation system produced significantly higher RMSE than both third generation systems across all types of motion (All P < 0.02). Conversely, during mounted and held measurements the second generation system produced significantly lower standard deviations (All P < 0.03). During movement the second generation produced significantly lower standard deviation than the LifeCam system (P = 0.01), but significantly higher than the research camera system (P < 0.01).

The LifeCam system was shown to produce higher RMSE than the research camera system when not mounted, with moving measurements producing a significant decrease factor of 0.44 (P < 0.01). The research camera system also provided a significant reduction in standard deviation of 0.30 while moving, over the LifeCam system (P < 0.01). However, the LifeCam system produced the lower variance while held (P = 0.01).

The effect of camera-marker separation is illustrated by Figure 8.42 below.

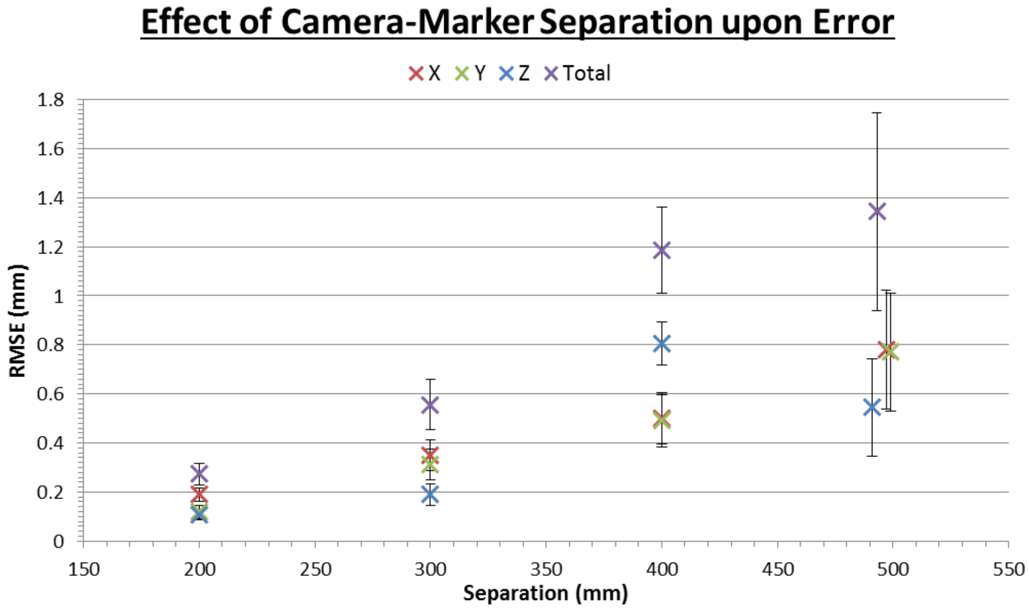


Figure 8.42: Effect of camera-marker separation upon the accuracy of the third generation research camera system

Both RMSE and standard deviation were seen to increase with separation as expected. The standard deviation was found to increase significantly for each increment across all components (All P < 0.01). Total RMSE was found to increase only for the initial two increments (Both P < 0.01). The final increment produced significantly larger X and Y components (P = 0.03 and P = 0.02), but a significantly reduced Z component (P = 0.01).

Figure 8.43 demonstrates the effect of target marker size upon error for the research camera system.

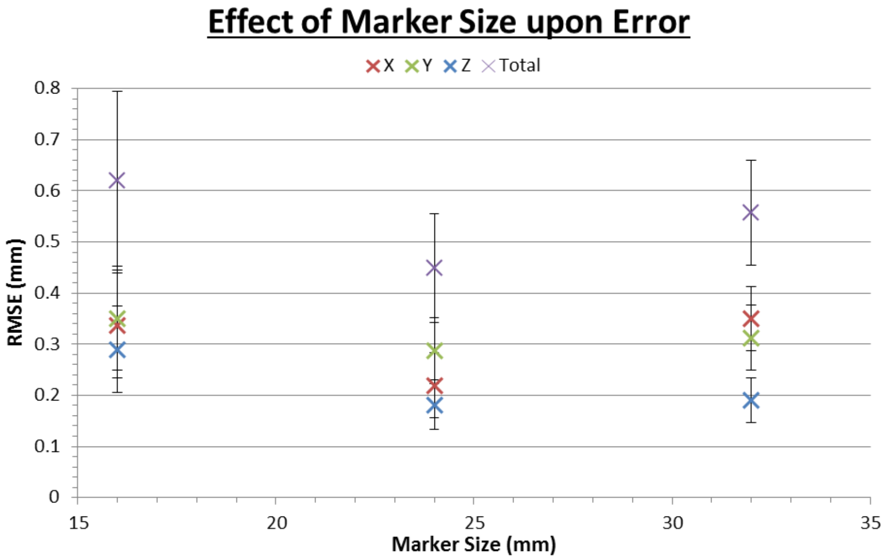


Figure 8.43: Effect of target marker size upon error of the third generation research camera system

Figure 8.43 shows the 24 mm marker produced the lowest total error at 0.45±0.00 mm. This was shown to be significantly smaller than the largest 32 mm marker by a factor of 1.24 (P = 0.03). The significance was attributed to the X component which was larger for the 32 mm marker by a factor of 1.60 (P < 0.01).

The smallest 16 mm marker was found to produce significantly larger standard deviations, for all components, than both the medium and larger marker (All P < 0.01).

The alternative probe marker design was found to produce reduced RMSE and standard deviation across all components. The alternative probe results were on average 0.84 and 0.95 times those of the standard marker, for RMSE and standard deviation respectively. However, only the Z component of standard deviation reached significance (P = 0.02).

8.3.3.3 Discussion

The second generation LifeCam system produced a mean error of 0.54±0.08 mm, while the research camera system produced an error of 0.55±0.10 mm. These both readily surpass the second generation system and importantly the 1.0 mm accuracy target of the system defined in Chapter 4. These results are again compared to the observations of Wiles et al. for the Polaris system. With an 80 mm probe length the tip accuracy of the Polaris system is estimated to be 0.77±0.43 mm, approximately

40% less accuracte and 350% less precise than the third generation systems. Again, the Polaris system provides this accuracy over a greater volume. However, much of this 1.6 m^3 volume may go unused during knee arthroplasty.

Badiali *et al.* report upon a similar visible spectrum stereoscopic augmented navigation system, wearable augmented reality for medicine (WARM), for use in maxillary repositioning (Badiali et al. 2014). The WARM system produced a mean error of 1.70±0.51 mm during maxillary repositioning. This is not directly comparable to the results of the third generation system, as this is not a probing error and includes both user and probable ground truth error. However, comparison still holds significance due to the similarities of the systems.

Both systems produced the expected increase in both RMSE and standard deviation at increased marker-probe separations. As discussed previously, in Section 7.3.3.3, this is most likely the result of pose orientation errors being amplified by the increased separation. It is noted that unlike the second generation system, there was no sudden increase in error with the final level. The increased camera-marker separation afforded by the alternative design allowed all markers to remain visible at all levels without adjusting the camera angles, which the increase of error of the second generation system was attributed to.

The accuracy was improved, as discussed in Section 8.3.2.3, as a result of the additional 3D information. However, it is noted that in spite of this the standard deviation is increased. This is accounted for by two compounding factors. Initially, due to the increased camera-marker separation under which the third generation systems operate, the marker corners are detected to a lower spatial resolution. As such marker corner detection errors result in larger pose estimation errors. In addition to this the third generation system utilises two markers. Therefore, in comparison to the second generation system, errors are intrinsically doubled. As such the small inter-frame corner detection discrepancies result in substantially larger standard deviations, despite the improved accuracy.

It is noted that the Z component of the research camera system remained relatively consistent in contrast to the LifeCam system. This is potentially indicative of the increased viewing angle required by the research camera system to accommodate the larger camera-marker separation used. This increased angle resulted in the reported Z component, within the coordinate system of the target, aligning to a lesser degree with the Z-axis of the camera system. The Z-axis of stereoscopic systems incurs greater error than the remaining two axes due to the disparity error as shown by Equation 8.14. Therefore, this increased angle would reduce the effect of Z errors in the coordinate system of the camera upon Z errors in the coordinate system of the marker.

The effect of the increased error along the optical axis of the camera system is also present in the target orientation data present by Figures Figure 8.30, Figure 8.31, Figure 8.34, and Figure 8.35. The probe was incident upon the target at a 90° lag to the camera, about the Z-axis of the marker. Therefore, while probing at 0° the camera was imaging the target at 90°.

While probing between 315 and 45° the X-axis of the target became well aligned with the Z-axis of the camera system. Therefore, an increase in the X component of error would be expected. A significant increase in the X component RMSE may be observed in this region for the LifeCam system from Figure 8.30. The increase is not as notable from the research camera data set.

Conversely, probing between 90 and 135° and 225 and 270° the X-axis becomes perpendicular to the optical axis of the camera, and as such a decrease in the X error component would be expected. Again, this may be well observed for the LifeCam system from Figure 8.30, where significantly smaller errors were observed. An inverted relation is observed for the Y component, as expected. It is probable that tip calibration, which would also be affected by the increased optical axis error, may mask the RMSE error pattern to some extent. However, this is not expected to affect the standard deviation of the system, and as such the effect of angle is expected to be more prevalent within the standard deviation measurements. As may be seen from Figures Figure 8.31 and Figure 8.35 the described patterns are indeed readily visible across the full range of angles for both the X and Y components.

Figure 8.40 compared the RMSE of the three latest systems. Both of the third generation systems provided nearly a twofold RMSE improvement over the second generation system. As previously discussed in Section 8.3.2.3 above, this improvement is the product of the direct extraction of 3D information from the imaging scene, reducing the reliance of pose estimation through projection.

During probe movement Figure 8.41 illustrated a significant difference between the standard deviation of the three systems. The LifeCam system produced the highest standard deviation, followed by the second generation and research camera systems respectively. This order is indicative of the deficit of commercial web cameras for use in tracking systems. The standard deviation of both LifeCam systems were increased under motion due to their use of rolling shutters, as previously discussed in Section 7.3.3.3. This was compounded in the third generation LifeCam system by a lack of camera synchronisation. This allowed a considerable temporal difference between the stereo image pairs. Therefore, under motion, markers may move between the capture of the two images, resulting in a distortion of pose estimation, and increased standard deviation.

Conversely, the third generation research camera system utilised global shutters and software triggered inter-camera synchronisation. Additionally, the comparatively short exposure time of 10 ms helped to reduce the blurring of the image under motion.

Increased camera-marker separation, as shown by Figure 8.42, negatively affected both the RMSE and standard deviation of the system. This was the expected behaviour, resulting from reduced spatial resolution and Equation 8.14, as discussed by Section 8.3.2.3. Figure 8.44 below illustrates how small the 32 mm markers appeared in the image frame at 500 mm separation. It also shows the target prompt such that the normal of the marker was less perpendicular to the optical axis of the camera. This was required to ensure consistent marker detection. It is noted that as the marker was orientated increasingly parallel to the optical axes the Z error component became proportionally larger. Measurements at 300 mm were least parallel and were found to produce the smallest proportional Z component (0.58, $P < 0.05$). The 400 mm measurements were most parallel and produced the largest Z component (1.68, $P < 0.01$). This supports the previous discussion of increased error along the optical axis of the camera system.

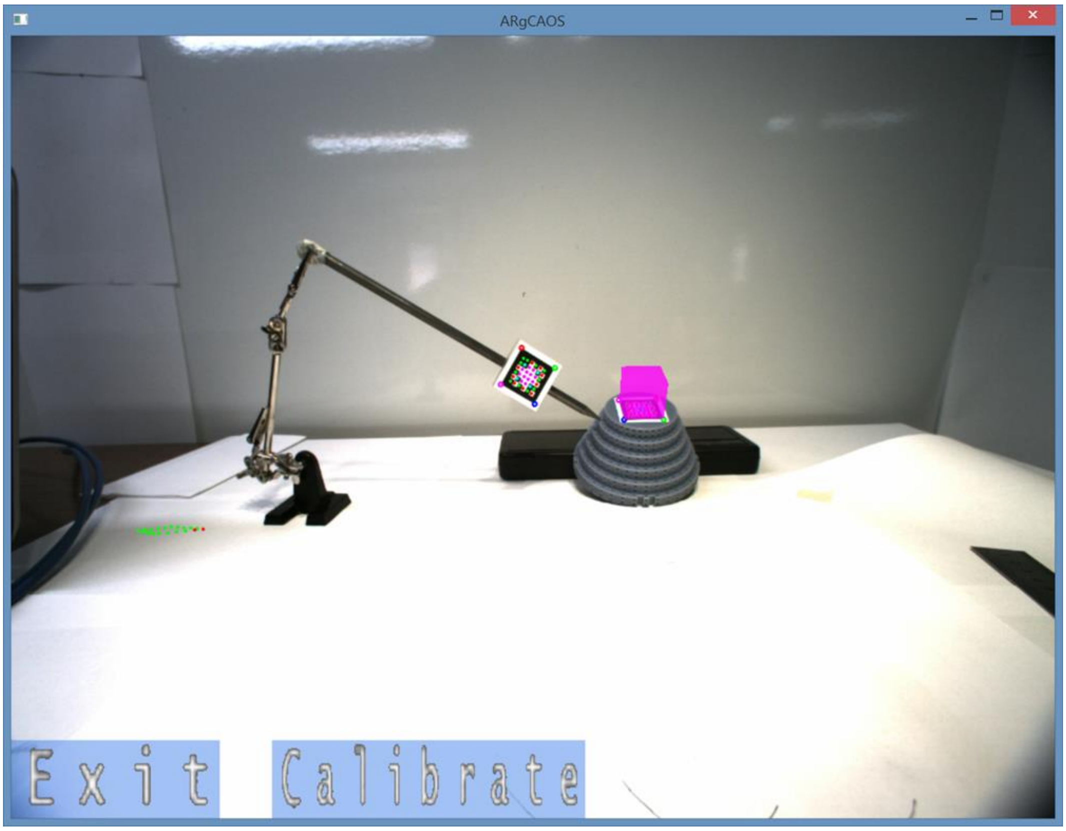


Figure 8.44: Third generation research camera probing experiment performed at 500 mm

Based upon photographic measurement of hand-to-eye separation during assisted knee surgery, a working distance of approximately 350 mm was estimated. Linear regression of Figure 8.42 ($R^2 = 0.95$) indicates an error of 0.83±0.13 mm at this separation. Therefore, the research camera system would still provide the required 1.0 mm accuracy at a practical separation applicable to the desired application.

The effect of target marker size did not produce the error distribution expected, with the mid-size marker producing the lowest mean error, as shown by Figure 8.43. At a constant distance, reduced marker size resulted in fewer edge pixels and therefore the accuracy of corner detection would be expected to decrease. However, this is a statistical effect only, and as such fewer edge pixels of good quality may still result in equally, or more accurate, corner detection. As the marker size was decreased further the statistics became weighted more strongly and an increase in error was observed.

Larger markers were also likely to be more susceptible to engrave edge warping during manufacture. Figure 8.45 shows an example engraved marker edge at 5x magnification.

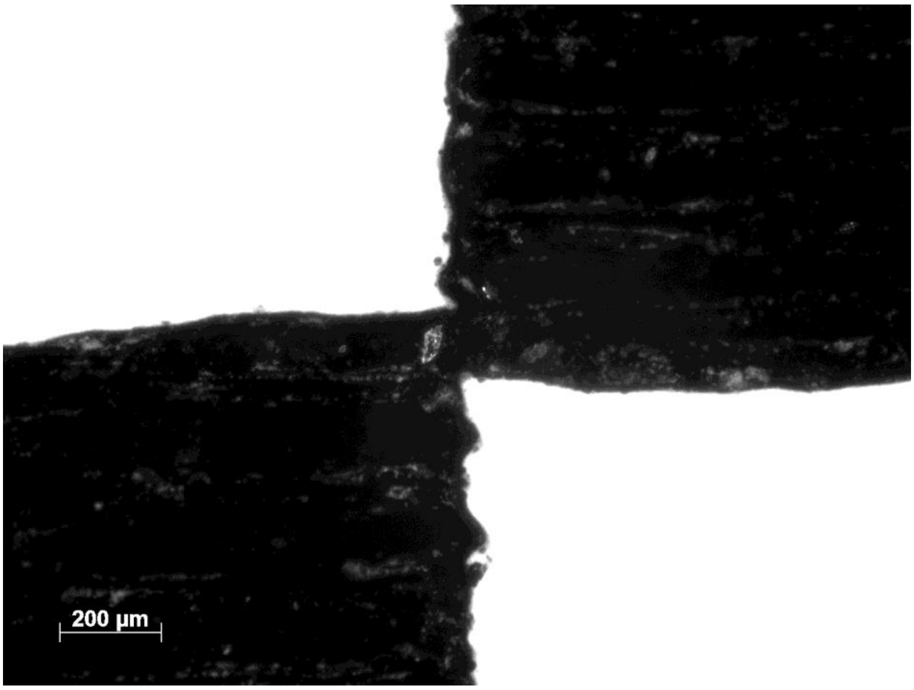


Figure 8.45: Example marker edge imaged a 5x magnification

The horizontal black-white interfaces are expected to be linear and parallel. However, visible warping of the edges is observable. This warping may occur across an entire marker edge. For larger markers this may be sufficiently severe to be detected by the camera system and thus induce minor errors during the linear interpretation of edges.

This warping may also have contributed to the reduced errors observed for the alternative marker. Warping was most prevalent in edges along the X-axis of the laser cutting. The edges of the alternative marker did not align with the axes of the laser cutter and therefore macroscopic edge warping did not occur, as shown by Figure 8.46.

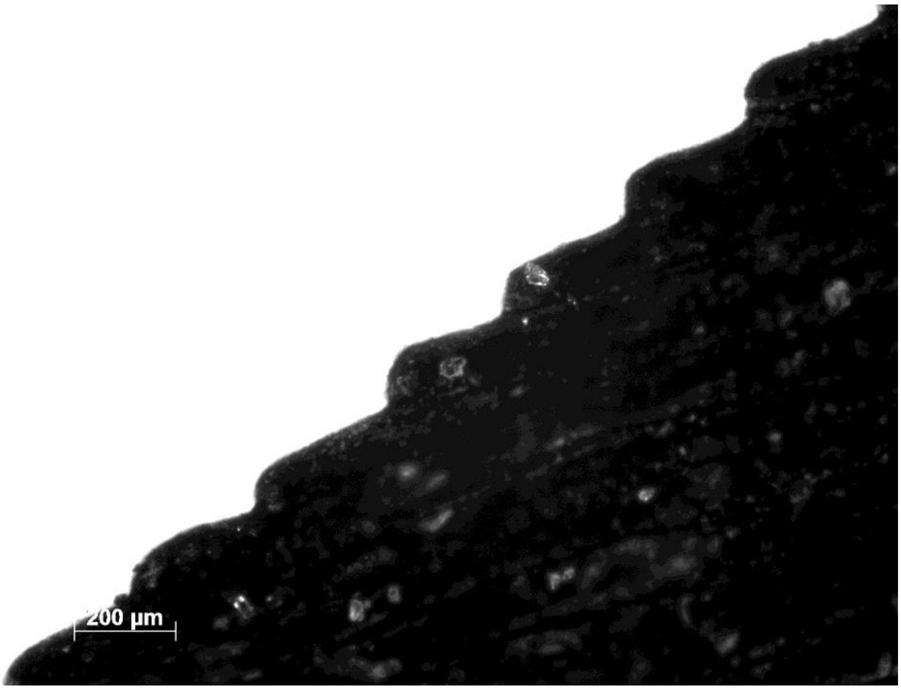


Figure 8.46: Alternative marker edge imaged at 5x magnification

However in the presented case it is likely the increased marker edge length was the predominant factor for improved accuracy.

8.3.4 Resection Analysis

As with the second generation system resection analysis was performed to characterise the performance of both third generation systems.

8.3.4.1 Materials and Methods

Resection analysis was performed using the same equipment as Section 7.3.4.1. However, the camera system was mounted at a nominal distance of 250 and 300 mm for the LifeCam and research camera system respectively. A 32 mm square marker was affixed to the resection burr, with a centre-tip separation of 80 mm. The burr was calibrated using the 90% central band of 200 samples collected during moderate motion with the burr calibration target shown by Figure 7.60.

Two sets of resections are presented for the LifeCam system. As discussed in Appendix A2.5, the initial implementation of the resection code did not enact control until the burr centre was within the volume. The second data set presents the rectification of this omission.

The research camera system was modified to allow multiple tools, identified by distinct markers, to be tracked. To allow the resection of the previously inaccessible peg holes an additional 3 mm spherical burr was calibrated. The main resection was performed with the 6 mm burr. The burr and sheath were then replaced by the 3 mm burr and appropriate sheath with an alternative marker. The peg holes were then resected. Additional resection of the smallest trench was also performed.

8.3.4.2 Results

Figure 8.47 presents the absolute errors of the third generation LifeCam system prior to correcting the resection implementation.

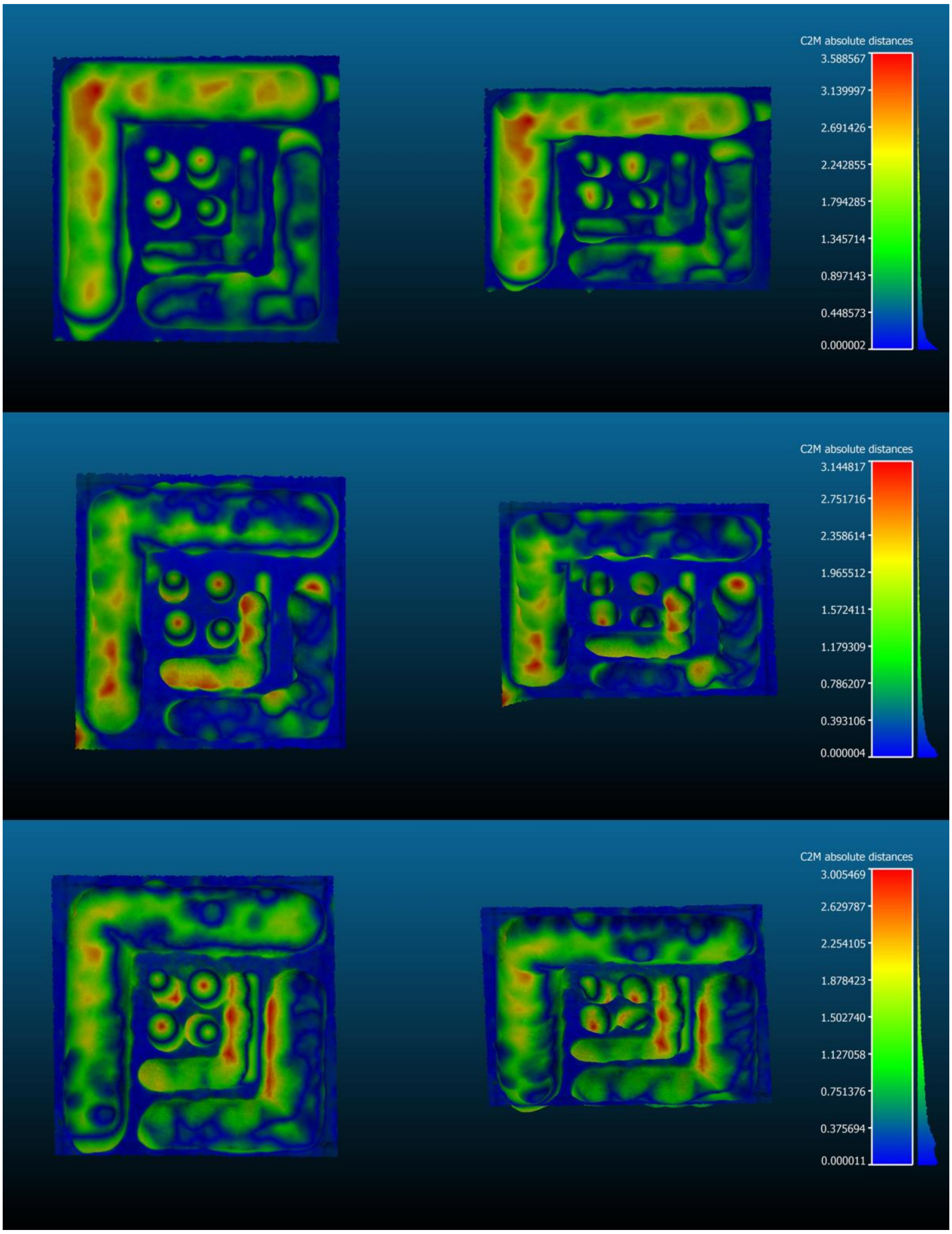


Figure 8.47: Absolute resection errors of third generation LifeCam system prior to resection implementation correction

As indicated by Figure 8.47, several large errors were produced by the original stereoscopic implementation. These are most evident where shallow resection was required, such as the edges of the trenches. A mean absolute resection error of 0.73±0.08 mm was produced, with a mean non-peg hole maximum error of 3.24±0.25 mm.

Figures Figure 8.48 and Figure 8.49 illustrate the absolute and signed errors, respectively, of the LifeCam system after correction of the resection implementation.

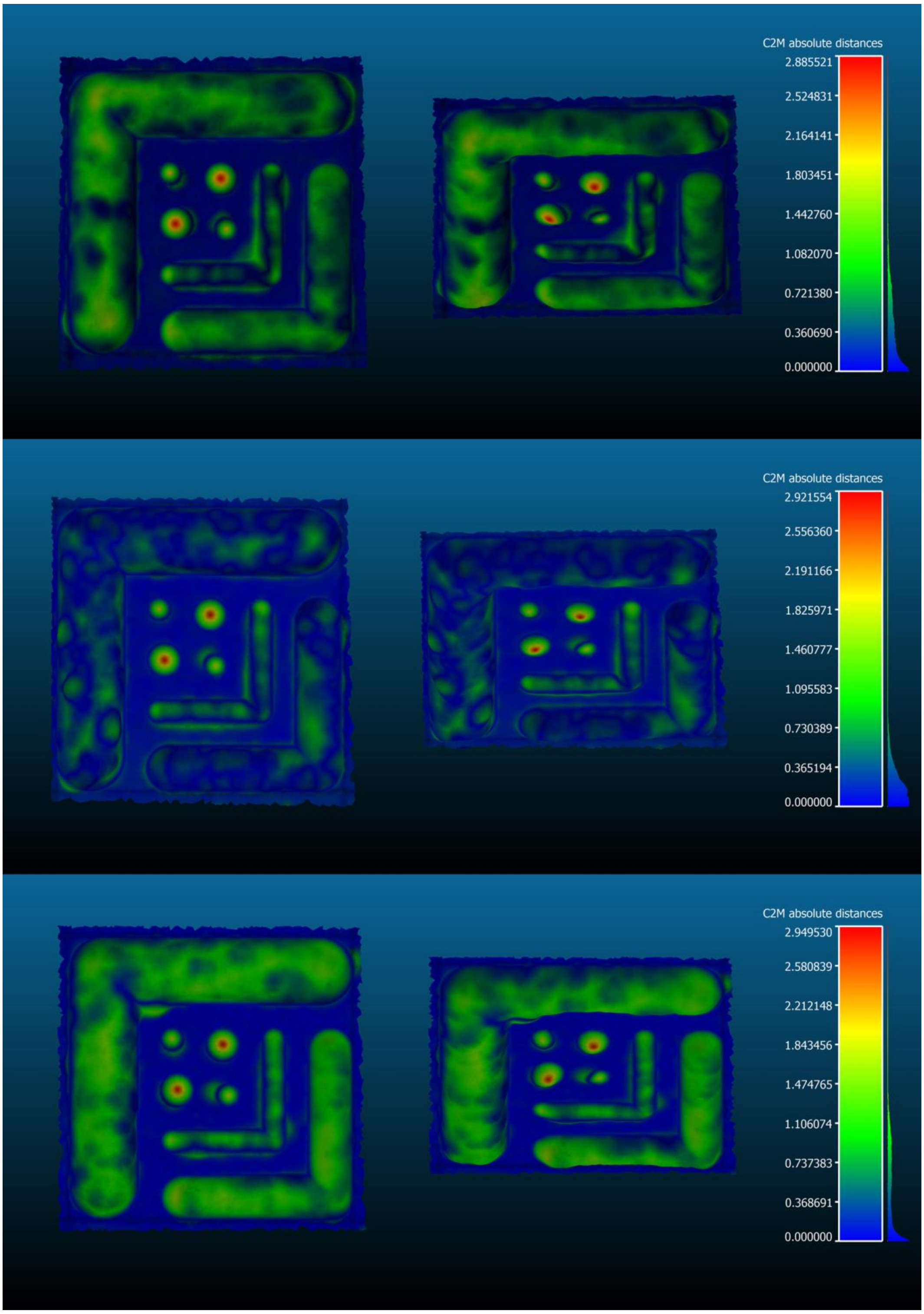


Figure 8.48: Absolute resection error of third generation LifeCam system after correction of resection implementation

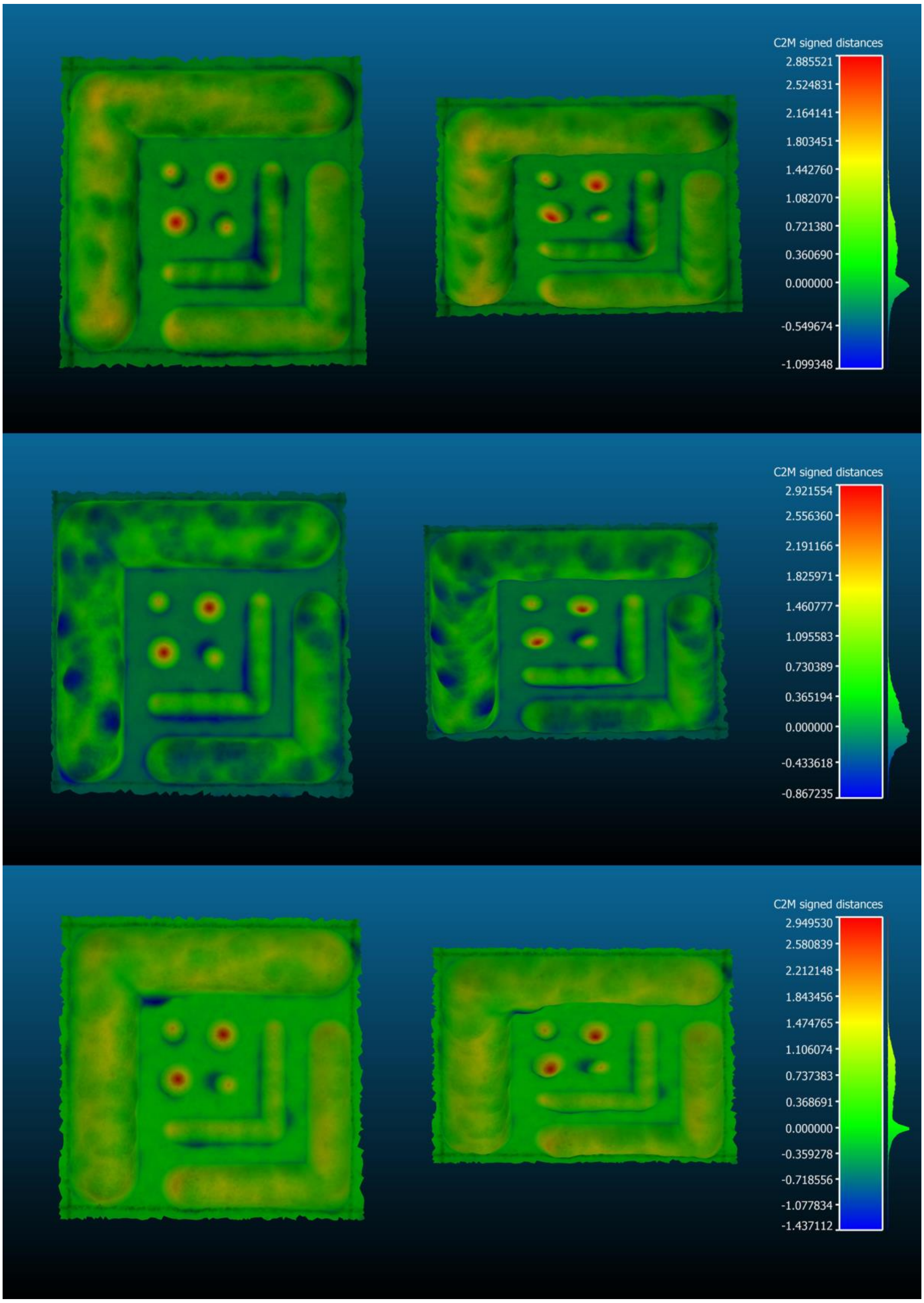


Figure 8.49: Signed resection error of third generation LifeCam system after correction of resection implementation

Figure 8.48 illustrates the improvement of the corrected resection implementation. Far fewer large errors are seen and the trench edges are straight and consistent. A mean absolute error of 0.37±0.11 mm was obtained, significantly better than the original implementation by a factor of 1.96 ($P = 0.01$). This was found to be insignificantly larger than the second generation system by a factor of 1.03 ($P = 0.44$). The maximum non peg whole error was calculated as 1.37±0.18 mm, a significant improvement of a factor of 2.38 ($P < 0.01$). Figure 8.49 indicated a general trend of under-resection producing a mean signed error of 0.25±0.15 mm.

Finally, Figures Figure 8.50 and Figure 8.51 show the absolute and signed errors of the third generation research camera system. The first resection shown was performed with only the 6 mm burr, and as such the peg holes were not fully resected. However, the geometry capture method was unable to image the peg holes and as such the results were not affected.

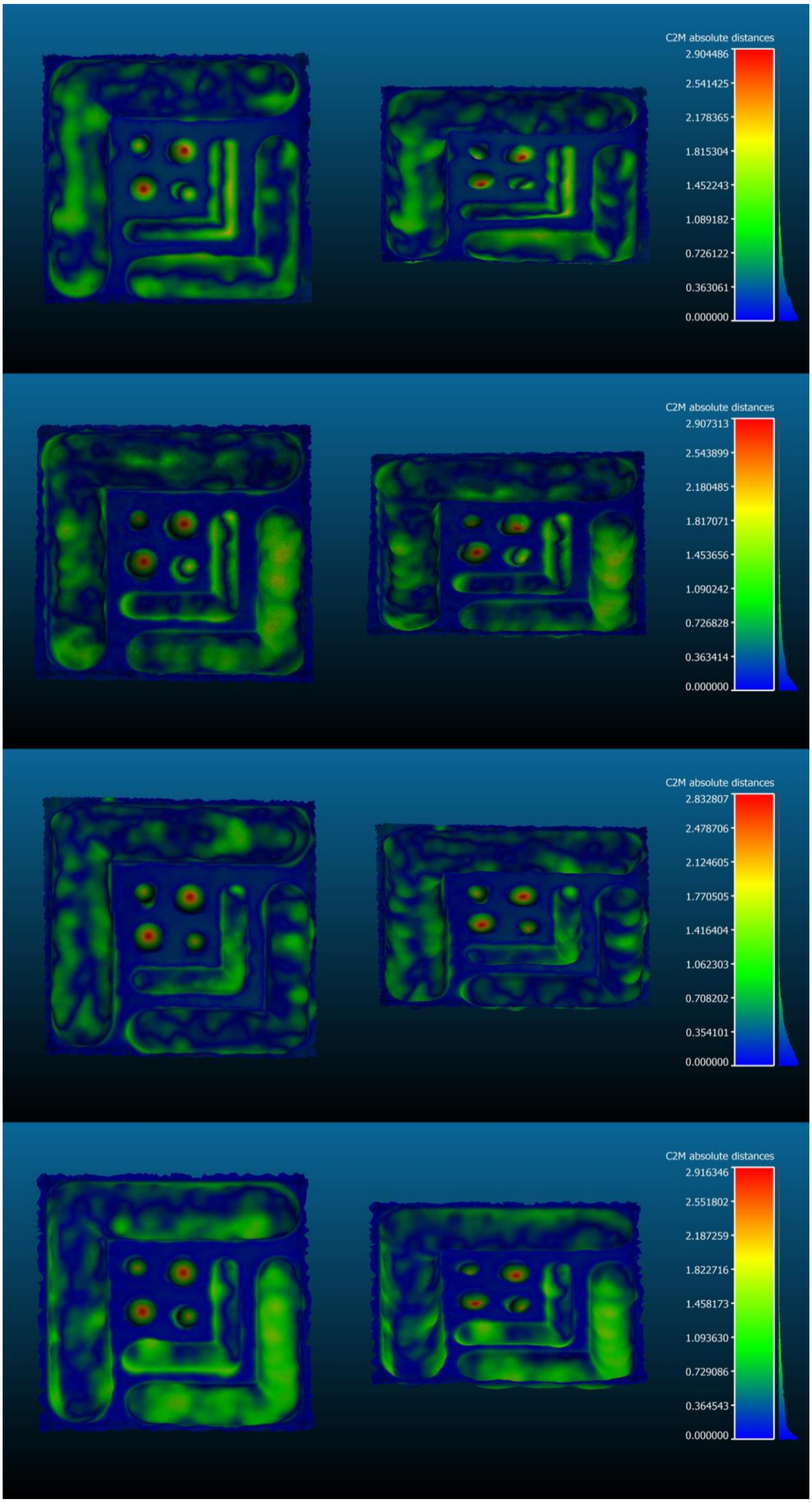


Figure 8.50: Absolute resection error of third generation research camera system

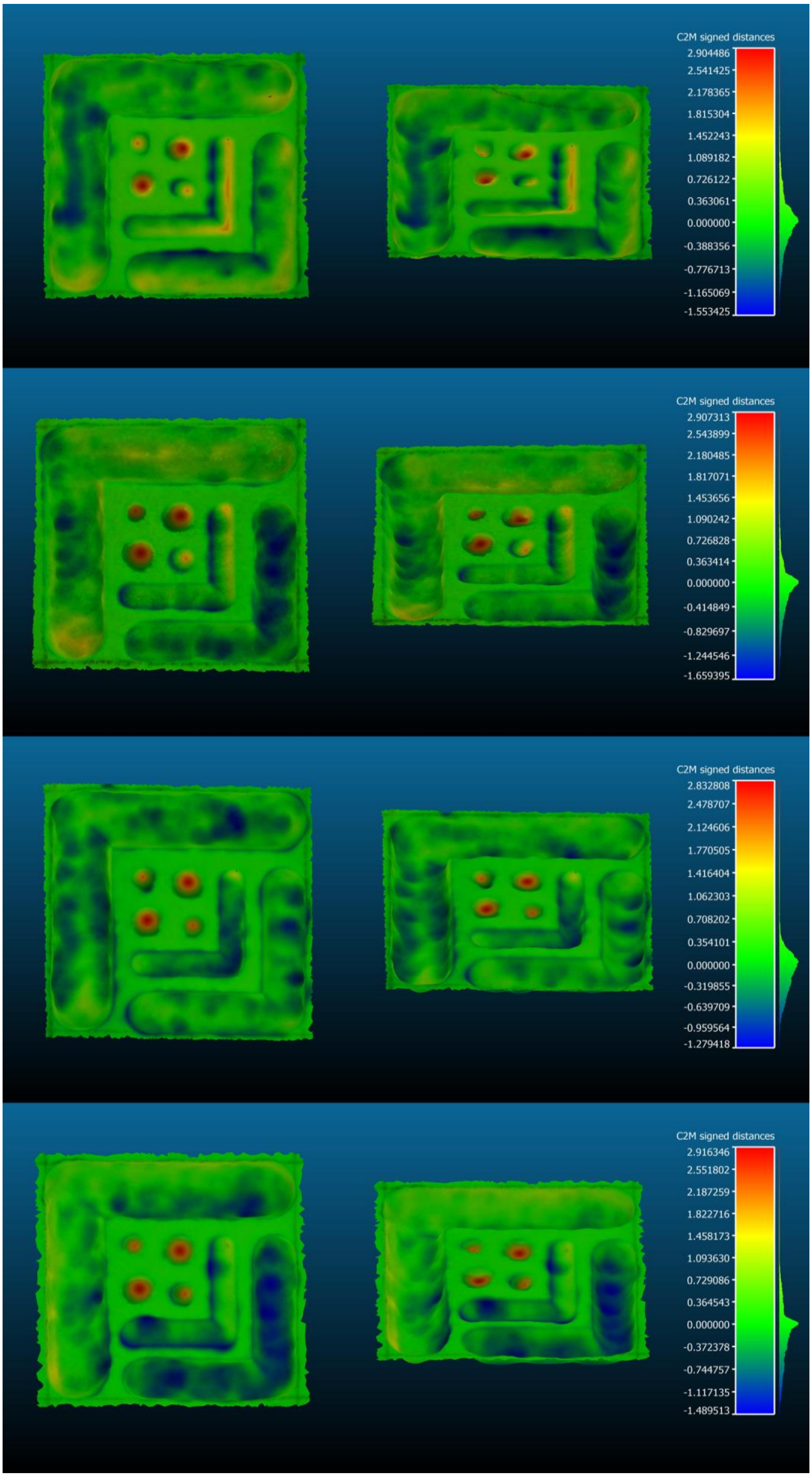


Figure 8.51: Signed resection error of third generation research camera system

The research camera system produced a mean absolute error of 0.34±0.04 mm. This was a non-significant improvement of a factor of 1.08 over the LifeCam system (P = 0.35). The maximum non-peg hole error was found to be 1.47±0.18 mm, non-significantly worse than the LifeCam system by a factor of 1.07 (P = 0.29).

Figure 8.51 indicated a general over-resection of -0.10±0.07 mm. The initial resection performed with only a 6 mm burr produced a general under-resection of 0.01 mm. This may be seen from Figure 8.51 to stem predominately from the under resection of the smallest trench, which produced a peak error of 1.82 mm.

The initial implementation of the 3 mm burr utilised a search radius of 0.25 mm, as used by the 6 mm burr. During resection it was only possible to resect the 6 mm peg holes. These both presented under-resection with a mean absolute depth error of 1.50±0.82 mm. The diameters, measured at a depth of approximately 8 mm, were also under-resected with a mean absolute error of 1.67±0.11 mm.

The second implementation reduced the search radius to 0.20 mm. One 4 mm peg hole was successfully resected. However, the second hole could not be progressed past a few millimetres. The 6 mm holes were both under-resected with a mean absolute depth and diameter error of 0.63±0.01 and 0.25±0.05 mm, respectively. The 4 mm hole showed errors of 0.13±0.06 and 0.06±0.01 mm for depth and diameter respectively.

The final two resections were performed using a 3 mm burr search radius of 0.1 mm. All peg holes were successfully resected. Both 4 and 6 mm holes were over-resected with regard to depth. The diameter of 4 mm holes was over-resected, while the 6 mm holes were typically under-resected. The mean absolute depth and diameter errors were calculated to be 0.37±0.29 and 0.09±0.05 mm and 1.12±0.04 and 0.08±0.07 mm for the 4 and 6 mm peg holes respectively. The depth error difference between the 4 and 6 mm peg holes was found to be significant (P = 0.01). However, the diameter difference was not significant (P = 0.39).

8.3.4.3 Discussion

The initial implementation of controlled resection failed to assert control over the upper 3 mm of the volume surface. This resulted in substantial over-resection, as illustrated by Figure 8.47. The correction of this saw a nearly twofold improvement in mean absolute error. As seen by Figure 8.49 upper surface over-resection was greatly reduced.

Despite the significant improvement of the third generation system over the second generation system during probing, reported in Section 8.3.3 above, the mean absolute resection error was found to be greater for the third generation system by a factor of 1.04, although this did not reach significance (P = 0.44). This is possibly due to the dynamic nature of probe motion during resection, both from operation movement and tool vibration. This movement would lead to an increased error due to the unsynchronised nature of the stereoscopic cameras. The research cameras did present a slight improvement, over the second generation system, of 1.08. However, this also did not reach significance (P = 0.36). As previously demonstrated the research camera system presented good resistance to movement induced errors.

