Multimedia Motion Analysis for Remote Health Monitoring

Cheng Yang

University of Strathclyde

A dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electronic and Electrical Engineering

April 2017

To my parents,

and to my elder brother

Declaration

This dissertation is the results of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Cheng Yang

April 2017

Acknowledgements

I would like to first thank Dr. Vladimir Stankovic for endless guidance and inspiration in terms of both technical knowledge and research methodology. This work would have been impossible without his great support and advice. I would also like to thank Dr. Lina Stankovic, Prof. Philip Rowe, Dr. Ukadike C. Ugbolue, Dr. Andrew Kerr and Dr. Bruce Carse, for their thoughtful advice and comments.

I would like to thank Dr. Gene Cheung, with whom I had the pleasure working during my internships at National Institute of Informatics, Tokyo, Japan (Jun.-Sept. 2013, Jul.-Oct. 2014 and Mar.-Jun. 2016).

The author was supported by University of Strathclyde Faculty of Engineering Scholarship (2011-2014), and University of Strathclyde Department of Electronic and Electrical Engineering Scholarship (2011-2014). This work was also supported by QoSTREAM - Video Quality Driven Multimedia Streaming in Mobile Wireless Networks, an FP7 Marie Curie International Research Staff Exchange Scheme project (2013-2014), and National Institute of Informatics International Internship Program Scholarship (2016).

Finally, I would like to thank my parents and my elder brother, for years of support and encouragement.

Abstract

Substantial amount of research in home-use health monitoring techniques has emerged given growing global health awareness and ageing population in recent decades. These sensor-driven home-use healthcare applications encourage patient involvement at home during daytime activities and nighttime sleep, effectively help assess patients conditions away from clinics and hospitals, and significantly reduce the number of infirmary visits. However, there are two main issues in current wearable/remote sensor-based home-use health monitoring applications: 1) portable human motion analysis systems that are commercially available still require substantial amount of manual effort to process the measurements, which is time consuming and thus impractical for long-term home-use health monitoring, and 2) current sleep-related health monitoring applications are intrusive to the body, limited to measuring the respiration rate and sleep duration, or not clinically validated to demonstrate their efficacy.

In this dissertation, we overcome the drawbacks of current health monitoring systems as follows. For lower limb motion analysis, we propose an alternative to state of the art optical motion analysis systems, cost-effective and portable, single-camera system. For upper limb motion analysis, we track all relevant body joints simultaneously, and classify the post-stroke recovery levels based on features extracted from the tracked body-joint trajectories. For abnormal respiratory event detection during sleep, we propose to record video and audio of a patient using a depth camera during his/her sleep, and extract relevant features to train a classifier for detection of the abnormal respiratory events scored manually by a scientific officer based on data collected by a clinical-use sleeping device.

The main contribution of this dissertation lies in proposing new application-driven algorithms for advancing cost-effective human limb motion analysis and sleep monitoring healthcare techniques, including an autonomous detection scheme for finding the initial and final frames that are of interest for video analysis, a single marker tracking scheme that is based on the Kalman filter and Structural Similarity image quality assessment, an autonomous gait event detection scheme that is based on the features of the relative positions of the markers, a scheme classification of the post-stroke recovery level by minimization of graph total variation with graph-based signal processing, an alternating-frame depth video coding scheme, a depth video temporal denoising scheme using a motion vector graph smoothness prior, and a dual-ellipse model that can efficiently track the torso motion during a person is sleeping. Experimental results show that, both the autonomous frame-of-interest detection and gait event detection show high detections rates. The validation of tracking in terms of the knee angle, shoulder movement, trunk tilt and elbow movement with a gold standard optical motion analysis system shows R-squared value larger than 0.95. The graph-based classification scheme has the potential to accurately classify participants into different stroke groups. Our depth video coding scheme outperforms a competitor that records only the 8 most significant bits. Our temporal denoising scheme reduces the flickering effect without ever-smoothing. Finally, our trained classifiers can deduce respiratory events with high accuracy. Overall, our proposed limb motion analysis system offers an alternative, inexpensive and convenient solution for clinical gait and upper limb motion analysis, and our proposed sleep monitoring system can reliably detect abnormal respiratory events using our extracted video and audio features.

Table of contents

Glossary

α shoulder-elbow-wrist angle. 54

β shoulder-elbow-wrist angle. 54

$\forall$ universal quantification. 29

γ shoulder-elbow-wrist angle. 54

$\mathbf{A}$ an adjacency matrix for a graph of a depth image block. 81

$\mathbf{B}$ process noise covariance matrix. 32

$\mathbf{E}$ measurement error covariance matrix. 32

$\mathbf{H}$ a temporal-activity matrix from non-negative matrix factorization. 90

$\mathbf{I}$ an indentity matrix. 71

$\mathbf{J}$ a wegithed adjacency matrix for a graph of a joint-angle signal. 59

$\mathbf{K}$ Kalman gain. 32

$\mathbf{L}$ a graph Laplacian for a graph of a depth image block. 81

$\mathbf{P}$ a degree matrix for a graph of a depth image block. 81

$\mathbf{Q}$ observation matrix. 32

$\mathbf{R}$ state transition matrix. 32

$\mathbf{W}$ a spectral-feature matrix from non-negative matrix factorization. 90

$\mathbf{X}$ a non-negative nonsingular matrix. 91

$\mathbf{X}_i^-$ posteriori covariance matrix. 32

$\mathbf{Y}$ a spectrogram matrix. 90

$\mathbf{Z}$ a depth frame. 78

$\hat{\mathbf{s}}_i$ a posteriori estimate for a Kalman filter. 31

$\hat{\mathbf{s}}_i^-$ a priori estimate for a Kalman filter. 32

$\hat{\mathbf{u}}_i$ an observation model for single marker tracking. 32

$\tilde{\mathbf{E}}$ a video feature vector. 90

$\tilde{\mathbf{U}}$ an audio feature vector. 92

$\mathbf{h}$ a vector with class labels. 60

$\mathbf{o}$ a set of depth observations. 87

$\mathbf{p}$ a smooth region boundary pixel location. 78

$\mathbf{v}$ a motion vector. 80

$\mathbf{x}$ a 1D ellipse signal segment. 89

$\mathbf{y}$ an audio signal segment. 89

$\mathcal{B}$ an image block within a bit-recovered depth image. 80

$\mathcal{N}$ a set of adjacent pixels. 79

$\mathcal{X}$ a graph signal. 59

$\odot$ elementwise product. 130

τ heuristically set threshold for peak detection. 29

θ an ellipse. 87

ϖ a combined motion vector. 83

ζ a graph signal. 59

AASM American Academy of Sleep Medicine. 19

AET average execution time. 49

ALS an alternating least-square update rule. 91

AL augmented Lagrangian. 95

BA Bland-Altman plot. 49

BF bilateral filtering. 95

BSNM a combination of the Bi-section and Nelder-Mead simplex method. 103

CI confidence interval. 49

CSP camera scene plane. 65

CV cross-validation. 97

DKF-SSIM discrete Kalman filter and Structural Similarity. 5

D a video sequence. 29

EEG Electroencephalography. 109

FDR frame difference rate. 30

FF Foot Flat. 37

FN false negative. 101

FP false positive. 42

F a video frame. 28

GT ground truth. 41

GUI graphical user interface. 16

G grayscale. 49

HJC hip joint centre. 48

HR Heel Raise. 37

H the histogram of a frame segment. 29

IC Initial Contact. 37

I intercept. 49

JCTH joint colour texture histogram. 15

LCI lower 95% confidence interval. 49

LF linear fit. 49

LOA limit of agreement. 65

LSB least significant bits. 78

MD mean difference. 49

ME motion estimation. 79

MO max error. 49

MSB most significant bits. 78

MST Mid-Stance. 37

MSW Mid-Swing. 37

MS Microsoft$^{\text{TM}}$. 21

MV motion vector. 79

NMF non-negative matrix factorization. 76

NM Nelder-Mead simplex method. 89

NN neural network. 76

OAA one-against-all classification. 58

OAO one-against-one classification. 58

OFA once-for-all classification. 58

OS ordinal scale. 68

PCA Principal component analysis. 76

PCM Pulse-code modulation. 75

PDM perfectly detected marker. 42

PMR perfect marker rate. 42

PSNR peak signal-to-noise ratio. 94

PWS piecewise smooth. 77

P a set of detected peaks. 29

RAM random-access memory. 41

RGS regularization on a graph signal. 18

RMSE root mean square error. 49

RPS reference picture selection. 78

RSV R-squared value. 49

RTG reach-to-grasp motion. 17

SA a search area. 31

STFT short-time Fourier transform. 90

STR “Struck” tracker. 15

SVD singular value decomposition. 104

SVM support vector machine. 17

S a frame segment. 28

TC Terminal Contact. 37

TLD Tracking-Learning-Detection. 15

TMF temporal median denoising. 95

TN true negative. 101

TP true positive. 42

ToF Time of Flight. 21

UCI upper 95% confidence interval. 49

VAR variance. 97

VICON VICON Motion Systems. 13

WMF weighted mode filtering. 95

WPT wavelet packet transform. 76

WSA a tracking scheme without the update of the search area. 61

acc classification accuracy. 70

B an image block within an original depth image. 79

G a graph signal. 59

M number of quantization bins of a histogram. 29

d total number of pixels in a quantization bin. 29

rbf Gaussian Radial Basis Function kernel. 68

List of figures

List of tables

Chapter 1

Introduction

The proper study of mankind is the science of design.