Several additional factors may partially account for the lack of error reduction observed during resection in comparison to the probing analysis. The requirement of two markers, both of which were proximal to the resection burr, made the system more susceptible to impared marker visibility due to bone dust. Due to the dry nature of the saw bone material, dust built up upon the markers and periodically required removal. This dust is expected to be detrimental to accurate marker detection, as it blurs the distinction between black and white regions of the markers. In analogue to image thresholding discussed in Section 6.4.6.1, the fine dust may either result in marker drop, or more problematically cause subtle edge occlusion that results in erroneous corner detection and pose estimation.

It was observed that the third generation research camera system was the first system to show a general trend of over-resection. Initial experiments with the system indicated a larger under-resection than previous systems. It was concluded that this resulted from the reduced latency of the system afforded by the rapid image capture of the research cameras. To compensate for this reduced latency the search radius of the research camera system was reduced to 0.25 mm. This allowed the burr to resect closer to non-resection regions before the system deactivated. It is suggested that optimised tuning of the search radius may allow control of the mean resection error.

The addition of the 3 mm burr allowed the research camera system to successfully resect both the 6 and 4 mm peg holes. To facilitate this, the search radius of the 3 mm burr was set to 0.1 mm. Larger search radii increased the likelihood of resection failure. With the original 0.25 mm search radius the user was required to correctly place the burr to a positional accuracy of only 0.25 mm, assuming an error free system. This proved overly challenging to complete the 4 mm resections. The reduction to a search radius of 0.2 mm afforded an accuracy of 0.3 mm, and permitted the successful resection of one peg hole. However, this was attributed to luck as it was performed in nearly a single action on the first attempt. Conversely, several attempts were made upon the second peg hole, from a range of approaches, to no effect. The final radius of 0.1 mm allowed for a positional error of 0.4 mm, and all peg holes were successfully resected. However, the reduced search radius also produced increased over-resection as previously theorised. The significantly increased over-resection observed for the 6 mm peg holes is attributed to the increased rate of resection. The initial resection of the 6 mm peg holes, which predominately defined the final depth, was typically performed in a single motion. However, due to the reduced clearance, and therefore increased precision required, the 4 mm peg holes took several motions to complete resection, resulting in a much slower rate of resection. Therefore, the 4 mm holes were less susceptible to over-resection as a result of system latency.

The ease with which peg hole resection was performed should be noted. Experience with the commercial Blue Belt system has demonstrated that peg hole resection proved the most challenging stage of resection for novice users. The augmented guidance allowed the burr to be placed on target accurately and intuitively. The use of speed control over the standard exposure control of the Blue Belt system also appeared to simplify the process, as retraction often required more substantial repositioning of the burr. It is highlighted that the planar surface of the resection volume used here may have simplified resection of the peg holes. Humans are inherently sensitive to perpendicular angles, such as that produced by the tool when correctly aligned with the surface for resection.

In conclusion both third generation systems performed well during resection, producing mean absolute errors of 0.37±0.11 and 0.34±0.04 mm respectively. The addition of an additional 3 mm burr also allowed the research camera system to perform all resections including peg holes.

8.3.5 Hip Centre Analysis

To demonstrate the advantage of the increased FOV and hands free nature of the third generation system the original hip centre algorithm presented for the first generation system was re-implemented. Furthermore, an alternative, pivoting, algorithm, based upon least square fitting was also implemented (Siston & Delp 2006).

8.3.5.1 Materials and Methods

To demonstrate the principle of the system the existing probe and known geometry target were used to emulate and miniature femur and pelvis, respectively. Each was fitted with a standard 32 mm marker with manufacturing errors accounted for.

The effect of femur and pelvic marker separation from the hip centre were investigated, as were sample count, inter-sample spacing, and magnitude of femoral rotation. Finally, the influence of camera-marker separation was examined.

Each measurement was performed three times and the two algorithms were run simultaneously upon the same data.

8.3.5.2 Results

The results of the hip centre experiments are shown below in Figure 8.52.

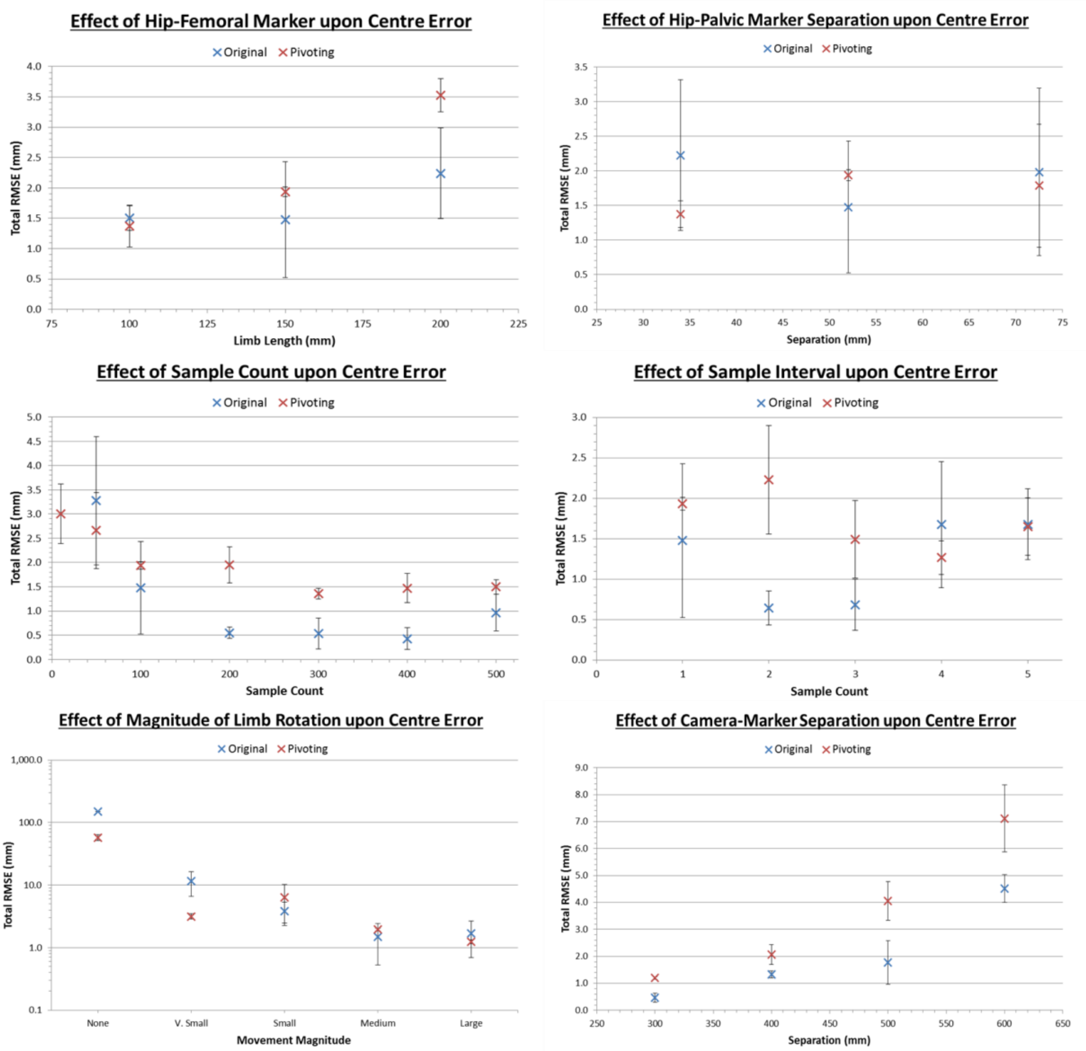


Figure 8.52: Results of hip centre experiments

The pivoting algorithm showed a significant increase in error with increase hip-femoral marker separation (P = 0.01 and P < 0.01). The original algorithm did not produce a significant increase, with the second measurement being insignificantly smaller than the first (P = 0.31). The original system was found to provided significantly better accuracy at 200 mm separation (P = 0.03). The pelvic marker separation did not have a significant effect on the accuracy of either algorithm. Neither algorithm provided significantly better results (All P > 0.17).

Both systems report an accuracy improvement with increased sample collection. All individual increments failed to reach significance (All P > 0.09). However, the original algorithm produced significantly improved results with 200 or more samples (P = 0.04). The pivoting algorithm reached significance at 300 samples (P < 0.01). At 200, 300, and 400 samples the original algorithm was significantly more accurate than the pivoting algorithm (All $P < 0.01$). Sampling interval was found to produce no significant difference upon centre error (All $P > 0.14$).

Both algorithms demonstrated a general accuracy improvement with increased limb movement. For both algorithms, medium and large movements were found to provide significantly better accuracies than very small or no movement ($P = 0.03$ and $P < 0.01$).

Finally, increased camera-marker separation was found to reduce the accuracy of both systems. Each 100 mm increment had a significant effect on the pivoting algorithm ($P = 0.04$, $P = 0.01$, and $P = 0.02$). The original algorithm reported no significant difference until 600 mm ($P = 0.01$). The pivoting algorithm became significantly less accurate with each increment ($P = 0.04$, $P = 0.01$, and $P = 0.02$). The original algorithm was found to outperform the pivoting system at each separation ($P < 0.01$, $P = 0.03$, $P = 0.02$, and $P = 0.03$).

8.3.5.3 Discussion

All experiments summarised by Figure 8.52, with the exception of pelvic marker separation, produced the expected results. Unexpectedly, however, the original system was found to produce improved accuracy for 63.0% of measurements, of which 52.9% demonstrated significance. Conversely, 67% of pivoting algorithm results produced lower standard deviations, 44.4% of which presented significance. This indicates that the original system provided the superior accuracy, while the pivoting algorithm offered improved precision.

The experiments illustrated that given sufficient samples and range of motion the original algorithm sufficiently managed signal noise to provide superior accuracy. Given the proposed proximal knee placement of the femoral marker, a marker-hip separation of approximately 400 mm may be assumed. To fully capture the range of motion desired a camera-marker separation of 600 mm would be required. Performing an exponential regression upon the original algorithm results under increased femoral-marker hip separation ($R^2 = 0.72$) provides a rough approximation for RMSE of 4.5 mm at 400 mm separation. This may be expressed as a factor increase of 3.0 from the error obtained at 150 mm separation. Applying this factor to the RMSE obtained at 600 mm camera-marker separation provides an approximate RMSE of 13.5 mm under practical conditions. This value is extremely approximate and does not account for additional error sources that may be incurred. Taking the average human femur length as 480 mm (Huang et al. 2012), and assuming a worst case error scenario, whereby the total RMSE is presented by a single component orthogonal to the mechanical axis of the femur, an error of 1.6° in the angle between the mechanical and anatomical axes of the femur is produced.

8.3.6 Conclusion

The four analysis sections above demonstrated the suitability of the third generation system for orthopaedic surgical guidance. The two systems provided the accuracy required during both probing, 0.52±0.12 and 0.56±0.10 mm respectively, and resection, 0.37±0.11 and 0.34±0.04 mm. Furthermore, unlike the second generation system, this accuracy was provided over a practical FOV for the investigated application.

The increased consistency achieved with the research cameras, particularly during motion, was attributed predominately to inter-camera capture synchronisation. However, the use of global shutters and mechanically locked lens parameters also improved the system, as shown by Section 8.3.1. For these reasons the third generation research camera system is the preferred system for accuracy above all other implementations.

8.4 Conclusion

The third generation systems provided a good level of accuracy, meeting both the probing and resection accuracy targets. However, further optimisation and validation is required to ensure the system provides sufficient accuracy during hip centre location.

In addition to accuracy, usability and intuitiveness were cited during discussion of the second generation system as key motivators for the development of the third generation system. The hands-free operation, afforded by the head-mounted approach, greatly improved usability of the system. This allowed several tasks, such as locating the hip centre, to be performed without assistance. Furthermore, the independence of tool and tracking system allowed for much more versatile use. The angle of the probe could be varied to a far greater degree while probing difficult features, such as the ankle joint or postural elements of the knee.

Again, although not quantified, the third generation system was believed to be notably more intuitive than the second generation system. This was attributed partially to the 3D display of both physical and virtual components. Tasks that required the tool to be position based upon an augmented guide, such as resection, were greatly improved by the depth perception the stereoscopic view provided. However, the natural view provided the most notable improvement to the intuitiveness of the system. Irrespective of the mounting modality of the cameras, be it head or bench-mounted, the appearance of tools moving within the FOV is more natural than that of the FOV moving with the tool, as experienced with the second generation system, and as such is more intuitive to the novice user.

In conclusion, the third generation system met the tasks presented by Chapter 7.4. It is believed that with several refinements the presented system could provide a platform for augmented reality guidance within UKA procedures. The suggested refinements and a number of proposed expansions and applications for the system are presented below in Section 9.

9 Discussion

This final chapter concludes the work presented within this thesis. The three generations and their principle findings are briefly summarised in Section 9.1. Suggested further work, including system refinements are then discussed in Section 9.1.1. This section and thesis are then concluded by discussion of potential future applications and functionality that a finalised ARgCAOS system may offer in Section 9.2.2.

9.1 Presented Systems

Initial investigations, described by Chapter 5, concluded that it would be extremely challenging to reach the required accuracy with a markerless system. The required accuracy could most likely be achieved, particularly in light of more recent projective type imaging systems. However, the operational environment presented a substantial challenge. The visibility of bone, which would be required for accurate tracking, would be restricted to capture the benefits of a minimally invasive approach. Furthermore, the limited visible bone may be further obscured by tools and bone debris during resection. Current systems present their minimally invasive nature as one of their chief attributes, with considerable device cost offsetting achieved through the reduced recovery times this allows. Therefore, it was concluded a markerless approach should not be sought at the expense of a minimally invasive approach.

Chapter 6 presented the first of three generations of real-colour planar fiducial marker tracking systems. The system was built upon ARToolKit, utilising the extrinsic estimation required for augmented reality to provide tool tracking. Accuracy was found to be lacking, however the concept proved promising and as such it was decided to produce a second generation system, providing improved accuracy and graphics. The constrained nature of ARToolKit complicated the implementation of the desired improvements

Chapter 7 established that due to this complication a ground up approach to the second generation system would be most efficient, redeveloping it upon the OpenCV and OpenGL libraries. The second generation system provided substantial improvements, falling just short of the 1 mm probing accuracy target, at 1.01±0.05 mm RMSE. It provided semi-active constrained resection to an accuracy of 0.36±0.34 mm, via a volumetric resection model and cutting burr speed control. However, the accuracy demonstrated orientation variability, and the limited FOV afforded by the tool mounted approach proved restrictive to the applications of the system. As such, a third generation was proposed.

Chapter 8 presented this third generation system. A HMD was utilised for display, and the tracking system was converted to a stereoscopic design which was moved from the resection tool to the HMD. This allowed a more intuitive and immersive experience, increasing the FOV and freeing the user's hands. The system was shown to meet the probing accuracy requirement, almost halving the 1 mm target, achieving 0.55±0.10 mm RMSE. This system was also able to provided semi-active constrained resection, to an absolute accuracy of 0.34±0.04 mm.

These results compare favourably to those reported in literature. Notably a comparison is drawn to the Polaris system based upon the work of Wiles *et al.* (Wiles et al. 2004). Applying the values of the experiments performed in this thesis to the work of Wiles *et al.* the Polaris system is estimated to produce an accuracy of 0.77±0.43 mm, 40% less than that of the presented system. The system also compares well to a similar AR guidance system utilising stereoscopic cameras reported upon by Badiali *et al.*, producing three-fold improvement in accuracy (Badiali et al. 2014). However, as previously noted, the work of Badiali *et al.* includes the effect of human error, as with much of the work on AR guidance. Therefore, the results should not be directly compared.

9.2 Limitations

A core limitation of the work presented within this thesis is the lack of saw bone or cadaveric experimentation. This prevents an implant and limb alignment accuracy figure from being define for the system, and as such limits the extent to which the system may be compared to literature and commercial systems.

Neither saw bone nor cadaveric experiments were performed as they would have required a more complete system to hold any value. While the presented system was able to perform controlled resection the planning system was extremely simplistic, allowing only manual positioning of implants with no feedback or guidance. The errors resulting from planning would have far outweighed the resection errors reported above, and as such the final implant alignment results would not have accurately reflected the performance of the system.

An additional limitation to this study was the risk of inaccuracy during analysis of the system. Due to the sub-millimetre and sub-degree accuracies investigated all measurements were susceptible to non-device errors. Throughout this work every effort was taken to minimise these errors, however a number of potential sources remained unmitigated. A chief example of this was the target object used for probing analysis. While the manufacturers of the 3D printer used report nominal accuracies of 20-85 μm, it is suggested this is unlikely to be reliable. Ideally a coordinate measurement machine could have been used to measure the actual position of each test divot to an accuracy of a few micrometres. A similar technique could also have been used to measure the final resection accuracy of the systems.

A final limitation was the lack of user testing, which is considered a two fold limitation. Firstly, without user testing all measures of intuativness, and core benefit of AR guidance, are based upon the arthors perception. Secondly, skilled user feedback, namely from orthopaedic surgeons, both with and without CAOS experience, would have helped target development to the needs and issues of the final potential customers throughout the development of the system.

In addition to these experimental limitations the final generation system also presented a number of limitations. Several of these and their proposed solutions are discussed below in 9.2.1 as optimisations of the system.

9.3 Future Work

A number of small scale system optimisations are first discussed by Section 9.2.1. Following this, the proposed more commercially viable iteration of the system is discussed, describing potential features and applications.

9.3.1 Optimisation

The presented third generation system would benefit from two forms of optimisation, accuracy and latency. Despite meeting the original accuracy targets as outlined in Chapter 4, the system was still found to be lacking. This was most notable from the reporting of hip centre accuracy of Section 8.3.5. Latency is also identified as a concern with the final implementation. The delay of the presented system was not readily perceivable to the user, with even fast actions such as the clicking of fingers appearing well synchronised with both the audible and tactile response. However, the overall system latency was believed to contribute to the over-resection observed for the final generation system in Section 8.3.4.

9.3.1.1 Accuracy

System accuracy, or the lack of, stemmed from a combination of both hardware and software sources, often interconnected. This section shall describe a number of these sources and potential optimisations where possible.

Sections 5.2.2.1 and 8.2.1.2.2 discussed camera and lens selection and the parameters, such as resolution, focal length, and distortion that would affect the accuracy of the system. For the final selection a number of compromises were made which negatively impacted the accuracy of the system. Cameras offering maximum resolution were found to be cost prohibitive. Additionally, as resolution increased so too did the bandwidth requirements, and as such the framerate was reduced. It was also noted that the image processing times, in particular thresholding, were considerably increased with image resolution. The focal length was selected such that the FOV of the lens approximately matched that of the HMD, such that immersion was maximised. However, this resulted in a very large FOV. As a result the sensor pixels were spread across a very large volume, resulting in a relatively low spatial resolution. Furthermore, large FOV, or wide angle, lenses are inherently susceptible to image distortion. The compound lenses selected compensated for this using an array of lenses and stops to inter-cancel aberrations. However, the may result in an extremely complex distortion pattern, as pincushion and barrel distortion inducing lenses are combined. Therefore, the five parameter distortion model implemented by OpenCV may not fully compensate for the physical distortion. Furthermore, the distortion model is applied irrespective of pixel colour. Therefore, any chromatic aberrations residual to the lens array are not corrected. This would have the effect of blurring marker edges during thresholding.

Colour also affects the hardware accuracy of the camera. The pixel elements of the CMOS sensor used by the research camera are insensitive to the wavelength of light. Therefore, to capture the chromatic information a Bayer filter is placed over the sensors. The Bayer filter, as shown by Figure 9.1 below covers each pixel elements with either a red, green, or blue pass filter.

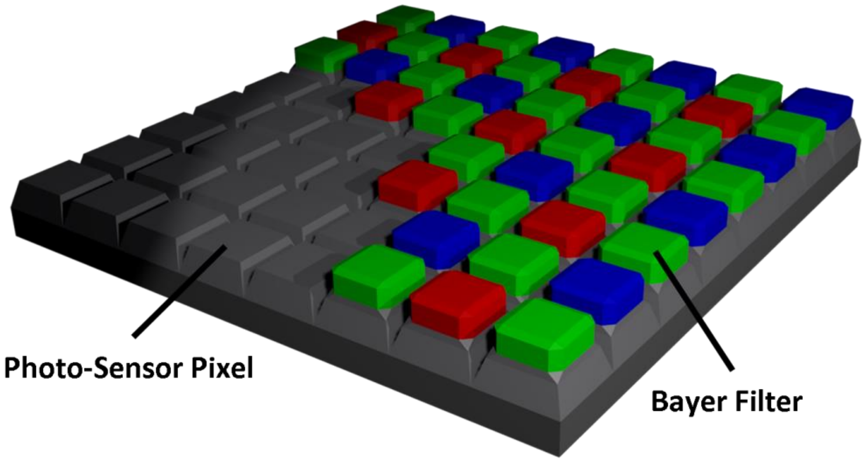


Figure 9.1: Colour CMOS sensor Bayer filter

The raw format of these sensors returns the light intensity of each pixel coloured by the filter pattern. This is not ideal for black and white edge detection as the edge appears spread over several pixels, as the change is detected by the different elements. On-chip processing may be used to merge the pixel values to produce an image of equal size where each pixel has an RGB value. However, this reduces the spatial resolution of the image as pixels are combined.

Commercial optical tracking systems, such as the Vicon or Polaris systems, utilise IR or near-IR cameras and therefore avoid the issue by not requiring a Bayer filter. However, clearly to provide video see-through AR, colour images are required, and as such this small reduction in accuracy cannot be avoided.

Stereoscopically the cameras were synchronised using software triggers. This provided good synchronisation, as shown by Section 8.3.3. However, this could be further improved via hardware trigging. Hardware triggering utilises a physical connection between the two cameras to ensure near simultaneous image capture. This would avoid the potential inter-capture delays of software triggers and allow superior and consistent motion capture.

An additional hardware concern is noted from Equation 8.14, whereby depth error is shown to be inversely proportional to the base line of the stereo system. Therefore, the relatively narrow base line utilised by both systems, approximately a tenth of the Polaris Spectra, was detrimental to their performance. However, this distance was selected to allow immersive augmented reality by matching the average IPD of users. This effectively limits the useable range of the system for accurate tracking. However, the head mounted modality of the presented system reduces the required maximum separation making the relatively limited range sufficient.

In addition to the camera system itself, marker manufacturing errors were highlighted at several stages of both Chapters 7 and 8. Initially, markers were found to be incorrectly sized. This was compensated for by measuring the produced markers using a highly accurate microscope and feeding these values into the system. However, some marker edges were later found to be non-linear, showing slight warping. One method found to reduce warping was to avoid the horizontal axis of the laser engraver during production. As seen by Figure 8.46, however, this produced relatively rough edges with 0.1 mm undulation. While this is unlikely to affect the accuracy of the system, alternative manufacturing methods, such as CNC or etching, could be investigated.

Alternatively to manufacturing improvement, the software of the system could be made more robust to errors. The current system performs simple linear regression upon the edges of the markers to determine the corner positions to sub-pixel accuracy. To compensate for potential warping parabolic regression could instead be utilised. Furthermore, this may compensate for edge wrapping resulting from lens distortion.

To further improve corner detection from distorted images, undistortion mapping should first be applied. As previously discussed, this was originally omitted to improve computational efficiency, and as initial experiments showed a reduction in accuracy. However, it is suspected that further work utilising remapping could reach or surpass the accuracy currently achieved. Furthermore, in comparison to the costly adaptive thresholding used, a remapping operation would not induce significant latency.

An additional consideration for accuracy improvement would be the use of higher order fiducial markers. The system currently employs four corner markers, one additional corner than required to determine the pose of the marker to 6 DOF. The additional corner is utilised to improve the robustness of the system. With only three corners any positional error in any one of the corners directly translates into a pose error. However, the fourth corner provides additional data and a serves to reduce the effect of positional errors. This may be extended by the addition of further corners, or identifiable points, therefore minimising the effect of single positional errors and potentially allowing the identification of outliers. Figure 9.2 below shows a number of concept markers considered.

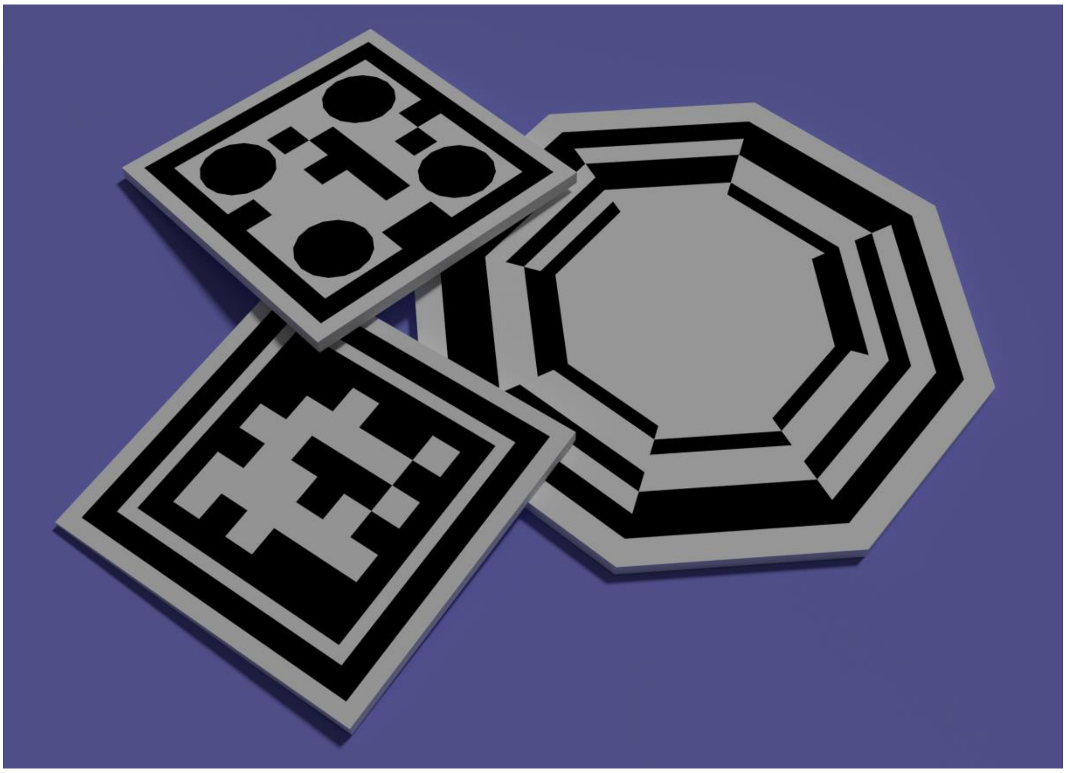


Figure 9.2: Increase measurement point concept markers

The top left marker would utilise a combination of the existing edge detection method and a blob centre detection algorithm to determine a total of eight measurement points, while marker identification and orientation are handled by a cross of units. The bottom left marker features two black perimeters. Including the inside edge of the outer perimeter a total of twelve corners could be detected. Finally the octagonal marker presents eight corners with identification performed using the inner elements.

A related technique was implemented, whereby instead of increasing the count of measurement points per marker, a number of markers were combined to form a cluster. Using the known fixed relative position of each marker, stored within an XML document, the pose of each marker was transformed into the common origin of the cluster. This was primarily implemented to provide marker obstruction protection. Providing a single marker remained visible at all times the pose of the cluster could be determined. However, this could provide an enhancement to accuracy through a similar mechanism as discussed above.

The final method discussed regarding accuracy improvement is pose filtering. Filtering is typically employed to remove or reduce an unwanted component of a system signal, here, pose errors. The original implementation of filtering was motivated by the significant jitter observed with the second generation system at near-parallel orientations. It was observed that the mean error taken over several frames was typically less than the error for a single frame as the results were relatively symmetrically distributed. Therefore, a running mean was implemented based upon variants of Equation 9.1 below.

$$\overline{[R|T]}_n = \sum \frac{6}{10}[R|T]_n + \frac{3}{10}[R|T]_{n-1} + \frac{1}{10}[R|T]_{n-2} \qquad (9.1)$$

During static imaging the averaging worked well, considerably reducing model jitter. However, under dynamic situations the above implementation induced latency in the model. Using three frames, including the current frame, producing a total time period of 150 ms for the 20 fps LifeCam system the latency was very evident, particularly during rotation where fast angular changes could readily occur.

The desire to filter results without incurring latency resulting from temporal averaging is a major concern in tracking systems. Kalman filters are a popular approach often used by the aerospace industry for guidance and navigation applications (Schmidt 1981). A Kalman filter is a real-time error reduction methodology. In basic terms, based upon models of the system, the Kalman filter predicts the result. For example, in the presented case, the marker pose is predicted. The measured marker pose is then compared to the predicted pose. A weighted average of the predicted and measured pose is then returned. The weighting is based upon the certainty of the pose prediction, and the results are fed back into the filter to generate the next prediction and certainty values (Kalman 1960).

Initially a 2D Kalman filter was implemented. This was applied to the corner detection algorithms to reduce detection noise. However, this was found to have no visible effect upon model jitter, as it was concluded the corner detection error was considerably smaller than the pose estimation error at near-parallel orientations. Therefore, a Kalman filter was designed to directly predict and optimise the marker pose matrix. The model was based upon the equations of motion, both positional and angular, as shown by Equations 9.2 and 9.3 respectively.

$$s = s_0 + u_0 t + \frac{1}{2} a t^2 \tag{9.2}$$

$$\theta = \theta_0 + \omega_0 t + \frac{1}{2} \alpha t^2 \tag{9.3}$$

Due to the 6 DOF of the marker pose the system was modelled by an 18x18 transition matrix and a six element measurement matrix.

A standalone program was implemented to simulate a marker to 6 DOF. Translational and rotational motion and noise were applied to the model to simulate a tracked marker with pose estimation errors. Figure 9.3 illustrates a static marker with 20° of rotational noise and 10 pixels of translational noise applied to each axis, before and after Kalman filtering.

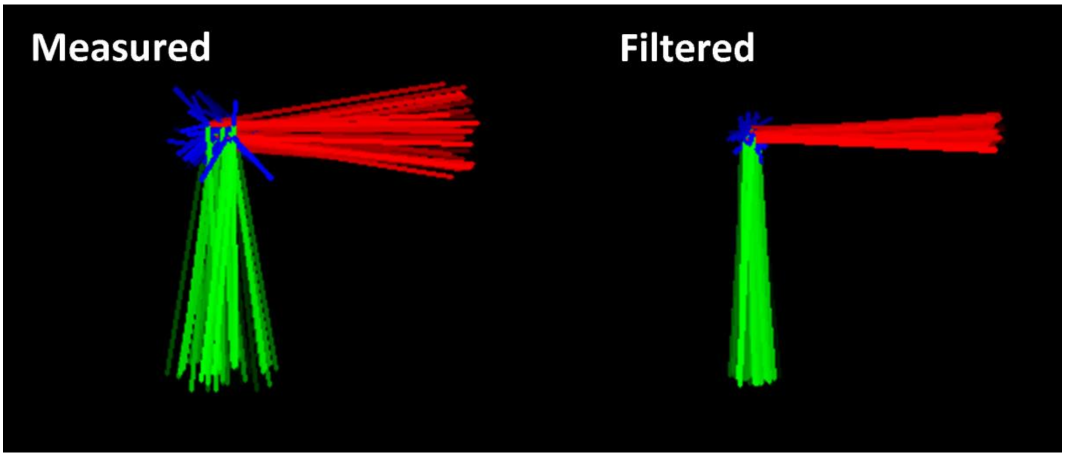


Figure 9.3: Effect of Kalman filter upon 6 DOF marker pose simulation

The standard colour code is used to represent the three axes. It may be seen from Figure 9.3 that filtering considerably reduced the noise present in the original measure. The effect upon translation may be more clearly seen in Figure 9.4 below.

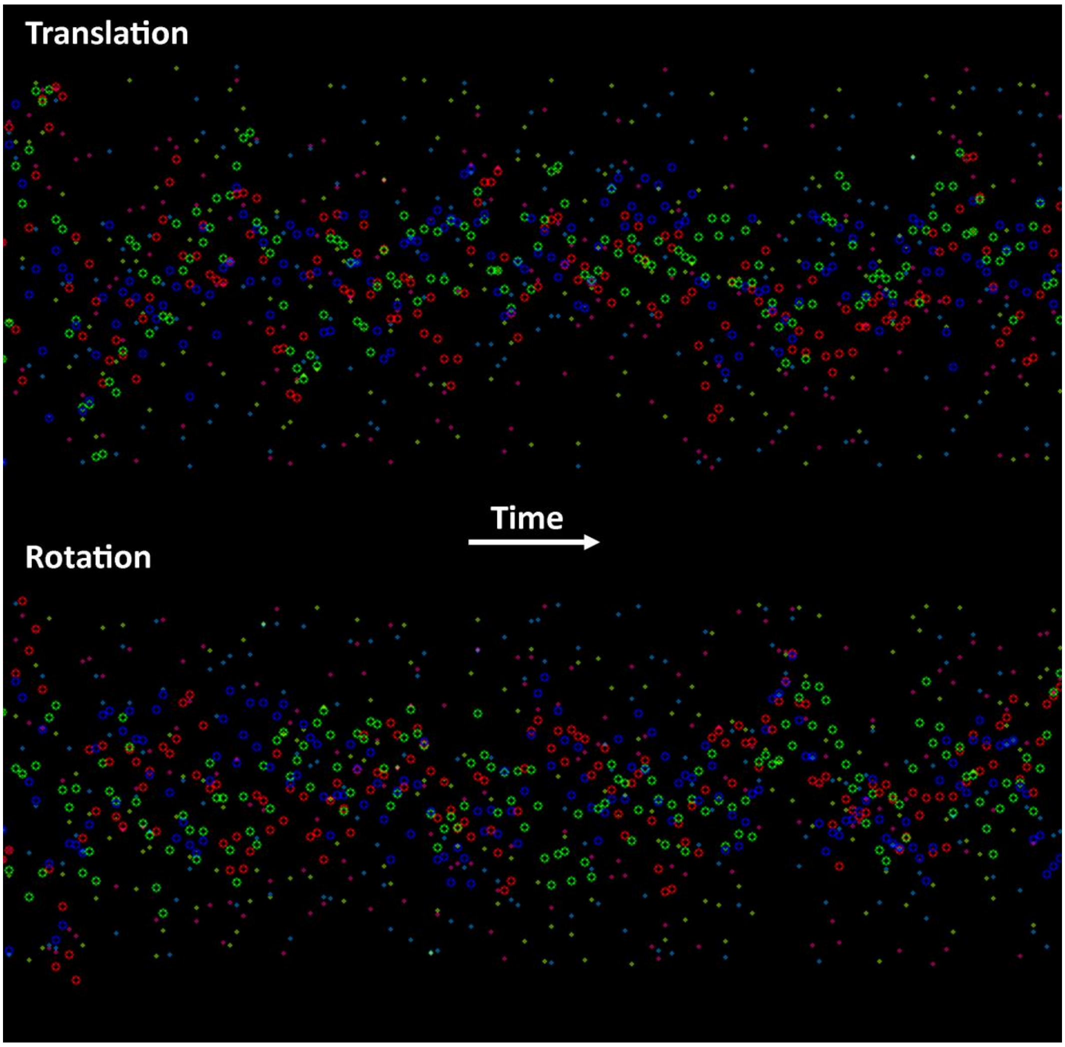


Figure 9.4: Effect of Kalman filter upon translation and rotation measurements. The small solid dots indicate measured values while the circles show their filtered equivalent.

It may be seen for both translational and rotational axes that the filtered results, shown by the hollow circles, produced the tighter grouping, indicating a reduction of noise.

Work upon the Kalman filter was begun to reduce the large jitter seen with the second generation system at near-parallel orientation. Before completion, the second generation system had been superseded by the third generation system. As discussed by Section 8.3.2, the third generation system did not present this near-parallel jitter, and as such the developed Kalman filter was not implemented into the main system. In light of the full analysis performed in Chapter 8 it is believed it would be beneficial to the system and should be implemented. Kalman filters typically require tuning to the system and noise. Therefore, the improvements of approximately 30% presented above may not be achieved. However, an improvement to the accuracy, and more so to the precision, that was seen to increase significantly over the second generation system, would be expected. It may also help further dampen the effect of vibration during resection.

Several sources of error were shown to be unavoidable with the current system, such as the limited base line and chromatic true resolution reduction. However, the proposed improvements, such as improved distortion characterisation and compensation, increased measurement points, and Kalman filtering would be expected to provide an accuracy improvement, ideally providing submillimeter accuracy across the full stereoscopic volume to a depth of at least one meter. This combined with the latency optimisation discussed below would provide a practical system for a wide range of guided orthopaedic procedures.

9.3.1.2 Latency

In addition to accuracy, latency was identified as a core system concern. Section 8.3.4 implicated latency as the cause of over-resection observed with the research camera system. Timing analysis identified two dominant sources of latency, image capture and thresholding.

The image capture latency was reduced by the use of the research cameras, as these offered both shorter exposure times and internal processing. However, the increased image resolution increased the image thresholding time. Commercial tracking systems, based upon IR cameras minimise both of these latencies. IR cameras, given sufficient lamination may have very short exposure times. Additionally, internal processing time is reduced as no processing of the Bayer layer is required. Furthermore, transmission and loading times are reduced as images contain only a single channel. Finally, due to the use of IR cameras the captured scene is considerably less complex and as such a basic thresholding algorithm may be applied, if required at all.

Hardware latency may not be readily reduced. More high-end cameras that provide larger image sensors with higher quantum efficiency would permit shorter exposure times. Additionally, they may offer more powerful internal processing. Using the existing research cameras, the exposure time may have been reduced and a gain applied to the image to maintain the level of illumination. However, this results in lower quality images as any noise is also amplified by the gain. Alternatively, the raw image format may be used. This removes the latency associated with Bayer processing while also reducing the data of each image, therefore reducing transmission times. Finally, raw images would not require preliminary conversion to greyscale prior to thresholding. However, as previously discussed, the raw format reduces the quality of edges and therefore corner extraction. Furthermore, the raw format is of a lower aesthetic quality and may reduce the immersion of the system. Several attempts to reduce the thresholding latency are described in Appendix A2.1.2. There it was concluded that the latency could not be readily reduced without reducing the quality and accuracy of the system. Two additional approaches, Kalman filtering and parallelisation, are discussed here.

As discussed in Section 9.2.1.1 above, a 2D Kalman filter was implemented for corner detection. One stage of the Kalman filter generates a pose prediction. In the presented solution this is used with the most recent measured pose to generate a pose estimation. However, it is possible to generate a prediction after the current measurement, which would predict the positions of the corners in the next frame. Using this information an approximate search region for the marker may be defined in the next image frame. Therefore, it would only be necessary to threshold the estimated image region, offering considerable latency reductions.

Parallelisation, as proposed here, would not directly reduce latency, however it would allow the system to operate at a higher framerate. For the third generation system each camera is run upon a separate processor thread, as shown by Figure 8.7. However, each thread completes the entire image processing method in series, including image capture and thresholding. This is logical as each stage requires the result of the previous. However, with parallelisation processing could be initialised for the next frame while the current frame is being completed. This is described by the activity diagram for a single camera system shown by Figure 9.5 below.

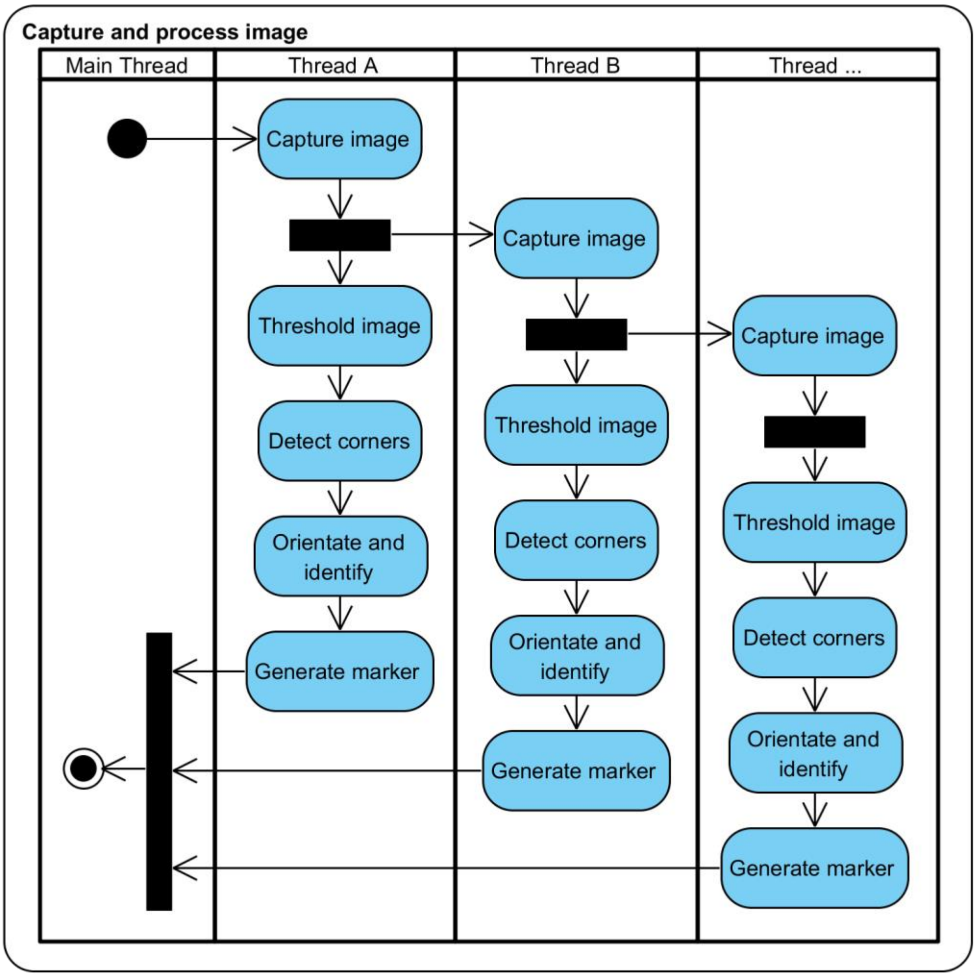


Figure 9.5: Activity diagram of parallelisation of image capture an processing

This initial frame would be processed by Figure 9.5 as normal. However, once the capture process was completed the next frame would begin processing on a new thread. As soon as capture was completed thresholding would be started.

As mentioned above, this approach would not decrease latency. Latency is defined as the time between image capture and display. The approach described by Figure 9.5 would not reduce this total time, merely interlace it. Furthermore, the approach may increase the latency, as multiple frames are being processed at once, reducing the resources available for any one frame. Furthermore, as illustrated by Figure 9.6 below, latency could be increased due to stages, such as display, that could not be interlaced. However, the framerate, that is the number of frames processed in a given time, would be increased.

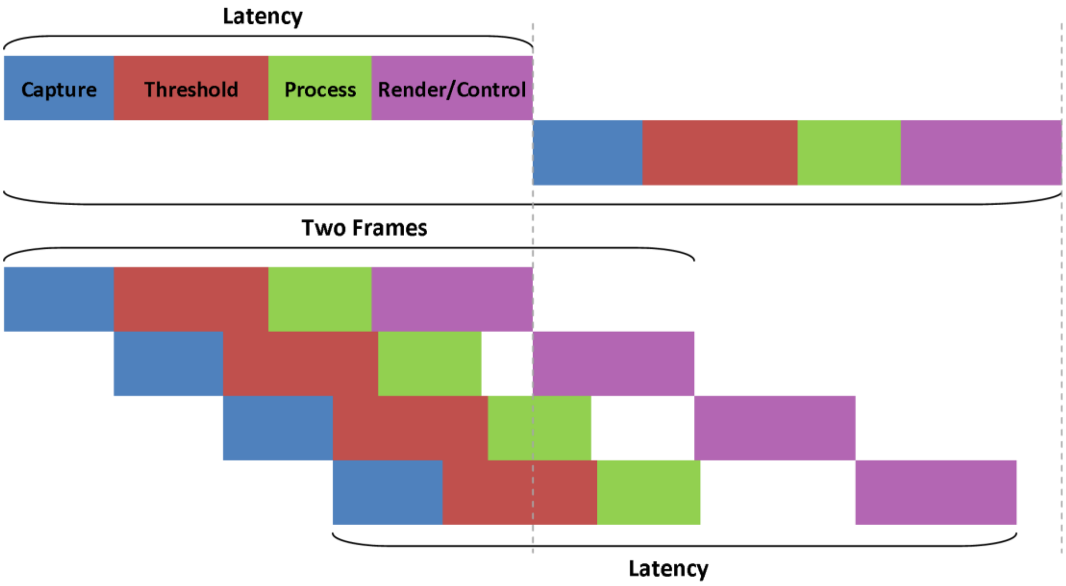


Figure 9.6: Multithreaded system example

This concludes discussion on immediate optimisations of the system. The following section presents more long term and hardware based imporvments to the system.

9.3.2 Potential system

This section discusses of a more commercially viable potential implementation of the system, describing hardware now avalible and the potential benefits afforded.

Section 8.2.1 described both the LifeCam and research camera systems as overly heavy and bulky for practical clinical use. The cameras and HMD were selected for their quality, affordability, and usability as opposed to weight or size. However, considerably more practical alternatives are available.

Many modern smartphones now contain very high quality image sensors with impressive optic quality, given their size. For example the flagship Samsung and Apple smartphones feature 16 and 12 MP main cameras respectively. Alternatively, the Nokia 808 boasts a 41MP camera. However, these cameras utilise relatively small sensors, resulting in pixel sizes of 1.12, 1.22, and 1.4 µm, respectively. This results in a pixel area approximately 28% the size of the research camera. As discussed previously, this reduces the ability of the sensor to capture light, requiring either an increased exposure time or gain, thus reducing the image quality. However, in the well-lit environment of an operating theatre these sensors would be expected to perform very well, producing high quality video at high framerates. The cameras also feature very high quality optics. Despite their incredibly slim form factor they utilise as many as ten elements to form a compound lens.

It is suggested that a commercial system could utilise these compact imaging technologies to greatly reduce the size and weight of the HMD tracking technology. To further scale the head mounted system the display should also be minimised.

The Oculus Rift HMD selected utilises lenses to focus a high resolution OLED smartphone display. This results in a large HMD system, as the headset must be large enough to fit the display and provide sufficient display-to-lens and lens-to-eye separation to allow the images to be focused. Several companies have mirrored this approach, with notable examples including the HTC Vive and Project Morpheus by Sony, shown in Figure 9.7 below. These types of HMD dominate the market due to the availability of relatively low cost displays resulting form the popularity of smartphone devices.

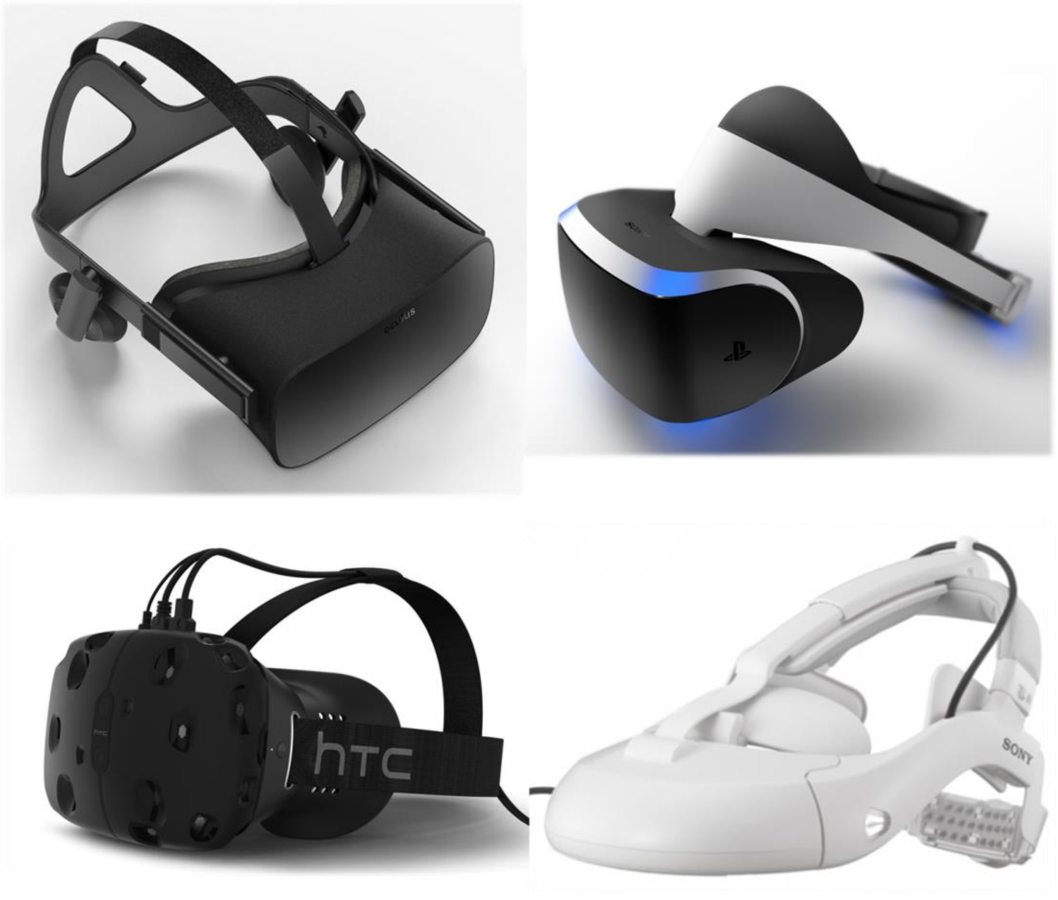


Figure 9.7: Current HMD devices – TL: Oculus Rift (consumer edition), TL: Sony Project Morpheus, BL: HTC Vive, BR: Sony HMS-3000MT

The leading commercial surgical HMD, the HMS-3000MT by Sony, deploys a similar approach, instead using two separate OLED display panels. This allows the device a somewhat sleeker design, as seen in Figure 9.7. However, the system is still relatively bulky, weighting approximately 480 g, 40 g more than the Oculus Rift DK2 used by the third generation system. Furthermore, the FOV is limited to only 45°, making the system impractical for AR applications.

With the increased interest in VR, research into alternative display technologies is expected to increase, with technologies such as near-eye light field displays (Wang et al. 2015; Lanman & Luebke 2013) potentially providing much lower profile HMDs. However, presently all high-resolution HMDs follow the relatively bulky designs depicted above. Therefore, even with the use of small smartphone cameras, the size of the HMD system may be unappealing to surgeons. However, it is hoped the potential advantages, some of which shall be discussed below, would be seen to outweigh the non-ideal physicality of the system.

Section 2.5.2 discussed OST HMDs as alternatives to the video see-through HMD selected in Section 8.2.1.1, due to cost and latency matching concerns. However, an optimal system would likely be deployed under an OST HMD. OST has several advantages as previously discussed, principally that the user views the physical world, causing no compromise to quality, nor camera-eye offset. In context of the work presented OST provides an additional advantage. The system would no longer be bound to the use of visible spectrum colour cameras. Furthermore, the baseline of the cameras would no longer be required to match that of the eyes of the user, increasing potential accuracy.

Commercial OST HMDs may be divided into two categories, augmentation or *head-up display* (HUD). Augmentation devices allow virtual information to be displayed across the majority of the user's view. Meanwhile, HUD devices only display to a small portion of the view of the user, typically towards a corner. Figure 9.8 below presents several examples of commercial, or near-commercial, systems.

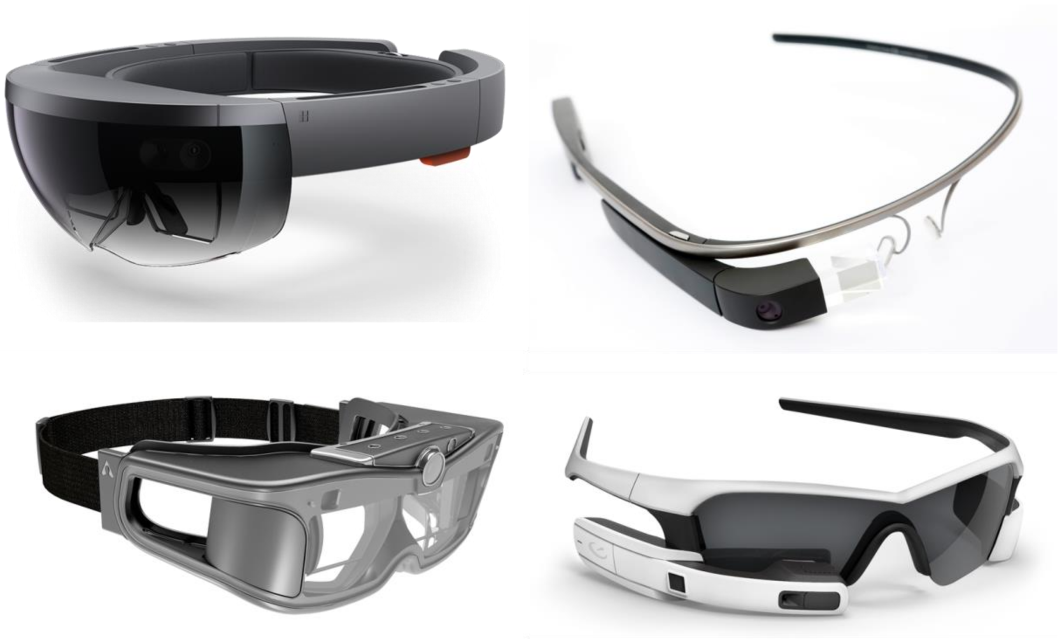


Figure 9.8: OST HMDs – Left: augmented devices – TL: Microsoft HoloLens, BL: Atheer AiR. Right: HUD devices – TR: Google Glass, BR: Recon Jet

It may be seen by comparison of Figure 9.7 and Figure 9.8 that OST devices are typically less bulky than VST systems, particularly so for HUD only platforms. However, HUD devices lack the display required for true augmented vision. The two presented augmented systems, the Microsoft HoloLens and Atheer Air, follow a similar design. They feature semi-transparent displays, that allow scene lighting and virtual light to be combined, and a series of cameras, to allow augmented overlay alignment and user interaction. This would make either device ideal for AR guidance of orthopaedic procedures. However, minor modifications would be required to ensure the devices were; comfortable for prolonged use, particularly in the typically downward looking stances of surgery; did not become overly warm under surgical conditions; and were resistant to spray and were compatible or substitutes to the eye and face protection warn by surgeons.

Therefore, a commercial version of the ARgCAOS system would likely utilise either a Microsoft HoloLens or Atheer Air to provide both visualisation and tracking. The remaining hardware would remain relatively unchanged. However, an additional breaking mechanism, such as an EM clutch, as discussed by Section 7.2.1.3, may be introduced to reduce control latency.

9.4 Clinical Relevance

Despite being a considerable distance from a commercially deployable system the prevent research may be considered an important milestone to the application of both intuitive AR guidance and compact head-mounted tracking to the area of knee replacement within orthopaedic surgery.

Cheifly the work demonstrated the ability of a low-cost, light-weight, and compact head-mounted visible spectrum tracking system to provide the sub-millimetre accuracy considered neccarserry to deliver consistently accuracte UKAs.

Furthermore, the applicability of AR to the guidance of complex bone resections was clearly indicated all be it not thoroughly quantified. The following section discusses the advantages of a refined AR implementation.

9.4.1 Augmented Reality Guidance

This section first discusses the primary use of the proposed system during the guided UKA procedure before looking at wider reaching applications.

9.4.1.1 Guided UKA Procedure

The system is proposed to provide continuous guidance, whereby the system assists with each stage of the procedure to ensure consistently accurate high quality outcomes, while reducing operating times to a minimum.

As described by Chapter 4, the guided UKA procedure is divided into six stages: setup, anatomy registration, planning, resection, trial, and final implant and closure. It is proposed the augmented system could guide each of these six stages.

During setup the augmented display could be used to review patient notes and joint X-rays. Various safeguards could ensure the avoidance of never events. The system would then highlight which components, such as screws and markers, should be used during setup. Augmented guides from the limb markers would ensure consistent marker orientation and separation relative to the joint.

The system would then guide the surgeon through registration of the patient anatomy by highlighting approximate locations of anatomical angles and guiding the surgeon through sufficient flexion. Bone model generation would be performed as described by Chapters 7 and 8, whereby the probe would be used to trace the dimensions of the condyles to generate a volumetric cuboid. The probe would then be traced across the surface of the condyles removing excess volume from the model, resulting in an accurate representation of the bone.

Planning would then be performed upon the physical limb of the patient. Viewing virtual implants upon the joint of the patient would allow easy implant positioning and sizing as any overhang would be readily visible as the surgeon examined the joint from different angles. Additional information, such as post-operative joint angles, contact points, and ligament tension, calculated from the patient's anatomy and proposed implant position, could be imaged directly on the limb as the surgeon articulated the leg through its range of motion. An example of such guidance is given below by Figure 9.9, which shows tibial alignment and collateral ligament tension during flexion. The left most image depicts a well aligned limb with even force distribution. However, at mid-flex in the central image the lateral collateral ligament is predicted to experience excessive force. Finally, as the knee flexion is increased further, in the right most image, the alignment of the tibia in the coronal plane is expected to deviate from plan above the desired threshold.

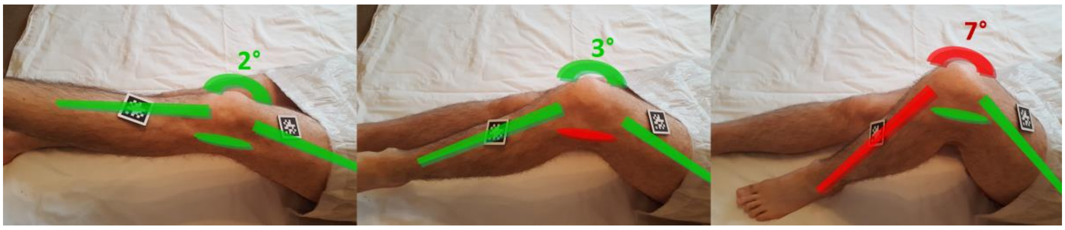


Figure 9.9: Augmented guidance showing tibial alignment in the coronal plane and collateral ligament tension during knee flexion

Resection would then be performed as with Section 8.3.4. The current system would allow both tibial and femoral resection to be performed simultaneously, allowing maximum freedom for the surgeon. Full speed controlled resection originally implemented, as described in Appendix A2.5.3, could also be utilised if desired. The hot-swapping of burrs and large available marker set would also allow surgeons to perform different stages of the resection using the tools they prefer. For example an oscillating saw could easily be incorporated into the tracking system to allow rapid initial resection of the tibial component. This would require either compensation for blade flexion or the use of a tip-only oscillating saw, such as those produced by Stryker.

Once resection was complete, trial components could be fitted and compared with the augmented planned position overlay. Additionally, a marker could be incorporated onto the non-contact edge region of the trial implants. This would allow direct measurement of the physical seated position of the implant. Post-operative range of motion and limb alignment would be compared to the predicted results using augmented overlays. This comparison would also be performed with the finally implant, so ensure any use of bone cement was not detrimental to alignment.

9.4.1.2 Extended Applications

In addition to the core use of the system described above, the inclusion of AR guidance could be hugely beneficial to other fields of orthopaedic surgery, both where accuracy is required or where the traditional view is obstructed. For example, during arthroscopic ligament repair pre-operative images and real-time arthroscopic camera imagery could be augmented on to the surgeons view.

Another significant area of application for the presented system would be in education of surgical practice. As the system uses head-mounted cameras to provide tracking, the view of the surgeon could be recorded or broadcasted in real-time. This opens the possibility of several educational platforms. Firstly, a surgeon could request intra-operative advice from a remote colleague. The remote colleague would receive the surgeon’s view and be able to offer advice on how to proceed. This could provide an additional safety net for junior surgeons during their early training. Additionally, the recorded procedure could be reviewed post-operatively to enhance training by highlighting areas of improvement.

Alternatively, surgical procedures may be shown in classrooms and presented as case studies either as they are being performed or from pre-recorded footage. Due to the tracking information gained, the surgery could be re-lived virtually using haptic feedback devices, thereby allowing surgeons to practice on actual patient data. Alternatively, if CT models are available, 3D printing could be utilised to produce physical models of the limb. The surgery could then be repeated, with augmented information from the actual procedure to improve training.

Finally, with an increase risk to both surgeons and healthcare providers of litigation, the data captured by the proposed system could provide invaluable evidence to protect against false claims.

9.5 Conclusion

In conclusion, the work presented within this thesis has shown the potential of AR within the field of guided UKA. A head-mounted combined display and tracking system and associated control software were implemented that were able to produce sub-millimetre accuracy during both probing and controlled resection. Following the optimisations considered this accuracy is expected to be observed across a full practical volume for the system. This would, therefore, offer a system that is able to provide truly intuitive guidance of the UKA procedure, its training, and orthopaedic surgery in general.

Bibliography

Abawi, D.F., Bienwald, J. & Dorner, R., 2004. Accuracy in Optical Tracking with Fiducial Markers: An Accuracy Function for ARToolKit. In *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE, pp. 260–261.

Abe, Y. et al., 2013. A novel 3D guidance system using augmented reality for percutaneous vertebroplasty: technical note. *Journal of neurosurgery. Spine*, 19(4), pp.492–501.

Acar, C. & Shkel, A., 2008. Fundamentals of micromachined gyroscopes. In *MEMS Vibratory Gyroscopes: Structural Approaches to Improve Robustness*. Springer, pp. 17–43.

Aglietti, P. et al., 2005. Comparison of Mobile-Bearing and Fixed-Bearing Total Knee Arthroplasty. *The Journal of Arthroplasty*, 20(2), pp.145–153.

Aigner, T. et al., 2004. Aging theories of primary osteoarthritis: from epidemiology to molecular biology. *Rejuvenation research*, 7(2), pp.134–45.

Andrianakos, A.A. et al., 2006. Prevalence of symptomatic knee, hand, and hip osteoarthritis in Greece. The ESORDIG study. *The Journal of rheumatology*, 33(12), pp.2507–13.

Argenson, J.-N.A. & Flecher, X., 2004. Minimally invasive unicompartmental knee arthroplasty. *The Knee*, 11(5), pp.341–7.

Armenise, M.N. et al., 2010. Physical effects in gyroscopes. In *Advances in Gyroscope Technologies*. Springer, pp. 9–17.

Ateshian, G.A. et al., 2003. The Role of Osmotic Pressure and Tension-Compression Nonlinearity in the Frictional Response of Articular Cartilage. *Transport in Porous Media*, 50(1-2), pp.5–33.

Athanasiou, K.A. et al., 2013. Structure and function of cartilage. In *Articular Cartilage*. CRC Press, pp. 1–50.

Azuma, R., 1997. A survey of augmented reality. *Presence*, 6(4), pp.355–385.

Azuma, R. et al., 2001. Recent advances in augmented reality. *IEEE Computer Graphics and Applications*, 21(6), pp.34–47.

Van Baar, M.E. et al., 1999. Effectiveness of exercise therapy in patients with osteoarthritis of the hip or knee: a systematic review of randomized clinical trials. *Arthritis and rheumatism*, 42(7), pp.1361–9.

Badiali, G. et al., 2014. Augmented reality as an aid in maxillofacial surgery: validation of a wearable system allowing maxillary repositioning. *Journal of cranio-maxillo-facial surgery : official publication of the European Association for Cranio-Maxillo-Facial Surgery*, 42(8), pp.1970–6.

Bae, D.K. & Song, S.J., 2011. Computer assisted navigation in knee arthroplasty. *Clinics in orthopedic surgery*, 3(4), pp.259–67.

Baker, P.N. et al., 2007. A randomised controlled trial of cemented versus cementless press-fit condylar total knee replacement: 15-year survival analysis. *The Journal of bone and joint surgery. British volume*, 89(12), pp.1608–14.

Banerjee, P.P. et al., 2007. Accuracy of ventriculostomy catheter placement using a head- and hand-tracked high-resolution virtual reality simulator with haptic feedback. *Journal of neurosurgery*, 107(3), pp.515–21.

Bao, M. & Wang, W., 1996. Future of microelectromechanical systems (MEMS). *Sensors and Actuators A: Physical*, 56(1), pp.135–141.

Barnes, M., 2003. Speed measurement transducers. In *Practical Variable Speed Drives and Power Electronics*. Newnes, pp. 248–261.

Barrett, D., Cobb, A. & Bentley, G., 1991. Joint proprioception in normal, osteoarthritic and replaced knees. *J Bone Joint Surg Br*, 73-B(1), pp.53–56.

Bay, H. et al., 2008. SURF:Speeded Up Robust Features. *Computer Vision and Image Understanding*, 110(3), pp.346–359.

Beasley, R. a., 2012. Medical Robots: Current Systems and Research Directions. *Journal of Robotics*, 2012, pp.1–14.

Bellemans, J., Ries, M.D. & Victor, J., 2005. *Total Knee Arthroplasty: A Guide To Get Better Performance*, Heidelberg: Springer.

Bennell, K.L. & Hinman, R.S., 2011. A review of the clinical evidence for exercise in osteoarthritis of the hip and knee. *Journal of science and medicine in sport / Sports Medicine Australia*, 14(1), pp.4–9.

Berend, K.R. & Lombardi, A. V, 2009. Distal femoral replacement in nontumor cases with severe bone loss and instability. *Clinical orthopaedics and related research*, 467(2), pp.485–92.

Berger, R.A. et al., 2005. Results of unicompartmental knee arthroplasty at a minimum of ten years of follow-up. *The Journal of bone and joint surgery. American volume*, 87(5), pp.999–1006.

Van den Bergh, M. & Van Gool, L., 2011. Combining RGB and ToF cameras for real-time 3D hand gesture interaction. In *2011 IEEE Workshop on Applications of Computer Vision (WACV)*. IEEE, pp. 66–72.

Bergström, P. & Edlund, O., 2014. Robust registration of point sets using iteratively reweighted least squares. *Computational Optimization and Applications*, 58(3), pp.543–561.

Berthold, K. & Richter, D., 2009. Simultaneous Stereo-Optical Navigation of Medical Instruments for Brachytherapy. *Federation for Medical*, 22, pp.911–913.

Bhan, S. et al., 2005. A comparison of fixed-bearing and mobile-bearing total knee arthroplasty at a minimum follow-up of 4.5 years. *The Journal of bone and joint surgery. American volume*, 87(10), pp.2290–6.

Bhosale, A.M. & Richardson, J.B., 2008. Articular cartilage: structure, injuries and review of management. *British medical bulletin*, 87, pp.77–95.

Bichlmeier, C. et al., 2007. Contextual Anatomic Mimesis Hybrid In-Situ Visualization Method for Improving Multi-Sensory Depth Perception in Medical Augmented Reality. In *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE, pp. 1–10.

Bistolfi, A. et al., 2013. Comparison of fixed and mobile-bearing total knee arthroplasty at a mean follow-up of 116 months. *The Journal of bone and joint surgery. American volume*, 95(12), p.e83.

Blagojevic, M. et al., 2010. Risk factors for onset of osteoarthritis of the knee in older adults: a systematic review and meta-analysis. *Osteoarthritis and cartilage / OARS, Osteoarthritis Research Society*, 18(1), pp.24–33.

Blood, E.B., 1989. Device for quantitatively measuring the relative position and orientation of two bodies in the presence of metals utilizing direct current magnetic fields.

Bloomfield, P.M. et al., 2003. The design and implementation of a motion correction scheme for neurological PET. *Physics in medicine and biology*, 48(8), pp.959–78.

Blum, J.R., Bouchard, M. & Cooperstock, J.R., 2012. What's around Me? Spatialized Audio Augmented Reality for Blind Users with a Smartphone. In A. Puiatti & T. Gu, eds. *Mobile and Ubiquitous Systems: Computing, Networking, and Services*. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 49–62.

Blum, T. et al., 2009. Advanced training methods using an Augmented Reality ultrasound simulator. In *2009 8th IEEE International Symposium on Mixed and Augmented Reality*. IEEE, pp. 177–178.

Bolognesi, M.P., Fang, D.M., et al., 2009. Coronal Alignment in Total Knee Arthroplasty. *The Journal of Arthroplasty*, 24(6), pp.39–43.

Bolognesi, M.P., Namba, R.S., et al., 2009. Outcomes of Routine Use of Antibiotic-Loaded Cement in Primary Total Knee Arthroplasty. *The Journal of Arthroplasty*, 24(6), pp.44–47.

Bonnin, M. et al., 2011. *The knee joint: Surgical techniques and strategies*, Springer.

Bonnin, M. & Chambat, P., 2008. *Osteoarthritis of the knee: Surgical treatment*, Springer.

Bonutti, P., Dethmers, D. & Stiehl, J.B., 2008. Case report : femoral shaft fracture resulting from femoral tracker placement in navigated TKA. *Clinical orthopaedics and related research*, 466(6), pp.1499–502.

Börner, M., Wiesel, U. & Ditzen, W., 2004. Clinical Experiences with ROBODOC and the Duracon Total Knee. In J. B. Stiehl, W. H. Konermann, & R. G. Haaker, eds. *Navigation and Robotics in Total Joint and Spine Surgery*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 362–366.

Bradski, G. & Kaehler, A., 2008. *Learning OpenCV: Computer Vision with the OpenCV Library*,

Brin, Y.S. et al., 2011. Imageless computer assisted versus conventional total knee replacement. A Bayesian meta-analysis of 23 comparative studies. *International orthopaedics*, 35(3), pp.331–9.

Brown, T.G. et al., 2001. Strap-down microelectromechanical (MEMS) sensors for high-g munition applications. *IEEE Transactions on Magnetics*, 37(1), pp.336–342.

Brys, D.A. et al., 1991. A comparison of intramedullary and extramedullary alignment systems for tibial component placement in total knee arthroplasty. *Clinical orthopaedics and related research*, (263), pp.175–9.

Buckwalter, J. & Mankin, H., 1997. Articular cartilage II. Degeneration and osteoarthrosis, repair, regeneration and transplantation. *J Bone Joint Surg.*, 79(4), pp.612–632.

Buckwalter, J.A., Saltzman, C. & Brown, T., 2004. The impact of osteoarthritis: implications for research. *Clinical orthopaedics and related research*, (427 Suppl), pp.S6–15.

Bull, D., 2014. *Communicating Pictures: A Course in Image and Video Coding* 1st ed., Oxford: Elsevier Ltd.

Cakmakci, O. & Rolland, J., 2006. Head-Worn Displays: A Review. *J. Display Technol.*, 2(3), pp.199–216.

Carney, S.L. & Muir, H., 1988. The structure and function of cartilage proteoglycans. *Physiological reviews*, 68(3), pp.858–910.

Caudell, T.P. & Mizell, D.W., 1992. Augmented reality: an application of heads-up display technology to manual manufacturing processes. *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences*, 2, pp.659–669.

Chabay, R.W. & Sherwood, B.A., 2011. Electric and magnetic interactions. In *Matter and Interactions*. John Wiley & Sons, pp. 553–1041.

Chaljub, G. et al., 2001. Projectile cylinder accidents resulting from the presence of ferromagnetic nitrous oxide or oxygen tanks in the MR suite. *AJR. American journal of roentgenology*, 177(1), pp.27–30.

Chandler, J. & Fryer, J., 2013. Autodesk 123D catch: how accurate is it. *Geomatics World*.

Chauhan, S.K. et al., 2004. Computer-assisted total knee replacement. A controlled cadaver study using a multi-parameter quantitative CT assessment of alignment (the Perth CT Protocol). *The Journal of bone and joint surgery. British volume*, 86(6), pp.818–23.

Chen, X. et al., 2013. Precise X-ray and video overlay for augmented reality fluoroscopy. *International journal of computer assisted radiology and surgery*, 8(1), pp.29–38.

Cheung, C.L. et al., 2010. Fused video and ultrasound images for minimally invasive partial nephrectomy: a phantom study. *Medical image computing and computer-assisted intervention : MICCAI ... International Conference on Medical Image Computing and Computer-Assisted Intervention*, 13(Pt 3), pp.408–15.

Chockalingam, S. & Scott, G., 2000. The outcome of cemented vs. cementless fixation of a femoral component in total knee replacement (TKR) with the identification of radiological signs for the prediction of failure. *The Knee*, 7(4), pp.233–238.

Choi, J. & Kim, G.J., 2012. Usability of one-handed interaction methods for handheld projection-based augmented reality. *Personal and Ubiquitous Computing*, 17(2), pp.399–409.

Choong, P.F., Dowsey, M.M. & Stoney, J.D., 2009. Does Accurate Anatomical Alignment Result in Better Function and Quality of Life? Comparing Conventional and Computer-Assisted Total Knee Arthroplasty. *The Journal of Arthroplasty*, 24(4), pp.560–569.

Christensen, K. et al., 2009. Ageing populations : the challenges ahead. *The Lancet*, 374(9696), pp.1196–1208.

Cobb, J. et al., 2006. Hands-on robotic unicompartmental knee replacement: a prospective, randomised controlled study of the acrobot system. *The Journal of bone and joint surgery. British volume*, 88(2), pp.188–97.

Cobb, J. & Pearle, A., 2013. Unicompartmental Knee Arthroplasty: Robotics. In F. Catani & S. Zaffagnini, eds. *Knee Surgery using Computer Assisted Surgery and Robotics*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 95–104.

Coggon, D. et al., 2001. Knee osteoarthritis and obesity. *International journal of obesity and related metabolic disorders : journal of the International Association for the Study of Obesity*, 25(5), pp.622–7.

Comport, A.I., Marchand, É. & Chaumette, F., 2003. A real-time tracker for markerless augmented reality. , p.36.

Cooke, T.D., Saunders, G.A.B. & Siu, D., 1986. Orthopaedic bone cutting jig and alignment device. , p.6.

Coon, T.M., Driscoll, M.D. & Conditt, M.A., 2010a. EARLY CLINICAL SUCCESS OF NOVEL TACTILE GUIDED UKA TECHNIQUE. *Journal of Bone & Joint Surgery, British Volume*, 92-B(SUPP I), p.116.

Coon, T.M., Driscoll, M.D. & Conditt, M.A., 2010b. ROBOTICALLY ASSISTED UKA IS MORE ACCURATE THAN MANUALLY INSTRUMENTED UKA. *Journal of Bone & Joint Surgery, British Volume*, 92-B(SUPP I), p.157.

Darling, E.M. & Athanasiou, K. a., 2003. Biomechanical Strategies for Articular Cartilage Regeneration. *Annals of Biomedical Engineering*, 31(9), pp.1114–1124.

Darling, E.M. & Athanasiou, K.A., 2003. Articular cartilage bioreactors and bioprocesses. *Tissue engineering*, 9(1), pp.9–26.

Davey, S.M. et al., 2012. Surgeon opinion on new technologies in orthopaedic surgery. *Journal of Medical Engineering & Technology*.

Davis, J.A., Hogan, C. & Dayton, M., 2015. Postoperative Coronal Alignment After Total Knee Arthroplasty: Does Tailoring the Femoral Valgus Cut Angle Really Matter? *The Journal of arthroplasty*, 30(8), pp.1444–8.

Decking, J. et al., 2004. Robotic total knee arthroplasty: the accuracy of CT-based component placement. *Acta orthopaedica Scandinavica*, 75(5), pp.573–9.

DeGroot, J. et al., 1999. Age-related decrease in proteoglycan synthesis of human articular chondrocytes: the role of nonenzymatic glycation. *Arthritis and rheumatism*, 42(5), pp.1003–9.

Dennis, D.A. et al., 1993. Intramedullary versus extramedullary tibial alignment systems in total knee arthroplasty. *The Journal of Arthroplasty*, 8(1), pp.43–47.

Dixon, B.J. et al., 2014. Inattentional blindness increased with augmented reality surgical navigation. *American journal of rhinology & allergy*, 28(5), pp.433–7.

Dixon, B.J. et al., 2013. Surgeons blinded by enhanced navigation: the effect of augmented reality on attention. *Surgical endoscopy*, 27(2), pp.454–61.

Dodgson, N.A., 2004. Variation and extrema of human interpupillary distance. In A. J. Woods et al., eds. *Electronic Imaging 2004*. International Society for Optics and Photonics, pp. 36–46.

Dorrington, A.A., Payne, A.D. & Cree, M.J., 2010. An evaluation of time-of-flight range cameras for close range metrology applications. *Photogrammetry, remote sensing and spatial information science*, 38(5), pp.201–206.

Douglas, D.H. & Peucker, T.K., 1973. Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2), pp.112–122.

Duffy, G.P., Berry, D.J. & Rand, J.A., 1998. Cement Versus Cementless Fixation In Total Knee Arthroplasty. *Clinical Orthopaedics & Related Research*, 356, pp.66–72.

Duh, H.B.-L. & Billinghurst, M., 2008. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. In *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*. IEEE, pp. 193–202.

Dwarakanath, D. et al., 2014. Study the Effects of Camera Misalignment on 3D Measurements for Efficient Design of Vision-Based Inspection Systems. *Artificial Intelligence: Methods and Applications*, 8445, pp.150–163.

Elfring, R., de la Fuente, M. & Radermacher, K., 2010. Assessment of optical localizer accuracy for computer aided surgery systems. *Computer aided surgery : official journal of the International Society for Computer Aided Surgery*, 15(1-3), pp.1–12.

El-Sana, J. & Billinghurst, M., 2013. Markerless 3D gesture-based interaction for handheld Augmented Reality interfaces. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, pp. 1–6.

Engesaeter, L.B. et al., 2003. Antibiotic prophylaxis in total hip arthroplasty: effects of antibiotic prophylaxis systemically and in bone cement on the revision rate of 22,170 primary hip replacements followed 0-14 years in the Norwegian Arthroplasty Register. *Acta orthopaedica Scandinavica*, 74(6), pp.644–51.

Engh, G.A. & Petersen, T.L., 1990. Comparative experience with intramedullary and extramedullary alignment in total knee arthroplasty. *The Journal of Arthroplasty*, 5(1), pp.1–8.

Epinette, J.-A. et al., 2012. Unicompartmental knee arthroplasty modes of failure: wear is not the main reason for failure: a multicentre study of 418 failed knees. *Orthopaedics & traumatology, surgery & research : OTSR*, 98(6 Suppl), pp.S124–30.

Erickson, M.S., Bauer, J.J. & Hayes, W.C., 2013. The Accuracy of Photo-Based Three-Dimensional Scanning for Collision Reconstruction Using 123D Catch. *SAE*, p.10.

Fan, Z., Wang, J. & Liao, H., 2014. A Spatial Position Measurement System Using Integral Photography Based 3D Image Markers. *1st Global Conference on Biomedical Engineering & 9th Asian-Pacific Conference on Medical and Biological Engineering*, 47, pp.150–153.

Fang, D. & Ritter, M.A., 2009. Malalignment: Forewarned is Forearmed. *Orthopedics*, 32(9).

Fargie, W., Friend, J. & Wilson, J.L., 1988. Tibial cutting jig. , p.5.

Fassbender, H.G., 1987. Role of chondrocytes in the development of osteoarthritis. *The American Journal of Medicine*, 83(5), pp.17–24.

Felson, D.T., 1988. Obesity and Knee Osteoarthritis. *Annals of Internal Medicine*, 109(1), p.18.

Felson, D.T., 2000. Osteoarthritis: New Insights. Part 1: The Disease and Its Risk Factors. *Annals of Internal Medicine*, 133(8), p.635.

Felson, D.T. et al., 2005. Osteophytes and progression of knee osteoarthritis. *Rheumatology (Oxford, England)*, 44(1), pp.100–4.

Felson, D.T. et al., 1987. The prevalence of knee osteoarthritis in the elderly. the framingham osteoarthritis study. *Arthritis & Rheumatism*, 30(8), pp.914–918.

Fernandez, E. et al., 2002. Mapping of the human visual cortex using image-guided transcranial magnetic stimulation. *Brain Research Protocols*, 10(2), pp.115–124.

Fischler, M.A. & Bolles, R.C., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *ACM*, 24, pp.381–395.

Fleute, M. & Lavallée, S., 1998. Building a complete surface model from sparse data using statistical shape models: Application to computer assisted knee surgery. In W. M. Wells, A. Colchester, & S. Delp, eds. *Medical Image Computing and Computer-Assisted Interventation*. Lecture Notes in Computer Science. Berlin/Heidelberg: Springer-Verlag, pp. 879–887.

Forsyth, D.A. & Ponce, J., 2011. Stereopsis. In *Computer Vision: A Modern Approach*. Pearson Education, Limited, pp. 229–252.

Foxlin, E., 2005. Pedestrian Tracking with Shoe-Mounted Inertial Sensors. *IEEE Computer Graphics and Applications*, 25(6), pp.38–46.

Foxlin, E. & Harrington, M., 2000. WearTrack: a self-referenced head and hand tracker for wearable computers and portable VR. In *Digest of Papers. Fourth International Symposium on Wearable Computers*. IEEE Comput. Soc, pp. 155–162.

Foxlin, E., Harrington, M. & Pfeifer, G., 1998. Constellation: A Wide-Range Wireless MotionTracking System for Augmented Reality and Virtual Set Applications. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques - SIGGRAPH '98*. New York, New York, USA: ACM Press, pp. 371–378.

Fraenkel, L. et al., 2004. Treatment options in knee osteoarthritis: the patient's perspective. *Archives of internal medicine*, 164(12), pp.1299–304.

Garvin, K.L., Barrera, O.A. & Haider, H., 2011. NAVIGATION IN TKA: ARGUMENTS AGAINST THE USE OF NAVIGATION. *Journal of Bone & Joint Surgery, British Volume*, 93-B(SUPP IV), pp.456–457.

Geller, J.A., Yoon, R.S. & Macaulay, W., 2008. Unicompartmental knee arthroplasty: a controversial history and a rationale for contemporary resurgence. *The journal of knee surgery*, 21(1), pp.7–14.

Gilson, S.J., Fitzgibbon, A.W. & Glennerster, A., 2006. Quantitative analysis of accuracy of an inertial/acoustic 6DOF tracking system in motion. *Journal of neuroscience methods*, 154(1-2), pp.175–82.

Glossop, N.D., 2009. Advantages of optical compared with electromagnetic tracking. *The Journal of bone and joint surgery. American volume*, 91 Suppl 1(Supplement_1), pp.23–8.

Goldring, M.B., 2000. The role of the chondrocyte in osteoarthritis. *Arthritis and rheumatism*, 43(9), pp.1916–26.

Guccione, A.A. et al., 1994. The effects of specific medical conditions on the functional limitations of elders in the Framingham Study. *American journal of public health*, 84(3), pp.351–8.

Hagag, B. et al., 2011. RIO: Robotic-Arm Interactive Orthopedic System MAKOplasty: User Interactive Haptic Orthopedic Robotics. In J. Rosen, B. Hannaford, & R. M. Satava, eds. *Surgical Robotics*. Boston, MA: Springer US, pp. 219–246.

Hannah, M.J., 1974. *Computer Matching of Areas in Stereo Images*. Stanford University.

Hannan, M.T., Felson, D.T. & Pincus, T., 2000. Analysis of the discordance between radiographic changes and knee pain in osteoarthritis of the knee. *The Journal of rheumatology*, 27(6), pp.1513–7.

Hanssen, A. & Scott, W.N., 2009. *Operative technique: Total knee replacment*, Elsevier Health Sciences.

Hartley, R. & Zisserman, A., 2003. Epipolar Geometry and the Fundamental Maxtrix. In *Multiple View Geometry in Computer Vision*. Cambridge University Press, pp. 239–326.

Harvie, P., Sloan, K. & Beaver, R.J., 2012. Computer Navigation vs Conventional Total Knee Arthroplasty. *The Journal of Arthroplasty*, 27(5), pp.667–672.e1.

Hashimoto, S. et al., 2002. Development and regulation of osteophyte formation during experimental osteoarthritis. *Osteoarthritis and Cartilage*, 10(3), pp.180–187.

Hayward, K., 2009. Graphics Runner: Volume Rendering 101. Available at: http://graphicsrunner.blogspot.co.uk/2009/01/volume-rendering-101.html [Accessed June 9, 2015].

Healy, W.L. et al., 2002. Impact of cost reduction programs on short-term patient outcome and hospital cost of total knee arthroplasty. *The Journal of bone and joint surgery. American volume*, 84-A(3), pp.348–53.

Henckel, J. et al., 2006. Very low-dose computed tomography for planning and outcome measurement in knee replacement. The imperial knee protocol. *The Journal of bone and joint surgery. British volume*, 88(11), pp.1513–8.

Hernigou, P. & Deschamps, G., 2004. Alignment influences wear in the knee after medial unicompartmental arthroplasty. *Clinical orthopaedics and related research*, (423), pp.161–5.

Hilliges, O. et al., 2012. HoloDesk. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12*. New York, New York, USA: ACM Press, p. 2421.

Himanen, A.-K. et al., 2005. Survival of the AGC total knee arthroplasty is similar for arthrosis and rheumatoid arthritis. Finnish Arthroplasty Register report on 8,467 operations carried out between 1985 and 1999. *Acta orthopaedica*, 76(1), pp.85–8.

Hiniduma, S., Gamage, U. & Lasenby, J., 2002. New least squares solutions for estimating the average centre of rotation and the axis of rotation. *Journal of Biomechanics*, 35(1), pp.87–93.

Hirschmuller, H., 2005. Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. IEEE, pp. 807–814.

Hofer, M. et al., 2009. Navigated Control for Skull Base Surgery: Part of the Surgical Cockpit. *Skull Base*, 19(01), p.A235.

Howarth, P.A. & Costello, P.J., 1997. The occurrence of virtual simulation sickness symptoms when an HMD was used as a personal viewing system. *Displays*, 18(2), pp.107–116.

Howell, R.E. et al., 2015. Unicompartmental Knee Arthroplasty: Does a Selection Bias Exist? *The Journal of arthroplasty*.

Hsieh, Y.C., McKeown, D.M. & Perlant, F.P., 1992. Performance evaluation of scene registration and stereo matching for cartographic feature extraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), pp.214–238.

Huang, B.W. et al., 2012. Dynamic characteristics of a hollow femur. *Life Science Journal*, 9(1), pp.723–726.

Huang, C.-H. et al., 2002. Osteolysis in Failed Total Knee Arthroplasty: A Comparison of Mobile-Bearing and Fixed-Bearing Knees. *The Journal of Bone & Joint Surgery*, 84(12), pp.2224–2229.

Huang, C.-H., Liau, J.-J. & Cheng, C.-K., 2007. Fixed or mobile-bearing total knee arthroplasty. *Journal of orthopaedic surgery and research*, 2(1), p.1.

Hughes, J.S. et al., 2005. Review of the radiation exposure of the UK population. *Journal of radiological protection : official journal of the Society for Radiological Protection*, 25(4), pp.493–6.

Insall, J.N. et al., 1989. Rationale of the Knee Society clinical rating system. *Clinical orthopaedics and related research*, (248), pp.13–4.

Isaac, S.M. et al., 2007. Does arthroplasty type influence knee joint proprioception? A longitudinal prospective study comparing total and unicompartmental arthroplasty. *The Knee*, 14(3), pp.212–217.

Jackson, D.W., Simon, T.M. & Aberman, H.M., 2001. Symptomatic articular cartilage degeneration: the impact in the new millennium. *Clinical orthopaedics and related research*, (391 Suppl), pp.S14–25.

Jacobs, W. et al., 2004. Mobile bearing vs fixed bearing prostheses for total knee arthroplasty for post-operative functional status in patients with osteoarthritis and rheumatoid arthritis. *The Cochrane database of systematic reviews*, (2), p.CD003130.

Jacquot, F., 2007. Total knee prosthesis radiograph.

Jaramaz, B., Hafez, M.A. & DiGioia, A.M., 2006. Computer-Assisted Orthopaedic Surgery. *Proceedings of the IEEE*, 94(9), pp.1689–1695.