— Herbert Simon

In this chapter, we first overview the state-of-the-art home-use healthcare applications in terms of portability, cost, ease of use, near real-time processing, and visualization. We then highlight some of the issues in current research in home-use healthcare that we will try to address, list contributions of the thesis, and give an outline for the remainder of this thesis.

1.1 Sensor-driven home-use healthcare

Home-use sensor-driven healthcare techniques have emerged and grown rapidly with growing global health awareness and ageing population in recent decades [1]; the percentage of the elderly population to the overall population increased from 9.2% in 1990 to 11.7% in 2013 and is expected to reach 21.1% by 2050, and 40% of the elders live independently in their own homes [2]. The purpose of home-use sensor-driven healthcare techniques is to encourage patient involvement at home [3] during daytime activities and nighttime sleep and effectively help assessing patients and elders condition away from clinics and hospitals — frequent transportation from home to infirmary can be time-consuming, uncomfortable, and expensive. In general, clinical health monitoring systems require lots of devices (*e.g.*, multiple cameras to be fixed within a large laboratory for motion analysis [4]) and operational expertise to take measurement. The main functionality of these healthcare techniques includes motion analysis (*e.g.*, fall detection, gait[1] and posture analysis) and vital signs (*e.g.*, respiratory rate, body temperature, heart rate and blood pressure) monitoring. For example, portable limb motion analysis systems with inertial sensors [5] are able to track both lower and upper limb motions, and thus have potential for stroke patients[2] to assess their level of recovery at home. Similarly, vital signs tracking devices, such as smart watches and wrist bands, track activity patterns by measuring heart rate and body movement, which is desirable for quality self-assessment of a person's physical exercise and sleep.

Existing home-use healthcare applications can be divided into two groups based on sensor types. *Wearable*-sensor based applications [6] generally require one or more inertial sensors attached to the body to take measurement of the body motion and vital signs, *e.g.*, abdominal/chest strips are used for respiratory motion tracking in portable sleep monitoring systems; *Remote*-sensor based applications including those using smartphones usually require imaging and inertial sensors for body motion [7] and vital signs tracking, *i.e.*, they are less intrusive to the human body than wearable sensors. However, there are two main issues in current home-use health monitoring applications: 1) portable human motion analysis systems that are commercially available still require substantial amount of manual effort to process the measurements, which is time consuming and thus impractical for long-term home-use health monitoring, and 2) current commercially available sleep-related health monitoring applications are

[1] Gait refers to lower limb motion.

[2] Stroke patients refer to people who survived from a stroke, a sudden event when part of one's brain cells are deprived of oxygens due to block or burst of blood vessels.

intrusive to the body, limited to measuring the respiration rate and sleep duration, or not clinically validated to demonstrate their efficacy.

The motivation for this work is to build remote-sensor based, at-home health monitoring systems that are portable, cost-effective and easy-to-use, while reliably tracking vital signs during people's daily activities, with three targeted applications: 1) human gait analysis in *colour* videos captured by a single high-speed camera, 2) upper limb motion analysis in colour videos for post-stroke recovery assessment, and 3) abnormal respiratory event detection for sleep monitoring in *depth* videos and audio captured by a single Microsoft Kinect$^{\mathrm{TM}}$. This work seeks to address the gap between user-friendly at-home health monitoring systems and reliable, but expensive, clinical health monitoring platform. The main challenges lie in the interdisciplinary field of signal processing and machine learning, including *body joint* tracking with occlusion handling, image quality enhancement, *body part* modelling and tracking, and feature extraction for event detection. The technical challenges will be listed in detail in Section 1.2.

In this thesis, we try to overcome the main drawbacks of current home-use healthcare applications, with the goal to develop image and signal processing techniques for at-home healthcare applications using cost-effective imaging sensors. We do this by using a single imaging sensor for image sequence collection, and developing object detection, motion tracking, feature extraction and classification methods to our targeted applications on gait analysis, post-stroke recovery level classification and abnormal respiratory event detection.

1.2 Contributions of the thesis

First, the main technical challenges of this work are listed below:

- For Targeted Application 1 on gait analysis:

1) individual body joint tracking in colour videos;

2) partial/full occlusion of the body joints due to arm swing;

3) gait event detection based on body joint trajectories;

- For Targeted Application 2 on upper limb motion analysis for post-stroke recovery level classification:

4) simultaneous body joint tracking in colour videos;

5) classification of post stroke recovery levels based on body joint trajectories;

6) overall easy-to-use upper limb motion analysis system for home use with feedback from practitioners;

- For Targeted Application 3 on abnormal respiratory event detection for sleep monitoring:

7) depth image enhancement in the presence of high acquisition noise;

8) model formulation for body movement tracking in depth videos during sleep;

9) feature extraction in depth videos and audio for abnormal respiratory event detection.

The methodologies to address the above challenges for the three targeted applications are as follows. For gait analysis, we propose an alternative to state-of-the-art optical motion analysis systems, cost-effective and portable, single-camera system. The system consists of video acquisition, marker-based individual body-joint tracking, data analytics for calculating relevant kinematics parameters, visualization, and gait event detection. For upper limb motion analysis, we build on our gait analysis system, track all relevant body joints simultaneously, and classify the recovery levels based on features extracted from the tracked body-joint trajectories. For both gait and upper limb motion analysis systems, the experimental results are validated with a state-of-the-art optical motion analysis system; the only manual effort is the designation of marker templates for marker-based body-joint tracking. For abnormal respiratory event detection during sleep, we propose to record video and audio of a patient using a depth camera during his/her sleep, and extract relevant features to train a classifier for detection of the abnormal respiratory events scored manually by a scientific officer based on data collected by a clinical-use sleeping device. The proposed, clinically validated sleep monitoring system, is non-intrusive to human body and can operate in complete darkness.

The main contribution of this thesis is in addressing the above challenges and advancing remote-sensor-driven home-use healthcare applications in three aspects via: 1) autonomous gait analysis with gait event detection, 2) autonomous post-stroke recovery level assessment via upper limb motion analysis, and 3) non-intrusive sleep monitoring. Specifically, the following new application-driven algorithms are proposed to achieve this. For gait analysis, we propose an autonomous frames-of-interest detection scheme before designation of marker templates, followed by a discrete-Kalman-filter (a recursive solution to the discrete-data linear filtering problem) [8, 9]+Structural-Similarity (an image quality assessment algorithm, see Appendix B.1)[10]-based (DKF-SSIM) individual marker tracking scheme with non-linear interpolation based occlusion handling, and a video-frame-identification-based gait event detection scheme. For upper limb motion analysis, we build on our gait analysis system, where we track all relevant body joints simultaneously using our DKF-SSIM body-joint tracking scheme, and classify the stroke recovery level by minimization of graph total variation with graph-based signal processing. For abnormal respiratory event detection during sleep, we first propose an alternating-frame video coding scheme. Next, we perform temporal denoising on the decoded depth video using a motion vector graph smoothness prior to remove flickering effect in the video. Then we track patient's chest and abdominal movements based on a dual-ellipse model. Finally, we extract ellipse model features via a wavelet packet transform, extract audio features via non-negative matrix factorization, both of which are used for training and testing our abnormal-respiratory-event-classifier.

1.3 Organization of the thesis

The remainder of the thesis is organized as follows. In Chapter 2, we review state-of-the-art health monitoring systems and image and video processing methods that are related to the above three targeted applications.

In Chapter 3, we propose a single-camera gait analysis system for cost-effective home-use healthcare Targeted Application 1 (see Section 1.1). The proposed system includes 1) algorithm design for autonomous frame-of-interest detection, DKF-SSIM lower-limb-joint individual-marker tracking (*i.e.*, separately tracking the required hip, knee and ankle joints) and gait event detection, and 2) performance comparison of the proposed tracking scheme in colour and grayscale video sequences. Chapter 3 is largely based on the work that appeared in 2013 IEEE International Conference on Image Processing [11] and Journal of Sensors [12].

In Chapter 4, building on the single-camera gait analysis system in Chapter 3, we propose an upper limb motion analysis-based post-stroke recovery assessment system for cost-effective home-use healthcare Targeted Application 2, including 1) simultaneous DKF-SSIM upper-limb-joint tracking (*i.e.*, tracking the required waist, neck, shoulder, elbow and wrist joints simultaneously), and 2) classification of the stroke recovery level by minimization of graph total variation with graph-based signal processing, where the ground truth labels were marked by a biomechanics expert. This chapter is largely based on the work that appeared in 2014 IEEE International Conference on Image Processing [13] and IEEE Access [14].

In Chapter 5, we present our proposed abnormal sleep event detection system for cost-effective home-use healthcare Targeted Application 3, which consists of 1) algorithm design of video coding, video denoising and respiratory movement tracking, and 2) feature extraction from the tracked movement for classification of abnormal respiratory events scored manually by a scientific officer based on data collected by a clinical-use sleeping device. This chapter is largely based on the work that appeared in 2014 IEEE International Workshop on Hot Topics in 3D [15], 2014 IEEE International Workshop on Multimedia Signal Processing [16], and the work to appear in IEEE Transactions on Multimedia [17].