Jaramaz, B. & Nikou, C., 2012. Precision freehand sculpting for unicondylar knee replacement: design and experimental validation. *Biomedizinische Technik. Biomedical engineering*, 57(4), pp.293–9.

Jaramaz, B., Picard, F. & Gregori, A., 2013. EVALUATING JOINT LAXITY IN UKR USING NAVIO-PFS. *Bone & Joint Journal Orthopaedic Proceedings Supplement*, 95-B(SUPP 28), p.74.

Jeer, P.J.S., Keene, G.C.R. & Gill, P., 2004. Unicompartmental knee arthroplasty: an intermediate report of survivorship after the introduction of a new system with analysis of failures. *The Knee*, 11(5), pp.369–74.

Jeffery, R., Morris, R. & Denham, R., 1991. Coronal alignment after total knee replacement. *J Bone Joint Surg Br*, 73-B(5), pp.709–714.

Jensenius, A.R. et al., 2012. A Study of the Noise-Level in Two Infrared Marker-Based Motion Capture Systems. In *Proceedings of the 9th Sound and Music Computing Conference*. pp. 258–263.

Jonna, V. & Tria, A., 2009. Unicompartmental knee arthroplasty. In T. Brown et al., eds. *Arthritis & Arthroplasty: The knee*. Elsevier Health Sciences, pp. 62–69.

Jorn, L.P., Lindstrand, A. & Toksvig-Larsen, S., 2009. Tourniquet release for hemostasis increases bleeding: A randomized study of 77 knee replacements.

Josefsson, T., 2002. Method and system for motion analysis. , p.19.

Jung, K.A. et al., 2010. Accuracy of implantation during computer-assisted minimally invasive Oxford unicompartmental knee arthroplasty. *The Knee*, 17(6), pp.387–391.

Kahlmann, T., Remondino, F. & Ingensand, H., 2006. Calibration for increased accuracy of the range imaging camera Swissranger. *Image Engineering and Vision ...*, 36(5), pp.136–141.

Kalfas, I.H., 2001. Image-guided spinal navigation: application to spinal metastases. *Neurosurgical focus*, 11(6), p.e5.

Kalman, R.E., 1960. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1), pp.35–45.

Kanade, T. et al., 1995. Development of a Video-Rate Stereo Machine. In *Proc. of International Robotics and Systems Conference (IROS '95), Human Robot Interaction and Cooperative Robots*. pp. 95–100.

Kang, X. et al., 2014. Towards a Clinical Stereoscopic Augmented Reality System for Laparoscopic Surgery. *Clinical Image-Based Procedures*, 8361(1), pp.48–56.

Kao, F.-C. et al., 2009. Surgical planning and procedures for difficult total knee arthroplasty. *Orthopedics*, 32(11), p.810.

Kaplan, R.J., 2008. Total knee replacment. In W. R. Frontera, J. K. Silver, & T. D. Rizzo, eds. *Essentials of Physical Medicine and Rehabilitation: Musculoskeletal Disorders, Pain, and Rehabilitation*. Elsevier Health Sciences, pp. 399–406.

Karia, M. et al., 2013. ROBOTIC-ASSISTANCE ENABLES INEXPERIENCED SURGEONS TO PERFORM UNICOMPARTMENTAL KNEE ARTHROPLASTIES ACCURATELY ON THEIR FIRST ATTEMPT. *Bone & Joint Journal Orthopaedic Proceedings Supplement*, 95-B(SUPP 28), p.28.

Kasodekar, V.B., Yeo, S.J. & Othman, S., 2006. Clinical outcome of unicompartmental knee arthroplasty and influence of alignment on prosthesis survival rate. *Singapore medical journal*, 47(9), pp.796–802.

Kato, H. & Billinghurst, M., 1999. Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*. IEEE Comput. Soc, pp. 85–94.

Keene, G., Simpson, D. & Kalairajah, Y., 2006. Limb alignment in computer-assisted minimally-invasive unicompartmental knee replacement. *The Journal of bone and joint surgery. British volume*, 88(1), pp.44–8.

Khadem, R. et al., 2000. Comparative tracking error analysis of five different optical tracking systems. *Computer aided surgery*, 5(2), pp.98–107.

Khamene, A. et al., 2003. An Augmented Reality system for MRI-guided needle biopsies. *Studies in health technology and informatics*, 94, pp.151–7.

Kilgard, M., 1996. The OpenGL Utility Toolkit (GLUT).

Kim, Y.-H., Yoon, S.-H. & Kim, J.-S., 2007. The long-term results of simultaneous fixed-bearing and mobile-bearing total knee replacements performed in the same patient. *The Journal of bone and joint surgery. British volume*, 89(10), pp.1317–23.

Klatt, B.A., Lonner, J.H. & Kerr, G.J., 2012. Robotically Assisted Unicompartmental Knee Arthroplasty. *Operative Techniques in Orthopaedics*, 22(4), pp.182–188.

Knappe, P. et al., 2003. Position control of a surgical robot by a navigation system. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*. IEEE, pp. 3350–3354.

Kon, E. et al., 2012. Non-surgical management of early knee osteoarthritis. *Knee surgery, sports traumatology, arthroscopy : official journal of the ESSKA*, 20(3), pp.436–49.

Koschan, A., Rodehorst, V. & Spiller, K., 1996. Color stereo vision using hierarchical block matching and active color illumination. In *Proceedings of 13th International Conference on Pattern Recognition*. IEEE, pp. 835–839 vol.1.

Koskinen, E. et al., 2009. Comparison of survival and cost-effectiveness between unicondylar arthroplasty and total knee arthroplasty in patients with primary osteoarthritis: A follow-up study of 50,493 knee replacements from the Finnish Arthroplasty Register. *Acta Orthopaedica*, 79(4), pp.499–507.

Krackow, K.A. et al., 1999. A new technique for determining proper mechanical axis alignment during total knee arthroplasty: progress toward computer-assisted TKA. *Orthopedics*, 22(7), pp.698–702.

Krevelen, D. Van & Poelman, R., 2010. A survey of augmented reality technologies, applications and limitations. *... Journal of Virtual Reality*, 9(2), pp.1–20.

Lang, J.E. et al., 2011. Robotic systems in orthopaedic surgery. *The Journal of bone and joint surgery. British volume*, 93(10), pp.1296–9.

Lanman, D. & Luebke, D., 2013. Near-eye light field displays. *ACM Transactions on Graphics*, 32(6), pp.1–10.

Lawrence, R. & Felson, D., 2008. Estimates of the prevalence of arthritis and other rheumatic conditions in the United States: Part II. *Arthritis & ...*, 58(1), pp.26–35.

Lee, S., Choi, O. & Horaud, R., 2012. *Time-of-Flight Cameras: Principles, Methods and Applications*, Springer.

Lee, T. & Hollerer, T., 2007. Handy AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking. In *2007 11th IEEE International Symposium on Wearable Computers*. IEEE, pp. 1–8.

Lee, T. & Höllerer, T., 2009. Multithreaded hybrid feature tracking for markerless augmented reality. *IEEE transactions on visualization and computer graphics*, 15(3), pp.355–68.

Levenberg, K., 1944. A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Applied Mathmatics*, II(2), pp.164 – 168.

Li, C.-H. et al., 2008. Periprosthetic femoral supracondylar fracture after total knee arthroplasty with navigation system. *The Journal of arthroplasty*, 23(2), pp.304–7.

Lide, D.R., 2004. *CRC Handbook of Chemistry and Physics* 85th ed., CRC Press.

Lim, J.W. et al., 2014. Oxford unicompartmental knee arthroplasty versus age and gender matched total knee arthroplasty - functional outcome and survivorship analysis. *The Journal of arthroplasty*, 29(9), pp.1779–83.

Liming, C. et al., 2012. Sensor-Based Activity Recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6), pp.790–808.

Lin, Y.-K. et al., 2015. A novel dental implant guided surgery based on integration of surgical template and augmented reality. *Clinical implant dentistry and related research*, 17(3), pp.543–53.

Liow, R.Y. & Murray, D.W., 1997. Which primary total knee replacement? A review of currently available TKR in the United Kingdom. *Annals of the Royal College of Surgeons of England*, 79(5), pp.335–40.

Litwic, A. et al., 2013. Epidemiology and burden of osteoarthritis. *British medical bulletin*, 105, pp.185–99.

Liu, H., Holt, C. & Evans, S., 2007. *Accuracy and repeatability of an optical motion analysis system for measuring small deformations of biological tissues*, Liu, S., Hua, H. & Cheng, D., 2010. A novel prototype for an optical see-through head-mounted display with addressable focus cues. *IEEE transactions on visualization and computer graphics*, 16(3), pp.381–93.

Loeser, R., 2006. Molecular mechanisms of cartilage destruction: mechanics, inflammatory mediators, and aging collide. *Arthritis & Rheumatism*, 54(5), pp.1357–1360.

Lombardi, A. V et al., 2009. Is recovery faster for mobile-bearing unicompartmental than total knee arthroplasty? *Clinical orthopaedics and related research*, 467(6), pp.1450–7.

Lombardi, A. V, 2004. Total Knee Arthroplasty. In D. E. Brown & R. D. Neumann, eds. *Orthopedic Secrets*. Elsevier Health Sciences, pp. 356–361.

London, N.J., Miller, L.E. & Block, J.E., 2011. Clinical and economic consequences of the treatment gap in knee osteoarthritis management. *Medical Hypotheses*, 76(6), pp.887–892.

Longstaff, L.M. et al., 2009. Good Alignment After Total Knee Arthroplasty Leads to Faster Rehabilitation and Better Function. *The Journal of Arthroplasty*, 24(4), pp.570–578.

Lonner, J.H. et al., 2015. High degree of accuracy of a novel image-free handheld robot for unicondylar knee arthroplasty in a cadaveric study. *Clinical orthopaedics and related research*, 473(1), pp.206–12.

Lonner, J.H., John, T.K. & Conditt, M.A., 2010. Robotic arm-assisted UKA improves tibial component alignment: a pilot study. *Clinical orthopaedics and related research*, 468(1), pp.141–6.

Lutz & Stefan, 2012. PMDMDK Multithreading.

Lutz, W. & K C, S., 2010. Dimensions of global population projections: what do we know about future population trends and structures? *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 365(1554), pp.2779–2791.

Lützner, J. et al., 2009. Surgical options for patients with osteoarthritis of the knee. *Nature reviews. Rheumatology*, 5(6), pp.309–16.

Ma, B. et al., 2009. Computer-assisted FluoroGuide navigation of unicompartmental knee arthroplasty. *Canadian journal of surgery. Journal canadien de chirurgie*, 52(5), pp.379–85.

Maestro, A. et al., 1998. Influence of intramedullary versus extramedullary alignment guides on final total knee arthroplasty component position. *The Journal of Arthroplasty*, 13(5), pp.552–558.

Magnus Manske, 2012. Knieprothese vom Typ Schlittenprothese (Uni-Knie).

MAKO, S.C., 2013. *MAKO Surgical Corp. Reports Operating Results for the Third Quarter 2013 (NASDAQ:MAKO),*

Mankin, H.J. & Lippiello, L., 1979. Acute Metabolic Effects of Nitrogen Mustard and Thiotepa on Rabbit Articular Cartilage and Synovium. *Arthritis & Rheumatism*, 22(6), pp.579–585.

Marmor, L., 1979. Marmor modular knee in unicompartmental disease. Minimum four-year follow-up. *The Journal of bone and joint surgery. American volume*, 61(3), pp.347–53.

Marquardt, D.W., 1963. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2), pp.431–441.

Mascott, C.R., 2005. Comparison of magnetic tracking and optical tracking by simultaneous use of two independent frameless stereotactic systems. *Neurosurgery*, 57(4 Suppl), pp.295–301; discussion 295–301.

Mattess, T. & Decking, R., 2014. The Role of Navigation in Hip Arthroplasty. In *European Surgical Orthopaedics and Traumatology*. pp. 2483–2493.

Mattoccia, S., 2010. Improving the accuracy of fast dense stereo correspondence algorithms by enforcing local consistency of disparity fields. In *3D Data Processing, Visualization and Transmission*.

Matyas, J.R. et al., 2002. Regional quantification of cartilage type II collagen and aggrecan messenger RNA in joints with early experimental osteoarthritis. *Arthritis and rheumatism*, 46(6), pp.1536–43.

McKeever, D.C., 1960. Tibial plateau prosthesis. *Clinical orthopaedics*, 18, pp.86–95.

Meneghini, R.M. & Sheinkop, M.B., 2004. Unicondylar Arthroplasty. In B. J. Cole & M. M. Malek, eds. *Articular Cartilage Lesions*. New York, NY: Springer New York, pp. 183–192.

Miettinen, S.S.A. et al., 2015. Mid-term results of Oxford phase 3 unicompartmental knee arthroplasties at a small-volume center. *Scandinavian journal of surgery : SJS : official organ for the Finnish Surgical Society and the Scandinavian Surgical Society*, p.1457496915577022.

Mofidi, A. et al., 2012. Assessment of Accuracy of Robotic Assisted Unicompartemental Arthroplasty (Makoplasty). *Journal of Bone & Joint Surgery, British Volume*, 94-B(SUPP XXV), p.150.

Mohamed-Ali, H., 1991. Proteolytic enzymes and the destruction of articular cartilage in arthritis and chronic polyarthritis. *Wiener medizinische Wochenschrift*, 141(4), pp.77–85.

Montgomery, A.J. et al., 2006. Correction of Head Movement on PET Studies: Comparison of Methods. *J. Nucl. Med.*, 47(12), pp.1936–1944.

Mroz, F. & Breckon, T.P., 2012. An empirical comparison of real-time dense stereo approaches for use in the automotive environment. *EURASIP Journal on Image and Video Processing*, 2012(1), p.13.

Mullaji, A.B., Sharma, A. & Marawar, S., 2007. Unicompartmental knee arthroplasty: functional recovery and radiographic results with a minimally invasive technique. *The Journal of arthroplasty*, 22(4 Suppl 1), pp.7–11.

Munoz-Salinas, R., 2012. ArUco: A minimal library for Augmented Reality applications based on OpenCV.

Munzinger, U.K., Boldt, J.G. & Keblish, P.A., 2004. *Primary Knee Arthroplasty*, Heidelberg: Springer.

Murray, D.W., Goodfellow, J.W. & O'Connor, J.J., 1998. The Oxford medial unicompartmental arthroplasty: a ten-year survival study. *The Journal of bone and joint surgery. British volume*, 80(6), pp.983–9.

Mutto, C.D., Zanuttigh, P. & Cortelazzo, G.M., 2012. *Time-of-Flight Cameras and Microsoft Kinect*^{TM}, Springer.

Myers, P.T., 2005. Specific issues in surgical techniques for mobile-bearing designs. In J. Bellemans, M. D. Ries, & J. M. K. Victor, eds. *Total knee arthroplasty: A guide to get better performance*. Heidelberg: Springer, pp. 217–222.

Nabighian, M.N. et al., 2005. The historical development of the magnetic method in exploration. *GEOPHYSICS*, 70(6), pp.33–61.

Nafis, C. et al., 2006. Method for estimating dynamic EM tracking accuracy of surgical navigation tools. In K. R. Cleary & R. L. Galloway, Jr., eds. *Medical Imaging*. International Society for Optics and Photonics, p. 61410K–61410K–16.

National Audit Office, 2011. *Managing High Value Capital Equipment in the NHS in England: Department of Health*,

Navab, N., Heining, S.-M. & Traub, J., 2010. Camera augmented mobile C-arm (CAMC): calibration, accuracy study, and clinical applications. *IEEE transactions on medical imaging*, 29(7), pp.1412–23.

Nguyen, T.T., Jung, H. & Lee, D.Y., 2013. Markerless tracking for augmented reality for image-guided Endoscopic Retrograde Cholangiopancreatography. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2013, pp.7364–7.

Nicolau, S.A. et al., 2009. An augmented reality system for liver thermal ablation: Design and evaluation on clinical cases. *Medical Image Analysis*, 13(3), pp.494–506.

Nigel Arden, D. et al., 2006. Osteoarthritis: Epidemiology. *Best Practice & Research Clinical Rheumatology*, 20(1), pp.3–25.

Nikou, C. et al., 2013. Navigated freehand surgical tool and kit.

Nolte, L.-P. et al., 1995. Clinical evaluation of a system for precision enhancement in spine surgery. *Clinical Biomechanics*, 10(6), pp.293–303.

Novák-Marcinčin, J. & Fečová, V., 2010. The options of motion tracking in virtual reality. In *14th International Research/Expert Conference "Trends in the Development of Machinery and Associated Technology"*. pp. 361–364.

Novelli, C., Costa, J. & Souza, R., 2012. Effects of aging and physical activity on articular cartilage : a literature review. *J. Morphol. Sci.*, 29(1), pp.1–7.

De Onis, M., Blössner, M. & Borghi, E., 2010. Global prevalence and trends of overweight and obesity among preschool children. *The American journal of clinical nutrition*, 92(5), pp.1257–64.

Ossendorf, C., Fuchs, B. & Koch, P., 2006. *Femoral stress fracture after computer navigated total knee arthroplasty*,

Palmer, C. et al., 2015. Mobile 3D augmented-reality system for ultrasound applications. In *2015 IEEE International Ultrasonics Symposium (IUS)*. IEEE, pp. 1–4.

Panerai, F. et al., 1999. A 6-dof device to measure head movements in active vision experiments: geometric modeling and metric accuracy. *Journal of Neuroscience Methods*, 90(2), pp.97–106.

Pang, H.-N. et al., 2013. Joint line changes and outcomes in constrained versus unconstrained total knee arthroplasty for the type II valgus knee. *Knee surgery, sports traumatology, arthroscopy : official journal of the ESSKA*, 21(10), pp.2363–9.

Parratte, S. et al., 2010. Effect of postoperative mechanical axis alignment on the fifteen-year survival of modern, cemented total knee replacements. *The Journal of bone and joint surgery. American volume*, 92(12), pp.2143–9.

Parratte, S. et al., 2007. Reproducing the Mechanical Axis did not Improve the 15-Year Survival of 398 Modern TKA. In *American Association of Hip and Knee Surgeons annual meeting*. p. 18.

Peersman, G. et al., 2014. Cost-effectiveness of unicondylar versus total knee arthroplasty: a Markov model analysis. *The Knee*, 21 Suppl 1, pp.S37–42.

Petersen, T.D., 1985. Surgical knee alignment method and system. , p.12.

Picard, F. et al., 2004. Clinical classification of CAOS systems. In A. Digioia et al., eds. *Computer and Robotic Assisted Hip and Knee Surgery*. Oxford University Press, p. 336.

Picard, F., 2014. Computer Assisted Knee Replacement Surgery: Is the Movement Mainstream? *Orthopedic & Muscular System*, 03(02).

Porter, M. et al., 2013a. *National joint register for England, Wales and Northern Ireland 10th annual report*,

Porter, M. et al., 2013b. *National joint register for England, Wales and Northern Ireland 10th annual report: Prostheses used in hip, knee, ankle, elbow and shoulder replacement procedures 2012*,

Pour, A.E. et al., 2007. Rotating hinged total knee replacement: use with caution. *The Journal of bone and joint surgery. American volume*, 89(8), pp.1735–41.

Preim, B. & Botha, C.P., 2013. Image-guided surgery and augmented reality. In *Visual Computing for Medicine: Theory, Algorithms, and Applications*. Newnes, pp. 625–661.

Price, A.J. et al., 2003. A mobile-bearing total knee prosthesis compared with a fixed-bearing prosthesis. *The Journal of Bone and Joint Surgery*, 85(1), pp.62–67.

Raab, F. et al., 1979. Magnetic Position and Orientation Tracking System. *IEEE Transactions on Aerospace and Electronic Systems*, AES-15(5), pp.709–718.

Radin, E.L. & Rose, R.M., 1986. Role of subchondral bone in the initiation and progression of cartilage damage. *Clinical orthopaedics and related research*, (213), pp.34–40.

Radke, R.J., 2012. Dense correspondence and its applications. In *Computer Vision for Visual Effects*. Cambridge University Press, pp. 148–207.

Ramer, U., 1972. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3), pp.244–256.

Ranawat, A.S. & Ranawat, C.S., 2012. The history of total knee arthroplasty. In M. Bonnin et al., eds. *The Knee Joint*. Paris: Springer Paris, pp. 699–707.

Rand, J.A. et al., 2003. Factors affecting the durability of primary total knee prostheses. *The Journal of bone and joint surgery. American volume*, 85-A(2), pp.259–65.

Reed, S.E. et al., 2014. Shaping Watersheds Exhibit: An Interactive, Augmented Reality Sandbox for Advancing Earth Science Education. *American Geophysical Union*.

Ren, H. & Kazanzides, P., 2012. Investigation of Attitude Tracking Using an Integrated Inertial and Magnetic Navigation System for Hand-Held Surgical Instruments. *IEEE/ASME Transactions on Mechatronics*, 17(2), pp.210–217.

Ribo, M., Pinz, A. & Fuhrmann, A.L., 2001. A new optical tracking system for virtual and augmented reality applications. In *IMTC 2001. Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference. Rediscovering Measurement in the Age of Informatics (Cat. No.01CH 37188)*. IEEE, pp. 1932–1936.

Ritter, M. et al., 2007. A landmark-based 3D calibration strategy for SPM. *Measurement Science and Technology*, 18(2), pp.404–414.

Ritter, M.A. et al., 2011. The effect of alignment and BMI on failure of total knee replacement. *The Journal of bone and joint surgery. American volume*, 93(17), pp.1588–96.

Roche, M.W., Augustin, D. & Conditt, M.A., 2010a. ACCURACY OF ROBOTICALLY ASSISTED UKA. *Journal of Bone & Joint Surgery, British Volume*, 92-B(SUPP I), p.127.

Roche, M.W., Augustin, D. & Conditt, M.A., 2010b. ONE YEAR OUTCOMES OF ROBOTICALLY GUIDED UKA. *Journal of Bone & Joint Surgery, British Volume*, 92-B(SUPP I), pp.156–157.

Rodriguez, F. et al., 2005. Robotic clinical trials of uni-condylar arthroplasty. *The international journal of medical robotics + computer assisted surgery : MRCAS*, 1(4), pp.20–8.

Rolland, J.P. & Fuchs, H., 2000. Optical Versus Video See-Through Head-Mounted Displays in Medical Visualization. *Presence: Teleoperators and Virtual Environments*, 9(3), pp.287–309.

Rönn, K. et al., 2011. Current Surgical Treatment of Knee Osteoarthritis. *Arthritis*, 2011, pp.1–9.

Rosneck, J. et al., 2007. Managing knee osteoarthritis before and after arthroplasty. *Cleveland Clinic Journal of Medicine*, 74(9), pp.663–671.

Ruijters, D. et al., 2014. Latency optimization for autostereoscopic volumetric visualization in image-guided interventions. *Neurocomputing*, 144, pp.119–127.

S., N., A., D. & Carlos, J., 2012. Mechanical Behavior of Articular Cartilage. In *Injury and Skeletal Biomechanics*. InTech.

Saber-Sheikh, K. et al., 2010. *Feasibility of using inertial sensors to assess human movement*,

Sackier, J.M. & Wang, Y., 1994. Robotically assisted laparoscopic surgery. *Surgical Endoscopy*, 8(1), pp.63–66.

Salter, R.B., 1999. *Textbook of Disorders and Injuries of the Musculoskeletal System: An Introduction to Orthopaedics, Fractures, and Joint Injuries, Rheumatology, Metabolic Bone Disease, and Rehabilitation*, Lippincott Williams & Wilkins.

Samosky, J.T. et al., 2012. BodyExplorerAR. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction - TEI '12*. New York, New York, USA: ACM Press, p. 263.

Santoro, M., Alregib, G. & Altunbasak, Y., 2012. Misalignment correction for depth estimation using stereoscopic 3-D cameras. *2012 IEEE 14th International Workshop on Multimedia Signal Processing, MMSP 2012 - Proceedings*, pp.19–24.

Schai, P.A., Thornhill, T.S. & Scott, R.D., 1998. Total knee arthroplasty with the PFC system: RESULTS AT A MINIMUM OF TEN YEARS AND SURVIVORSHIP ANALYSIS. *J Bone Joint Surg Br*, 80-B(5), pp.850–858.

Schauwecker, K., Klette, R. & Zell, A., 2012. A new feature detector and stereo matching method for accurate high-performance sparse stereo matching. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 5171–5176.

Schmalstieg, D. & Wagner, D., 2007. Experiences with Handheld Augmented Reality. In *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE, pp. 1–13.

Schmidt, S.F., 1981. The Kalman filter - Its recognition and development for aerospace applications. *Journal of Guidance, Control, and Dynamics*, 4(1), pp.4–7.

Schoenfelder, R. & Schmalstieg, D., 2008. Augmented Reality for Industrial Building Acceptance. In *2008 IEEE Virtual Reality Conference*. IEEE, pp. 83–90.

Schreer, O., Kauff, P. & Sikora, T., 2005. Tracking and user interface for mixed reality. In *3D Videocommunication: Algorithms, concepts and real-time systems in human centred communication*. John Wiley & Sons, pp. 315–334.

Scuderi, G.R. & Tria, A.J. eds., 2006. *Knee Arthroplasty Handbook*, New York, NY: Springer New York.

Sculco, T.P. & Martucci, E.A., 2001. *Knee Arthroplasty*, Springer.

Seon, J.K. et al., 2009. Functional comparison of total knee arthroplasty performed with and without a navigation system. *International orthopaedics*, 33(4), pp.987–90.

Shin, S. et al., 2014. A single camera tracking system for 3D position, grasper angle, and rolling angle of laparoscopic instruments. *International Journal of Precision Engineering and Manufacturing*, 15(10), pp.2155–2160.

Siebert, W. et al., 2002. Technique and first clinical results of robot-assisted total knee replacement. *The Knee*, 9(3), pp.173–180.

Simmons, S. et al., 1996. Proprioception following total knee arthroplasty with and without the posterior cruciate ligament. *The Journal of Arthroplasty*, 11(7), pp.763–768.

Sinha, R.K., 2009. Outcomes of robotic arm-assisted unicompartmental knee arthroplasty. *American journal of orthopedics (Belle Mead, N.J.)*, 38(2 Suppl), pp.20–2.

Sinha, R.K., Plush, R. & Weems, V.J., 2010. UNICOMPARTMENTAL ARTHROPLASTY USING A ROBOTIC TACTILE GUIDANCE SYSTEM. *Journal of Bone & Joint Surgery, British Volume*, 92-B(SUPP I), pp.157–158.

Sinram, O. et al., 2002. Calibration of an SEM, using a nano positioning tilting table and a microscopic calibration pyramid. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Proceedings ISPRS Commission V Symposium Corfu*, pp.210–215.

Siston, R. a. & Delp, S.L., 2006. Evaluation of a new algorithm to determine the hip joint center. *Journal of Biomechanics*, 39(1), pp.125–130.

Skwara, A. et al., 2008. Damages of the tibial post in constrained total knee prostheses in the early postoperative course - a scanning electron microscopic study of polyethylene inlays. *BMC musculoskeletal disorders*, 9, p.83.

Smith, J.R., Riches, P.E. & Rowe, P.J., 2013. Accuracy of a freehand sculpting tool for unicondylar knee replacement. *The international journal of medical robotics + computer assisted surgery : MRCAS*.

Song, E.-K. et al., 2013. Robotic-assisted TKA reduces postoperative alignment outliers and improves gap balance compared to conventional TKA. *Clinical orthopaedics and related research*, 471(1), pp.118–26.

Song, E.-K. et al., 2011. Simultaneous bilateral total knee arthroplasty with robotic and conventional techniques: a prospective, randomized study. *Knee surgery, sports traumatology, arthroscopy : official journal of the ESSKA*, 19(7), pp.1069–76.

Song, S., Mor, A. & Jaramaz, B., 2009. HyBAR: hybrid bone-attached robot for joint arthroplasty. *The international journal of medical robotics + computer assisted surgery : MRCAS*, 5(2), pp.223–31.

Song, W. et al., 2007. Computer-aided modeling and morphological analysis of hip joint. *2007 1st International Conference on Bioinformatics and Biomedical Engineering, ICBBE*, (863), pp.1218–1221.

Sorger, J.I. et al., 1997. The posterior cruciate ligament in total knee arthroplasty. *The Journal of Arthroplasty*, 12(8), pp.869–879.

Sparmann, M. et al., 2003. Positioning of total knee arthroplasty with and without navigation support: A PROSPECTIVE, RANDOMISED STUDY. *J Bone Joint Surg Br*, 85-B(6), pp.830–835.

Srinivasan, K. et al., 2009. Effects of acoustic source and filtering on time-of-flight measurements. *Applied Acoustics*, 70(8), pp.1061–1072.

Stefanescu, D.M., 2011. *Handbook of Force Transducers: Principles and Components*, Springer.

Stewart, D., 1966. A Platform with Six Degrees of Freedom: A new form of mechanical linkage which enables a platform to move simultaneously in all six degrees of freedom developed by Elliott-Automation. *Aircraft Engineering and Aerospace Technology*, 38(4), pp.30–35.

Strobel, N. et al., 2009. 3D Imaging with Flat-Detector C-Arm Systems. In M. F. Reiser et al., eds. *Multislice CT*. Medical Radiology. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 33–52.

Stuart L. Axelson, J., Bono, J. V. & Wartel, D.J., 1999. Surgical apparatus for use in total knee arthroplasty and surgical methods for using said apparatus. , p.11.

Stuart, M.J. et al., 1990. Late recurrence of varus deformity after proximal tibial osteotomy. *Clinical orthopaedics and related research*, (260), pp.61–5.

Stukenborg-Colsman, C. et al., 2001. High tibial osteotomy versus unicompartmental joint replacement in unicompartmental knee joint osteoarthritis: *The Knee*, 8(3), pp.187–194.

Stulberg, S.D., Loan, P. & Sarin, V., 2002. Computer-Assisted Navigation in Total Knee Replacement: Results of an Initial Experience in Thirty-five Patients. *The Journal of Bone & Joint Surgery*, 84(suppl_2), pp.S90–S98.

Sutherland, I., 1965. The ultimate display. In *IFIP '65*. pp. 506–508.

Sutherland, I.E., 1968. A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I on - AFIPS '68 (Fall, part I)*. New York, New York, USA: ACM Press, p. 757.

Suzuki, S. & be, K., 1985. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1), pp.32–46.

Szeliski, R., 2010. Stereo Correspondence. In *Computer Vision: Algorithms and Applications*. Springer, pp. 471–500.

Tamura, Y. et al., 2005. Surface-based registration accuracy of CT-based image-guided spine surgery. *European spine journal : official publication of the European Spine Society, the European Spinal Deformity Society, and the European Section of the Cervical Spine Research Society*, 14(3), pp.291–7.

Tarwala, R. & Dorr, L.D., 2011. Robotic assisted total hip arthroplasty using the MAKO platform. *Current reviews in musculoskeletal medicine*, 4(3), pp.151–6.

Tateishi, H., 2000. Indications for Total Knee Arthroplasty and Choice of Prosthesis. *Journal of the Japanese Medical Association*, 123(4), pp.470–474.

Terzidis, I. et al., 2012. Gender and Side-to-Side Differences of Femoral Condyles Morphology: Osteometric Data from 360 Caucasian Dried Femori. *Anatomy Research International*, 2012(1), pp.1–6.

Thienpont, E., 2012. *Improving Accuracy in Knee Arthroplasty*, JP Medical Ltd.

Tingart, M. et al., 2008. Computer-assisted total knee arthroplasty versus the conventional technique: how precise is navigation in clinical routine? *Knee surgery, sports traumatology, arthroscopy : official journal of the ESSKA*, 16(1), pp.44–50.

Tola, E., Lepetit, V. & Fua, P., 2010. DAISY: an efficient dense descriptor applied to wide-baseline stereo. *IEEE transactions on pattern analysis and machine intelligence*, 32(5), pp.815–30.

Trattnig, S., 1997. Overuse of hyaline cartilage and imaging. *European Journal of Radiology*, 25(3), pp.188–198.

Turkiewicz, A. et al., 2014. Current and future impact of osteoarthritis on health care: a population-based study with projections to year 2032. *Osteoarthritis and cartilage / OARS, Osteoarthritis Research Society*, 22(11), pp.1826–32.

Valenzuela, G.A. et al., 2013. Implant and Limb Alignment Outcomes for Conventional and Navigated Unicompartmental Knee Arthroplasty. *The Journal of Arthroplasty*, 28(3), pp.463–468.

Vazquez-Alvarez, Y., Oakley, I. & Brewster, S.A., 2011. Auditory display design for exploration in mobile audio-augmented reality. *Personal and Ubiquitous Computing*, 16(8), pp.987–999.

Vercher, J.L. & Gauthier, G.M., 1992. Oculo-manual coordination control: ocular and manual tracking of visual targets with delayed visual feedback of the hand motion. *Experimental brain research*, 90(3), pp.599–609.

Vetter, S.Y. et al., 2014. Computer assisted surgery in trauma and osteotomy. *Orthopaedics and Trauma*, 28(5), pp.286–293.

Vince, K.G. & Malo, M., 2008. Choice of constraint in revision total knee arthroplasty. In M. Bonnin & P. Chambat, eds. *Osteoarthritis of the knee: Surgical treatment*. Springer, pp. 457–472.

Van der Voort, P. et al., 2013. A systematic review and meta-regression of mobile-bearing versus fixed-bearing total knee replacement in 41 studies. *The bone & joint journal*, 95-B(9), pp.1209–16.

Vos, T. et al., 2012. Years lived with disability (YLDs) for 1160 sequelae of 289 diseases and injuries 1990 – 2010 : a systematic analysis for the Global Burden of Disease Study 2010. *Lancet*, 380, pp.2163–2196.

Waldschmidt, J.G. et al., 1997. In vitro and in vivo MR imaging of hyaline cartilage: zonal anatomy, imaging pitfalls, and pathologic conditions. *Radiographics : a review publication of the Radiological Society of North America, Inc*, 17(6), pp.1387–402.

Van de Walle, P. et al., 2012. Mechanical energy estimation during walking: Validity and sensitivity in typical gait and in children with cerebral palsy. *Gait & Posture*, 35(2), pp.231–237.

Wang, X. et al., 2015. Apparent resolution enhancement for near-eye light field display. In *SIGGRAPH ASIA 2015 Mobile Graphics and Interactive Applications on - SA '15*. New York, New York, USA: ACM Press, pp. 1–5.

Wang, Y.C. et al., 2011. Obesity 2 Health and economic burden of the projected obesity trends in the USA and the UK. *The Lancet*, 378(9793), pp.815–825.

Welch, G. & Foxlin, E., 2002. Motion tracking: no silver bullet, but a respectable arsenal. *IEEE Computer Graphics and Applications*, 22(6), pp.24–38.

Wiesel, U. & Boerner, M., 2004. First Experiences Using a Surgical Robot for Total Knee Replacment. *Journal of Bone & Joint Surgery, British Volume*, 86-B(SUPP IV), p.439.

Wiles, A.D., Thompson, D.G. & Frantz, D.D., 2004. Accuracy assessment and interpretation for optical tracking systems. In R. L. Galloway, Jr., ed. *Medical Imaging 2004*. International Society for Optics and Photonics, pp. 421–432.

Williams, D.H., Garbuz, D.S. & Masri, B.A., 2010. Total knee arthroplasty : *British Columbia Medical Journal*, 52(9), pp.447–454.

Willis-Owen, C.A. et al., 2009. Unicondylar knee arthroplasty in the UK National Health Service: an analysis of candidacy, outcome and cost efficacy. *The Knee*, 16(6), pp.473–8.

Windolf, M., Götzen, N. & Morlock, M., 2008. Systematic accuracy and precision analysis of video motion capturing systems--exemplified on the Vicon-460 system. *Journal of biomechanics*, 41(12), pp.2776–80.

Wishart, N. et al., 2014. National Joint Registry: 11th Annual Report. , (December 2013).

Wixted, A. et al., 2011. Detection of throwing in cricket using wearable sensors. *Sports Technology*, 4(3-4), pp.134–140.

Woellner, C. et al., 2010. Do musicians synchronise better with a prototypical conductor? *11th International conference on Music Perception and Cognition (ICMPC 11)*, pp.173–174.

Wolf, A. et al., 2005. MBARS: mini bone-attached robotic system for joint arthroplasty. *The international journal of medical robotics + computer assisted surgery : MRCAS*, 1(2), pp.101–21.

Woo, S.-K. et al., 2003. Development of event-based motion correction technique for PET study using list-mode acquisition and optical motion tracking system. In M. Sonka & J. M. Fitzpatrick, eds. *Medical Imaging 2003: Image Processing. Edited by Sonka*. pp. 1300–1307.

Woodman, O. & Harle, R., 2008. Pedestrian localisation for indoor environments. In *Proceedings of the 10th international conference on Ubiquitous computing - UbiComp '08*. New York, New York, USA: ACM Press, p. 114.

Woolf, A.D. et al., 2007. Strategies for the prevention and management of osteoarthritis of the hip and knee. *Best Practice & Research Clinical Rheumatology*, 21(1), pp.59–76.

Woolf, A.D. & Pfleger, B., 2003. Policy and Practice Burden of major musculoskeletal conditions. , 81(03).

Wu, J.Z. & Herzog, W., 2002. Elastic anisotropy of articular cartilage is associated with the microstructures of collagen fibers and chondrocytes. *Journal of Biomechanics*, 35(7), pp.931–942.

Wysocki, R.W. et al., 2008. *Femoral Fracture Through a Previous Pin Site After Computer-Assisted Total Knee Arthroplasty*,

Yao, L. et al., 2012. Accurate real-time stereo correspondence using intra- and inter-scanline optimization. *Journal of Zhejiang University SCIENCE C*, 13(6), pp.472–482.

Yoganandan, N. et al., 2011. Comparison of head-neck responses in frontal impacts using restrained human surrogates. *Annals of advances in automotive medicine / Annual Scientific Conference ... Association for the Advancement of Automotive Medicine. Association for the Advancement of Automotive Medicine. Scientific Conference*, 55, pp.181–91.

Yoganandan, N. et al., 2012. Upper and lower neck loads in belted human surrogates in frontal impacts. *Annals of advances in automotive medicine / Annual Scientific Conference ... Association for the Advancement of Automotive Medicine. Association for the Advancement of Automotive Medicine. Scientific Conference*, 56, pp.125–36.

Yun, X. et al., 2007. Self-contained Position Tracking of Human Movement Using Small Inertial/Magnetic Sensor Modules. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, pp. 2526–2533.

Zhang, W. et al., 2010. OARSI recommendations for the management of hip and knee osteoarthritis: part III: Changes in evidence following systematic cumulative update of research published through January 2009. *Osteoarthritis and cartilage / OARS, Osteoarthritis Research Society*, 18(4), pp.476–99.

Zhang, Y.Z. et al., 2011. Preliminary application of computer-assisted patient-specific acetabular navigational template for total hip arthroplasty in adult single development dysplasia of the hip. *The international journal of medical robotics + computer assisted surgery : MRCAS*, 7(4), pp.469–74.

Zhao, W., Chai, J. & Xu, Y.-Q., 2012. Combining marker-based mocap and RGB-D camera for acquiring high-fidelity hand motion data. , pp.33–42.

Zhao, W. & Nandhakumar, N., 1996. Effects of camera alignment errors on stereoscopic depth estimates. *Pattern Recognition*.

Zhou, H. et al., 2008. *Use of multiple wearable inertial sensors in upper limb motion tracking*,

Zhou, H. & Hu, H., 2008. Human motion tracking for rehabilitation—A survey. *Biomedical Signal Processing and Control*, 3(1), pp.1–18.

Appendix 1
Generation One Implementation

The following sections describe the implementation of the first generation system. A general overview is given before discussing more specific elements.

As described in the design section above the system was built upon the ARToolKit library and its architecture was modelled around the examples included with the library. These utilise a single source file containing; initialisation, a main continuous loop, a mouse and keyboard event, and de-initialisation functions. The rest of the system then expanded upon this using an OOP class based design.

The program initialised by connecting to an external USB web camera and providing a GUI to allow the users to configure the camera parameters. Following successful connection, camera calibration parameters and marker patterns were loaded via separate files, and their identification parameters appropriately stored.

The main display window, scaled to the resolution of the camera, was then launched and the program entered the main processing loop. This main loop performed image acquisition, image rendering, marker detection, pose estimation, and augmented overlay rendering. The loop also allowed for user interaction, such as adjusting the image thresholding value to suit lighting conditions, which was set by default to 100.

During marker detection a list of possible markers, defined by the ID numbers, was produced. Each potential marker was compared to each marker instance loaded into the systems at initialisation. If a match was found the marker class stored the array index value of the matched marker. If the marker instance had previously been matched the confidence factor of the two potential matchers were compared and the higher of the two stored.

Rendering was performed as a three stage process. Firstly 2D background elements were rendered, such as the video stream and UI buttons. After which, the rendering pipeline was switched to 3D mode to allow the augmented elements to be rendered, using the calculated marker pose as the view matrix and calibration parameters to derive a projection matrix. In the case of multiple markers the view matrix would be switched to the pose matrix of the marker, relative to which the current object was to be positioned. Finally the pipeline reverted to 2D mode to allow any 2D augmentation to be render, such as the side scales shown in Figure A2.1 below.

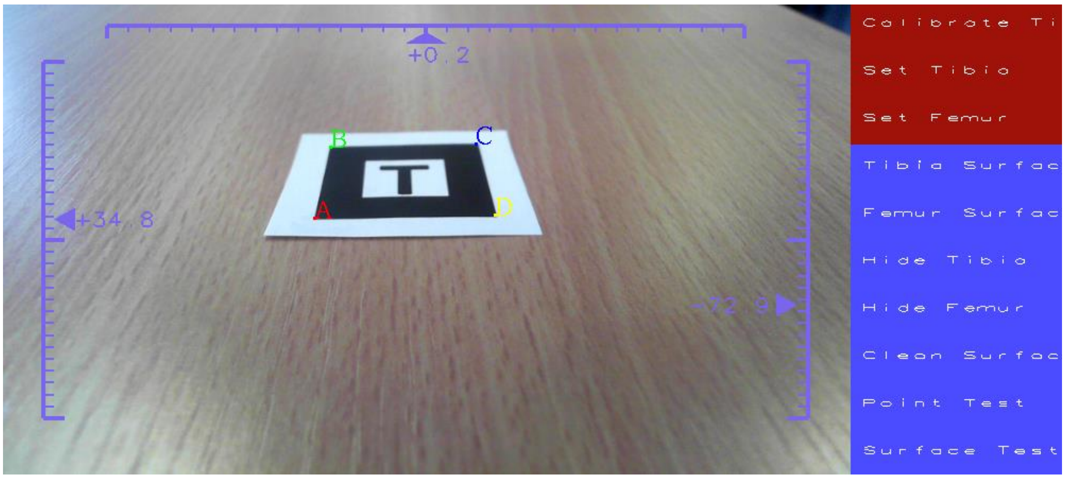


Figure A2.1: 2D rendered elements showing: marker corners, soft buttons, and relative position indicators.

Figure A2.2 below illustrates the augmented node that was rendered onto each marker to indicate successful detection. The colour of the marker signifies the quality of the marker detection, based upon the confidence factor.

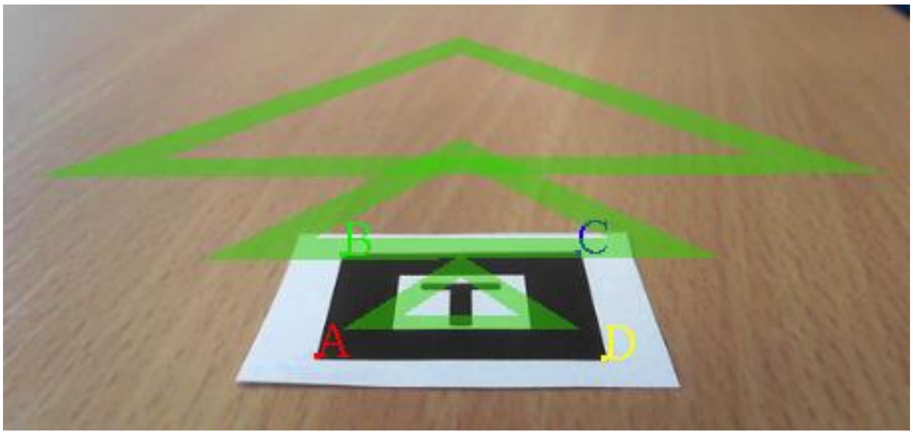


Figure A2.2: augmented node marker overlay

All rendering was completed upon a back buffer before being switched to give the user a smooth experience.

After each render keyboard and mouse functions were processed to allow low latency user interface.

This concludes the general structure and implementation of the first generation system. What follows below is a discussion of the implementation of several specific system elements. These elements: tool tracking, surface generation, and joint centre location, were highlighted in Chapter 4 as core requirements of a guided UKA system. Most of the elements below underwent a number of iterative steps, building upon faults of the initial implementation to provide a better system. These iterations, and the problems or short comings prompting them, form the core of the discussion below.

A1.1 Tool Tracking

Chapter 4 concluded that many of the functions required for guided UKA, such as bone model generation, required the tracking of a probe tool. To this end, the first generation system provided the tracking of a probe tool attached to the camera. The pose of the camera, relative to a marker, may be given by inverting the marker pose matrix. Therefore, with the tool rigidly attached to the camera a simple transformation of the camera pose provided the tool position. As the tool was to be used as a probe it was decided that only the position of the probe tip, relative to a marker, was required, as opposed to a full pose including orientation. Given this generalisation the situation is illustrated by Figure A2.3 below.

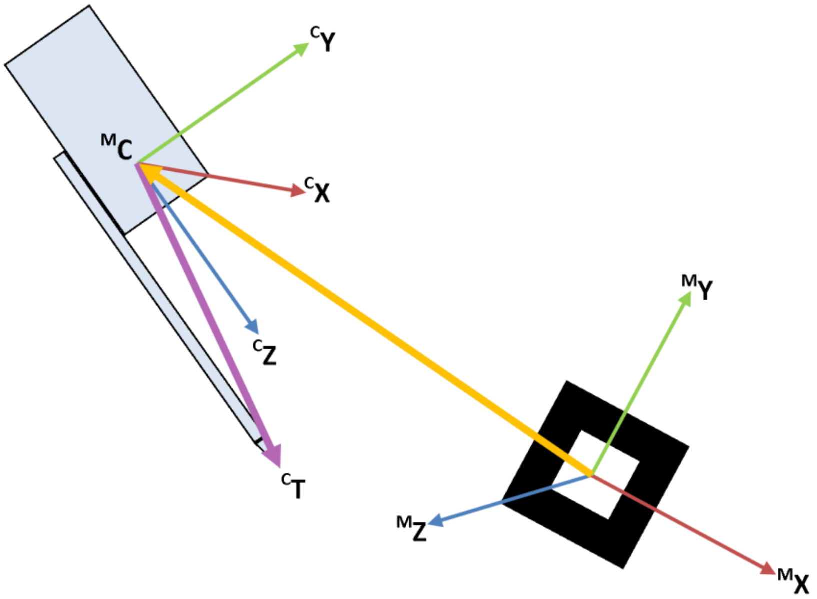


Figure A2.3: Tool tip, camera, and marker spatial relation.

The position vector, or offset, of the probe tip ($^{C}\vec{T}$) is known in the camera's coordinate system. Furthermore, the position and orientation of the camera (^{M}C) is known relative to the marker is the marker's coordinate system. Therefore, the position of the probe tip may be derived using Equation A2.2 below.

$$^{M}\vec{T} = {}^{M}C * {}^{C}\vec{T} \tag{A2.2}$$

The tip position was calculated relative to each marker detected, using the $cam2tip$ method of the Drill class. As non-homogeneous pose matrices were used Equation A2.2 was implemented as the two stage process shown by Equation A2.3, where the rotational and translation elements of the camera pose are treated separately.

$$^{M}\vec{T} = {}^{M}R_{C} * {}^{C}\vec{T} + {}^{M}T_{C} \tag{A2.3}$$

The design of the probe did not ensure a constant tool tip position relative to the camera. Furthermore, it was not possible to physically measure the displacement. The centre of the camera system, which defines both the position of the camera in the coordinate system of the marker and the centre of the coordinate system of the camera, is within the centre of the closed lens and chip system. Therefore, it became necessary to calculate the offset of the tool tip in the coordinate system of the camera. Equation A2.2 is solved for the tool tip offset in the coordinate system of the camera to give equation A2.4.

$$ {}^{C}\vec{T} = {}^{M}C^{-1} * {}^{M}\vec{T} \qquad (\text{A2.4}) $$

The initial implementation was found to be error prone, using only a single sample to derive the offset that would be used throughout the procedure. Therefore, a second iteration was performed. This second iteration gathered multiple camera pose samples. For each sample the offset was calculated and summed into the offset member. Once all samples had been captured the offset was divided by the sample count to produce a mean offset. Typical sample counts used were between 100 and 250 samples. This was found to reduce the effect of random sample errors, especially in comparison to the original implementation.

To calibrate the system the user placed the probe at a known position, relative to a specific marker. The calibration button was then clicked and the system would begin collecting samples. Once all samples had been collected the mean tip to camera offset was stored. This allowed the position of the tip to be calculated, relative to each marker.

A1.2 Surface Generation

As described by Chapter 4, the proposed system was to operate under the image-free modality. Therefore, it was required to generate a model intra-operatively. As such a surface generation module was designed. Several iterative and incremental loops were performed to bring this module to a practical level, and these shall be discussed below.

As discussed in Chapter 4, the surface was to be generated by tracing the tip of the probe across the articulating surface of the joint while tracking the tool relative to the bone marker, akin to the Navio® system. This would enable the system to record a series of points in 3D space which lie on the surface of the joint, relative to the bone marker. Some method of processing would then be used to form a digital model of the joint which could be displayed to the user by overlaying the model onto the physical joint.

The initial iteration generated a raw point cloud and displayed this as a series of small dots. In addition to the Main module three classes were involved in this iteration; Drill, Bone, and Draw. The interaction of these three classes is shown below in Figure A2.4 illustrates the interaction between these four modules during surface generation.

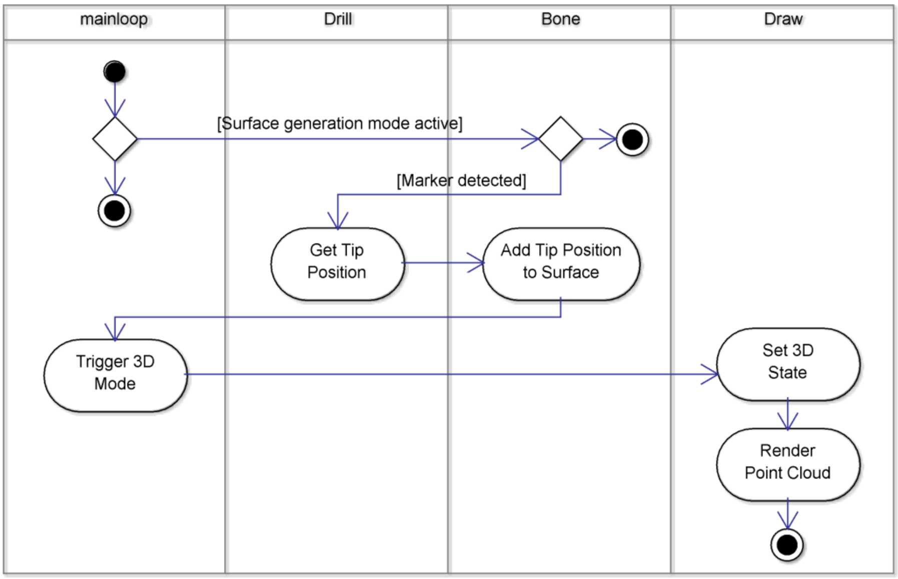


Figure A2.4: Activity diagram for original surface generation implementation

As no post processing was required it was possible to display the resulting surface as it was generated. The results of this surface generation method may be seen below in Figure A2.5 where the articulating surface of a synthetic tibia saw bone has been traced.

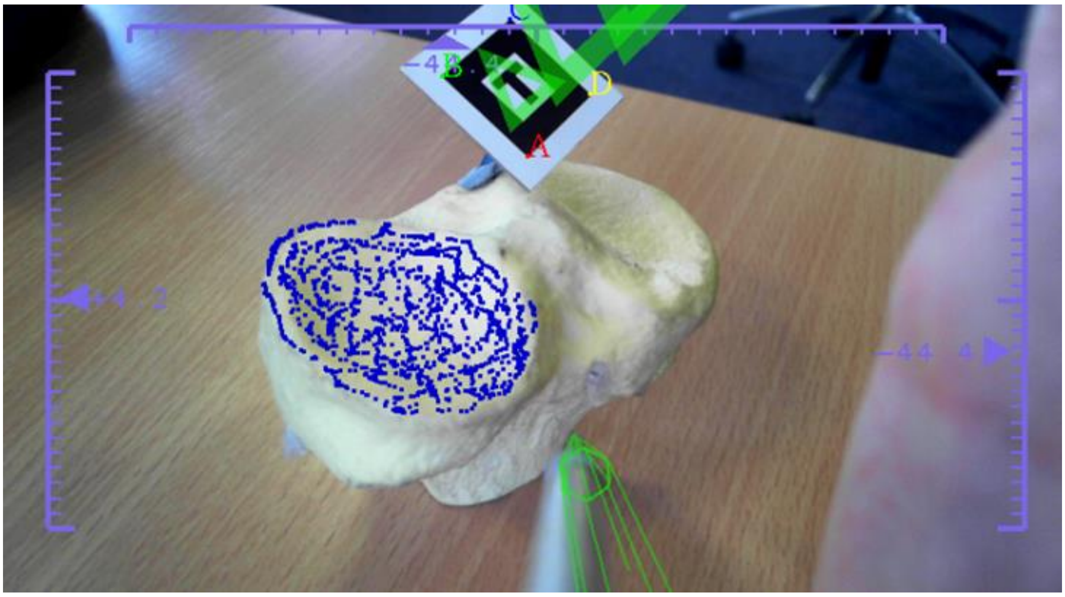


Figure A2.5: Point cloud tibial surface

This initial error had two clear problems. Firstly, if the tool tip was accidently removed from the surface before releasing the capture button, a point *tail* was produced. This was remedied with a point remove function. Secondly, noise within the tracking system caused a relatively rough surface. To smooth the capture method was updated. As opposed to appending the new tip position (P_N) directly to the surface vector, the new method calculated the weighted mean of the tip with the previous two entries using Equation A2.5 below.

$$P_{n+1} = \frac{P_N + 0.5P_n + 0.25P_{n-1}}{1.75} \qquad \text{(A2.5)}$$

Despite the easy with which this method was implemented it had a number of short comings. Firstly, the data was stored in an unordered fashion. Therefore, to remove or edit the position of points, as would be required during resection, each element would require inspection. This could quickly present a bottleneck to the system, as surfaces could contain several hundred points. Inconsistent point density was also a concern. As different surface regions could have considerably different point densities the performance of any subsequent system would likely be inconsistent as well. Finally, either system noise or operator error could cause point stacking. This is where there are several points from the same point on the bone surface, but at different tangential distances. Again, this was likely to affect the performance of any subsequent system.

To overcome these issues a new method was developed. The surface generation system would initialise with a grid of points predefined on a plane. As the tool tip was traced across the surface of the bone if the X and X-coordinates of the tip approximately matched those of a grid point the z coordinate of the grid point would be set to that of the tip. This approach would allow the surface points to be ordered, and prevent variable point densities and stacking. Figure A2.6 illustrates this approach whereby the blue points represent the original grid points. The black point indicates the tip centre and the surrounding circle to the search region. The point shown in red is within this search region, as such its Z-axis in increased to match that of the probe.

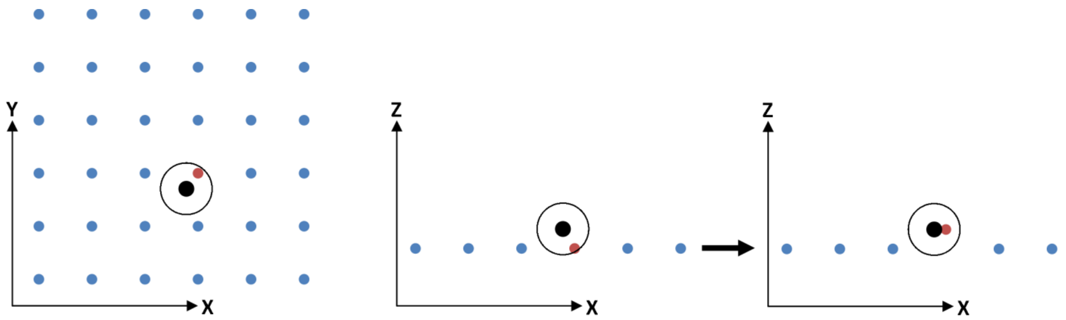


Figure A2.6: Grid point surface method

A planer grid of 700 by 700 points was initialised. As the camera was calibrated in millimetres a system unit was equal to 1 mm. Therefore, the grid measured 70 by 70 mm, with a resolution of 10 points/mm. The grid was centred upon the tool tip, with a Z-offset of -10 mm, in the marker's coordinate system. The grid was aligned such that it was coplanar with the bone marker. The grid was stored as a vector of float vectors, where each point contained an x, y, and z value. Therefore a 70x70 mm surface that resulted in a 700x700 point grid required 1.47 million float values for storage, requiring approximately 6 MB of memory on a 32-bit system. This alternative method of storage was used as it simplified access to any one point, without increasing memory overhead. Two methods edited the grid. The first allowed a grid point to be moved up or down, along the z-axis, to match the position of the tool tip. This allowed the surface to be generated by tracing the physical bone, as the surface was initialised below the surface of the bone. The second method was a restricted version of *build* that allowed grid points to move downwards only, to update the model during resection. Aside from this both functions were implemented the same. The procedure of the *built* method is illustrated by the activity diagram below in Figure A2.7.

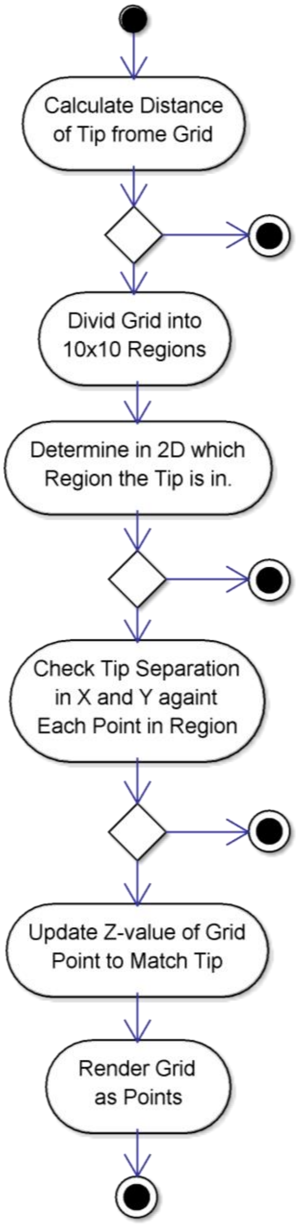

Calculate Distance of Tip frome Grid
Divid Grid into 10x10 Regions
Determine in 2D which Region the Tip is in.
Check Tip Separation in X and Y againt Each Point in Region
Update Z-value of Grid Point to Match Tip
Render Grid as Points

Figure A2.7: Activity diagram of grid surface method

Display of the surface was done as before, rendering each point as a dot. Figure A2.8 below shows this surface method at initialisation, on the right, and after surface tracing, on the left.

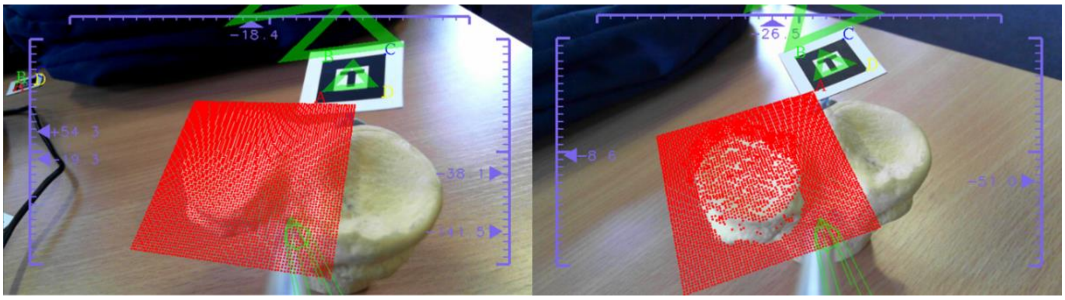


Figure A2.8: Second iteration surface generation grid method

Review of this approach indicated several potential improvements. The second iteration would convert the display of the points from point rendering to mesh render. As can be seen from Figure A2.8 above, in a 3D environment it is difficult to determine the relative depth of the dots used. As the size of the dots is irrespective to their depth and they are rendered with a constant colour it is difficult to fully interpret the structure of the surface. It was intended that mesh rendering would improve the depiction of relative depth information. In addition a colour vector was added so that separate grid points could be coloured individually. Again, this was designed to improve the interpretation of the surface model. All points would be initialised as transparent and only those that were edited would become visible. This would reduce the distraction caused by the unused grid points, such as those seen in Figure A2.8. A final iteration altered the pose of the surface grid at initialisation. The original method aligned the grid parallel to the bone marker. This resulted in poor results if the surface of the bone and bone marker were not near parallel. Therefore, the grid was aligned perpendicular to the optical axis of the camera, by applying the rotational section of the camera pose to the surface grid. This allowed the user to set the alignment of the grid based upon the tool. Figure A2.9 shows surface generation using this final iteration. The left image shows is taken part way through generation while the right image shows the finished surface.

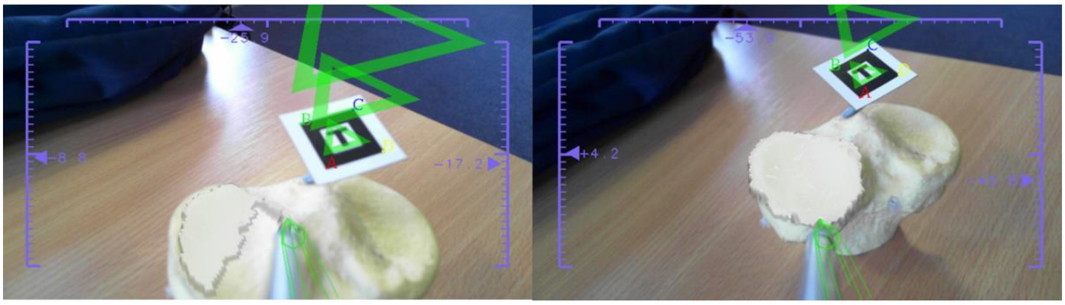


Figure A2.9: Final iteration of surface generation method

A1.3 Joint Centres

As discussed previously, HKA alignment is one of the fundamental properties used to plan implantations and gauge the success of a procedure. As such it was desirable for the system to be able to capture the positions of the three joints involved.

The knee centre is directly accusable during arthroplasty, while the ankle centre may be readily estimated as the mean centre of the two malleoli. However, the hip centre is impossible to measure directly as it is deep within the pelvis. The standard non-invasive approach seeks to locate the centre of the hip by measuring the movement of the limb about this joint. The existing commercial systems track the femoral marker as the hip joint is rotated. Assuming the pelvis remains relatively stationary the hip centre may be determined as the centre of rotation of the femur. However, this system relies on both the pelvis and tracking camera to remain stationary. If both remain stationary a relative movement of the femur within the camera reference frame is equivalent to a relative movement within the reference frame of the pelvis. As the current tracking system was tool mounted it was not stationary relative to the pelvis and as such it was not possible to equate the two reference frames. Two possibilities were considered, fix the tracking system during hip movement capture, or provide an additional fixed reference frame. Fixing the tracking system relative to the pelvis would allow the system to work as the commercial systems do. However, this approach would require a relatively strong stand able to hold the tracking system. This would likely be quite large, and setting up the stand and attaching the tracking system may be a relatively time consuming process. Alternatively, an additional reference frame would only require an additional marker. To keep the reference frame of this marker stationary to the pelvis the marker could be attached to the pelvis via additional bone pins, as used for the limb markers by the commercial systems. However, this would increase the risk of infection and cause additional discomfort to the patient. The commercial systems have already demonstrated that it is sufficient to assume the pelvis is stationary relative to the tracking system. For this to be the case the pelvis must be stationary to the operating table that in turn must be stationary to the floor, and thus the tracking system. Therefore, if a marker is rigidly attached to the operative table it is justified to consider its reference frame stationary relative to the pelvis.

It was decided to use the additional reference frame approach. However, this increases the potential error of the hip location method. The fixed camera approach requires only one marker to be tracked relative to the tracking camera. However, the additional reference frame approach requires two markers to be tracked relative to each other. Figure A2.10 illustrates the two possible marker systems. The two red arrows shown are equivalent, both representing the location of the femoral marker in a reference frame fixed relative to the pelvis.

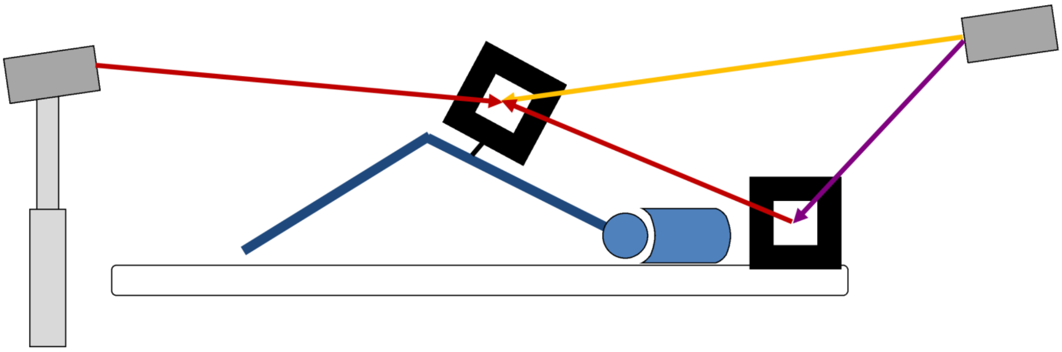


Figure A2.10: Two possible marker systems for hip centre location. Left shows fixed camera system and right shows dual marker system.

As the additional reference frame approach requires two markers to be tracked relative to each other the potential tracking error is doubled in comparison to the fixed camera approach.

The hip is a ball and socket joint, with the head of the femur rotating inside of the acetabular of the pelvis. This rotation may be closely approximated as spherical about the hip centre (Song et al. 2007). By assuming the rotation to be spherical the centre of the hip is equated to the centre of the sphere of rotation. If a fixed point on the femur is measured, relative to the pelvis, multiple times as the hip joint is rotate all the measurements should lie on the surface of a single sphere – the centre of which gives the hip centre.

The implementation used was based upon the closed sphere fitting method reported by Hiniduma *et al.* (Hiniduma et al. 2002). The method may be summarised by Equation A2.6, where m is the centre of rotation and r_t^i is the vector of the i^{th} sample point within the t^{th} frame of N total frames.

$$\begin{gathered} Am = b \\ where\ A = 2\sum_{i=1}^{I}\left\{\frac{1}{N}\sum_{t=1}^{N} r_t^i\left(r_t^i\right)^T\right\} - \overline{r^\iota}\left(\overline{r^\iota}\right)^T \\ b = \sum_{i=1}^{I}\overline{(r^\iota)^3} - \overline{r^\iota}\,\overline{(r^\iota)^2} \end{gathered} \qquad (\text{ A2.6 })$$

To implement this sphere fitting 100 measurements of the position of the bone marker in the coordinate system of the bed marker, while the femur was rotated by the user, were collected. As seen in Figure A2.10, the transformation from the bed marker to the bone marker – i.e. the pose of the bone marker in the coordinate system of the bed – is equivalent to the camera in the coordinate system of the bed, plus the bone marker in the coordinate system of the bed. This is expressed mathematically by Equation A2.7 below.

$$^{B}F = {}^{B}C + {}^{C}F \qquad (\text{ A2.7 })$$

As discussed previously the reference frames may be reversed by taking the inverse of the pose matrix. Therefore, Equation A2.7 may be more conveniently expressed as Equation A2.8.

$$^{B}F = {}^{C}B + {}^{C}F \qquad (\text{ A2.8 })$$

Once all the samples had been collected the method automatically began hip centre calculation. The hip centre in the coordinate system of the bed was first calculated. This was then translated into the coordinate system of the bone.

Figure A2.11 shows the system during sample capture for hip centre location. It is seen that lines, in the X, Y, and Z-axes, jointing the bed and femur marker are rendered.

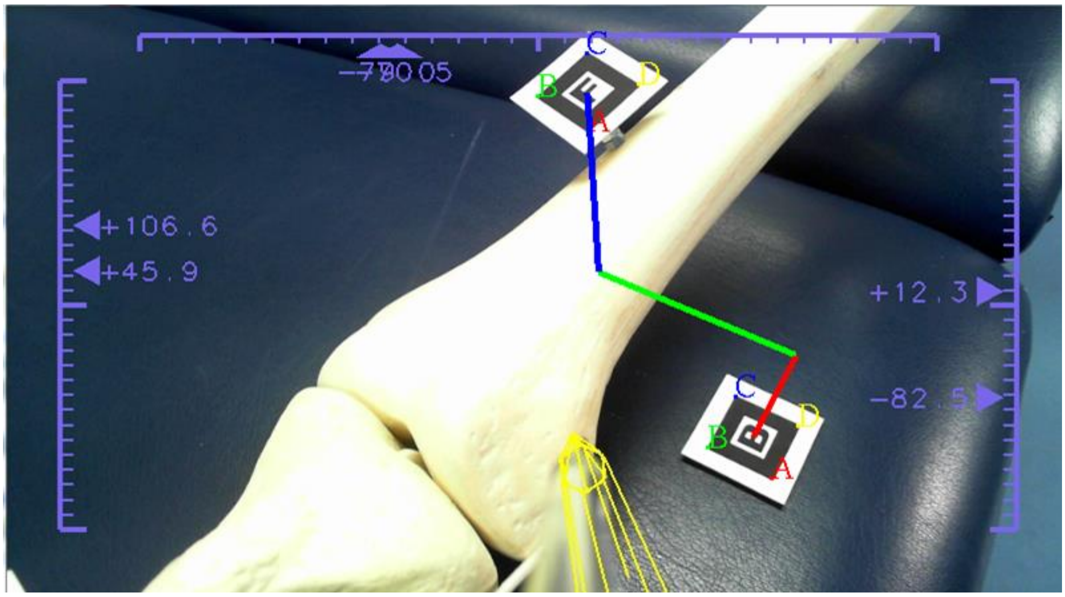


Figure A2.11: System during hip centre sampling, showing position of femur marker in coordinate system of bed marker.

Appendix 2
Generation Two Implementation

This chapter describes the core components of the second generation system, namely: marker detection, pose estimation, the resection tool, the graphical library, and the bone model.

As discussed in the main text, the second generation system was based upon the concept of stages, to simplify system expansion. Stages were derived from a common parent and provided overrides for the base rendering and UI methods. A stage manager allowed the current stage to be swapped for a new stage using a double buffer approach. The program initialised into a menu stage which allowed various testing stages, with associated world objects to be entered. The following sections describe the marker detection and pose estimation required by these world objects to perform augmented 3D rendering of virtual objects.

A2.1 Marker Detection

As discussed in Chapter 4, determining the pose of fiducial markers is a two stage process. Firstly the markers must be detected within the 2D image, before their 3D poses are estimated. This section looks at the steps taken to detect and identify potential markers within the 2D image. The section is sub-divided into the six main steps of marker detection and identification: image capture, where a frame is retrieved from the camera and loaded onto system memory; thresholding, whereby the original colour image of the scene is converted into a binary image; contour extraction, during which the edges within the binary image are found; corner detection, where the corners of any parallelepiped contours are determined; orientation, which inspects patterns upon the marker to correctly order the corners; and finally identification, which assigns an identification number to the marker based upon its markings.

A2.1.1 Image Capture

Image capture involves configuring the camera at the initialisation of the program and then extracting and saving to memory a new image frame for each iteration of the main loop of the system.

Camera initialisation and frame capture were handled by the $Camera$ class. Initialisation saw the calibration file, discussed in Section 7.3.1 loaded and the calibration parameters stored. OpenCV was then used to establish a connection to the USB camera. The camera settings used during calibration, stored within the calibration file, were then applied to the camera.

Once initialised the camera would continuously capture frames while running, replacing the current frame stored upon its internal memory. When the system required a new frame the camera passed the frame currently stored on this internal memory. This is categorised as continuous capture, as opposed to other capture methods that require a trigger. These are discussed in more detail in Chapter 8, where camera selection was again considered.

With each cycle of the program a new frame was captured from the camera, and checked for corruption. Once the frame was successfully captured it was passed by reference to the $Detector$ class, which was instantiated within the $Camera$ object.

A2.1.2 Thresholding

Threshold was the first of five image processing stages contained within the $Detector$ object. As described previously, thresholding is the process of converting the original colour image into a binary image. During the analysis of the first generation system the thresholding approach taken was presented as a probable cause of inaccuracy within the system. As such four different thresholding approaches were investigated, namely: global - as used in the first generation system, adaptive mean, adaptive Gaussian, and Otsu.

Each thresholding method operates upon a single channel intensity image. Therefore, the original three channel colour image is first converted to an intensity image by averaging the red, green, and blue channels.

Global thresholding was performed as with the first generation system. Each pixel within the image was compared against a single threshold value, typically a value of 120 was used. If the intensity of the pixel was above the threshold value the associated pixel within the resultant binary image was set high, if not the pixel was set low.

Adaptive mean and adaptive Gaussian thresholding adjust the threshold value across the image. This reduces the effect of changing lighting conditions across the system, therefore better preserving true contrasts. Adaptive mean calculates a local threshold value for each pixel based upon the mean of the surrounding pixels in a square block. The Gaussian approach is similar, however, it apples Gaussian weighting to the mean calculation. Both approaches typically subtract a constant from the calculated mean to derive the threshold value.

Finally, Otsu thresholding performs global thresholding using an automatically calculated threshold value. An image is considered bimodal if a histogram of the intensity values across the whole image contains two peaks. Otsu calculates the image histogram and sets the threshold value as being between these two peaks.

The four thresholding methods were each implemented and Figure A3.3 below shows the four resulting binary images of the same scene using standard parameter values. It may be seen from the original image of Figure A3.3 that the scene contained several non-marker items and presented a gradient lighting with the top left corner being brighter than the bottom right.

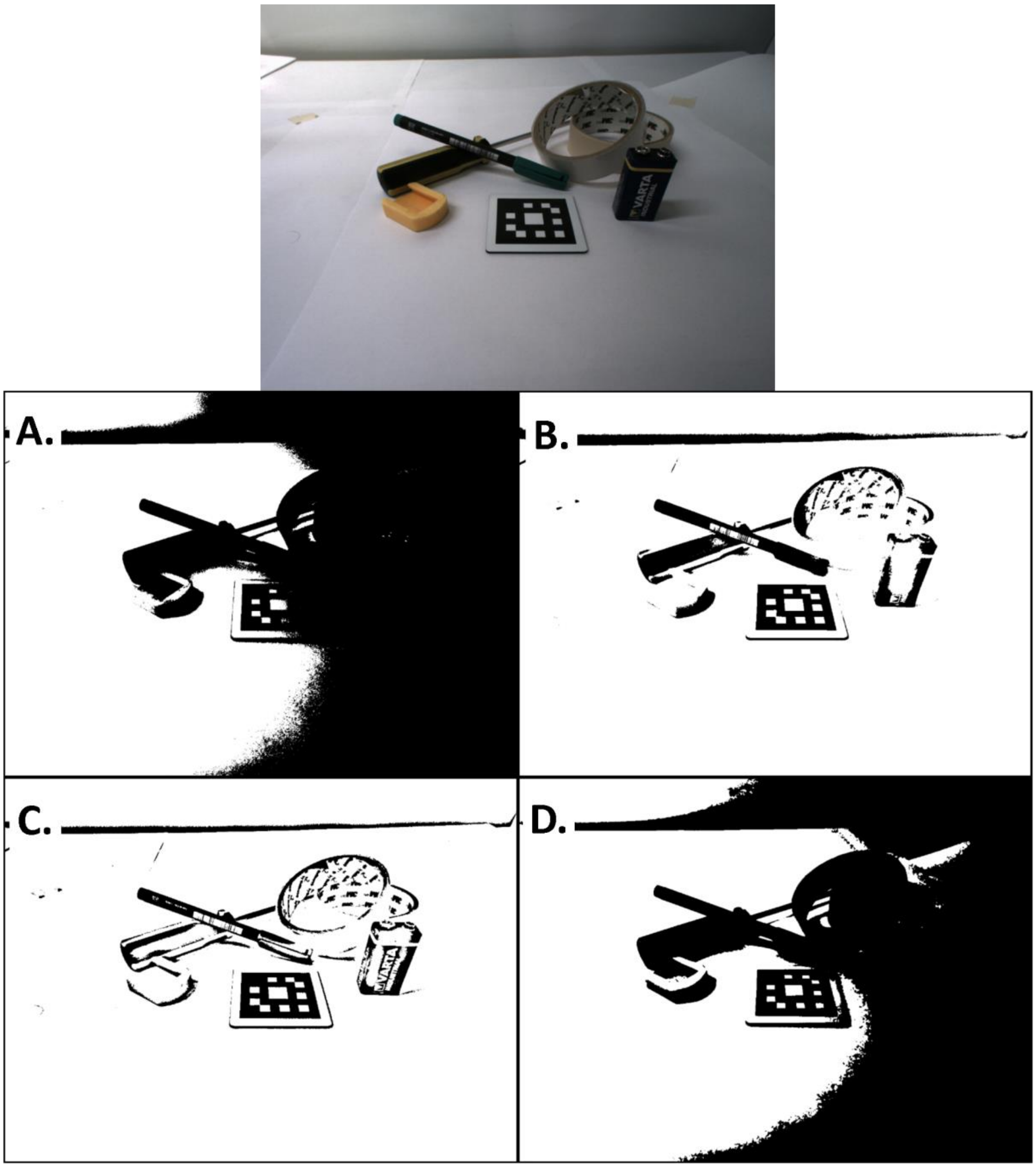


Figure A3.3: Comparison of four thresholding methods. A. Global, B. Adaptive mean, C. Adaptive Gaussian, and D. Otsu

Global thresholding produced the poorest results, preserving only half of the marker. This was expected due to the changing light intensity. Otsu also performed relatively poorly, failing to preserve the full marker. The Otsu method calculates the optimal threshold value for bimodal images only. While the markers themselves are bimodal the remaining scene dilutes the intensity distribution, leading to a poorly calculated threshold value.

Both forms of adaptive thresholding performed well, being able to preserve the marker fully. However, they are considerably more computationally intensive, as before thresholding may be performed a threshold value for each pixel must be calculated. Table A3.2 below shows the average processing times of the four thresholding methods.

Method	Global	Adaptive Mean	Adaptive Gaussian	Otsu
Time (ms)	17.1±2.4	22.9±1.6	135.1±12.2	23.5±2.3

Table A3.2: Average processing times of thresholding methods

As predicted by the complexities of their algorithms, global provides the significantly fastest solution (P < 0.01) followed by adaptive mean and Otsu that show an insignificant difference (P = 0.5). Lastly adaptive Gaussian provided the slowest result (P < 0.01), being approximately six times slower than adaptive mean or Otsu.

Neither global nor Otsu methods consistently preserved all the features of the marker, despite their speed. Conversely, adaptive Gaussian provided good results but required over a tenth of a second. Adaptive mean produced results typically on par with the Gaussian equivalent, while requiring a sixth of the processing time. Therefore, adaptive mean thresholding was selected as the default thresholding method. Experimentation indicated a block size of 75 and threshold offset of 25 produced consistently good results across a broad range of marker separation and orientations, and lighting conditions.

A2.1.3 Contour Extraction

After image thresholding had generated the binary image the $Contour$ method was called. The OpenCV $findCountours$ function was used without hierarchical relations nor approximation. Contours are extracted from the binary image based upon the boarder following algorithms of Suzuki and Abe (Suzuki & be 1985).

The $findContours$ function generates a list of contours. These are stored within a vector of vectors, where each vector contains an ordered list of 2D pixel coordinates that make up the contour. This structure is illustrated below by Figure A3.4.

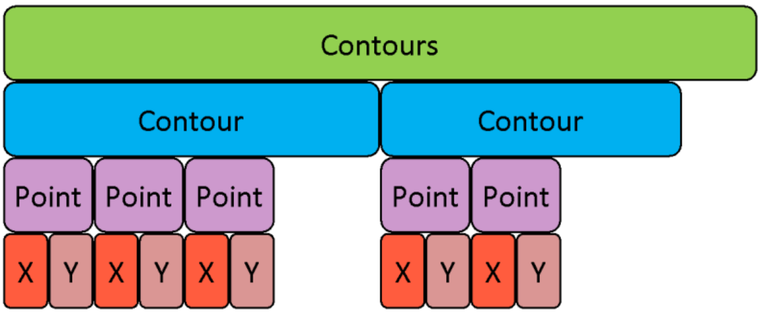


Figure A3.4: Contour structure

Each detected contour would be classed as a possible marker and have to undergo a relatively computationally intensive identification process to determine if it corresponds to a marker. As shown by Figure A3.5 below, a great deal of contours may be detected. Approximately 300 are shown in a relatively uncluttered scene.

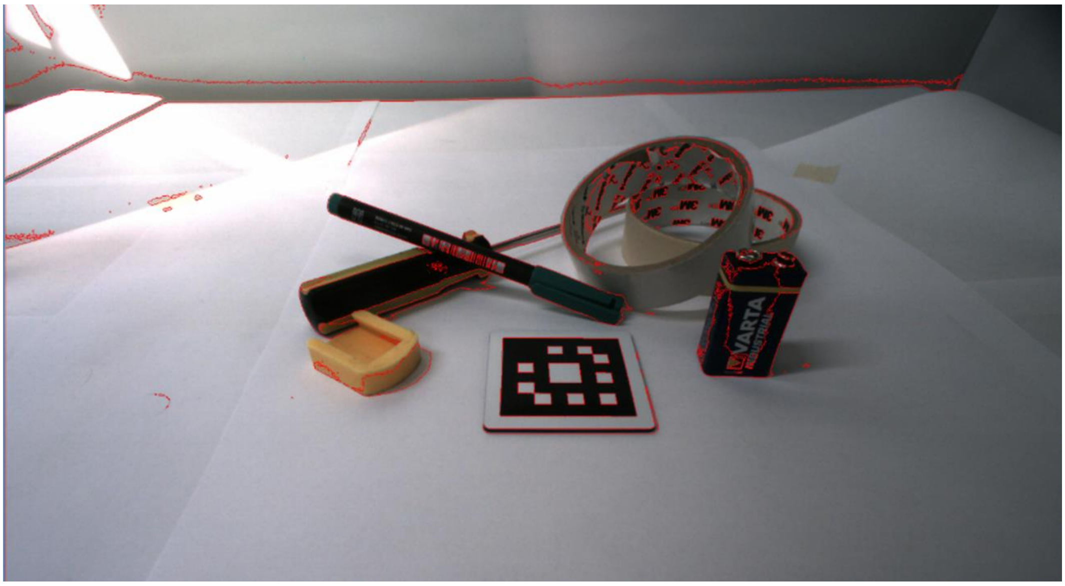


Figure A3.5: Contours detected from scene image

As may be seen in Figure A3.5 above many of the detected contours are far too small to be markers. Therefore, to avoid the computational load associated with applying the identification tests, that are bound to fail, small contours were filtered out at this stage. The list of contours was iterated through. The length then area of each contour was checked against threshold values. This order allowed contours to be discarded with the least computationally intensive test first. Only contours that passed both these tests were further evaluated. The size thresholds for these filtered was set relative to the image resolution, typically using values equivalent to a maximum size of half the image and a minimum size of a 25th of the image, although this was reduced to 100th for later work.

Figure A3.6 below shows the same scene as Figure A3.5 above, after the application of contour filtering. The original contours are shown in red, while the contours maintained after filtration are shown in blue. It may be seen that of the approximately 300 original contours only three passed initial filtering. This greatly reduced the computational cost of the corner detection stage that follows.

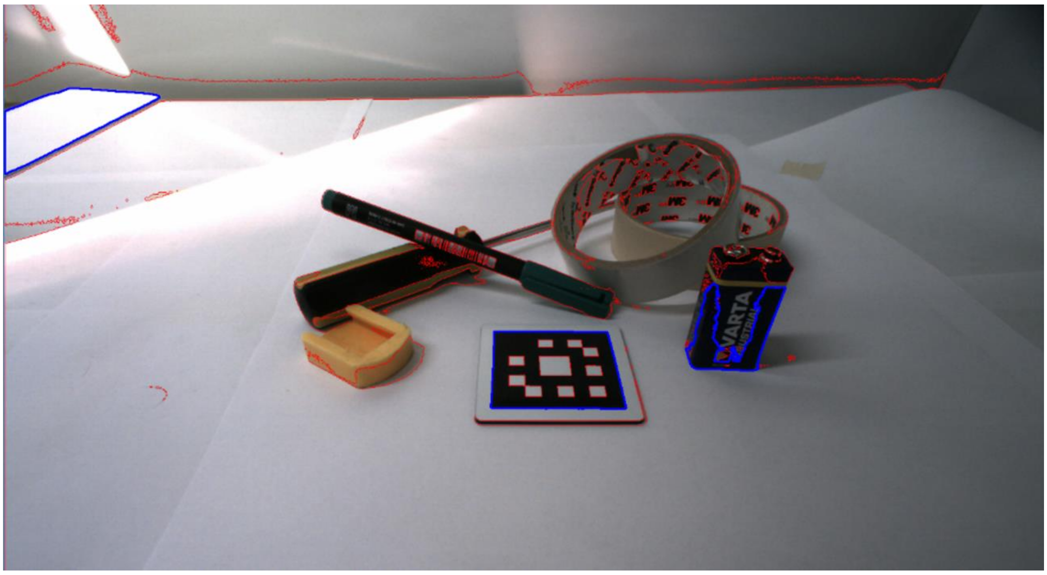


Figure A3.6: Scene contours after length and area filtration

A2.1.4 Corner Detection

Corner detection was designed as a three stage process. It was designed such that all three stages could be used or either of the last two stages omitted. Each stage provided additional accuracy to the corner detection method at the cost of increased computational intensity. While it was not expected that the system would ever be required to run upon a system not able to support the full three algorithms the stages were segmented in this manner to provide a more versatile solution. The three stages are illustrated below by Figure A3.7.