Note that, Chapters 3, 4 and 5 will bring focus on Targeted Applications 1, 2 and 3, respectively. Each of these chapters include the proposed methodology, the experimentation and a summary.

Finally, in Chapter 6, we conclude remarks of this thesis, and justify the challenges in home-use healthcare systems that still remains.

1.4 Ethics approval and publications

1.4.1 Ethics approval

The data collection procedure performed using single imaging sensor for lower limb motion analysis and post-stroke recovery level classification with upper limb motion analysis has passed ethical committee in National Health Service, UK and University of Strathclyde. See Appendix A.1 for details of the pilot protocol; the experimental procedure performed using captured depth video and audio for abnormal respiratory event detection during sleep has passed the ethical committee in National Institute of Informatics, Tokyo, Japan.

1.4.2 Publications

Journal articles

1. C. Yang, G. Cheung, and V. Stankovic, “Estimating heart rate and rhythm via 3D motion tracking in depth video,” *IEEE Transactions on Multimedia*, in press.

2. C. Yang, G. Cheung, V. Stankovic, K. Chan, and N. Ono, "Sleep apnea detection via depth video & audio feature learning," *IEEE Transactions on Multimedia*, vol. 19, no. 4, pp. 822-835, Apr. 2017.

3. M. Ye, C. Yang, V. Stankovic, L. Stankovic, and A. Kerr, "A depth camera motion analysis framework for tele-rehabilitation: Motion capture and person-centric kinematics analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 5, pp. 877-887, Aug. 2016.

4. C. Yang, A. Kerr, V. Stankovic, L. Stankovic, P. Rowe, and S. Cheng, "Human upper limb motion analysis for post-stroke impairment assessment using video analytics," *IEEE Access*, vol. 4, pp. 650-659, Jan. 2016.

5. C. Yang, U. Ugbolue, A. Kerr, V. Stankovic, L. Stankovic, B. Carse, K. Kaliarntas, and P. Rowe, "Autonomous gait event detection with portable single-camera gait kinematics analysis system," *Journal of Sensors*, vol. 2016, Jan. 2016.

Conference papers

1. M. Ye, C. Yang, V. Stankovic, L. Stankovic, and A. Kerr, "Gait analysis using a single depth camera," *IEEE Global Conference on Signal and Information Processing*, Orlando, FL, Dec. 2015.

2. M. Ye, C. Yang, V. Stankovic, L. Stankovic, and A. Kerr, "Kinematics analysis multimedia system for rehabilitation," *Workshop on Image and Video Processing for Quality of Multimedia Experience*, Genova, Italy, Sep. 2015.

3. C. Yang, G. Cheung, and V. Stankovic, "Estimating heart rate via depth video motion tracking," *IEEE International Conference on Multimedia and Expo*, Torino, Italy, Jul. 2015. (1 of 8 best paper finalists out of 524 submitted papers.)

4. C. Yang, A. Kerr, V. Stankovic, L. Stankovic, and P. Rowe, “Upper limb movement analysis via marker tracking with a single-camera system,” *IEEE International Conference on Image Processing*, Paris, France, Oct. 2014.

5. A. Kerr, C. Yang, V. Stankovic, and P. Rowe, “Accuracy of a 2D video system for measuring upper limb movement in stroke survivors,” *World Stroke Congress*, Istanbul, Turkey, Oct. 2014.

6. C. Yang, Y. Mao, G. Cheung, V. Stankovic, and K. Chan, “Graph-based depth video denoising and event detection for sleep monitoring,” *IEEE International Workshop on Multimedia Signal Processing*, Jakarta, Indonesia, Sep. 2014.

7. C. Yang, G. Cheung, K. Chan, and V. Stankovic, “Sleep monitoring via depth video recording and analysis,” *IEEE International Workshop on Hot Topics in 3D*, Chengdu, China, Jul. 2014.

8. C. Yang, A. Kerr, V. Stankovic, L. Stankovic, and P. Rowe, “Arm movement analysis via marker tracking with a single-camera system,” *World Congress of Biomechanics*, Boston, MA, Jul. 2014.

9. C. Yang, U. Ugbolue, B. Carse, V. Stankovic, L. Stankovic, and P. Rowe, “Multiple marker tracking in a single-camera system for gait analysis,” *IEEE International Conference on Image Processing*, Melbourne, Australia, Sep. 2013.

10. C. Yang, U. Ugbolue, B. Carse, V. Stankovic, L. Stankovic, and P. Rowe, “Multiple marker tracking in a single-camera system for gait analysis,” *Congress of the International Society of Biomechanics*, Natal, Rio Grande de Norte, Brazil, Aug. 2013.

Other publications

1. C. Yang, Y. Mao, G. Cheung, V. Stankovic, and K. Chan, “Non-intrusive apnoea / hypopnoea detection system via a graph-signal analysis of Microsoft Kinect captured depth video,” *Congress of the European Sleep Research Society*, Tallinn, Estonia, Sep. 2014.

2. A. Kerr, C. Yang, P. Rowe, and V. Stankovic, “Accuracy of a 2D video system for measuring upper limb movement in stroke survivors,” *Annual Meeting of the Clinical Movement Analysis Society UK and Ireland*, Oswestry, UK, Apr. 2014.

Chapter 2

Multimedia Motion Analysis

2.1 Introduction

In this chapter, we review state-of-the-art health monitoring systems and image and video processing methods that are related to the three targeted applications in Section 1.2: gait analysis, upper limb motion analysis for post-stroke recovery level classification, and abnormal respiratory event detection for sleep monitoring. We do this by first reviewing in Section 2.2 health monitoring systems and image and video processing algorithms that are related to limb motion analysis and briefly describing our solutions, and then in Section 2.3 systems and algorithms that are related to respiratory motion analysis during sleep with our solutions.

2.2 Limb motion analysis

Stroke is a worldwide healthcare problem which often causes long-term motor impairment, handicap, and disability [18, 19]. Advanced objective clinical gait analysis on stroke patients can generate quantified, standardised, and more reliable gait measurements [19] compared to traditional, semi-subjective [20], observational gait analysis methods [21, 22], while being minimally intrusive to the stroke patients [20, 23]. Some examples include acoustic gait analysis systems [24], optical non-wearable motion analysis systems [4, 20] based on strategically located infrared cameras to capture three dimensional (3D) limb motion by tracking retroreflective markers adhered to the skin overlying anatomical landmarks of the study participants (see Fig. 2.1 for an illustration), and markerless systems that are completely contact-less to patients, such as Organic Motion OpenStage 2.0 (Organic Motion HQ, New York, NY).

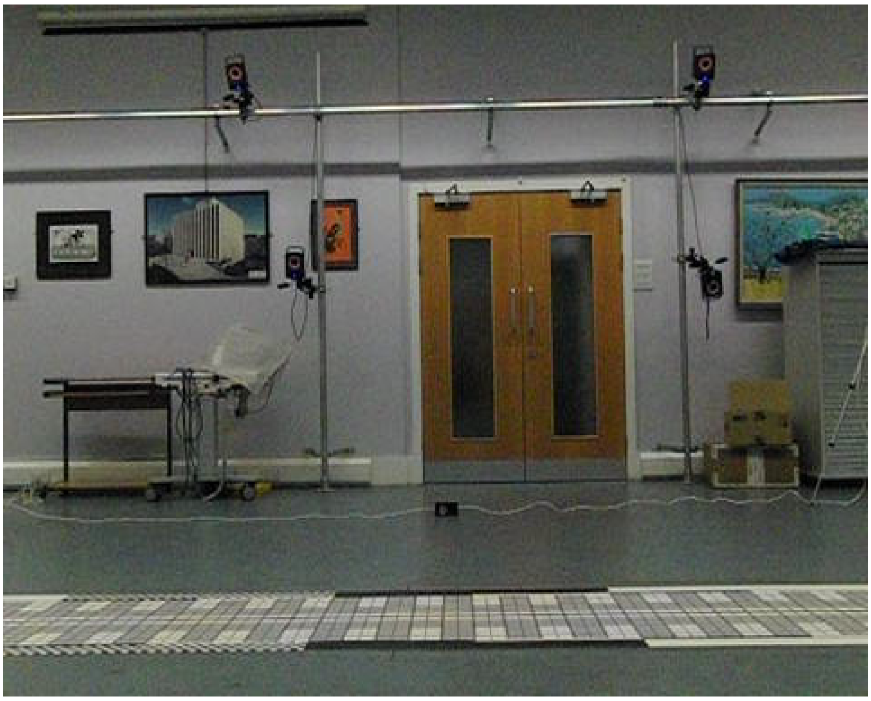


Fig. 2.1 Illustration of a multi-camera motion analysis system. Note that multiple cameras are installed on the wall in a laboratory.

However, all these systems have downsides, such as: (1) they require operational expertise and large laboratory space – hence patients need to be regularly transported to major clinics for assessment; (2) they do not facilitate easy comparison with results from the previous assessment in a longitudinal study; (3) they are expensive; and (4) they cannot distinguish between gradual and abrupt functional changes which negatively affect clinical intervention [25]. Additionally, markerless systems are particularly sensitive to the motion capture background and ambient lighting (*e.g.*, OpenStage 2.0 requires white fabric walls and strong stage lights), which could make patients uncomfortable.