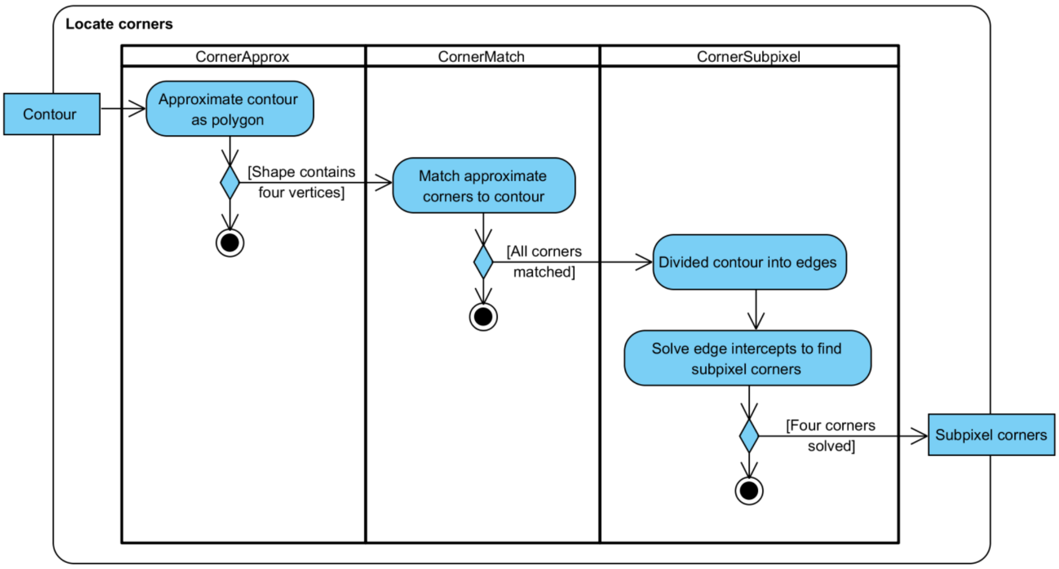

Locate corners
CornerApprox
CornerMatch
CornerSubpixel
Contour
Approximate contour as polygon
[Shape contains four vertices]
Match approximate corners to contour
[All corners matched]
Divided contour into edges
Solve edge intercepts to find subpixel corners
[Four corners solved]
Subpixel corners

Figure A3.7: The three stages of corner location

Each contour was sent to the three corner detection stages. Each stage involved filter processes designed to remove contours which were unlikely to be caused by markers. If the contour failed any of these filter processes the contour was dropped and the next contour analysed.

The first stage of contour detection approximated the contours into basic n-sided polygons. The *CornerApprox* method was passed the contour. The method approximates the shape of the contour using the Ramer-Douglas-Peucker algorithm (Ramer 1972; Douglas & Peucker 1973). Given an array of points the algorithm defines a straight line joining the first and last points, which are to be preserved. The furthest point from this line is then found and compared to the epsilon threshold value. If the distance of the point from the line is below the epsilon value the point and all other unpreserved points are removed, approximating the contour. If the point is above the threshold the point is marked to be preserved and further lines, between this point and the existing preserved points are defined. The process is repeated with these lines until all contour points are either preserved or discarded and the full contour has been approximated to the precision defined by the epsilon value. An epsilon base value of 0.01 was typically used. This was multiplied by the contour's perimeter length to give the epsilon value.

As the markers were known to be square, and thus contain precisely four vertices, only shape approximations containing four points were kept.

For debugging purposes the approximated corners could be rendered. The corners were rendered as relatively large circles upon the original image frame. Figure A3.8 below shows these corners marked on a single marker. It is noted, from comparison with Figure A3.6, that the rough contour of the battery failed to produce a four corner approximation and was thus dropped. However, the smooth contour of the paper in the top left corner produced a four corner approximation and as such is still considered by the system as a potential marker.

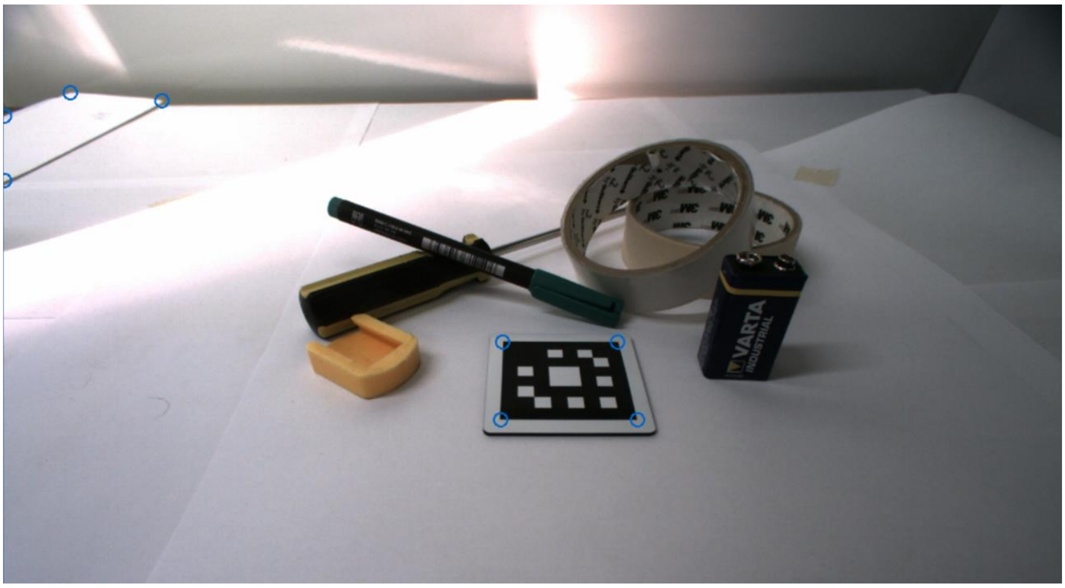


Figure A3.8: Contour corner approximation

The second stage of the corner detection process attempted to find the index of the original contour points that match the approximated corners. These indices were required by the final stage of the corner detection process, where they were used to determine sub-pixel corners. The indices were found by determining the contour point that produced to smallest Euclidian distance to each approximated corner.

Contours that failed to match all four corners were dropped. The four matched corners of passed contours were sorted. As discussed above in Section A3.1.3 contour extraction initialises in the top left of the image and progresses clockwise about contours. Therefore, sorting the indices of the matched corners helped to ensure the order of the matched corners started in the top left corner and also progressed clockwise. The matched corners are shown by the smaller circles in

Figure A3.9 below. The indices of these matched corners were returned to the main *Run* method.

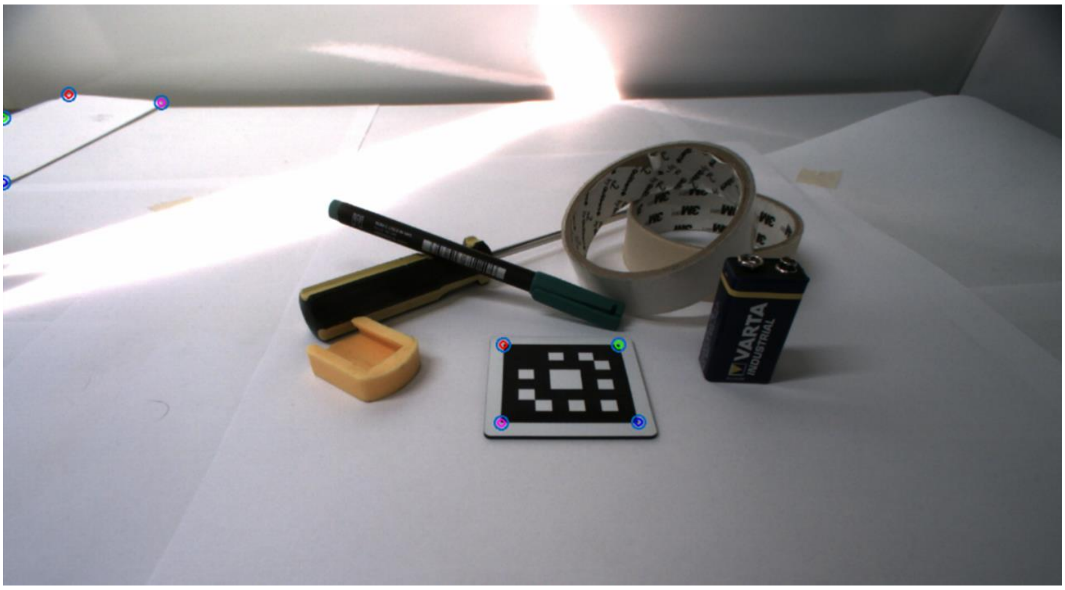


Figure A3.9: Contour corners matched to approximated corners

The final corner detection stage was designed to find the corners of the potential markers to subpixel accuracy. As discussed in Section 6.2, were the corners to be detected only to the accuracy of the nearest pixel the accuracy of the system would be severely limited, as dependent upon working distance a single pixel may represent several millimetres of space. Therefore, the system processed the discrete contour to obtain results to subpixel accuracy.

Interpellation was performed by treating the contour as four separate edges, divided by the four matched corners. Linear regression was then performed on the contour points making up the central 90% of each edge. This provided four line equations, which were solved to provide four intercepts that represent the corner positions to subpixel accuracy. This process is illustrated by Figure A3.10 below.

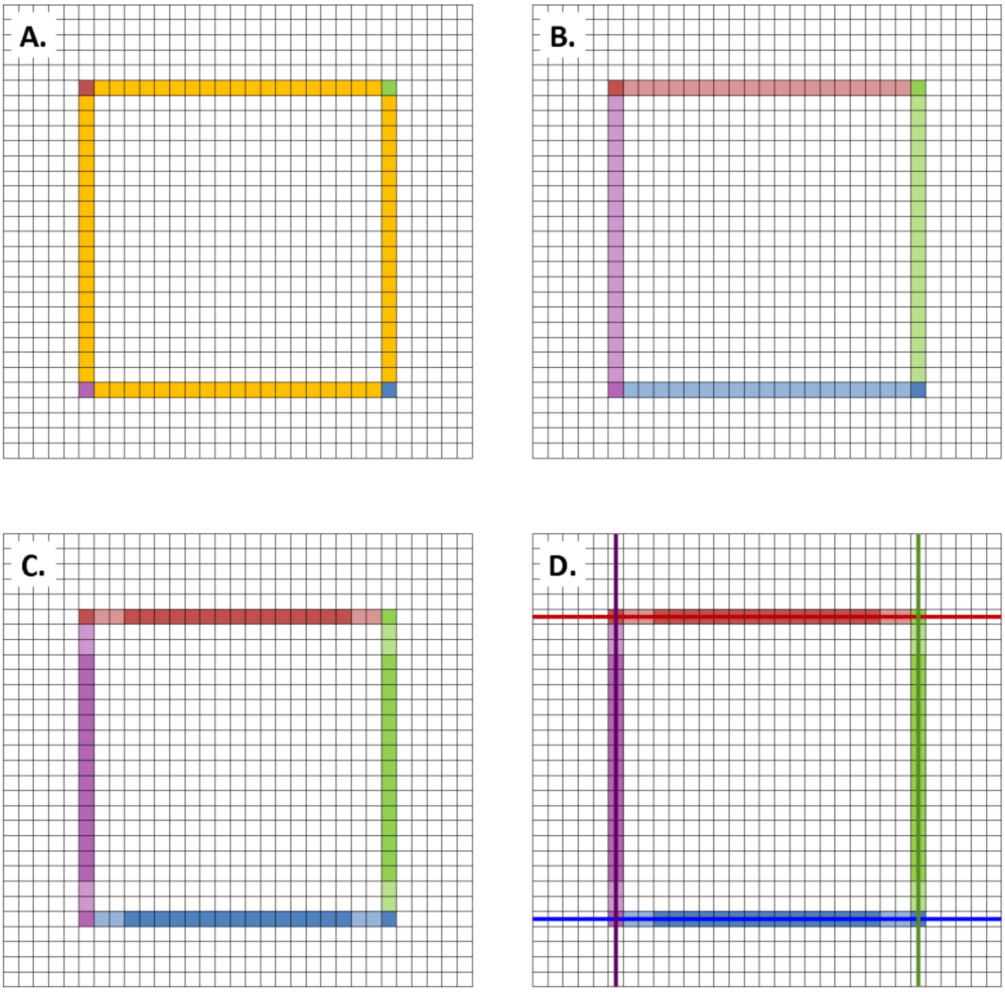


Figure A3.10: Subpixel corner detection method – The original contour (A) was divided into four edges based upon the four matched points (B). The central region of each edge (C) was then taken and a linear regression performed (D). These four lines were solved to find the four corners to subpixel accuracy.

The first, second, and thrid edges, refered to as north, east, and south respectivly, were determined using the same method. The length, defined as the number of contour points, of each edge were calculated as the difference between the indices of consecutive match corners. The edge contour was then defined as the points between the lower corner index plus 5% of the edge length and the higher corner index minus 5% of the edge length, giving the central 90% of the edge.

Ideally the first corner point should corrispond to the first contour point, that is to say the index of the first matched corner should equal zero. However, due to non-idealityies within the system, such as image distortion and thresholding errors, this could not be garenteed. Figure A3.11 below illistrates an exagurated example of a non-zero initial corner. The distortion of one of the edges causes a mid-edge point to be the first point contacted by the scanning pattern of the contour detection method, shown by the blue arrows.

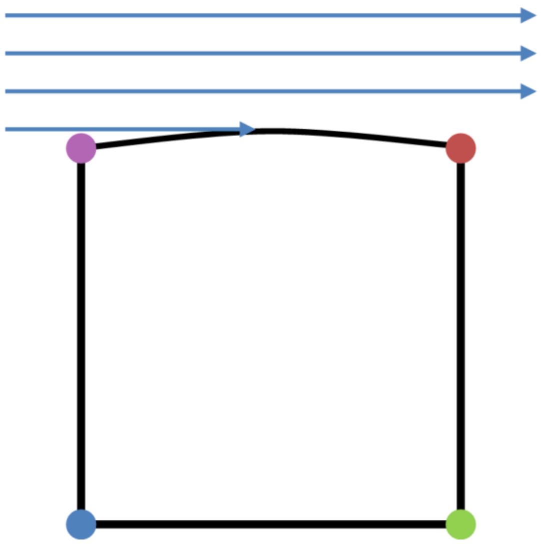


Figure A3.11: System nonidealities leading to a mid-edge point being detected as the first contour point.

Without the scenario depicted by Figure A3.11 above, the fourth edge would be defined as the length of contour between the fourth matched corner index and the final contour point. Instead, the final edge was defined as the fourth matched corner index, shown in purple, to the final contour point plus the first contour point, both shown by the arrow point, to the first matched corner index, shown in red.

Once all four contour edges were defined a linear regression was performed on each section using the OpenCV $fitLine$ function. The $fitLine$ function performed linear regression by minimising the distance function. The standard least-squares distance function, shown in Equation A3.1, was used, as it provided good accuracy without excessive computational demand.

$$\rho(r) = r^2/2 \qquad \text{(A3.1)}$$

The intercept of these four edge equations were then solved to provide corner positions with subpixel accuracy. A check was performed to ensure that each corner was within the range of the image and that four corners were solved. A final check was performed to ensure the first corner was defined as the left most corner. If not the corners were reordered. This ensured that the following orientation stage would read each marker image in a consistent fashion.

The final subpixel corners are shown by Figure A3.12 below. However, it is noted that the corner plotting function only has pixel accuracy. This often caused the subpixel corners to appear to wobble due to the value rounding. The paper in the top left corner is still seen to be considered a potential marker by the system, in addition to a new region in the top right, resulting from changing lighting conditions.

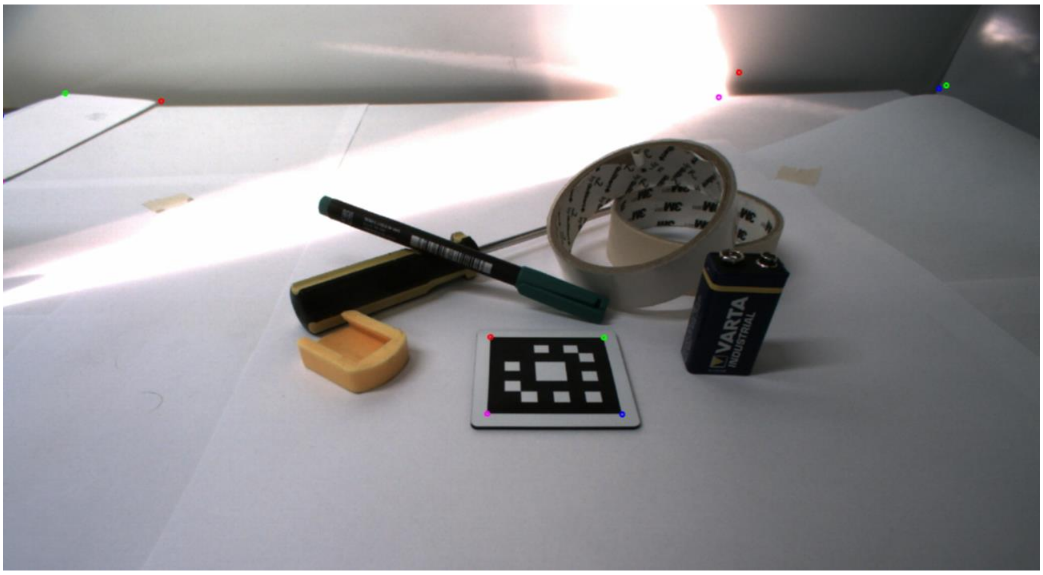


Figure A3.12: Subpixel accuracy corners with order indicated by RGBM colour

A2.1.5 Orientation and Identification

The processes of marker orientation and identification were closely linked and based upon the same grid inspection principle. The orientation region, discussed in Section 7.2.1.1, was first inspected to determine the orientation of the marker. This information was then used to read the identification region in the correct order to determine the identification of the marker.

At this stage the system had identified a contour as a potential marker and defined the four corners of this contour to subpixel accuracy. It may be seen from Figure 7.6 that a marker is divided into an 8x8 unit grid, where the orientation region forms a 6x6 grid with the identification region occupying the central 4x4 elements. It is possible to find the centre of each grid element in image space from the subpixel corner data determined by the previous sections.

The final stage of the corner detection process discussed above ensured that the corners were stored in clockwise order. Therefore, it was readily possible to pair the four corners such that they defined the start and end points of the four edges of the potential marker. Each of these edge lines were then divided into eight equal lengths, providing eight coordinates which represent the centre of each length. By matching these points between parallel edges, sixteen additional lines were produced, eight vertical and eight horizontal. The outermost lines may then be discarded. The intercept of the remaining six vertical lines with the remaining six horizontal lines were solved. This provided the centres of each orientation and identification grid unit as a series of 36 pixel coordinates in the image reference frame. This process is illustrated by Figure A3.13 below.

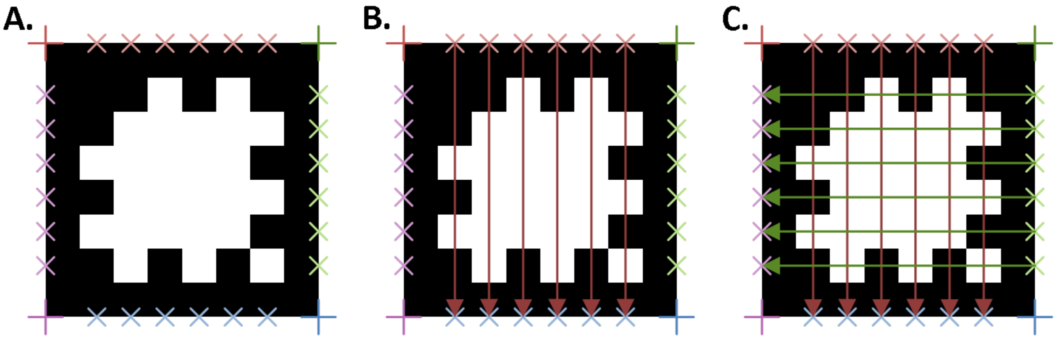


Figure A3.13: Calculation of orientation and identification unit centres. A. Marker depicted with calculated corner positions and edge points. B. Six vertical vectors determined by pairing north and south edge points. C. Six horizontal vectors determined by pairing east and west edge points.

By inspecting the thresholded binary image at each of the orientation points, using a small 2x2 region, in the correct order a 20 digit binary code is produced. This code has two functions, firstly it identifies the potential marker as a true marker or not. Secondly, it identifies the extracted corners and allows the orientation of the marker to be determined.

The vector intercept method resulted in the coordinates of the identification units being stored in the order shown by Figure A3.14 below.

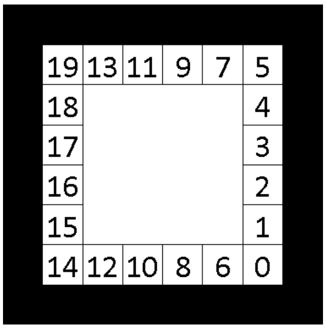


Figure A3.14: Read order of orientation units, whereby unit 19 is next to the most top left corner in the image reference frame.

Given this read order, and the design of the orientation region presented previously by Figure 7.5, a true marker may produce one of only four orientation codes as shown by Figure A3.15 below.

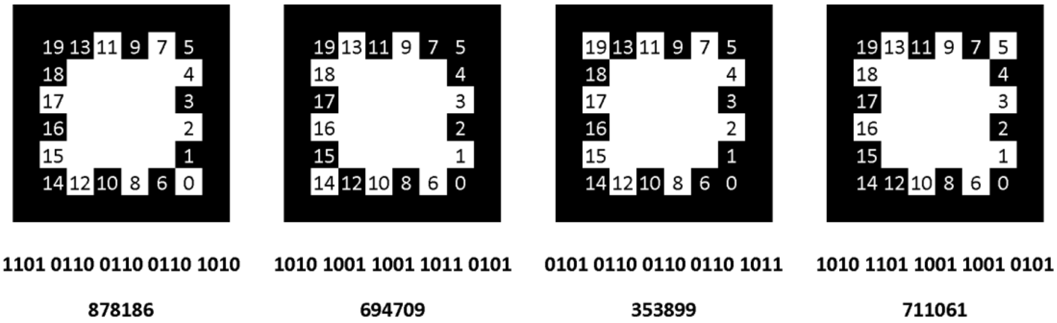


Figure A3.15: Four possible orientation results of a true marker.

The system may therefore determine the orientation of the marker, and corner and identification order adjustments required, based upon which of these four codes the marker produced. Evidently, a potential marker that does not return one of these four codes is classified as a non-marker and dropped by the system. This represents the last point at which a potential marker may be dropped under normal conditions. As such, all potential markers that pass this stage were considered full markers by the system.

The read order of the identification points were adjusted for non-vertically aligned markers, as shown in Figure A3.16. It may be seen that the corner order, and the read order of the identification region, has been rotated clockwise to match that of the physical marker.

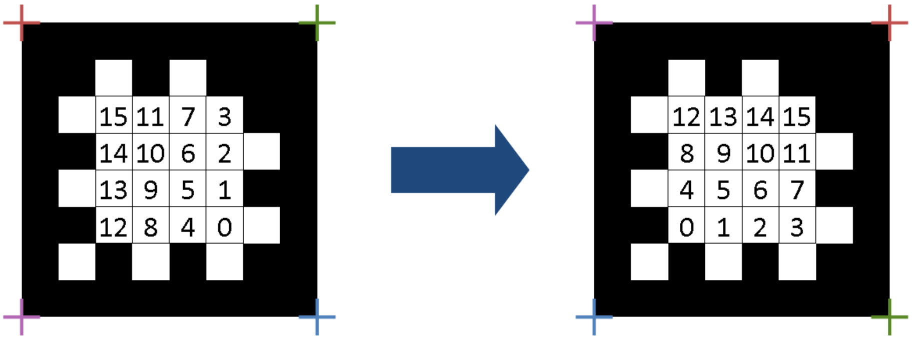


Figure A3.16: Reorientation of marker corners and identification read order.

Now orientated the same approach was used to read the identification units. The orientation and identification units are shown below in Figure A3.17 by red and green and blue and magenta points respectively. The paper in the top left corner produced an invalid orientation code, and as such identification was not performed.

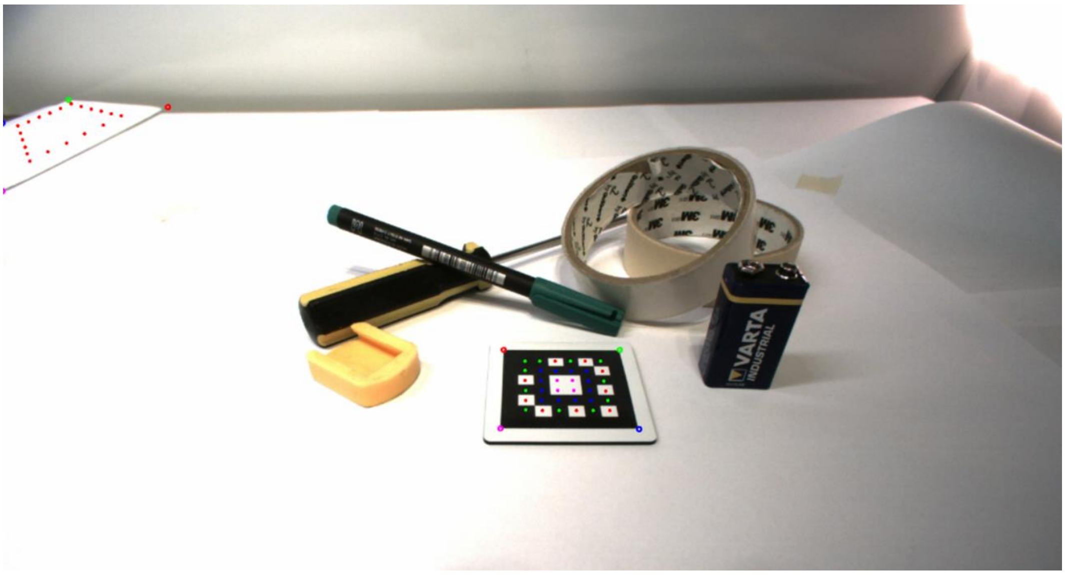


Figure A3.17: Marker orientation (red and green) and identification (blue and magenta) points

Figure A3.18 depicts several markers at different orientations. It may be seen that the system corners, shown as red, green, blue, and magenta points, are correctly orientated with each marker, whereby the red marker corresponds to the origin corner.

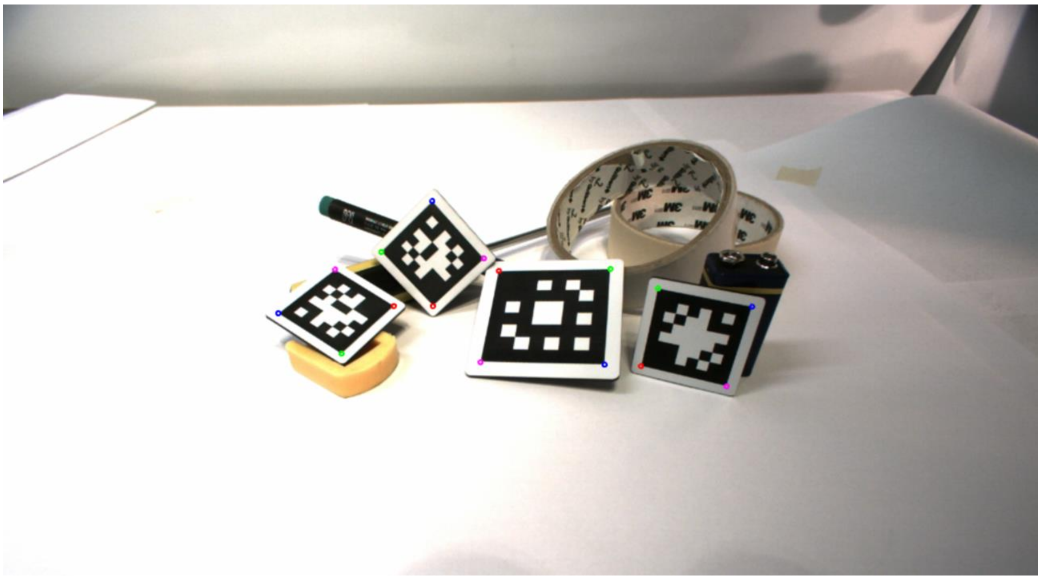


Figure A3.18: A set of differently orientated markers demonstrated the correctly orientated corners.

With this, the 2D marker extraction, corner detection, orientation, and identification from the image frame are completed. The obtained information may now be used to estimate the pose of the marker relative to the camera.

A2.1.6 Timing

Before discussing pose estimation the processing time required for the stages of marker detection are briefly discussed. Figure A3.19 below shows the timings of the five detection stages discussed above. Corner detection is displayed as its three separate stages, although corner approximation required negligible processing time.

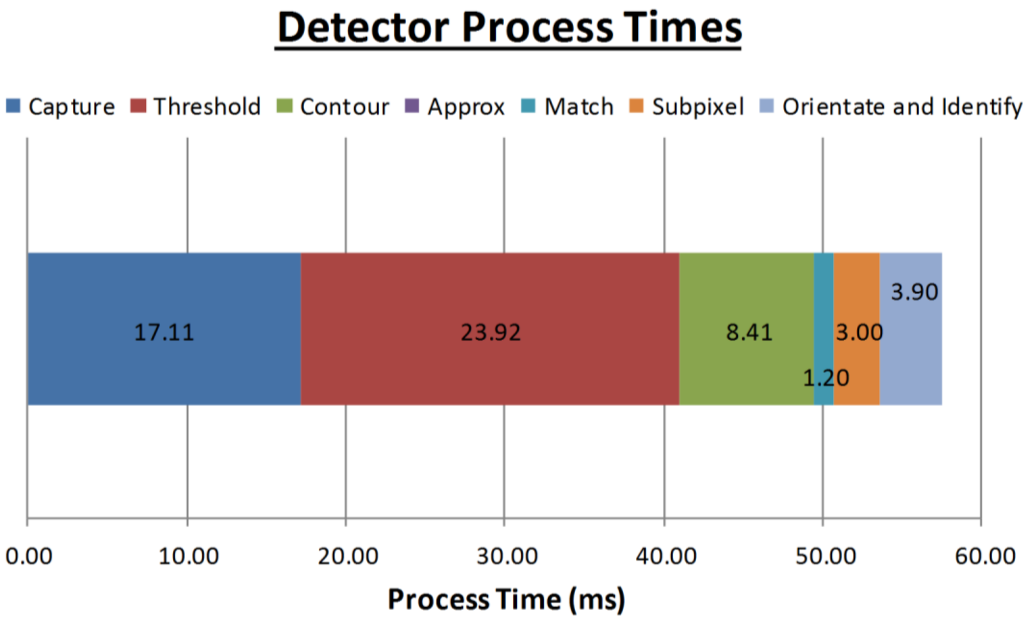


Figure A3.19: Process times of the detector stages

As seen from Figure A3.19 image thresholding is the most time consuming process ($P < 0.01$), following by image capture ($P < 0.01$) and contour extraction ($P < 0.01$). The stages over which most control was possessed produced the lowest process times.

These timings indicate that threshold should be primarily investigated to minimise the overall latency of the system. However, as discussed by Section A3.1.2, the adaptive mean thresholding method selected was the least computationally intensive method that produced adequate results. Even the most simplistic global approach provided a time saving of only 5 ms.

OpenCV provided a GPU based implementation of the *threshold* function. However, as the current implementation supported only global thresholding it was unable to provide the results required. Furthermore, the conversion from the standard image format to a GPU compatible equivalent was relatively costly, thus reduced the advantage of the GPU implementation.

The duration of thresholding could be reduced by down sampling the image, and thus thresholding a lower resolution image, containing fewer pixels. However, this would effectively reduce the spatial accuracy of the image, and therefore the pose estimation.

A final option to improve thresholding, and other stage, processing times would be improved hardware. An improved processor with a higher clock speed could considerably reduce the processing times.

Measurements of the total system latency were performed, defined as the period between image capture and final augmented display. A mean latency of 188±48 ms was calculated. A separate experiment performed using the camera purely as a video streaming device, without an image processing produced a mean latency of 122±30 ms. Therefore, it is concluded that the physical image acquisition and transmission of data was responsible for the majority of system latency observed.

The effect of multiple markers upon the final three stages – contour extraction, corner detection, and orientation and identification – of the marker detection process was investigated. Between zero and ten markers were images simultaneously and the process times of the final three stages recorded. The contour extraction time was found to be the dominant factor and also independent of marker count (P = 0.27). It was therefore omitted and the remaining times plotted in Figure A3.20 below.

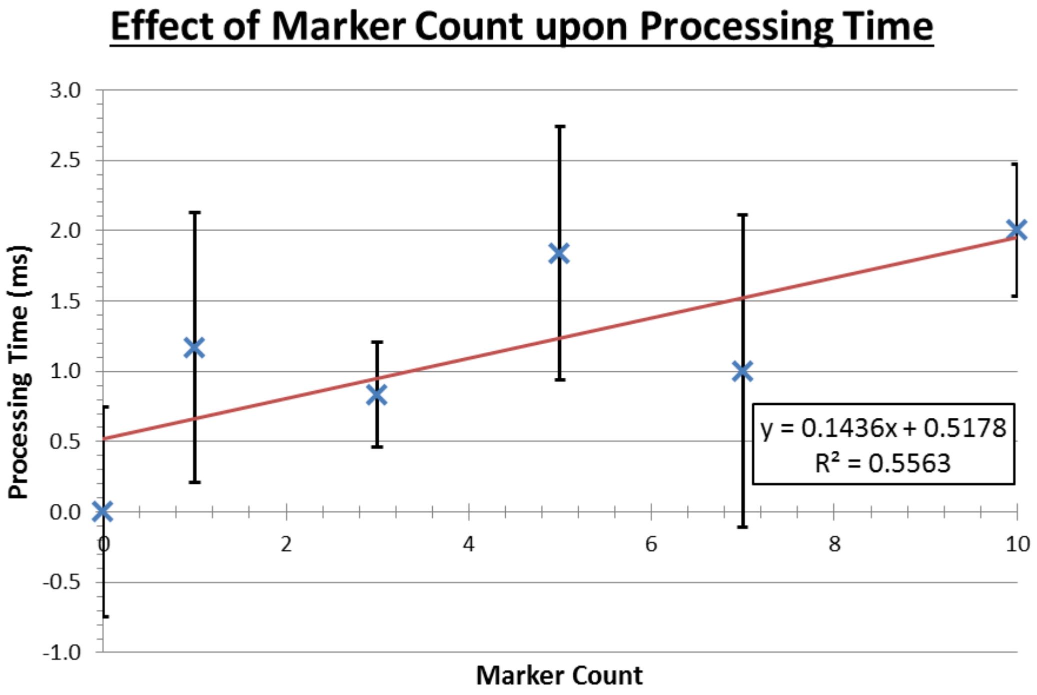


Figure A3.20: Effect of marker count upon corner detection and marker orientation and identification process time

A weak positive relation between marker count of process time was observed ($R^2 = 0.56$). However, all increases aside from three to five markers ($P = 0.03$) failed to reach significance. As did the time increase between one and ten markers ($P = 0.06$).

While increased markers tend to increased processing time this is unlikely to exceed the thresholding time. Based upon the linear regression of Figure A3.20 the process time of corner detection and marker orientation and identification are not predicted to exceed that of thresholding until approximately 160 markers are simultaneously imaged. Therefore, further optimisation of these methods appeared unnecessary, thus pose estimation could be performed.

A2.2 Marker Pose Estimation

As previously discussed marker pose estimation is the process of determining the rotation and translation of the marker in the reference frame of the imaging camera. The approach taken by the second generation system is equivalent to that of the first generation system, as described in Section 6.2, in that it is based upon reverse projection of the pinhole model.

The pose estimation was implemented upon the OpenCV $solvePnP$ function. The $solvePnP$ function is an extremely powerful tool. Operating under similar principles to those discussed in Section 6.2 the function may estimate the 3D pose of a marker in world space based upon the 2D location of its four corners in image space. The function offers several methods to perform this operation. However, basic experimentation indicated that the iterative approach produced the most robust and consistent results without substantially increasing computational effort.

The iterative approach performs a non-linear least-square fitting between the corner image points, found by Section A3.1 above, and the corner image points produced by projecting the corners of a virtual marker at the estimated pose in world space. The projection of the virtual marker into image space is performed using the intrinsic camera matrix as discussed in Section A1.5 while also incorporating the effects of lens distortion characterised by the distortion parameters. This ensures an accurate projection that closely mimics the projection undergone by a physical marker. The least-squares function is shown by Equation below, whereby K_D represents the intrinsic matric of the camera combined with the distortion model, $\hat{R}$ and $\hat{T}$ the optimised rotation and translation, respectively, combined to give the pose matrix, M_i the corners of the virtual marker in world space, and o_i the observed corner positions in image space.

$$\Sigma^2 = \sum_{i=1}^{4} \left\| K_D \left[\hat{R} \middle| \hat{T} \right] M_i - o_i \right\|^2 \tag{ A3.4 }$$

OpenCV implements a Levenberg-Marquardt (LM) optimisation of the iterative least-square fitting method (Levenberg 1944; Marquardt 1963). LM intelligently combines two fitting methods – gradient decent method and Gauss-Newton method - to benefit from a combination of their advantages. The gradient decent method calculates the gradient of the chi-squared error function with respect to the varied parameters. The parameters are then adjusted by the inverse of this gradient. This tends the chi-squared value towards its minimum. The Gauss-Newton method simplifies the Newton method by using an approximation of the second-order Hessian matrix normally required to converge upon the solution.

LM initially utilises the gradient decent method to locate the region of the minimum. Then as the solution is neared LM tends towards the use of the Gauss-Newton method.

The solvePnP function returns the estimate pose of the marker as a three element rotation vector and a three element translation vector. For both tracking and graphical purposes it is considered more convenient to transform these vectors into a single homogeneous transformation matrix. This may be achieved by first performing a Rodrigues conversion upon the rotation vector. The resulting rotation matrix is then combined with the translation vector into a 4x4 homogeneous matrix, as shown by Equation A3.5.

$$P = \begin{bmatrix} Rod(\vec{r}) & T^T \\ 0 & 1 \end{bmatrix} \tag{ A3.5 }$$

OpenCV and OpenGL use two different coordinate systems, as shown by Figure A3.21. While both systems utilise a right-handed coordinate system the direction of the axial vectors is inverted in both the Y and Z-axes. To simplify the system it was desirable to use a single coordinate system for all aspects. If the OpenCV system were used the pose would require conversion each time an object was rendered. However, using the OpenGL system throughout would require a single conversion per marker per frame. Therefore, it was decided to adopt the OpenGL coordinate system throughout the system.

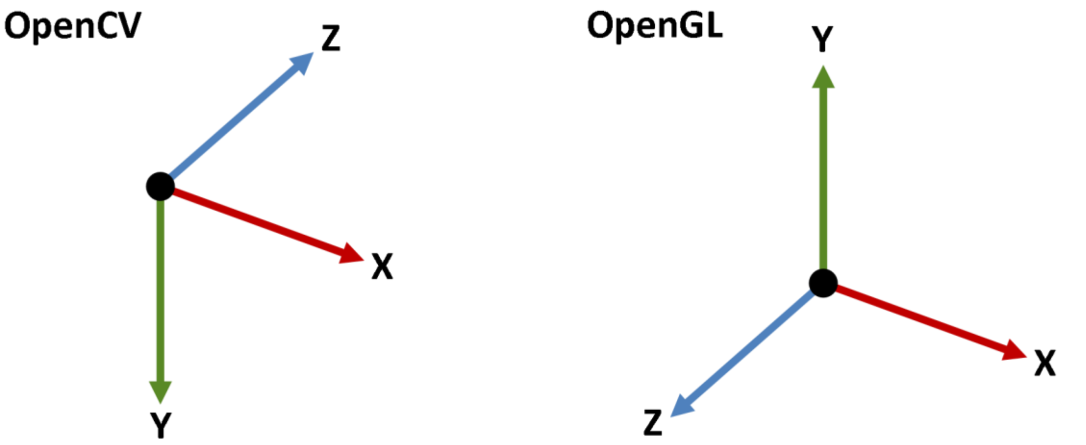


Figure A3.21: Coordinate systems used by OpenCV and OpenGL

Inspection of Figure A3.21 indicates that these inversions are equivalent to a rotation about the X-axis, which may be performed by the homogeneous rotation matrix of Equation A3.6, transforming both the rotational and translation elements

$$R_{CV\rightarrow GL} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{(A3.6)}$$

Once all 2D marker detection, as discussed above, the pose of each corner set was calculated using $solvePnP$. In addition to the marker corners found within image space the function required the physical coordinates of the corners in marker space. It was logical to define the centre of the marker as its origin. Therefore the coordinates of the four corners are given by Figure A3.22 below.

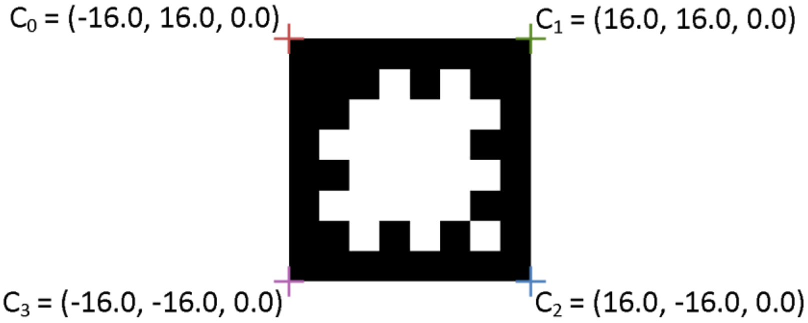

C_0 = (-16.0, 16.0, 0.0)
C_1 = (16.0, 16.0, 0.0)
C_3 = (-16.0, -16.0, 0.0)
C_2 = (16.0, -16.0, 0.0)

Figure A3.22: Physical corner coordinates of standard marker

In addition to the two versions of corner coordinates the $solvePnP$ function was also passed the intrinsic matrix and distortion parameters calculated for the camera system during calibration, discussed in Section 7.3.1. These allow the function to perform the projection realistically, based upon the characteristics of the physical camera

With the marker poses calculated and stored within easy access of all relevant classes, the various tracking and rendering operations may be performed. Figure A3.23 shows several markers, each with an augmented cube rendered using the pose matrix of each marker. The following section utilises the pose information for tool tracking.

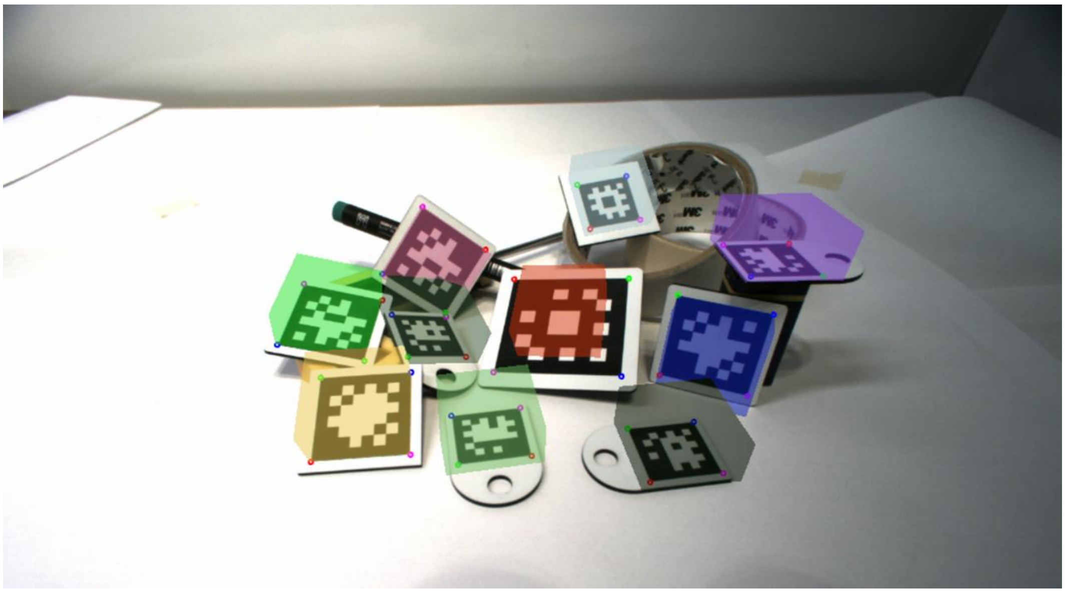


Figure A3.23: A selection of markers with randomly coloured cubes augmented on to them

A2.3 Tool

As described in both Chapter 4 and Section 7.2 the tool facilitates a number of functions. The following sections describe first the tracking and probing functionality of the tool before discussing the different modes of user interface, including the use of the external interface board.

A2.3.1 Tracking

The primary function of the system is to accurately determine the 3D location of the tool tip with respect to reference markers. Throughout this chapter the phrase tool is used to refer to the object tracked by the system. Including both the probe tool used during the probing experimentation discussed in Section7.3.3, and the resection tool used by Section 7.3.4. Both tools terminated in a spherical tip, and where thus tracked and calibrated in the same manner. The position returned by the tracking methods gives the position of the centre of the spherical tip relative to the reference marker. Due to the exclusive use of spherical tools the orientation of the tool tip was not required by the system and was thus omitted from calculations.