Motivated by cost and providing a convenient option to patients and health services, further research on the development of cost effective and portable systems has emerged. The portability of these systems would enable conducting gait analysis with adequate fidelity outside a gait laboratory, *e.g.*, at local clinics and homes. After measurements are taken, the stroke patient would send the analyzed gait parameters to physiatrists for near real-time clinical consultation, which has the potential to facilitate the development of increasingly popular tele-rehabilitation [26–30]. In particular, an automatic 2D single-camera gait kinematics analysis system is proposed in [31]. However, the requirement for a dark background and a dark suit with gloves to be worn by patients limits the flexibility of system usage; the system is validated without a gold standard on only one healthy volunteer with one walking trial. A detailed gait kinematic parameters evaluation of a 2D single-camera gait analysis system (approximate cost £700) is presented in [32]. The evaluated system is built on [33] with Pro-Trainer motion analysis software (Sports Motion, Inc., Cardiff, CA), showing excellent agreement with a gold standard VICON MX Giganet 6×T40 and 6×T160 (VICON Motion Systems Ltd., Oxford, UK, approximate cost £250,000) optical motion analysis system, and similar to Siliconcoach video analysis software (Siliconcoach Ltd., Dunedin, New Zealand) as used by [34, 35]. For all the above, operational expertise to manually process the measurements is a time consuming and impractical procedure for clinical use.

An autonomous object tracking framework is required motivated by the need for cost-effective tele-rehabilitation and initial findings [31, 32]. In particular, the overall task is to track the interest human joints and quantify the movement by joint angle calculation based on the tracked joint locations. This can be performed using or without using markers. Marker-based approaches require individual markers to be attached to the human joints. On the other hand, marker-less approaches can directly track the human joints. However, most of state-of-the-art marker-less body joint tracking algorithms still lack tracking stability, low in tracking accuracy, or require large dataset for body joint classifier training [7], which is not practical for small scale studies. In terms of practicability and comfort, a sub-optimal solution to human joint tracking is by tracking the markers that are attached to the joints. Specifically, these markers can be tracked individually (for accuracy) or simultaneously (for efficiency).

Simultaneously tracking all markers in colour videos is challenging due to the following marker features: 1) The markers clinically used in 2-D video-based kinematic analysis are *identical* and are in close proximity, which can easily cause tracking confusion. 2) The size, orientation, and appearance of each marker could change due to the joints movement, and thus the tracker should be capable of handling such non-rigid objects. 3) These markers move along with the limb motion of the subject, *i.e.*, small-size target objects move with a large moving object that can be assumed as the appearance-changing background [36], which potentially distracts marker tracking which reduces the tracking accuracy, and thus the tracker needs to address the *object-on-object* tracking problem.

There is a substantial amount of work on object tracking, and good surveys can be found in [36–39]. Next, we review the work most relevant to ours. A point-based tracking method is proposed in [40]. This method represents each object with object-correspondence-points. However, this approach cannot handle non-rigid objects. A silhouette tracking method is proposed in [41] that handles non-rigid objects well. This method consists of building online shape priors and implementing object contour evolvement using energy minimization in gradient descent direction for target objects.

However, this method is only capable of tracking objects that are very different. A kernel tracking method, proposed in [42], jointly applies local binary pattern texture with color histogram (JCTH) which effectively extracts the features of the edges and corners within the target region, and adopts mean-shift with the above JCTH approach and acquires robust performance for tracking objects that have similar color appearance to the background. However, the *object-on-object* problem significantly affects the tracking accuracy and can cause tracking failure. Another kernel tracking method in [43] applies online learning and binary classification within a Tracking-Learning-Detection (TLD) scheme to update the object template adaptively. *i.e.*, this learning-based kernel tracking method is robust for tracking non-rigid objects. However, online learning in [43] is achieved by searching a global frame, which means [43] cannot be directly used for simultaneously tracking multiple objects. "Struck" (STR) [44] is the best tracker among 19 state-of-the-art trackers tested in [39] and a highly competitive online tracker gauged in [37, 45, 46]. The tracking scheme in [44] is based on structured output prediction with kernels. Still, this kernel based method cannot handle out-of-plane rotation well, and there is no object-dynamic model incorporated into this adaptive tracking-by-detection framework. Furthermore, a particle swarm optimization method is proposed in [47], and a particle filter-mean shift joint tracking algorithm is proposed in [48], both of which achieve simultaneous multiple objects tracking. However, these two methods cannot address the *object-on-object* problem. Table 2.1 shows the disadvantages of above state-of-the-art object tracking methods.

In Chapter 3, we propose an alternative to state-of-the-art optical motion analysis systems, inexpensive and portable, 2D single-camera gait kinematics analysis system, including video acquisition, autonomous detection of the frame when tracking starts and ends, discrete-Kalman-filter+Structural-Similarity-based individual marker tracking scheme (DKF-SSIM) that shows a significant improvement with respect to JCTH track-

Table 2.1 Disadvantages of state-of-the-art object tracking methods when applied to tracking identical markers for human motion analysis.

literature	object tracking method	disadvantages
[40]	point-based, correspondence points	cannot handle non-rigid objects
[41]	silhouette-based, online shape priors	cannot distinguish objects with the same appearance
[42]	kernel-based, local binary pattern with color histogram	easily fails to track the small objects that are on top of a large moving object (object-on-object problem)
[43]	kernel-based, online learning, binary classification	cannot be directly used for multi-target tracking due to global search
[44]	kernel-based, structured output prediction	cannot handle out-of-plane rotation
[47]	particle swarm optimization	cannot handle the object-on-object problem
[48]	particle filter-mean shift	cannot handle the object-on-object problem

ing approach [42] and a Tracking-Learning-Detection (TLD) scheme [43], autonomous knee angle calculation, autonomous gait event detection, and result visualization. Our system addresses some of the drawbacks of [31], [32], namely: (1) Unlike [31], there are no colour restrictions on background or the participant's clothing; (2) Soda et al. [31] is validated on only one healthy volunteer with one walking trial with no gold standard benchmark. In contrast, we validate our proposed system's knee angle against VICON. (3) Unlike systems of [32] and Pro-Trainer and Siliconcoach (Siliconcoach Ltd., Dunedin, New Zealand) as used by [34] and [35] that require significant manual effort, our system autonomously tracks the markers attached to the joints with occlusion handling and calculates the knee angle; the only operational effort required is for marker-template selection for tracking initialization which can be done via a user-friendly graphical user interface (GUI).

Motivated by the fact that arm impairment is also a common outcome of stroke [19, 49], building on Chapter 3 that the marker tracking result is used for *manual* impairment assessment of stroke survivors via gait analysis, in Chapter 4, we propose a decision support system for upper limb motion analysis that simultaneously tracks a number of identical bullseye markers, and maps the trajectories of the tracked markers into meaningful information used for rehabilitation assessment. The system comprises a single high-speed camera together with a visualisation module that enables navigating through the captured frames, selecting parameters to present, and comparison with the previous results.

The data analytics part of our solution can be used independently of the capturing module to process autonomously existing reach-to-grasp (RTG) video datasets (see Section 4.3), that contain recordings of RTG movements in the sagittal plane with multiple bullseye markers adhered to the joints of a human body, which are a common alternative to 3D datasets. Similar to gait analysis in Chapter 3, we use black-and-white bullseye markers that are conventionally used in 2D video-based clinical kinematic analysis [50], in the RTG datasets, attached to the skin overlying anatomical landmarks of the subject's joints.

The motion of the subject's upper limb kinematics is captured by tracking the markers frame by frame and autonomously computing joint angles. Once the joint angles have been extracted in each frame, they are used as classification features to automatically estimate the level of impairment [51]. Data classification using regularization on graphs [52–54] is proposed in [55], where it is shown that graph-based supervised binary classification shows competitive performance to conventional classifiers, such as Support Vector Machine (SVM) [56, 57] and neural networks, and good robustness to noise in the training dataset. The main idea is to first represent the dataset to be classified as a signal indexed by a graph, whose vertices correspond to samples in the dataset and weighted edges reflecting similarities or correlation between vertices, then minimize total variation on a graph [58] based on a binary mapping of this graph. In Chapter 4 (see Section 4.2.3), we propose two regularization on graph signals (RGS) based multi-class classification methods, by first constructing graphs for the motion patterns obtained as a result of object tracking, and then designing binary mappings of these graphs using graph-based tools following [55] for minimization of the total variation on graphs [58]. We also propose a third RGS multi-class classification method, by first constructing a graph following [55], and then, designing a multi-class mapping of this graph, unlike binary mappings in [55, 58], and minimize the total variation on graph.

After the publication of [11–14], in [59] a single depth sensing device is used for a marker-based gait kinematics analysis system. This system consists of infrared and depth video capture with video data cleaning, scene calibration, marker identification, detection and tracking in video, and marker position mapping from image space to the real world space. Although the system in [59] can capture 3D limb motion with high accuracy, it requires an expensive high-performance laptop during the video capture phase.

2.3 Respiratory-during-sleep motion analysis

Sleep occupies approximately one third of people's daily activities. It is well understood that quantity and quality of sleep could significantly affect work productivity [60]. In particular, *obstructive sleep apnoea*, characterized by repetitive obstruction in the upper airway during sleep, is common in the general population [61] and can have significant negative effect on a person's sleep quality, and hence quality of life and cognitive functions. The condition is diagnosed via attended (in-laboratory) or unattended (ambulatory) diagnostic sleep studies. We address the problem of identifying the obstructive respiratory events in Chapter 5.

To detect different respiratory events (that characterize obstructive apnea, hypopnea and central apnea), there exist in-laboratory monitoring devices such as system Alice6 LDxS (Philips) that measure a patient's physiological parameters such as oxyhemoglobin saturation, oronasal airflow etc, using various sensors physically attached to the patient's body. In particular, according to the American Academy of Sleep Medicine (AASM) Manual 2007 [62], an apnea is defined by a drop in the peak respiratory airflow by $\geq 90\%$ from the baseline and the duration of the event lasts at least 10 seconds. An obstructive apnea is associated with continued or increased inspiratory effort throughout the entire period of absent airflow. In contrast, a central apnea is associated with absent inspiratory effort. A mixed apnea is associated with the initial portion of the event with absent effort followed by the resumption of such in the latter part of the event. A hypopnea is defined by a drop of $\geq 30\%$ airflow from the baseline and the event lasts for at least 10 seconds, and such change is associated with a 4% drop in oxyhemoglobin desaturations [62].

However, existing in-laboratory monitoring devices are cumbersome to use, expensive, and intrusive with multiple body straps and tubes that affect a patient's sleep quality during monitoring. On the other hand, less intrusive sleep monitoring units such as vibration-sensing wristbands (*e.g.*, Fitbit[1] and Jawbone UP[2]) mostly record sleep time, *i.e.*, the *quantity* rather than the *quality* of sleep, and are not equipped to detect respiratory events of different kinds as previously described during the night.

On the other hand, recent advances in wireless sensing and multimedia processing have led to the development of many novel sleep monitoring systems, using a variety of sensors such as force, temperature, audio, and image sensors. Most of these systems, however, require wearable sensors (hence not contact-less) or do not have sufficient precision necessary for clinical applications. Numerous smartphone-based systems for

[1] http://www.fitbit.com/

[2] https//jawbone.com/up/

sleep disorder detection have emerged recently (see Table 1 in [63]), based on audio recording and accelerometer measurements. However, there is no scientific evidence regarding clinical usability of these systems, with the exception of those that implement a simple clinically validated questionnaire [63]. Other recent methods not reviewed in [63] are either limited to measuring respiration rate (such as [64]) and sleep duration [65], or require wearable sensors [66, 67]. For example, the system proposed in [66] is capable of detecting sleep apnea, but it requires a smartphone and oximeter to be attached to the patient's body while sleeping. A wearable-sensor based system [67] successfully classifies the patients into those with apnea episodes and those without, with over 90% accuracy, but requires wearing an armband containing a phone, attaching a microphone on the face, and an oximeter to the wrist. Further, the classification scheme is limited to apnea / non-apnea subject classification, rather than detection of individual episodes of sleep apnea (medically defined 10-sec intervals) and types of apnea (central, obstructive and mixed).

Force sensors placed on top or under the mattress, have also been used for sleep monitoring and estimation of heart rate, respiration rate, snoring periods, etc [68–71]. There is no evidence, however, that such systems can differentiate among central, obstructive and mixed apnea.

The system in [72] estimates respiratory rate using received signals from wireless sensor nodes. However, the system requires a large number of wireless sensors to provide high accuracy (between 15 and 20 sensor nodes), only the test subject can be present in the room, and it is unclear if the system is accurate enough to detect apnea episodes based only on the detected breathing rate.

Video is used for non-contact sleep monitoring in [73–78]. Using video for sleep monitoring requires capturing the breathing action from the recorded images based on human pose estimation—a long-standing problem in computer vision [7, 79–81].

For sleep monitoring, since color images are usually not available (due to the typically dark sleeping environment), and there is no clear separation between foreground object (patient under a blanket) and background (bed), colour-image hypergraph-distance [82] and pairwise-distance [83] based detection methods, and generic pose estimation techniques such as [80, 84], are not suitable for estimating the sleep pose.

A Microsoft (MS) Kinect infrared sensor is adopted in [73] and [74] for video capture. However, the video based system in [73] is limited to respiration rate monitoring, and the other system [74] is only validated on simulated respiratory events. The depth-video based sleep monitoring system of [75] is limited to sleep-awake status detection. A Time of Flight (ToF) camera was used in [76] to detect chest and abdomen movements for apnea detection, but there is no description of which ToF camera was used and how chest and abdomen movements were deduced from the collected depth measurements. There is also no performance analysis of the proposal against ground truth data. This renders a direct comparison with [76] impossible.

The Kinect colour camera is adopted in [77], where chest movements are detected by tracking over time the closest depth measurement of the patient to a virtual camera directly above the patient. We differ from [77] in three respects. First, we use both audio and video to infer respiratory events, which improves detection accuracy and enables us to distinguish among central, obstructive and mixed apnea. Second, we propose a complete system that includes efficient depth video coding and denoising schemes. Third, unlike [77, 78, 85–88], we propose a more accurate dual-ellipse model, so that individual chest and abdominal movements can be tracked, even if the patient is sleeping sideway.

Motivated by the shortcomings of in-laboratory monitoring devices and consumer-level sleep monitoring units, in Chapter 5, our goal is to accurately but non-intrusively detect respiratory events as manually scored by a scientific officer based on data collected by system Alice6 LDxS. Towards this goal, we propose a *completely contactless* sleep monitoring system based on depth video and audio processing, suitable for home use. Not relying on the lighting condition of a dark sleeping room, we use an MS Kinect sensor projecting infrared light patterns to capture depth images of the sleep patient. See Fig. 2.2 for illustration of our proposed system.

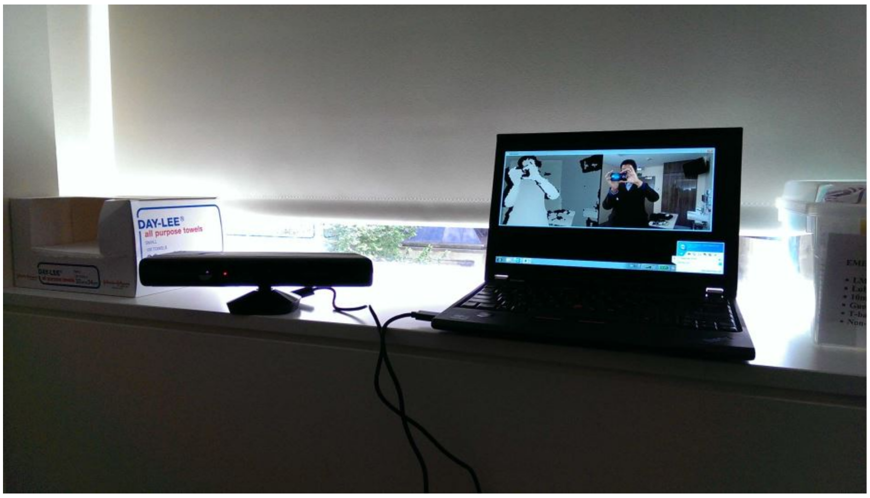


Fig. 2.2 Depth video capturing system at a sleep clinic: an MS Kinect camera is attached to a laptop computer. Example depth and infrared captured images are shown on the screen.

2.4 Summary

In Section 2.2, we analyse the downsides of state-of-the-art laboratory-based optical motion analysis systems for limb motion analysis, reivew cost-effective and portable motion analysis systems, propose a marker-based, single camera 2-D video motion analysis system, and review state-of-the-art object tracking algorithms and classification methods. In Section 2.3, we point out the inconvenience of use of existing in-laboratory sleep monitoring device for abnormal respiratory event detection, review state-of-the-art cost-effective and portable sleep monitoring systems, and propose a MS Kinect sensor based abnormal respiratory event detection system. In the following three chapters, we present in detail our targeted applications to gait analysis, upper limb motion analysis, and abnormal respiratory event detection during sleep, respectively.

Chapter 3

Gait Analysis in Colour Videos[1]

3.1 Introduction

Laboratory-based non-wearable motion analysis systems have significantly advanced with robust objective measurement of the limb motion, resulting in quantified, standardized and reliable outcome measures compared with traditional, semi-subjective, observational gait analysis. However, the requirement for large laboratory space and operational expertise makes these systems impractical for gait analysis at local clinics and homes. In this chapter, we propose a relatively inexpensive, and portable, single-camera gait kinematics analysis system. Our proposed system includes video acquisition with camera calibration, autonomous detection of frames-of-interest, Kalman-filter+Structural-Similarity-based marker tracking, autonomous knee angle calculation, video-frame-identification-based autonomous gait event detection, and result visualization. The only operational effort required is the marker-template selection for tracking initialization, aided by an easy-to-use graphical user interface. The evaluation of the autonomous frames-of-interest detection shows high accuracy compared with

[1] This chapter is largely based on the work that appeared in 2013 IEEE International Conference on Image Processing [11] and Journal of Sensors [12].