This section shall first describe the tracking methods of the tool before describing the calibration required to determine the tip position relative to the tool. As the second generation system utilises the same tool mounted camera approach of the first generation system the mathematics involved follow that of Section A2.1.

As with the first generation system, the second generation system calculates the pose of a marker relative to the camera. This pose is then inverted to provide the pose of the camera in the coordinate system of the marker. By applying the offset between the tool tip and the camera in the camera coordinate system it is possible to calculate the position of the tool tip in the coordinate system of the marker.

Tip calibration also followed the principles outlined in Section A2.1. The user would place the tip of the probe at a known position relative to a marker. Equation A2.4 was then applied using the camera pose and known tip position to calculate the tip offset in the coordinate system of the camera.

As with the first generation implementation the offset was initially calculated the mean value of several samples. However, in a further attempt to minimise the effect of sample errors, outliers were removed. The squared Euclidian distance of each offset from the mean offset was calculated. The mean offset was the again calculated using only the nearest 80% of the samples. This approach removed any outliers resulting from momentary errors within the system and helped to ensure an accurate tip calibration. The RMS distribution of the included samples was calculated and displayed to confirm a successful calibration. Values below 0.02 mm were typically considered acceptable. Figure A3.25 below shows the system immediately after calibration with the RMS result shown in millimetres in the top left corner.

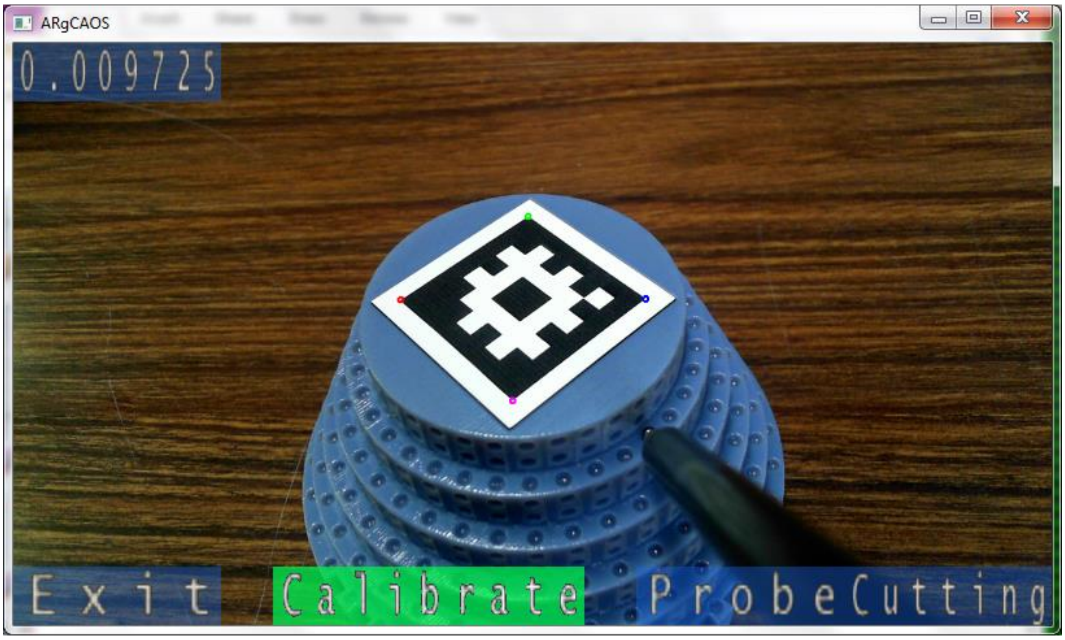


Figure A3.25: Second generation system calibration

A2.3.2 User Interface

In addition to the probe and resection burr, the term tool also refers to the UI elements associated with the hand-piece. As described by Section 7.2.1 the system made use of a USB interface board to facilitate interaction with external components. This includes both user input from external buttons, and system output to control the resection burr. The various components of the tool UI are demonstrated by Figure A3.26 below, which shows the activity diagram for setting a single digital output and for reading all digital inputs.

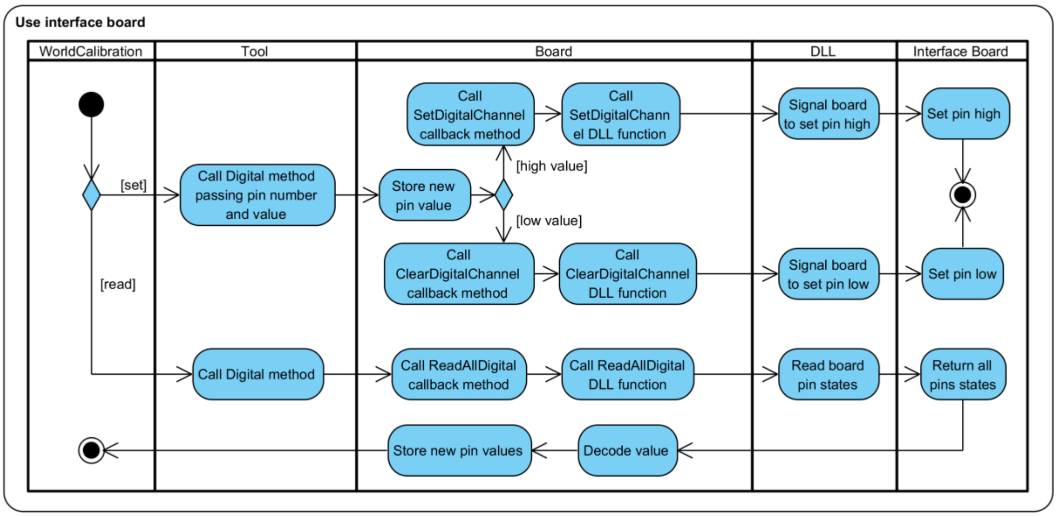

Use interface board
WorldCalibration
Tool
Board
DLL
Interface Board
[set]
[read]
Call Digital method passing pin number and value
Store new pin value
[high value]
[low value]
Call SetDigitalChannel callback method
Call SetDigitalChannel DLL function
Signal board to set pin high
Set pin high
Call ClearDigitalChannel callback method
Call ClearDigitalChannel DLL function
Signal board to set pin low
Set pin low
Call Digital method
Call ReadAllDigital callback method
Call ReadAllDigital DLL function
Read board pin states
Return all pins states
Store new pin values
Decode value

Figure A3.26: Activity diagram for setting a single digital output and reading all digital inputs

The user interface functionality detailed above shall be further explored in Section A3.5 while describing the use of the tool for resection. However, discussion of the graphical library of the system is first required.

A2.4 Graphic Library

As discussed by Section 7.1.2 the graphical elements of the second generation system were to be powered by the OpenGL library. In addition to OpenGL the closely associated GLUT library was used to provide windowing interface, while the AssImp library was later employed to provide higher-level model loading.

The graphical system may be broadly divided into two sections, primitives and shaders. Primitives represent the objects to be rendered, be they dots, lines, text boxes, or elaborate 3D meshes. Shaders include the code run upon the GPU in addition to the associated classes linking between the two systems.

A core task of the system to provide a convincing AR experience is the correct generation of the WVP matrix. The WVP is the amalgamation of three matrices; world, view, and projection, as shown by Equation A3.7.

$$WVP = P * V * W \qquad \text{(A3.7)}$$

The world matrix moves the vertex coordinates of a model from its local space to a global coordinate system, allowing an object to be placed within a world. The view matrix then transforms these coordinates into a camera coordinate system, such that they are relative to a viewer. Finally, the projection matrix applies a projective transformation, such that more distant objects appear smaller, as they do naturally, and moves the coordinates into image space. By combing these three matrices into one for vertices of an object, such as a 3D model, may be moved from their local coordinate system into image space, in a single operation.

In the case of the presented system, the world matrix defines where an object is relative to a marker. The view matrix defines how the camera is looking at the marker, and comes from the estimated pose. While the projection matrix attempts to estimate the transformation real objects underwent while forming the camera image.

The OpenGL projection matrix defines a frustum – a volume within which objects are rendered, as shown by Figure A3.31. This volume is defined mathematically by the perspective projection matrix shown by Equation A3.8. Where n and f are the distance of the near and far planes respectively from the origin, and r and l are the left and right limits of the near plane, respectively. Finally, t and b are the top and bottom limits.

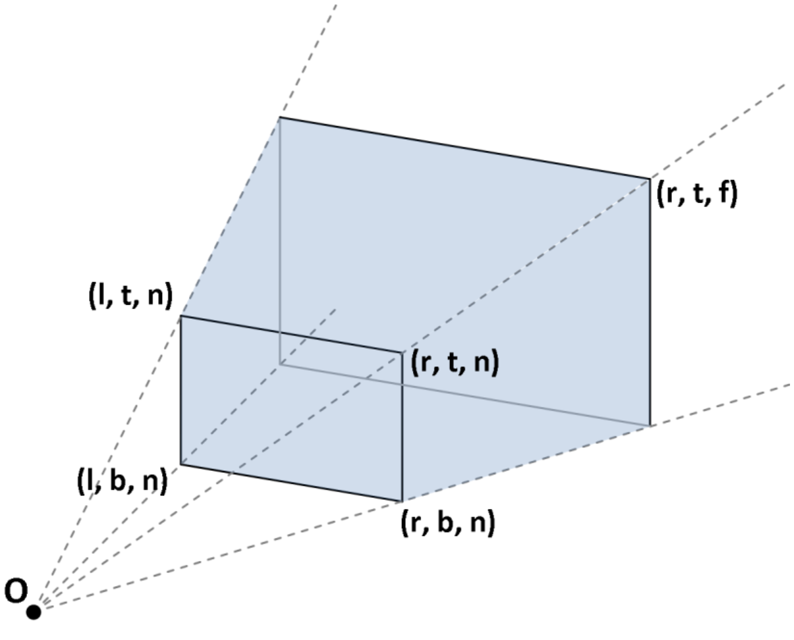


Figure A3.31: OpenGL perspective projection frustum

$$P = \begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix} \tag{A3.8}$$

It is observed that Figure A3.31 resembles the graphical representation of the intrinsic camera parameters. This is indicative of both figures presenting a perspective projective model. The OpenGL projection matrix appends near and fear clipping planes to control which objects are rendered based upon their distance from the origin. The remaining parameters of the projection matrix may be calculated by scaling the intrinsic parameters to match the resolution of the display

Primitive were originally manually programed, whereby the vertex and colour values were hardcoded. This limited the system to the use of very simple models based upon geometric shapes. Therefore, the system was expanded with the use of the Open Asset Import Library (AssImp 3.1.1). AssImp loads objects from a variety of model formats, such as Wavefront object (.obj) and standard template library (.stl), and converts them into a common intermediate format. A standard method was then implemented to load this intermediate format onto the GPU using OpenGL.

In addition to vertex information, such as position and normal angle, the new system also imported texture information, allowing models to be coloured in more complex ways, as shown by Figure A3.32 below.

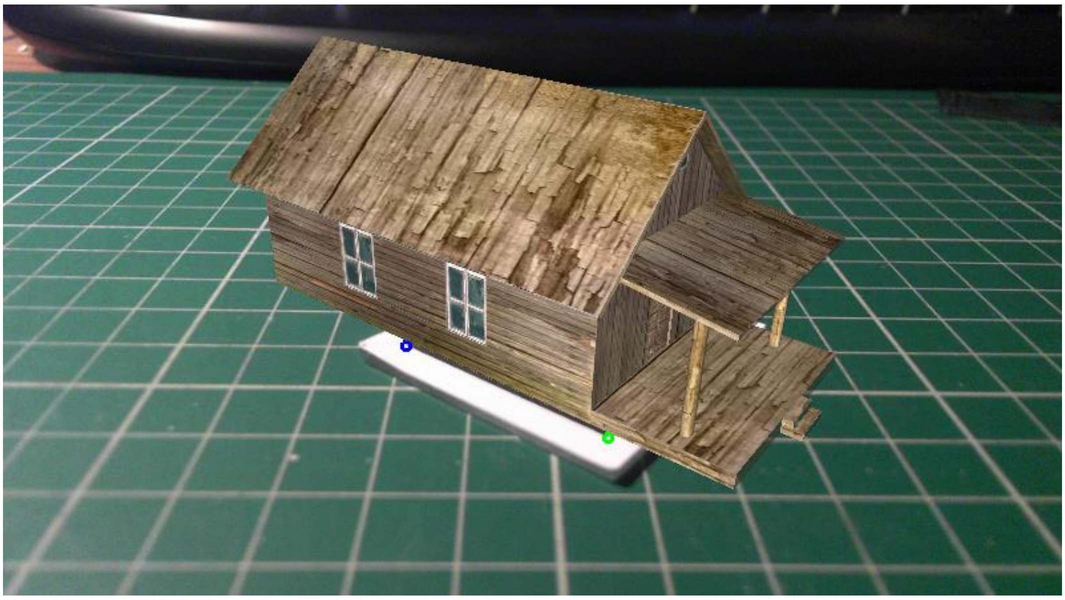


Figure A3.32: Example of complex textured 3D model imported via AssImp

With the primitive objects established the system required shader programs to correctly render the objects to screen. Shaders are programs written in GLSL which are compiled and run upon the many cores of the GPU to perform rendering tasks. Different types of shaders are designs to perform different roles within the render pipeline. The shaders developed as part of this research were either vertex or fragment shaders. As the names imply vertex shaders manipulate vertex data, while fragment shaders manipulate the fragments, pixel precursors, of the image to be written to the framebuffer. Several shaders were developed to provide different rendering effects, such a realistic lighting simulation provided by Phong lighting upon fully textured 3D objects, or basic flat lighting upon uniformly coloured 2D objects. An overview of s

With a general overview of the shader and primitive structure given the resection volume may now be discussed. The resection volume effectively builds upon many of the techniques discussed above.

A2.5 Resection Volume

This section describes the implementation of an improve bone and resection model, based upon ray-casting performed with modern programmable pipeline OpenGL.

A2.5.1 Background

Chapter 6 concluded that the bone surface model system implemented by the first generation system had several short comings. Therefore, an alternative approach was investigated for the second generation system. The results of the first generation system indicated that a planar model would not provide the functionality required by the system. As such it was concluded that a full 3D solution – as opposed to the 2.5D previously implemented – would be required. Based upon these findings the updated requirements list, shown in Table A3.3 below, was produced for the bone surface model.

Bone Surface Model Requirements
Support a fully 3D surface with overlaying areas
Support a surface with a surface normal range of 360°
Able to be built or expanded in real-time
Able to be resected or reduced in real-time
Support under-cutting during resection
Support resection control

Table A3.3: Updated bone surface model requirements

Based upon these updated requirements a review of literature provided three potential technologies: constructive solid geometry (CSG), block volumetric rendering, and direct volumetric rendering.

Constructive solid geometry uses standard rendering techniques, such as those already discussed above in Section A3.4, to render objects as meshes. However, CSG adds the ability to perform binary operations between two object meshes. Binary operations allow one mesh to modify the other. For example, two meshes may be added together to produce a single mesh whose outer surface corresponds to the non-intercepting region of the two constituent meshes. Subtraction, which would have great potential during resection, removes one mesh from the other, included any intersecting regions. These concepts are illustrated in 2D by Figure A3.36 below.

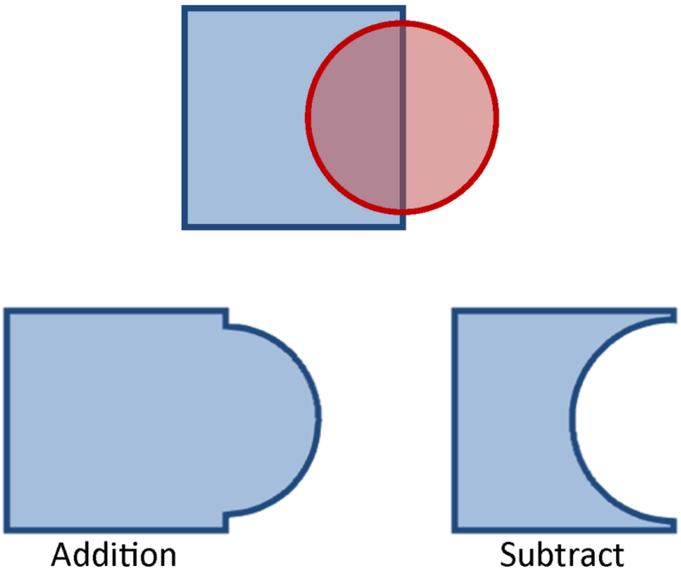


Figure A3.36: Binary addition and subtract operations of CSG methodology

Block rendering enables volumetric data to be rendered by first converting to an intermediate form which is then rendered. Block rendering is a relatively simple form of this approach whereby each volumetric data point in converted into a cube, with side lengths equivalent to the spacing between the data points, forming an array of voxels. These voxels are then rendered. To reduce the computational intensity of this task several optimisation methods are deployed to reduce the number of blocks that require rendering to the minimum. Figure A3.37 below illustrates this methodology.

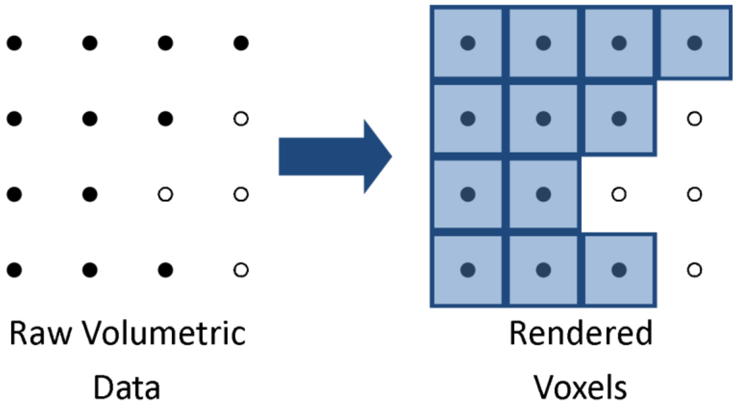


Figure A3.37: Voxel generation of block rendering methodology

Direct volume rendering renders volumetric data sets without conversion to an intermediate form. To allow an alternative form of rendering is often used. As opposed to the raster method implemented by OpenGL a form of ray-tracing is used.

Ray-tracing renders an image by producing many vectors between the system origin and each pixel of the image plane. The system then traverses along each of these vectors until an object to be rendered is reached. The properties of this object then dictate the colour of the pixel associate with that vector, either directly or by producing a secondary vector, simulating reflection. Figure A3.38 illustrates the principle of ray-tracing.

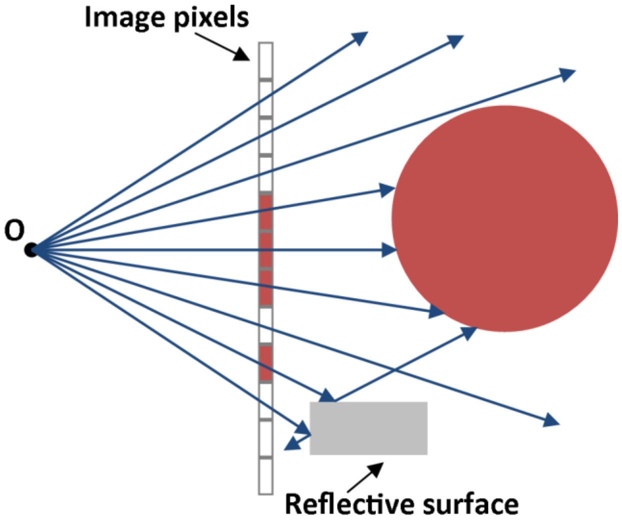


Figure A3.38: Principles of ray-tracing, showing both direct and reflected rays

These three potential technologies were compared and contrasted to select the best option. The comparison, shown in Table A3.4, defined seven key areas. Aesthetic quality refers to the appearance of the technology, while functional quality measures how well the system would perform. Intuitiveness gauges how naturally the system may guide the user through resection. Real-time guidance, and thus rendering, is defined by Section 4.3 as a core requirement. Therefore, the computational intensity of the technology must allow this. Data size is also a consideration. The memory requirements of volumetric data sets increase by the third power when increasing either resolution or dimensions. The complexity of implementing the general system is then evaluated, as well as the specific complexity of implementing the resection functionality.

Method	CSG	Direct Rendering	Block Rendering
Aesthetic Quality	Very high	high	Medium
Function Quality	high	high	High
Intuitiveness	high	high	High
Computational Intensity	high	medium	Medium
Data size	low	high	High
General complexity	Low	Medium	High
Resection complexity	Very high	medium	High
Notes	Compatible with tradition shader techniques	Directly compatible with CT and MRI data	compatible with CT and MRI data

Table A3.4: Resection volume technology comparisons.

Summarising the results of this comparison, all three systems offered a high quality finish; with CSG offering a slightly more aesthetics finish due to its natural compatibility with standard shader techniques. However, this was likely offset by the increased computational intensity. The two volumetric methods had a higher memory demand due to their use of volumetric data. However, individual data elements are smaller, and even large high resolution volumes could be easily handled by the large memory available on modern systems.

CSG would be compatible with the existing shaders discussed in Section A3.4. Therefore, it presents a relatively low general complexity. However, implementing the resection aspect is considerably more complex. Direct rendering would require new shaders and new shader techniques, such as double pass shading, and therefore would initially be more complex to implement. However, once implemented the updating of the model to accommodate resection would be notably more simplistic than that of CSG. Block rendering follows a similar form. The initial implementation would be relatively complex, potentially requiring use of the geometry shader discussed in Section 7.1.2.2. However once implemented model updating would be of a similar complexity. Potentially more complex than direct rendering, to ensure the intermediate model generation remained real-time.

A final advantage to the direct and block rendering technologies is their inherent compatibility with both CT and MRI data, which are volumetric data with similar format. Although as discussed in Chapter 4, this current system is to be developed as an image-less system, and thus does not require or make use of pre-operative imaging, the ability to render CT imagery in an augmented fashion may prove beneficial for future implementations of the system.

Several low complexity implementations and examples were investigated and an implementation based review of the literature was performed. Although, the initial candidate CSG proved difficult to implement in a reliable and consistent manner. There was a considerable amount of literature available for implementing a block rendering system, due to the popularity of the game Minecraft, which utilises block rendering as its core feature. However, it was desirable for the system to present a high model resolution, between 0.25 and 0.1 mm voxel sizes. Detail of the blocks was likely to be lost at this scale. Furthermore, many blocks would have to be visible at any one time. This would reduce the effectiveness of several of the optimisation methods deployed by block render systems. There were several high quality information sources for ray-tracing, again due to their use in the games industry for realistic lighting and reflection simulation. The desired resolution of the model was compatible with ray-tracing methods. Furthermore, as most of the heavy processing was performed upon the GPU the overall system was not overly computationally intensive and could easily meet real-time requirements. Finally, in comparison at least to CSG, direct rendering provided a very simple method to colour the model to provide intuitive resection guidance.

Therefore, direct volume rendering via a ray-tracing system was selected and the surface model methodology. The following sections discuss the design and implementation of the volume system.

A2.5.2 Design

The proposed volumetric rendering system would place the volumetric data directly into the scene. The ray-tracer may then transverse along the ray vectors into the volumetric data. Each time a vector was incident upon a data element the value would be inspected. The value would indicate if that voxel contained visible material or not. If visible the voxel value defined the colour of the fragment associated with the vector. If not the system would continue to transverse along the vector. This would allow the system to directly render the data stored within the volume. Figure A3.39 below briefly illustrates the proposed process.

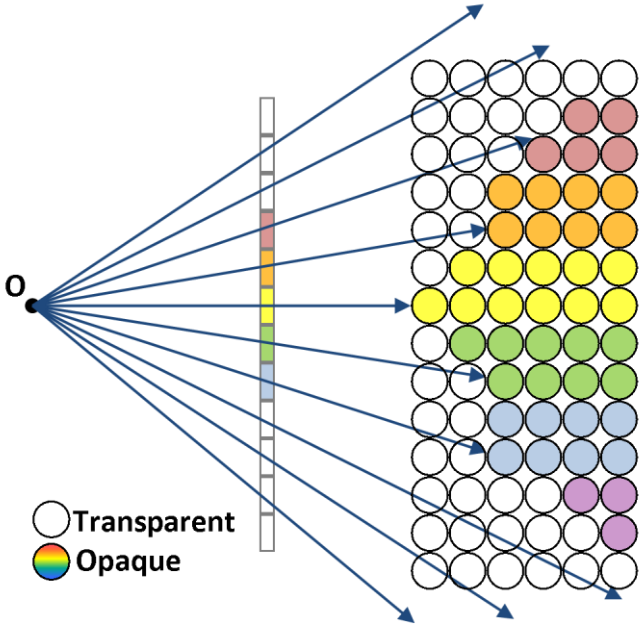


Figure A3.39: Illustration of voxel based ray-tracing

As stated OpenGL is a raster renderer. Therefore, a combination of different techniques is required to simulate ray-tracing using OpenGL. The presented solution utilised a two-pass render technique. Two-pass rendering performs two render cycles per frame. The first pass typically writes the resulting image to a user generated framebuffer, as opposed to the buffer displayed to screen. The information stored within this framebuffer is then used by the second render pass to generate the final image which is written to the back framebuffer, to be displayed to the screen.

The first render pass is used to help calculate the ray vectors. These are then transverse in the second render pass to render the volumetric data. This approach is based upon the excellent work of Kyle Hayward (Hayward 2009).

Only ray vectors which pass through the volume region are required, as any ray outside of this region is unable to intersect with a voxel. The volume data will be contained within a cuboidal region. Therefore the scene of Figure A3.40 is presented.

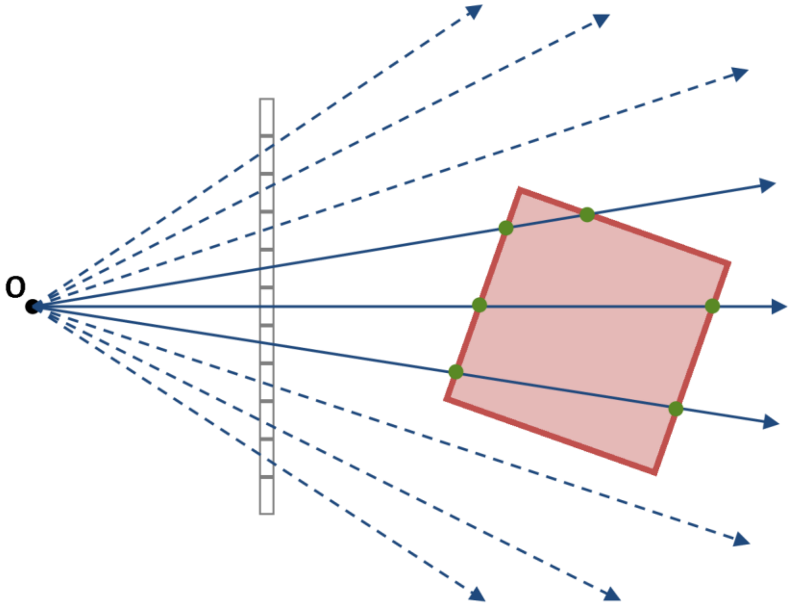


Figure A3.40: Ray vectors cast into the volume region

Figure A3.40 indicates that only the three central rays may intercept a voxel within the volume region. Therefore, it is only necessary to calculate the vectors for these rays. Figure A3.40 highlights the point at which each of these central ways in incident upon and exits the volume region. Simple geometry may thus be used to calculate the vector of each ray in the coordinate system of the volume region. Initially being within the volume coordinate space seems non-ideal, however, this in fact allows the volumetric data to be index directly. Therefore, this vector may be transverse along to inspect the volumetric data to detect an opaque voxel.

These two points, referred to as entry and exit point respectively, are calculated by the using the two pass rendering technique. The first shader pass is used to calculate the exit point while the entry point is calculated by the second pass before rendering.

As mentioned the volume data is rendered within a cuboidal volume. To determine the entry and exit points an automatic feature of OpenGL is exploited – interpretation. Each vertex may be applied a colour, if a rendered point is between two vertices of different colours OpenGL interpolates the two colours to produce their weighted mean. For a point on a face defined by three or more vertices the colour value of that point is defined by interpolating all vertices involved. The ray-tracer technique inverses interpolation and determines the position of a point relative to vertices of known colours. A cuboid consists of eight vertices, these were coloured such that their colours represented the coordinates of the vertices in coordinate space. Figure A3.41 below illustrates these eight vertices, with their coordinates and colours shown.

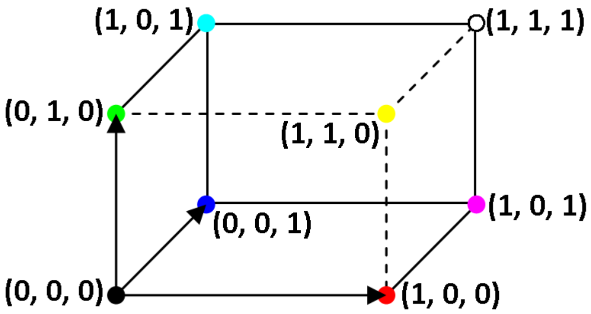


Figure A3.41: Colour encoded coordinate system of volume cuboid

Figure A3.42 below shows the same colour scheme rendering in OpenGL using interpolation. The top figure shows the cuboid rendered as lines only to show the point to point interpolation. The bottom figure shows the full interpolation across the three frontal faces of the cuboid.

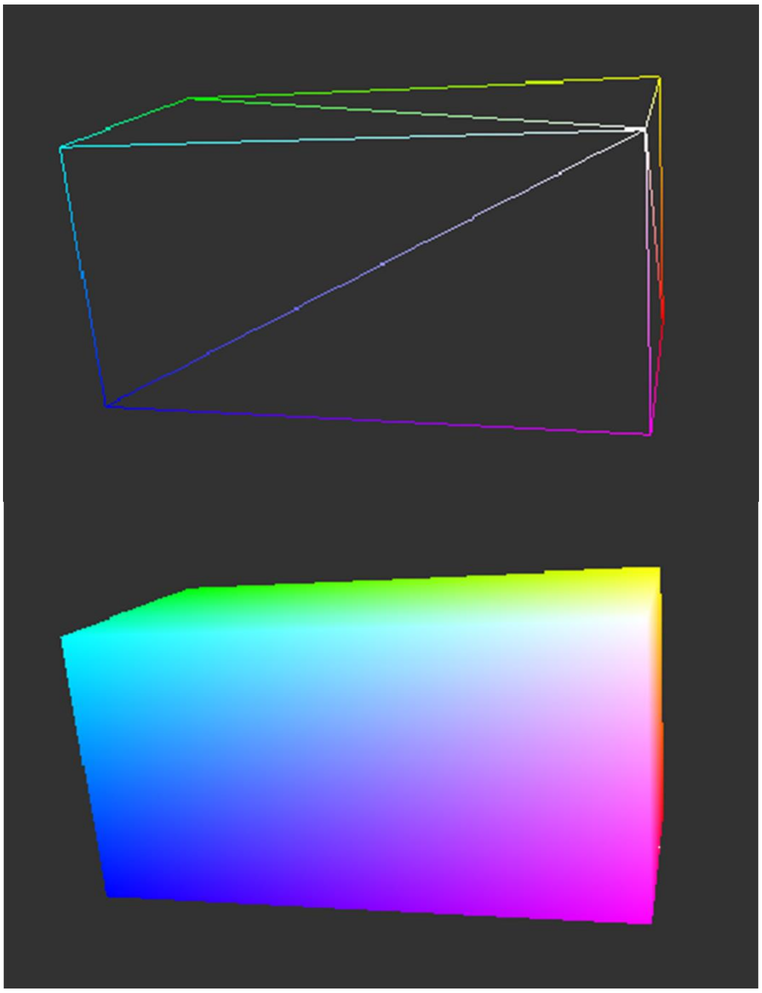


Figure A3.42: Colour encoded coordinate system showing linear and planar interpolation

When the volume cuboid is rendered as shown by Figure A3.42 above the coordinate of each pixel within the coordinate system of the volume may be readily determined from its colour. This provides the entry point, however, to determine the ray vector the exit point is also required. This is determined by the first render pass. Under normal rendering conditions OpenGL is set to cull, or not render, back faces. Each triplet of vertices is combined to produce a face. However, two faces are in fact produced. One, when viewing the triplet from the front and a second, when viewing them from behind. OpenGL is able to determine which face is the front and which is the back based upon the winding order of the vertices. The back face it typically culled to prevent OpenGL wasting resources rendering the inside of models that will not be visible. However, it is possible to flip face culling and instead cull the front faces. This causes the effect of making the front of an object transparent so that the inside of the back sides are visible. The effect of front face culling upon the same cuboid showed in Figure A3.42 in depicted in Figure A3.43 below.

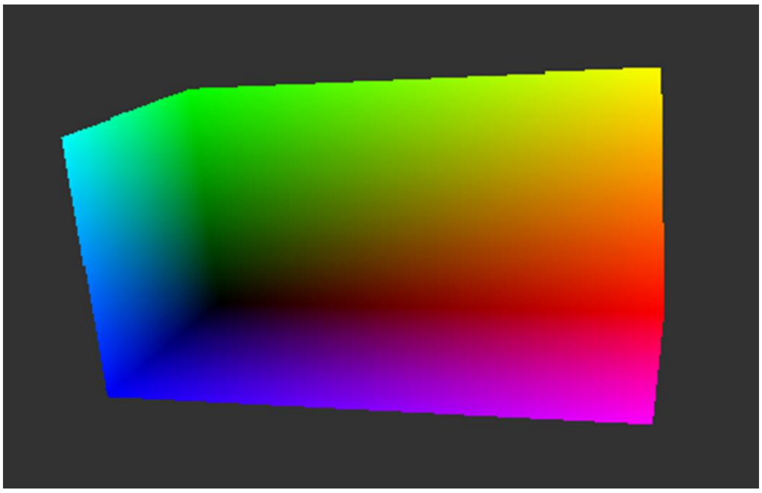


Figure A3.43: Render of colour coded cuboid with front face culling enabled

The system is thus able to determine the exit point of each pixel based upon the colour shown in Figure A3.43 above. The second render pass then renders the cuboid using standard rear face culling, as shown by Figure A3.42. Therefore, the entry and exit point may be determined for each fragment within the volume, allowing each ray vector to be calculated.

After calculating the ray vector the second render pass is able to traverse along it to render the volumetric data. The result of this is shown by Figure A3.44.

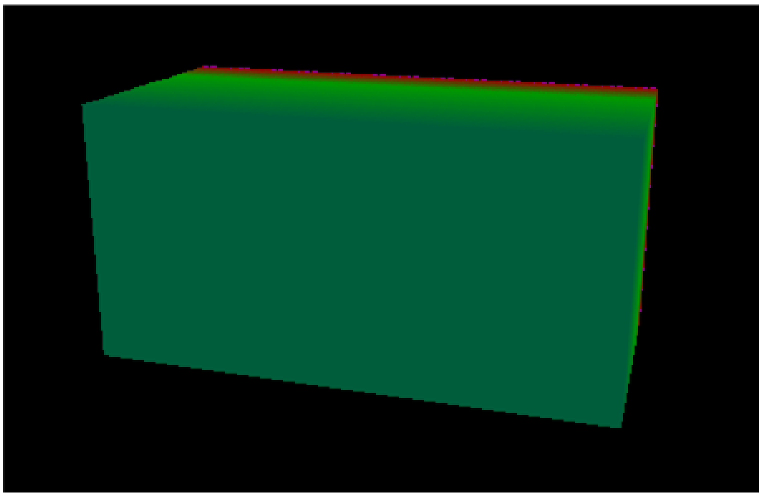


Figure A3.44: Directly rendered volumetric data

The front most face of Figure A3.44 appears flat. However, the volumetric data features a number of hemisphere indents. These aren't visible in the above figure as the surface within the indents is the same colour as the frontal surface. Therefore, both pixels within and out with the hemispheres are rendered the same colour. To show the relief of the surface lighting is required. Phong lighting varies the brightness of a surface based upon its orientation relative to the viewer and light source. Therefore this may be used to effectively show the relief of the surface.

Phong lighting requires the surface normal, a value not currently know. Typically normal values are loaded with the vertex data of the model and interpolated across faces. This is not possible with volumetric data. Furthermore, later implementations are to add the ability to modify the data. Therefore, it is necessary to calculate the normal at each surface point intercepted by a ray vector in real-time.

The normal is calculated within the second pass fragment shader. After determining which voxel is to be render the six surrounding orthogonal voxels are inspected. This information is used to generate X, Y, and Z-gradients which are combined to produce an approximate normal gradient. The calculated normals are shown for the volumetric dataset in Figure A3.45 below, where the RGB value is obtained from the three components of the normal vector.

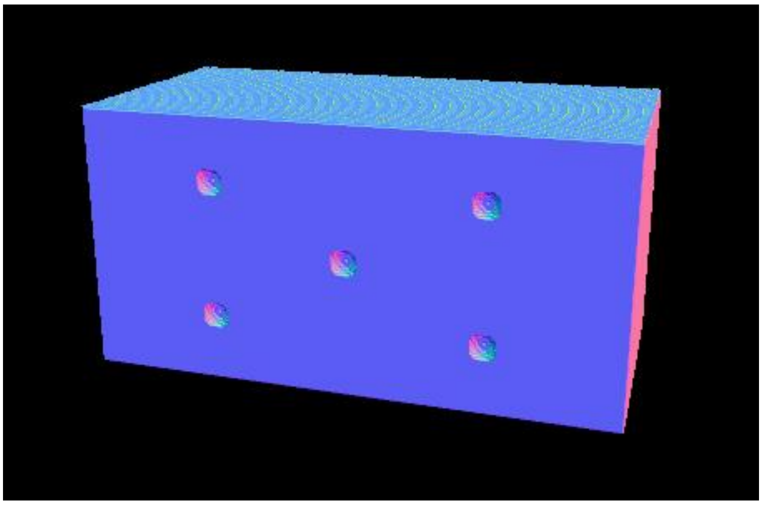


Figure A3.45: Estimated surface normals of volumetric data

The five hemispheres are visible within the normal map shown above in Figure A3.45. It is seen that the normals of the hemispheres match those of the total volume.

These normal values are then used in lighting calculations producing the final render shown by Figure A3.46 below

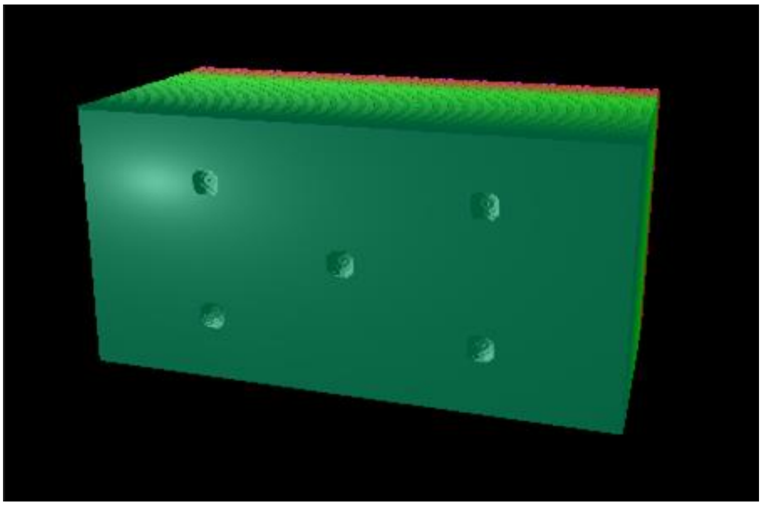


Figure A3.46: Directly rendered volumetric data with Phong lighting

While this approach only roughly approximates surface normal, it may be seen from Figure A3.46 that it is sufficient to make surface indentations of the same colour visible, while producing a relatively convincing effect of depth. The presented solution meets the first two requirements listed in Table A3.3; being able to render a full 3D model with overlaying regions and a normal range of 360°. The following discussion describes the volume editing approach required to meet the remaining four requirements.

Surface resection simulation is the key feature of the model volume as this provides the core functionality of the system. Therefore, this was designed as the central volume editing methodology from which other methods, such as surface generation, were derived.

The physical tool of the system, as discussed in Section 7.2.1.3, was a spherical cutting burr. Therefore, the volume system was required to remove spherical region from the volumetric data. To facilitate this, a second volume would be used to emulate the cutting burr. Using the tool tracking method, discussed in Section A3.3.1, this burr volume would be positioned within the main volume. The overlapping regions of the main volume would then be set to zero, simulating a resection. This principle is illustrated diagrammatically by Figure A3.47 and demonstrated in Figure A3.48 below.

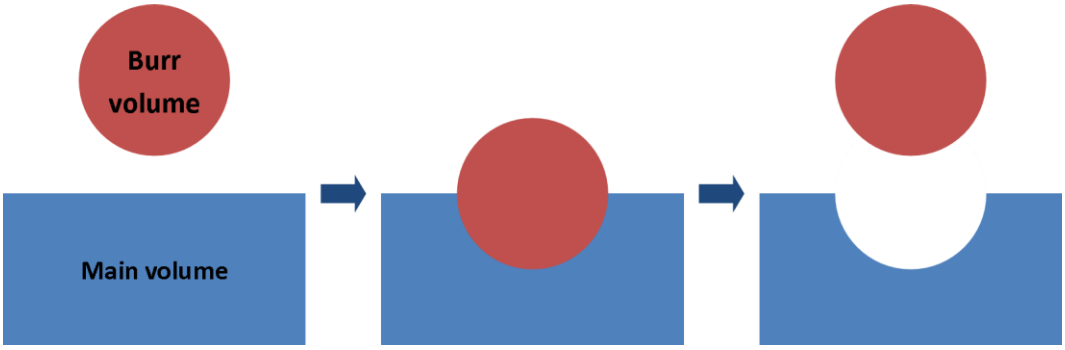


Figure A3.47: Volume resection simulation principle

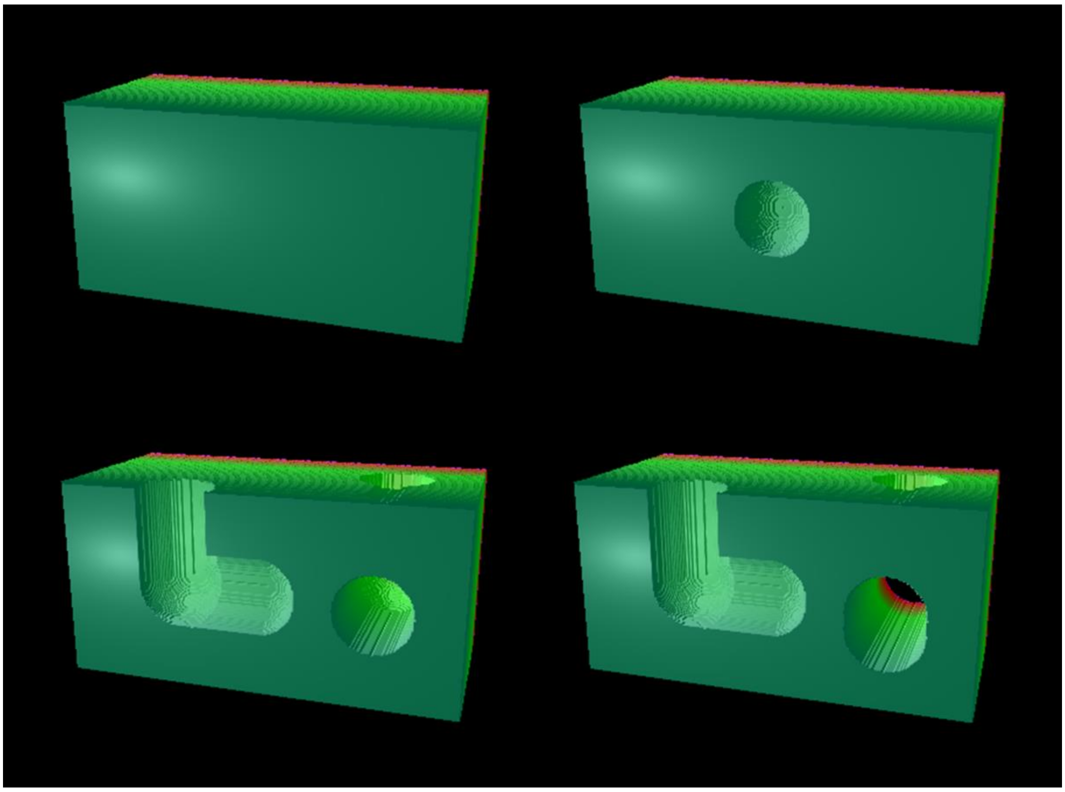


Figure A3.48: Demonstration of volume resection simulation

The resection simulation discussed above satisfies all but the final requirement listed by Table A3.3, the ability to control resection. To append this function an additional region was applied to the burr volume. Currently, the burr volume contains two values, one indicated to remove the overlaying main volume voxel, and another maintaining the voxel. A third value is applied which indicates for the system to check the overlaying voxel value of the main volume but to maintain it. The values of each overlaying voxel within the regain are sampled, and the lowest value recorded.