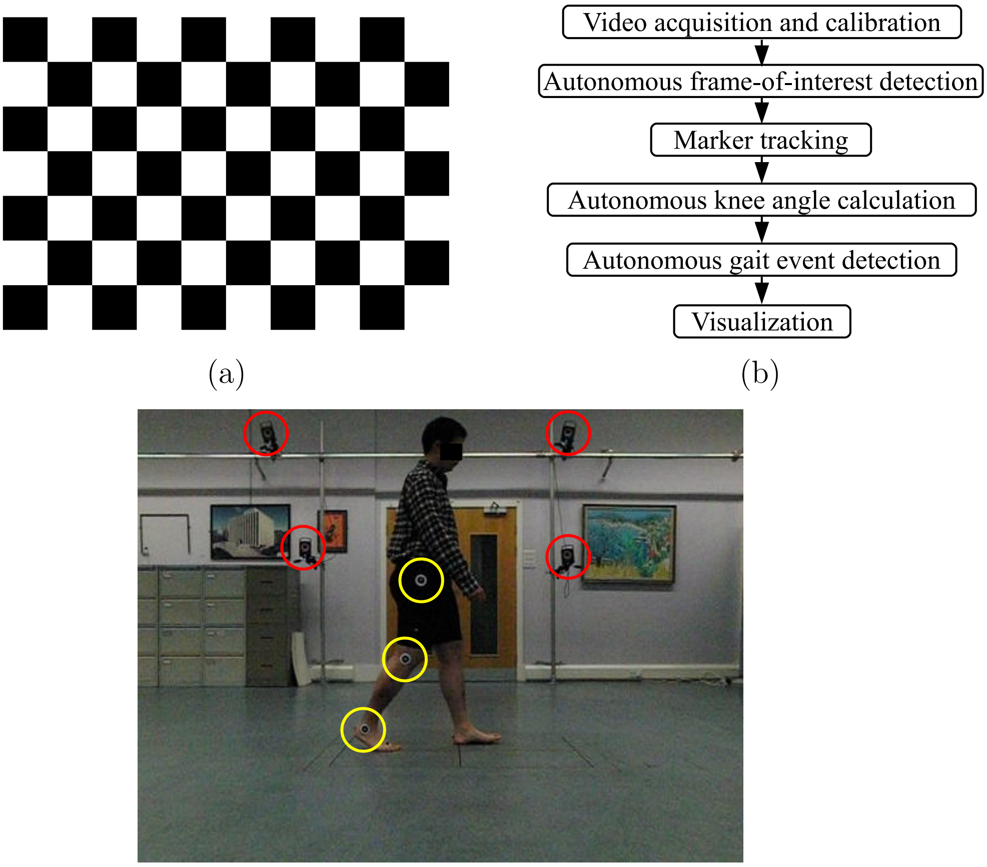
the ground truth. The knee angle validation on 10 stroke patients and 5 healthy volunteers against a gold standard optical motion analysis system shows R-squared value larger than 0.95 and Bland-Altman plot results smaller than 5 degrees mean difference. The autonomous gait event detection shows high detection rates for all gait events. Experimental results demonstrate that the proposed system can automatically measure the knee angle and detect gait events with good accuracy, and thus offer an alternative, cost effective and convenient solution for clinical gait kinematics analysis.

The remainder of this chapter is organized as follows. In Section 3.2 we give a detailed description of the proposed system configuration. In Section 3.3 we present the experimental results for each system block. We discuss the performance and potential improvements of the proposed system in Section 3.4 and summarize this chapter in Section 3.5.

3.2 Proposed gait analysis system

3.2.1 System overview

The system comprises a digital camera EX-FH20 EXILIM (Casio Computer Co., Ltd., Tokyo, Japan) with a tripod, 6 bulls-eye black-and-white paper markers [32], a 10×7 calibration checkerboard with square size of 23.3mm, shown in Figure 3.1(a), and a laptop with bespoke data processing software and a graphic user interface (GUI) developed in MATLAB R2014b (MathWorks, Inc., Natick, MA). The goal of the system is to autonomously analyse the study participant's gait kinematics indicated by knee angle. This is achieved by tracking three bulls-eye markers attached to the skin (or tight-fitting clothing for medical use) overlying the joint centres of hip, knee, and ankle of the study participant, in the sagittal plane. Note that we use black-and-white bullseye markers that are conventionally used in 2D video-based clinical kinematic analysis [50], in the proposed gait analysis system. As shown in Figure 3.1(b), the system procedure includes video acquisition and camera calibration, autonomous frame-of-interest detection, marker tracking, autonomous knee angle calculation and gait event detection, and result visualization. We describe each of these acquisition and processing steps in the following sections.


Video acquisition and calibration
Autonomous frame-of-interest detection
Marker tracking
Autonomous knee angle calculation
Autonomous gait event detection
Visualization

(a) (b)

(c)

Fig. 3.1 (a) Calibration checkerboard; (b) System procedure; (c) Sample single-camera scene.

3.2.2 Video acquisition and camera calibration

Before video acquisition, 6 bulls-eye markers are attached to the skin overlying the hip, knee, and ankle joint centres on both legs of a study participant in the sagittal plane. The study participants walk from left to right and back on a 6×0.7m mat using a similar approach to [32]. The digital camera is configured at 360×480 pixel resolution, 210 frames per second (fps), mounted on a tripod with 0.5-1.0m in height and positioned 1.5-3.0m away from a long-side centre of the mat, depending on the study participant, and calibrated using [89] with the 10×7 checkerboard to remove lens distortion in the video frames, which are then used for frame-of-interest detection, marker tracking and knee angle calculation.

For benchmarking with the gold standard VICON system, the SWIFT Cast trial protocol [90, 91] is applied in the stroke-patient group and Plug-in-Gait protocol [92] in the healthy-volunteer group. For each stroke patient, retroreflective markers (14mm-diameter) are fixed to the skin overlying the anatomical landmarks, as done in [91]. The knee flexion / extension axes are determined based on marker clusters at the femur and tibia and single calibration markers, followed by corresponding joint angle calculation as in [91]. For each healthy volunteer, 15 retroreflective markers (14mm-diameter) are fixed to the skin overlying the following anatomical landmarks adapted from the Plug-in-Gait protocol [92], denoted as: sacral wand marker, left (right) anterior superior iliac spine, knee, femur, ankle, tibia, toe, and heel markers, followed by joint angle calculation based on the Euler / Cardan angle determination algorithm with an y-x-z axis rotation sequence, namely flexion / extension, adduction / abduction, and internal / external rotation [92]. For both groups, all VICON motion-capture modules are calibrated.

Each study participant is simultaneously recorded using the proposed system and VICON. Figure 3.1(c) shows a sample single-camera scene for a healthy volunteer, where 4 out of 12 VICON infrared cameras are marked with red circles and 3 bulls-eye markers on the left leg of the study participant are marked with yellow circles. Note that, the VICON markers are attached on top of the proposed video system markers, and do not negatively affect the video tracking performance.

3.2.3 Autonomous frames-of-interest detection

Video recording starts when the study participant begins walking even though he/she is still not within the camera's field of view. The method needs to automatically recognize the first and last frames when all three markers are present (called "frames-of-interest") to start the marker tracking process. Due to a noticeable change of the frame histogram when a study participant walks into and out of the camera scene, we propose an image histogram-based frames-of-interest detection scheme that identifies at which frame the system starts tracking the markers and at which frame tracking stops. An image histogram shows the number of pixels of each intensity in a frame. To recognize entrance from both left and right sides, we define two frame segments as shown in Figure 3.2, denoted as S_1 and S_2, for all N frames, denoted as $\{\mathrm{F}_1, ..., \mathrm{F}_N\}$, in a video sequence.

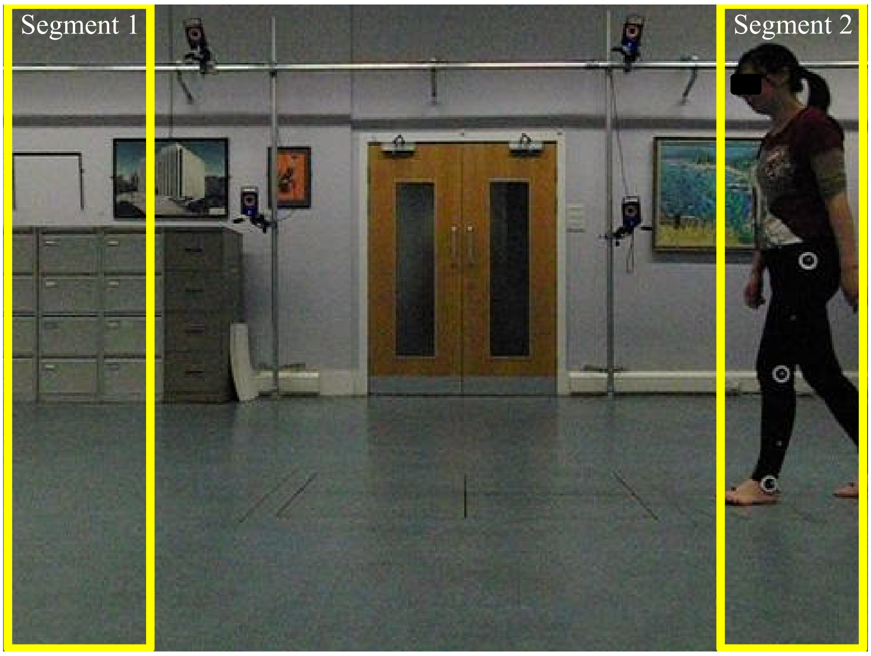


Fig. 3.2 Sample video frame when the study participant is walking into the camera scene with two frame-segments for the detection of the "frames-of-interest".

For each frame segment of the N frames, we compute the histograms, denoted as $\{\mathrm{H}_1^x, ..., \mathrm{H}_N^x\}$, where $x \in \{\mathrm{S}_1, \mathrm{S}_2\}$ and for each $\mathrm{H}_n^x \in \{\mathrm{H}_1^x, ..., \mathrm{H}_N^x\}$, $\mathrm{H}_n^x = \{d_{n1}^x, ..., d_{nm}^x\}$, where n is the frame number, $1 \leq n \leq N$, $1 \leq m \leq M$. M denotes the number of quantization bins of each histogram, and d_m is the total number of pixels in the m-th quantization bin. Next, we compute the difference between H_n^x and H_1^x, for all n, denoted as $\Delta\mathrm{H}_n^x$: $\Delta\mathrm{H}_n^x = \{|d_{n1}^x - d_{11}^x|, ..., |d_{nm}^x - d_{1m}^x|\}$, followed by forming the element sum of $\Delta\mathrm{H}_n^x$, $\sum_{j=1}^{m} \left| d_{nj}^x - d_{1j}^x \right|$, denoted as $\sum \Delta\mathrm{H}_n^x$. The $\{n, \sum \Delta\mathrm{H}_n^x\}$ plot is shown in Figure 3.3. We then perform peak detection in the $\{n, \sum \Delta\mathrm{H}_n^x\}$ plot for each frame segment with a heuristically set threshold τ, where the detected peaks with corresponding frame numbers are denoted as P^x: $\mathrm{P}^x = \{l^x, \sum \mathrm{H}_{l^x}^x\}$, where $\forall l^x$, $\{\sum \Delta\mathrm{H}_{l^x}^x \geq \tau\}$. The detected peaks are marked with red asterisks in Figure 3.3.

Each video sequence contains a pair of left-to-right (LtR) and right-to-left (RtL) walking trials – a study participant walks into and out of the camera scene twice, once from each direction, indicating that there exist two peak clusters, as shown in Figure 3.3. To separate P^x into two clusters, we first compute the difference of frame numbers between neighbouring peaks in P^x, denoted as Δl^x, with $\Delta l_{\max}^x = \arg\max_{l^x} \Delta l^x$, and let l_{m1}^x, l_{m2}^x be the two corresponding frame numbers, *i.e.*, $\Delta l_{\max}^x = l_{m2}^x - l_{m1}^x$. Then we separate P^x as follows:

$\mathrm{P}_1^x = \{\text{a subset of } \{l^x, \sum \mathrm{H}_{l^x}^x\},\ l^x \leq l_{m1}^x\}$;

$\mathrm{P}_2^x = \{\text{a subset of } \{l^x, \sum \mathrm{H}_{l^x}^x\},\ l_{m2}^x \leq l^x\}$.

Let the frame numbers associated with the first and last detected peaks in x be l_{first}^x and l_{last}^x, respectively. If $\Delta l_{\max}^{\mathrm{S}_1} > \Delta l_{\max}^{\mathrm{S}_2}$, which indicates the trial direction in a video sequence D is LtR → RtL, we designate the first and last frames of interest for the LtR trial as Frame $n_{\text{first}}^{\text{LtR}} = l_{m1}^{\mathrm{S}_1}$ and $n_{\text{last}}^{\text{LtR}} = l_{\text{first}}^{\mathrm{S}_2}$, respectively, and for the RtL trial as $n_{\text{first}}^{\text{RtL}} = l_{\text{last}}^{\mathrm{S}_2}$ and $n_{\text{last}}^{\text{RtL}} = l_{m2}^{\mathrm{S}_1}$, respectively. If $\Delta l_{\max}^{\mathrm{S}_1} < \Delta l_{\max}^{\mathrm{S}_2}$, which indicates the trial

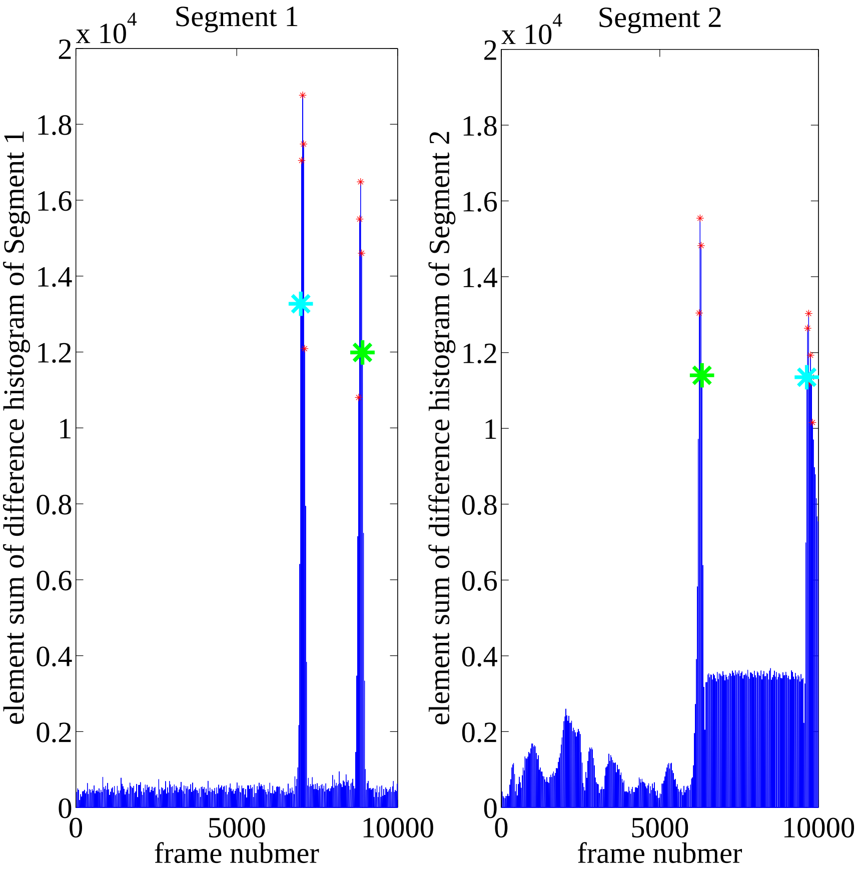


Fig. 3.3 The $\{n, \sum \Delta \mathrm{H}_n^x\}$ plots for two frame-segments with marked detected first (green asterisks) and last (cyan asterisks) frames of interest..

direction is RtL → LtR, then $n_{\text{first}}^{\text{RtL}} = l_{m1}^{\text{S}_2}$, $n_{\text{last}}^{\text{RtL}} = l_{\text{first}}^{\text{S}_1}$, and $n_{\text{first}}^{\text{LtR}} = l_{\text{last}}^{\text{S}_1}$, $n_{\text{last}}^{\text{LtR}} = l_{m2}^{\text{S}_2}$. The overall frames-of-interest detection scheme is summarized in Algorithm 1.

In all experiments, we set a size of 360×80 pixels for each frame-segment, and a threshold $\tau = 10000$ for peak detection. For evaluation, we manually label the frames where all three markers on the same leg first and last appear for each trial as the ground truth, and compare them with the corresponding detected frames using the following "frame difference rate (FDR)" measure:

$$\text{FDR} = \frac{|n_{\text{detected}} - n_{\text{labelled}}|}{n_{\text{labelled}}} \times 100\% \tag{3.1}$$

Algorithm 1: Frames-of-interest detection for marker tracking.

```
Input: D, $N$, $\mathrm{S}_1$, $\mathrm{S}_2$.
Output: $n^{\mathrm{LtR}}_{\mathrm{first}}$, $n^{\mathrm{LtR}}_{\mathrm{last}}$, $n^{\mathrm{RtL}}_{\mathrm{first}}$, $n^{\mathrm{RtL}}_{\mathrm{last}}$.
initialize $n = 1$;
for $n \leq N$ do
    $\mathrm{F}_n = \mathrm{D}(n)$;
    if $n = 1$ then
        $\mathrm{F}_1$, $\mathrm{S}_1$, $\mathrm{S}_2$, $\Rightarrow \mathrm{H}^x_1$;
    else
        $\mathrm{F}_n$, $\mathrm{S}_1$, $\mathrm{S}_2$, $\Rightarrow \mathrm{H}^x_n \Rightarrow \Delta\mathrm{H}^x_n \Rightarrow \sum \Delta\mathrm{H}^x_n$;
    $n = n + 1$;
$\{n, \sum \Delta\mathrm{H}^x_n\}$, peak detection $\Rightarrow \mathrm{P}^x$;
$\mathrm{P}^x$, difference of the frame numbers $\Rightarrow \Delta l^x \Rightarrow \Delta l^x_{\mathrm{max}} \Rightarrow \mathrm{P}^x_1$, $\mathrm{P}^x_2$;
if $\Delta l^{\mathrm{S}_1}_{\mathrm{max}} > \Delta l^{\mathrm{S}_2}_{\mathrm{max}}$ then
    $\mathrm{P}^x_1$, $\mathrm{P}^x_2$, LtR $\rightarrow$ RtL $\Rightarrow n^{\mathrm{LtR}}_{\mathrm{first}}$, $n^{\mathrm{LtR}}_{\mathrm{last}}$, $n^{\mathrm{RtL}}_{\mathrm{first}}$, $n^{\mathrm{RtL}}_{\mathrm{last}}$;
else
    $\mathrm{P}^x_1$, $\mathrm{P}^x_2$, RtL $\rightarrow$ LtR $\Rightarrow n^{\mathrm{RtL}}_{\mathrm{first}}$, $n^{\mathrm{RtL}}_{\mathrm{last}}$, $n^{\mathrm{LtR}}_{\mathrm{first}}$, $n^{\mathrm{LtR}}_{\mathrm{last}}$;
```

where n_{labelled} and n_{detected} denote the frame numbers of the manually labelled frame and detected frame, respectively.