The voxel values are set such that the value indicates how much more volume may be removed before the resection limit is reach. Therefore, the lowest value captured corresponds to how far the resection burr is from a no-cut region. Using this value the resection tool may be controlled, thus satisfying all requirements listed by Table A3.3.

The general implementation and incorporation of the volume system with the second generation system shall be discussed in the section below.

A2.5.3 Implementation

The volume system was relatively complex to implement, requiring multiple shaders and supporting classes and structures. Initially the class structure of the volume system is shown below by Figure A3.49.

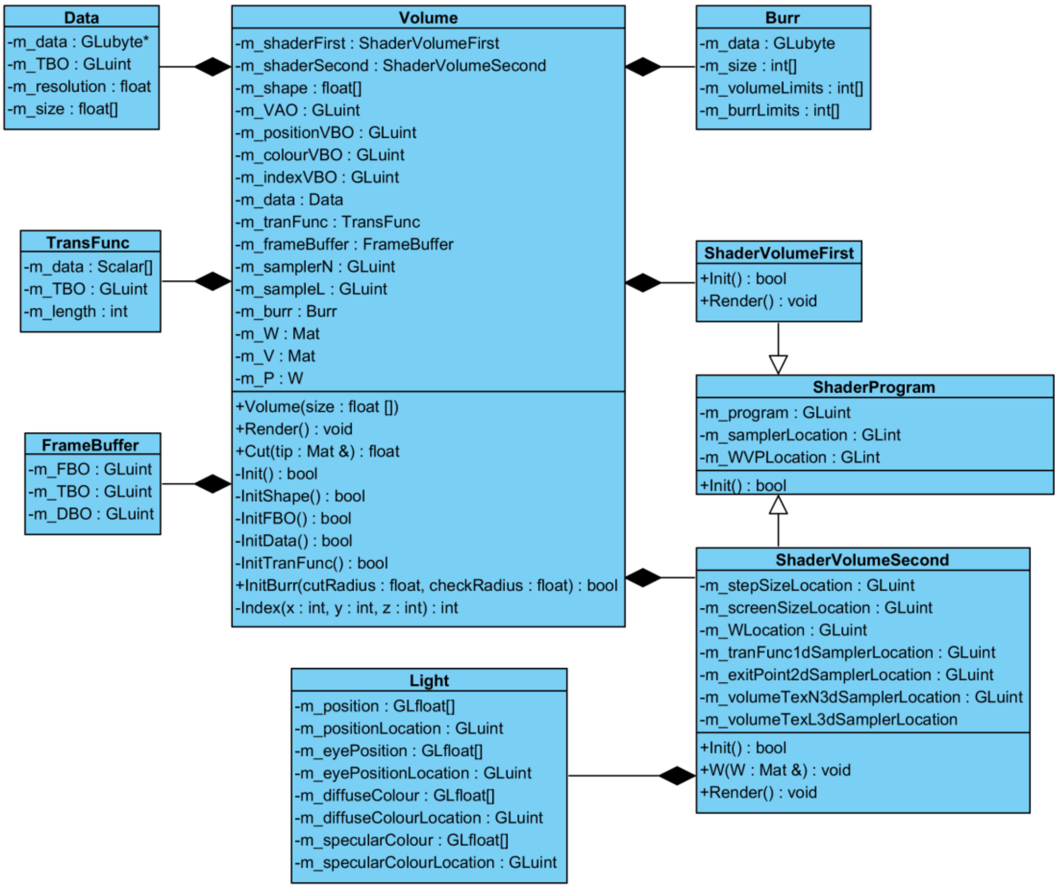


Figure A3.49: Volume system class and associated element structure

The elements depicted by Figure A3.49 are briefly summarised below to provide an overview of the system.

$Volume$ is the central class of the direct rendering system. It is the only class interacted with directly. It is placed within a $World$ class, as with any other model, and rendered. $Data$ is a structure used to group relivent information. It contains the raw volume data as well as size and resolution information required to decode the data into a useful structure. $TranFunc$ is a similar structure, containing information pertaining to the transfer function. The transfer function, as shall be disussed below, is used to convert the single byte value of the volume data into a colour RGBA value that may be rendered.

The $FrameBuffer$, as discussed above, may be used to store results of a render cycle, as opposed to displaying them to screen. The $FrameBuffer$ structure contains references to the framebuffer and associated data stores upon the GPU.

The two shaders, *ShaderVolumeFirst* and *ShaderVolumeSecond*, are both derived from the standard *ShaderProgram*. The second shader is seen to have a *Light* structure associated with it. This contains information required to render the final volume using Phong lighting.

Finally the *Burr* structure groups informations for a second volumetric data set. This infomartion was used to edit the main volume allowing it to be virtually resected in real-time.

Upon initialisation the system compiled both shader programs before generating the required data sets. Firstly the bounding cuboid position, colour, and index arrays were populated as described above. The main data was then procedurally generated as a large byte array. To allow a full range of colours, which would translate meaning of the model to the user, to be generated from byte values, thus reducing memory demands, a transfer function was also generated. This would be used by the final shader stage, and simply mapped each value between 0 and 255 to a four channel RGBA value. The mapping of the transfer function is documented by Table A3.5 below.

Intensity Value	Colour Value	Description
0	(0.0, 0.0, 0.0, 0.0)	Empty voxel
1	(1.0, 1.0, 1.0, 1.0)	Bone
2	(1.0, 0.0, 0.0, 1.0)	No-cut
3 – 4	(1.0, 0.0, 1.0, 1.0)	Reserved
5 – 34	(1.0, 0.0, 0.0, 1.0) (0.0, 1.0, 0.0, 1.0)	Overcut
34 – 255	(0.0, 1.0, 0.0, 1.0) (0.0, 0.0, 1.0, 1.0)	Undercut

Table A3.5: Transfer function colour scale

Using the transfer function approach a 70x30x30 mm volume at a resolution of 4 voxels/mm, such as shown by A3.50 below, required less than 4 MB of memory, and could be rendered at approximately 800 fps upon a Radeon R9 280X GPU.

With all pre-requisite data generate it was passed to the GPU to allow rendering. As described the first shader pass calculated the ray exit point, while the second pass calculated the entry point. This allowed a ray vector to be calculated and iterated along, thereby rendering the volume.

As described Phong lighting was to be used to emphasise curvature and depth. However, the initial implementation of the lighting within the volume rendering method resulted in a striation pattern across planar surfaces as shown by Figure A3.52 below.

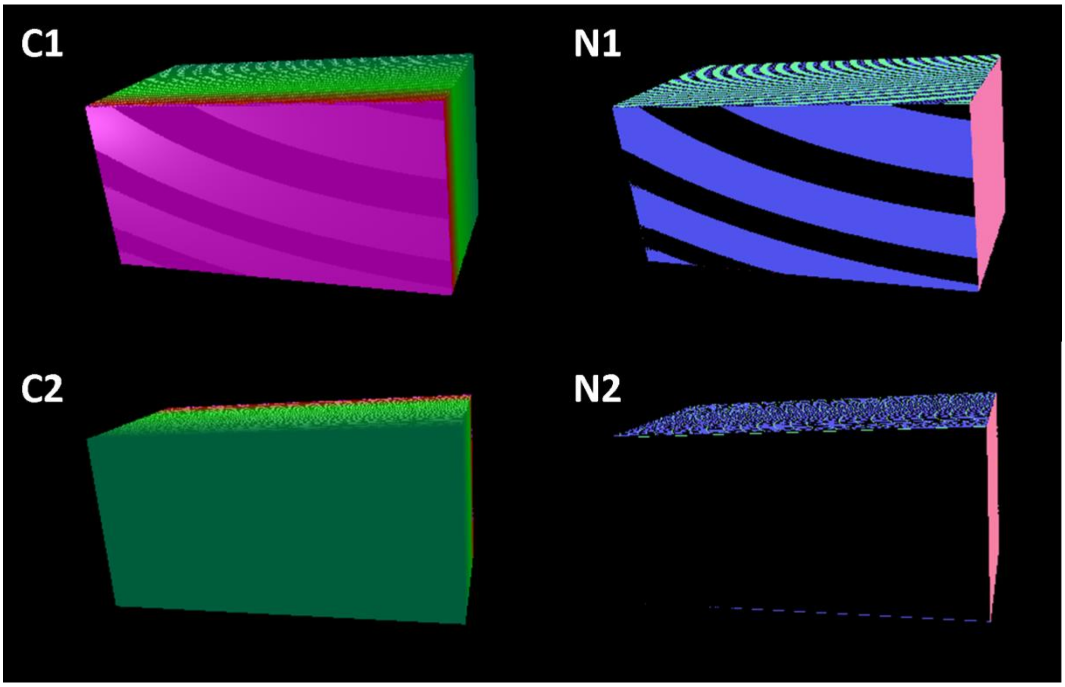


Figure A3.52: Error in the volume lighting. C1 and C2 show two different views with erroneous lighting, N1 and N2 show the respective normal calculated

It may be seen from Figure A3.52 that the lighting error was a result of erroneous normal calculation, resulting in either bands or entire faces of zero normal. This was found to be caused by the use of nearest interpretation while sampling the 3D volume texture. Using linear interpretation a ray that fell between a red no-cut region and green cut region would mix these two values, producing a yellow colour. To prevent this, the 3D texture was sampled using nearest interpretation, whereby the fragment was coloured by the value of the nearest voxel. However, nearest sampling was not compatible with the normal calculation method. Instead of sampling the six orthogonal voxels the nearest sampling often resulted in the central voxel being sampled each time, resulting in the frequent zero normal values seen in Figure A3.52.

To resolve this error, two separate samplers were generated. Firstly, a nearest sampler that was used to sample the voxel data to determine the fragment colour. Secondly, a linear sampler that was used to sample the data for the normal calculation. This allowed both the correct rendering of colours and lighting, as shown by A3.50 below.

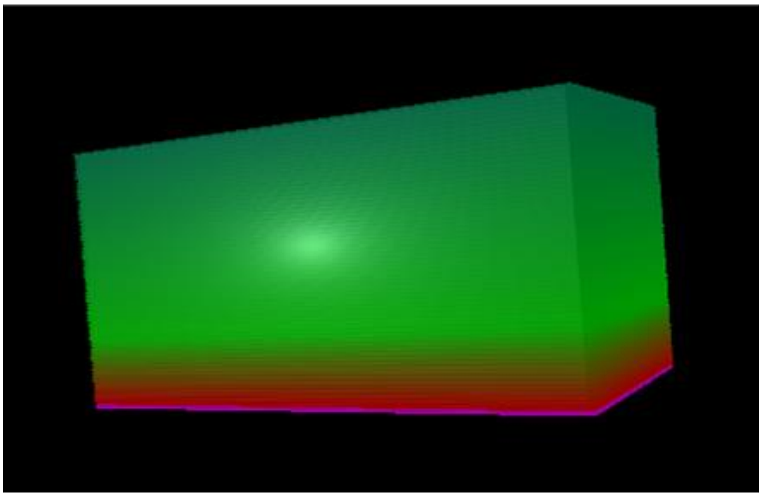


Figure A3.50: Procedurally generated volume

It was observed, however, that the quality and accuracy of the volume render could be improved by decreasing the step size by which the ray vectors were traversed. Large steps had the potential to miss several opaque voxels, potentially rendering the wrong colour in fast changing regions. They also caused notable flicker to the surface as the object was moved, again due to missing the true outer most opaque voxel. However, smaller steps put substantial strain upon the GPU, causing a notable reduction in framerate, particularly for larger volumes. Therefore a compromise was developed. Initially large steps were taken, equal to 1% of the length of the total ray vector within the volume region. Upon hitting an opaque vector a backwards step was taken, returning to the position previous to the collision. Ten smaller steps, of 0.1% total length were then taken until a hit occurred. This allowed the system to quickly locate the approximate area of the surface edge before determining it to a greater accuracy. This solution provided both optimal visual results and performance.

The volumetric rendering process is summarised in Figure A3.53 below using an activity diagram.

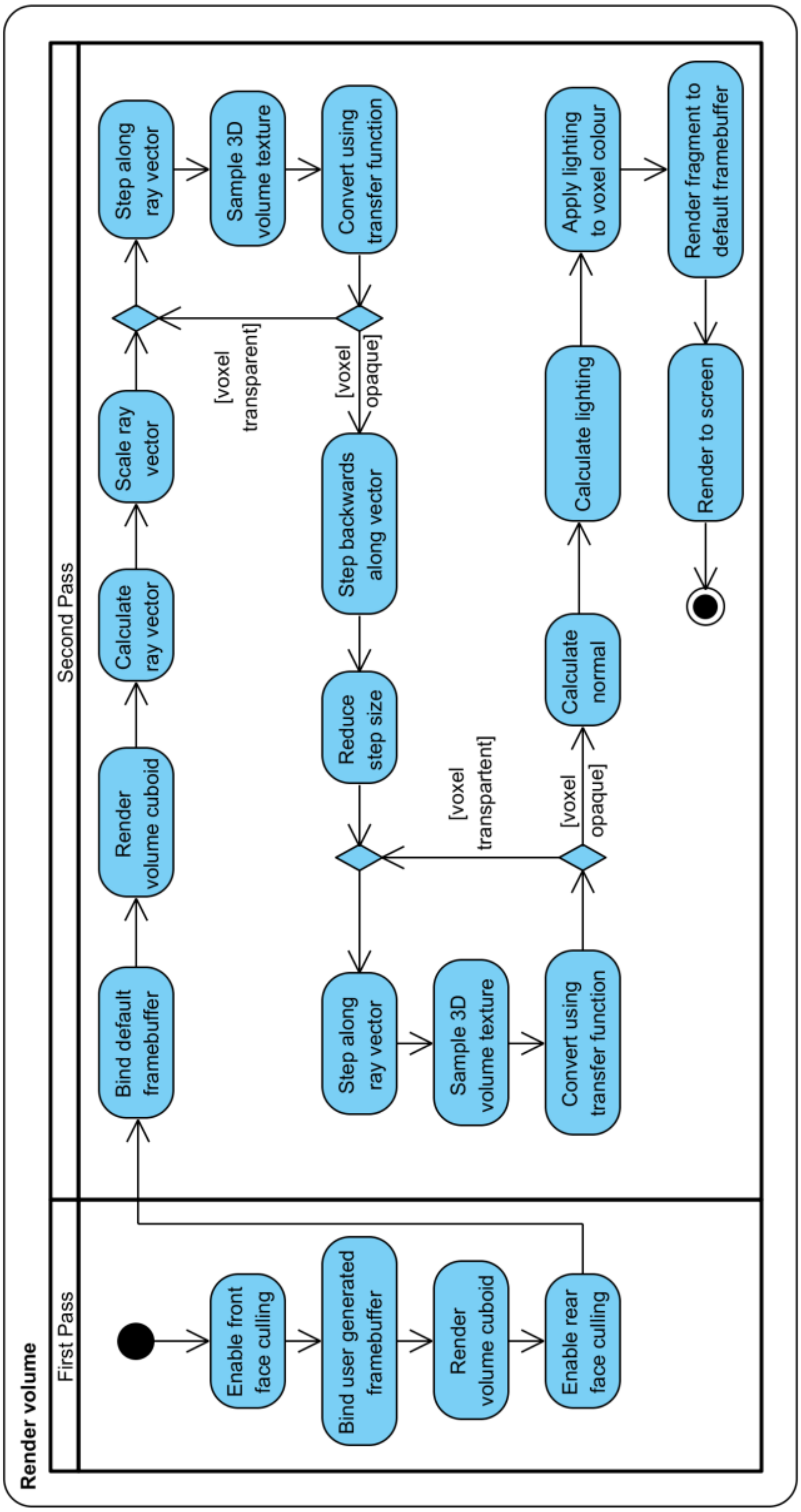


Figure A3.53: Summary activity diagram of direct volumetric rendering implementation

The final process to implement was the resection simulation method described by Section A3.5.2 above. The method received the tip position of the tool, relative to the same marker as the volume was rendered, and placed the burr volume at this point and compared overlaying voxels between the burr and main volumes.

The method first transformed the tip position from the coordinate system of the marker to that of the volume itself. This was a two stage process. The tip was first multiplied by the inverse of the world matrix of the volume. This moved the tip into the volume space. The origin of the volume, set during its initialisation, was then subtracted from the tip position. This brought the tip into the volume data space, such that the tip coordinate could be used to correctly index the volume data.

Three initial tests were performed to check that tip was within the volume. The original implementation simply checked the tip position against the dimensions of the volume. As discussed in Section 8.3.4, this resulted in resection errors. The tip was used to define the centre of the cutting burr. Therefore, the burr could be within the volume while the tip was out with the volume. As a result of this and the testing method above the resection control failed at the volume edges, leading to erroneous resection. The second implementation accommodated for the dimensions of the resection burr ensuring resection was performed if any portion of the two volumes were in contact.

If the tip position passed the contact test in the three dimensions it was known that at least some part of the burr volume was in contact with the main volume. However, much of burr volume was empty space, so true contact was not ensured.

To allow the tip position to correctly index the volume data it was transformed fully into the data space of the volume. The position, already relative to the volume data origin, was multiplied by the data resolution, before being converted into integer format. Therefore, a 1 mm displacement of the tip would translate into a displacement of four voxels.

The next stage of the method calculated burr and volume search regions. It was currently know that the burr volume and main volume overlapped to some extent.

The following stage determined which voxels within the burr volume coincided with the main volume. The method applied is illustrated by Figure A3.54 below.

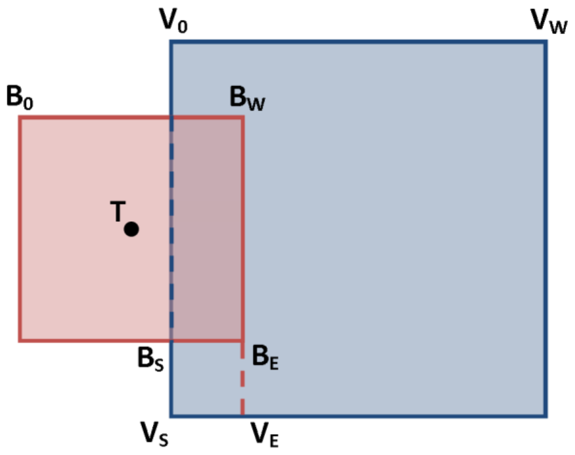


Figure A3.54: Volume resection simulation overlap region

For each dimension B_0 and B_W define the dimensions of the burr. Similarly, V_0 and V_W define the main volume. T indicates the tip position in indexable data coordinates of the volume. The region of overlay within the burr volume is bound by the start and end points, B_S and B_E respectively. Again, this region is bound within the volume data by V_S and V_E. Figure A3.55 below describes the calculations of these four boundary values.

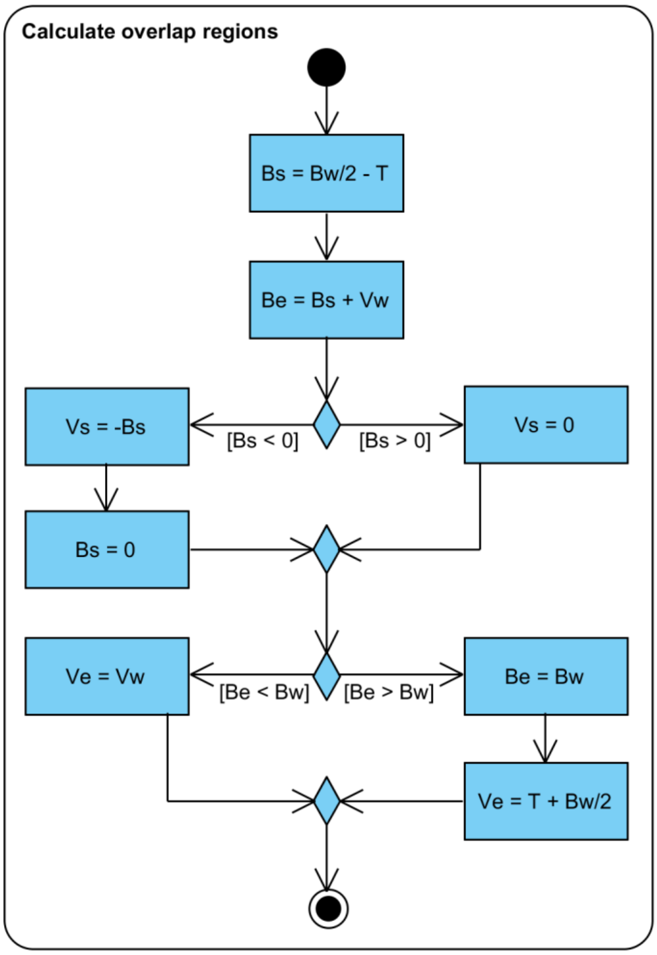


Figure A3.55: Calculation of overlap region bounding points

The burr start point is first defined as the centre of the burr minus the tip position, while the end point is defined as the start point plus the volume dimension. If the burr start point is greater than zero this indicates the burr is not completely within the volume, as the volume edge lies within the burr region. Therefore, the volume start point must be zero. However, if the burr start point is less than zero the burr must be fully within the volume, as the volume edge is to the left of the left edge of the burr region. The volume start point may thus be set as the inverse of the calculated burr start point, as if the volume edge is five voxels to the left of the burr, the burr edge must be five voxels to the right of the volume. As the burr is completely within the volume the burr start point must be zero.

Next, if the burr end point is less than the burr width the right edge of the volume must be within the burr region. Therefore, the volume region must extended to the volume edge, thus the volume end point is set to the volume width. If instead the burr end point exceeds the burr width, the right burr edge must be within the volume. Therefore, the burr end point must be equal to the burr width, and the volume end point equal to the tip position plus half the width of the burr volume.

This process is repeated for each of the three orthogonal dimensions, defining the overlap bounders for both the burr and main volume data. Empirically these two regions are of equal dimensions. A check was performed that the volume of this region was greater than zero before memory of equal size was reserved. The system then iterated through the two defined regions, comparing voxels to populate the newly initialised volume, as summarised by the activity diagram of Figure A3.56 below.

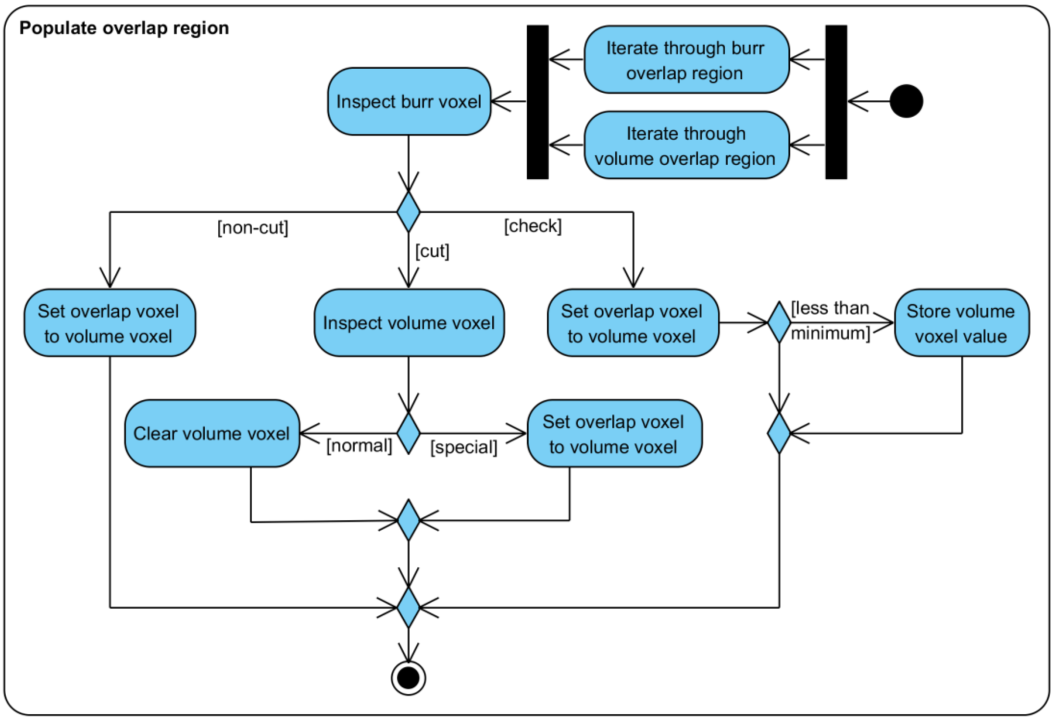

Populate overlap region
Iterate through burr overlap region
Iterate through volume overlap region
Inspect burr voxel
[non-cut]
[cut]
[check]
Set overlap voxel to volume voxel
Inspect volume voxel
Set overlap voxel to volume voxel
[less than minimum]
Store volume voxel value
Clear volume voxel
[normal]
[special]
Set overlap voxel to volume voxel

Figure A3.56: Activity diagram of overlap region population and resection simulation

Figure A3.56 show that both burr and volume overlap regions were iterated through simultaneously, using the bounding points previously calculated. Each burr voxel was inspected. It may have one of three potential values: non-cut, cut, and check. Non-cut indicates empty space within the burr volume around the actual burr sphere. For these voxels the value of the volume was loaded onto the new overlap volume. For cut voxels the overlapping volume voxel was inspected. Special reserved values could not be removed. This would allow permanent volume features to be generated. Standard voxels were resected by setting the volume voxel to zero. It was not necessary to set the overlap volume voxel as it was initialised to zero. As described in Section A3.5.2 the check value trigger the system to read the overlapping volume voxel but leave it unedited. The volume voxel was first copied to the overlap volume to preserve it. The lowest voxel value over the entire overlap region was stored.

Once the overlap region was populated it was loaded onto the GPU. Using the boundery points it was possible to replace the region within the volume data stored upon the GPU using the overlap region. This considerably reduced the data transfer requirements, helping allow real-time resection.

The resection method converted the minimum voxel value into a distance using Equation A3.10 below. This equation was based upon a voxel value of 35 representing 0 mm resection required, with each voxel value unit equivalent to 0.05 mm resection.

$$d(mm) = \min - 35 * 0.05 \qquad \text{(A3.10)}$$

This value was then used to activate or deactivate the resection tool, providing semi-active constraint.

Timing analysis of the resection method provided a mean processing time of 1.39±0.05 ms. Therefore, the resection time was significantly faster than all stages of marker detection ($P < 0.01$), excluding the corner matching sub-stage of corner detection, which was insignificantly faster ($P = 0.09$). As such, further optimisation of the method was deemed unnecessary.

Appendix 3
Generation Three Implementation

The implementation of the third generation system focuses only on the differences to the second generation system, namely; research camera interface, marker pose estimation, HMD interface and rendering, and hip centre location.

As discussed in Section 8.2.2, to allow interfacing with the HMD device the system was ported from the GLUT library to SDL2. To facilitate this, the system was restructured as shown by Figure A4.1 below.

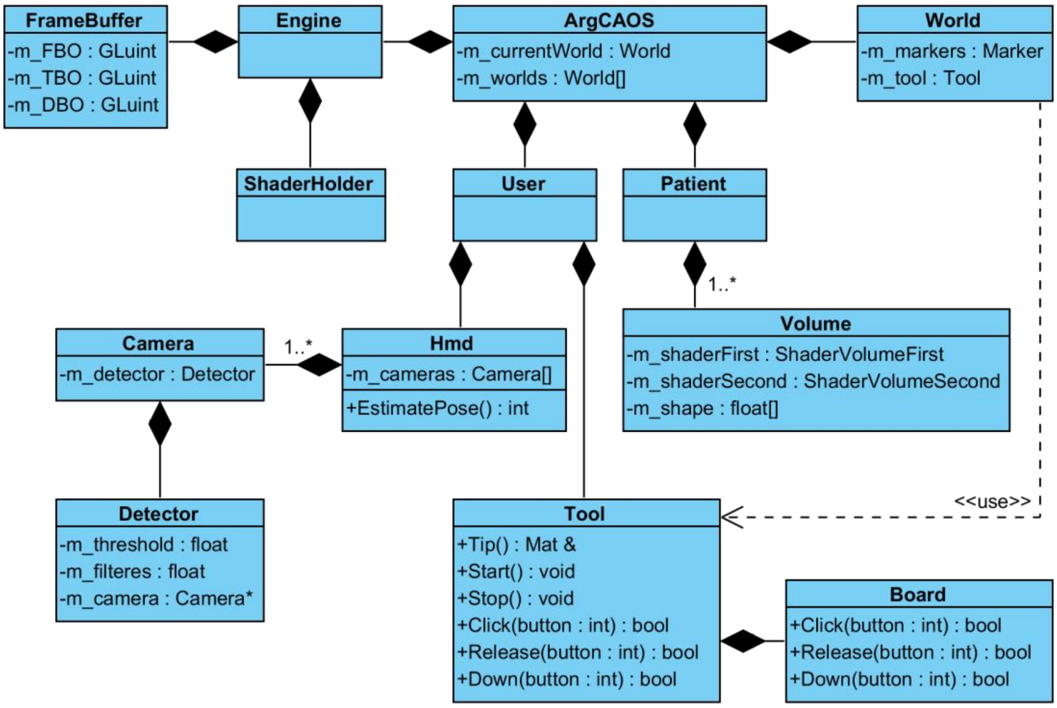


Figure A4.1: Third generation system class structure

As may be seen from Figure A4.1, the system was substantially restructured to facilitate the HMD and stereoscopic imaging. *ARgCAOS* represents the core system which manages and contains all other classes. The *Engine* class is responsible for managing the graphical elements of the system. This includes initialisation, storage, and binding of the shader programs used to render objects, in addition to managing the main system frame buffer. As shall be discussed in Section A4.2 below, all rendering was performed onto this framebuffer, the oculus SDK then performed a number of transformations to this image before displaying the results upon the HMD.

The $Patient$ class maintained all information pertaining to the patient. This most notably includes the resection $Volumes$ of the femoral and tibial segments, in addition to joint centres.

The $World$ classes, as before, allowed segmentation of the various stages of the system. However, the $World$ no longer contained the $Tool$ class, but instead merely had accesses to it via reference. The $Tool$ class was now contained within the $User$ class, as this allowed the tool to be preserved between $World$ instances without the use of static members. The $User$ class also contained the Hmd class. This was responsible for managing the cameras, triggering them to capture and process frames simultaneously before passing the resulting markers and images onto the $World$ and $Engine$ classes by reference, respectively. This process is described in more detail below.

A3.1 Marker Detection and Pose Estimation

As previously described, the first stage of marker pose estimation was image capture and processing. With the stereoscopic camera system presented it was logical to perform this process with both cameras simultaneously via multithreading, as shown by Figure 8.7.

Before image capture both cameras must be initialised. This was done as descried by Section A3.1.1 for the LifeCam system, with the addition of loading the stereoscopic parameter matrices. The research system omitted loading focus and exposure settings as this was managed either manually, via the lens, or using the separate Point Grey FlyCapture2 software (Version 2.8.3.1).

To ensure synchronous image capture, both research cameras were simultaneously software triggered. The LifeCams did not allow triggers and as such this step was omitted. Each camera stream was processed upon a separate thread, allowing simultaneous processing. The image processing method follows that of the second generation system, extracting the image from the camera before passing it through the various stages of image processing to detect and identify markers. The research cameras returned the image frame in a custom format which required conversion to the internal RGB matric format. However, this did not incur a notable increase in processing time.

Figure A4.3 below illustrates the functionality of marker generation for the third generation system. It is seen that the marker must be identified, and the corners located, in both images before the pose of the marker pose was estimated.

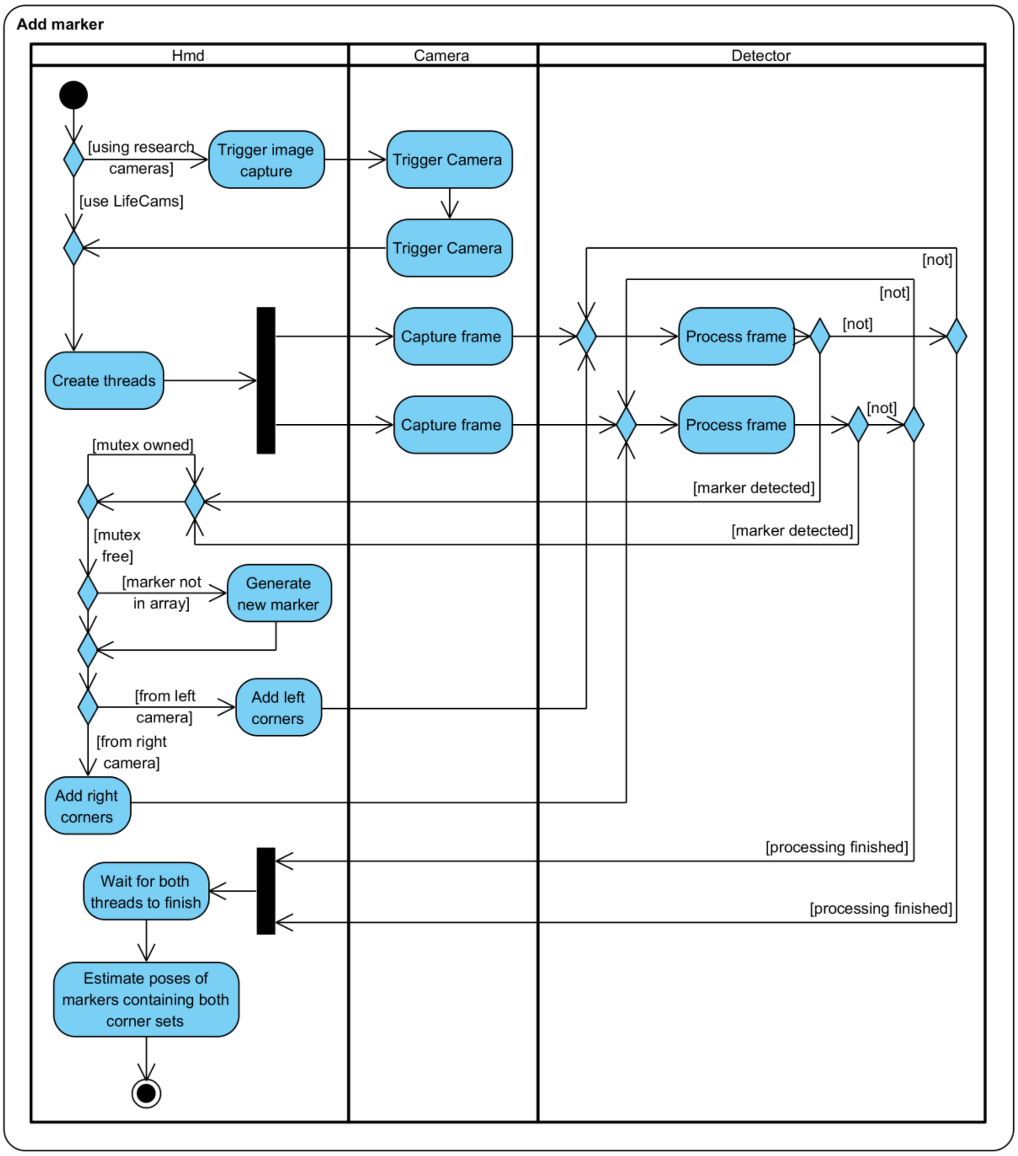


Figure A4.3: Activity diagram for third generation marker detection and generation process

Once the marker prototypes had been generated their poses were estimated using the SDV least squares method discussed in Section 8.2.2. If a marker contained only one set of corners it could be processed using monoscopic pose estimation as described by Section A3.2. However, this function was disabled for system analysis and only markers containing both corner sets were processed using the stereoscopic method detailed below.

To enable the accurate calculation of corner disparity and conversion to world space coordinates the extracted subpixel corner coordinates had to be undistorted. This firstly minimised the image distortion resulting from lens and camera aberrations, before rectifying the coordinates based upon the relative alignment of the two cameras determined during calibration. This was performed using the $undistortPoints$ method which was called for each corner set, being passed the intrinsic matrix, distortion parameters, rectification matrix, and rectified projection matrix of the relevant camera.

Once undistorted the disparity coordinates of each corner was calculated. This was achieved using the X and Y components of the left camera and the X component of the left camera minus the X component of the right to obtain disparity. The resulting disparity coordinates of each corner were multiplied by the reprojection matrix, before being divided by their w component to obtain an array of four three dimentional coordinates in world space relative to the left camera.

SVD, as described by Equations 8.2 to 8.12, was then performed to determine the pose of the marker based upon the four corner positions. This required comparison to a marker model. As discussed is Section 7.3.3.3, custom marker measurements were shown to be advantages, and as such were to be better implemented within the third generation system. As such an XML based library was implemented. The ID of each marker was read, this was then compared to the XML library, if the ID was present the accurate corner positions, as measured by microscope, were loaded and used by the remaining processes. If the ID was not present a standard marker model was instead loaded.

The centroids of the image-derived and model marker corners were calculated. These were then used to calculate the correlation matrix (H) as defined by Equation 8.8.

Once calculated, SVD was performed upon the correlation matrix using the $SVDecomp$ function. The rotation and translations matrices were then calculated as defined by Equations 8.11 and 8.12, respectively. As with the second generation system, described by Section A3.2, the pose matrix was composed from these two matrices with appropriate values inverted to perform rotation about the X-axis to translate the OpenCV coordinate system into that of OpenGL.

A3.2 HMD Rendering

Graphical rendering under the third generation system followed the basic principles of the second generation system presented in Section A3.4. The video stream was first rendered to provide the background image of the physical world. Virtual models, loaded into the $World$ instance, were then were then rendered on to the scene using the calculated marker poses to allow alignment with physical objects.

Two main modifications were made to the existing system to allow rendering to the HMD. Firstly, instead of rendering to the default framebuffer an intermediary user generated buffer was used. This was then processed by the Oculus SDK before being displayed to the HMD. Secondly, each object was rendered twice, to allow 3D simulating stereoscopic display.

The $Engine$ class managed both HMD interface and theSDL2 context upon which the system was run. Upon initialisation the $Engine$ instance initialised the Oculus OVR and OVR_CAPI_GL libraries. The physical HMD was then detected and bound to the system. Once bound SLD2 was used to generate a window based upon the resolution supported by the HMD. The GL context was then bound to this window.

A $FrameBuffer$ instance was then created, including both a colour and depth buffer, again based on the resolution supported by the HMD. The OVR library was then used to bind the $FrameBuffer$ to HMD, setting the dimension and depth of the framebuffer. Several additional parameters of the HMD were then set. These include the types of distortion correction that would be applied to the framebuffer before rendering. Both chromatic and vignette correction were utilised.

The initialisation of the *Engine* class concluded by setting several OpenGL states, such as face culling and depth tests, as discussed with the second generation system. This completed the initialisation of the system for HMD rendering.

Figure A4.4 below documents the process required to perform stereoscopic rendering.

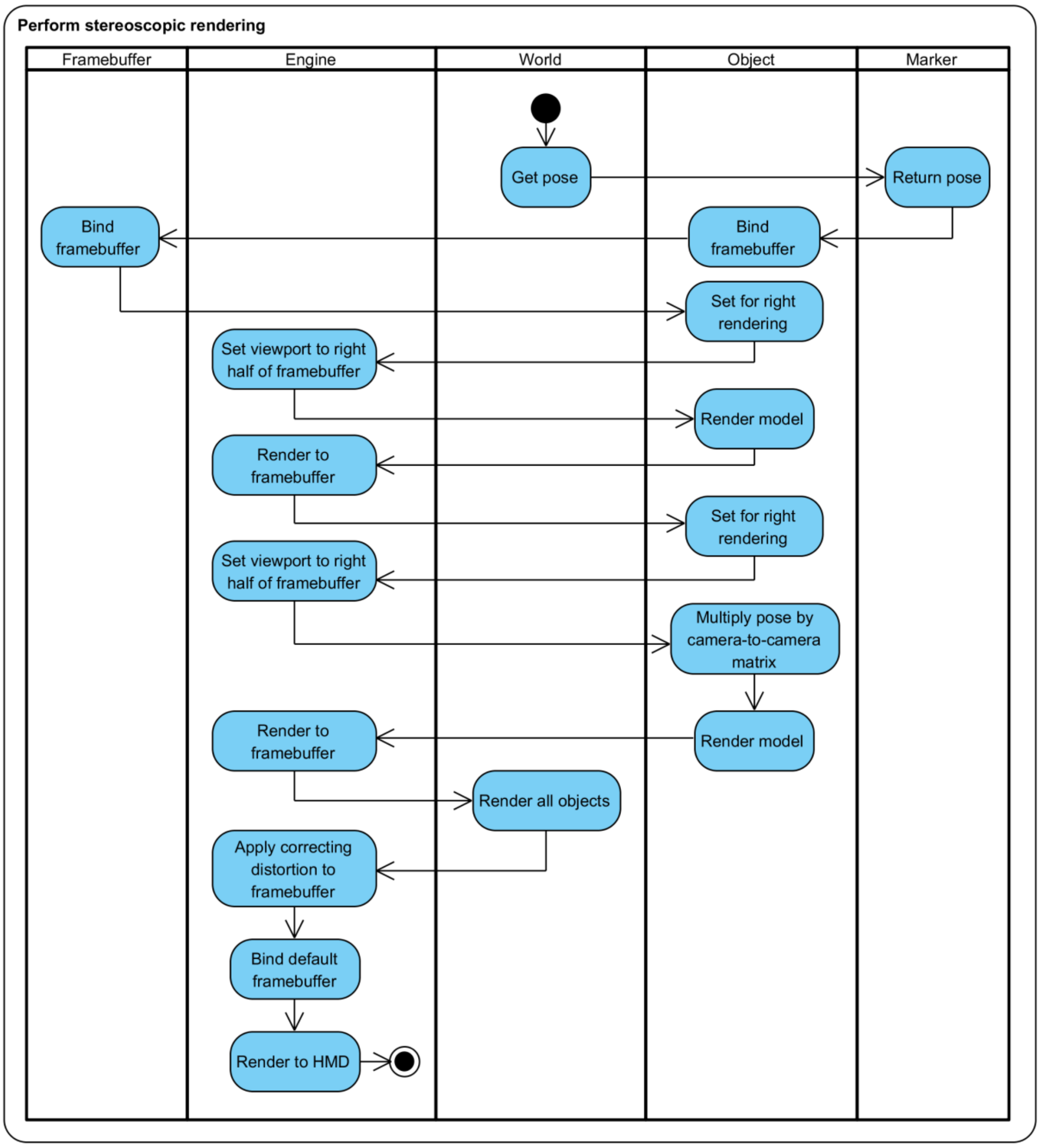


Figure A4.4: Activity diagram for third generation stereoscopic rendering

It may be seen from Figure A4.4 that prior to rendering the object to the right section of the framebuffer the pose matrix was multiplied by the camera-to-camera transformation matrix. This transformed the pose from the left camera coordinate system to that of the right. Therefore, when rendered the virtual object would still align with the physical world view through the right camera, giving the augmented scene a realistic and immersive appearance.

The stereoscopic approach discussed above was utilised by all virtual objects, including the volumetric resection model.