3.2.4 Marker tracking and autonomous knee angle calculation

We formulate the marker tracking task as automatically finding the centre coordinate of each marker on the camera-facing leg independently from other markers, frame by frame. For initialization, the marker-templates for hip, knee, and ankle markers are manually selected via mouse-click in the first frame of the video shown in the "Current frame" panel of the GUI in Figure 3.4. Three markers are individually tracked via a DKF-SSIM tracking scheme. The centre coordinate of each tracked marker in each frame is simultaneously determined for autonomous knee angle calculation.

In particular, for each marker, first a Search Area (SA) is set in each frame, where the position and size of the SA is determined by a DKF [9]. We first define the SA of size $h \times h$ pixels, and the centre coordinate and velocity given by $\hat{\mathbf{s}}_i$= $[f_i \; g_i \; u_i, v_i]^\top$, where f_i

and g_i denote the column and row centre coordinate of the SA in frame F_i, respectively, and u_i and v_i denote velocities along horizontal and vertical directions, respectively. The column and row centre coordinate of the marker in F_i are denoted as $\hat{\mathbf{u}}_i = [c_i \; r_i]^\top$. We adopt a DKF [9] with a four-dimensional constant-velocity model with additive discrete-time noise [93], whose dynamic and observation models are constructed by $\hat{\mathbf{s}}_i$ and $\hat{\mathbf{u}}_i$, respectively. This DKF consists of the prediction and correction phases:

$$\begin{aligned}
&\text{prediction phase}: \begin{cases} \hat{\mathbf{s}}_i^- = \mathbf{R}\hat{\mathbf{s}}_{i-1}, \\ \mathbf{X}_i^- = \mathbf{R}\mathbf{X}_{i-1}\mathbf{R}^\top + \mathbf{B}. \end{cases} \\
&\text{correction phase}: \begin{cases} \mathbf{K}_i = \mathbf{X}_i^-\mathbf{Q}^\top(\mathbf{Q}\mathbf{X}_i^-\mathbf{Q}^\top + \mathbf{E})^{-1}, \\ \hat{\mathbf{s}}_i = \hat{\mathbf{s}}_i^- + \mathbf{K}_i(\hat{\mathbf{u}}_i - \mathbf{Q}\hat{\mathbf{s}}_i^-), \\ \mathbf{X}_i = (1 - \mathbf{K}_i\mathbf{Q})\mathbf{X}_i^-. \end{cases}
\end{aligned} \tag{3.2}$$

where $\hat{\mathbf{s}}_i^-$ is the a priori estimate of $\hat{\mathbf{s}}_i$ in F_i, $\hat{\mathbf{s}}_{i-1}$ is the a posteriori estimate, $\mathbf{R}$ is the state transition matrix, $\mathbf{X}_i^-$ is the a posteriori covariance matrix, $\mathbf{B}$ is the process noise covariance matrix pre-computed by running the filter off-line based on the assumption that $\mathbf{B}$ is time invariant [9], $\mathbf{K}$ is the Kalman gain, $\mathbf{Q}$ is the observation matrix, and $\mathbf{E}$ is the measurement error covariance matrix pre-computed by running the filter off-line based on the assumption that $\mathbf{E}$ is constant across all frames [9]; v_i, which is in $\hat{\mathbf{s}}_i$, is determined by the DKF (K-velocity). In particular, $\mathbf{R}$, $\mathbf{B}$, $\mathbf{Q}$ and $\mathbf{E}$ are given by:

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & t & 0 \\ 0 & 1 & 0 & t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.3}$$

$$\mathbf{B} = \sigma_v^2 \begin{bmatrix} t^4/4 & 0 & t^3/2 & 0 \\ 0 & t^4/4 & 0 & t^3/2 \\ t^3/2 & 0 & t^2 & 0 \\ 0 & t^3/2 & 0 & t^2 \end{bmatrix}, \tag{3.4}$$

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \tag{3.5}$$

$$\mathbf{E} = \sigma_w^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{3.6}$$

The parameters are heuristically set as $t = 0.01s, \sigma_v = 0.032m/s^2, \sigma_w = 0.071m$.

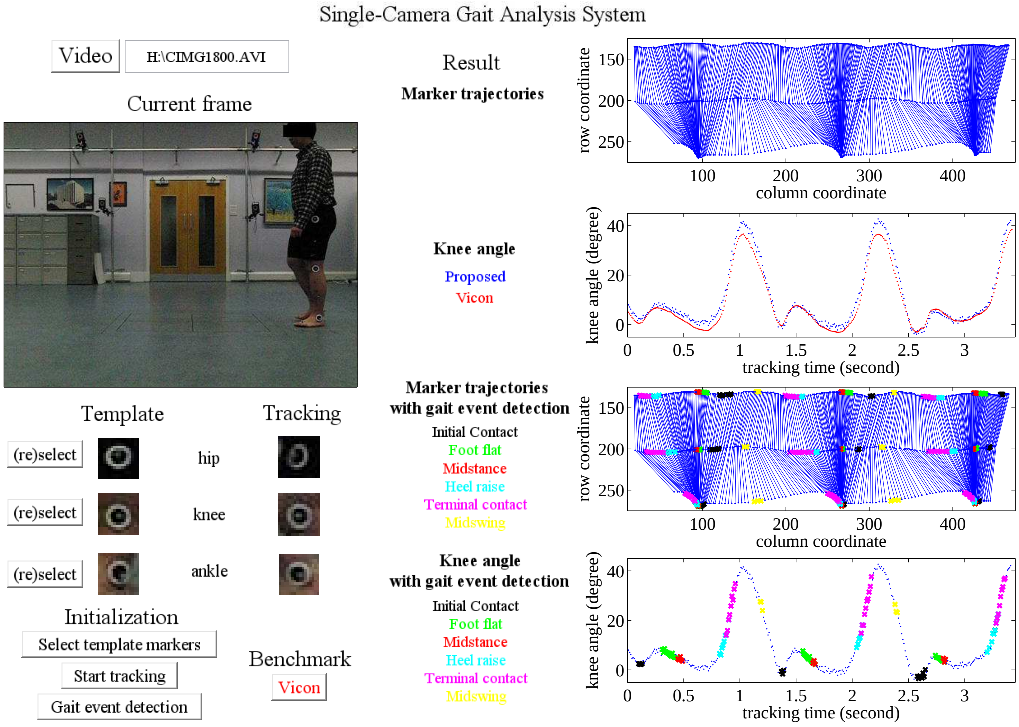


Fig. 3.4 GUI for the proposed gait analysis system.

The above filter is initialized by $\hat{\mathbf{s}}_1^- = [f_0 \; g_0 \; 0 \; 0]^\top$ and $\hat{\mathbf{u}}_1 = [c_1 \; r_1]^\top$, where $f_0 = c_1$, $g_0 = r_1$, and (c_1, r_1) denotes the centre coordinate of the marker-template. The size of each SA is heuristically initialized to $\lfloor 1.4q \rfloor \times \lfloor 1.4q \rfloor$ pixels, given Marker j's size is $q \times q$ pixels, and the four edges of the SA are dynamically updated in each frame based on $v_{i-1}f_{i-1}$ and $v_{i-1}g_{i-1}$. Given the duration of one frame is t seconds, if $v_{i-1}f_{i-1} \geq 0$, the right edge of $\hat{\mathbf{s}}_i$ is shifted to the right by $v_{i-1}f_{i-1}t$ pixels; otherwise, the left edge of the SA is shifted to the left by the same number of pixels. Similarly, if $v_{i-1}g_{i-1} \geq 0$, the bottom edge of the SA is shifted down by $v_{i-1}g_{i-1}t$ pixels; otherwise, the top edge of the SA is shifted up by the same number of pixels.

For template matching, we adopt SSIM with a motion full-search scheme to track the marker within the SA. SSIM is an image quality assessment metric shown in Appendix B.1. SSIM compares a candidate block within the updated SA ($\text{SA}_{\text{updated}}$) in F_n, denoted as $\mathbf{a}_n$, with its corresponding marker-template, denoted as $\mathbf{b}$, where $0 < \text{SSIM}(\mathbf{b}, \mathbf{a}_n) \leq 1$. The candidate block with the largest $\text{SSIM}(\mathbf{b}, \mathbf{a}_n)$, over all $\mathbf{a}_n$ in SA denoted as $\mathbf{a}_n^{\text{best}}$, is designated as the tracked marker; we denote its centre coordinate as $\hat{\mathbf{u}}_{n+1}$, which is used to update the observation and dynamic models in the above Kalman filter.

There exists several occlusion phases (OP) for the hip marker due to arm swing. We address this occlusion problem by setting a heuristically determined threshold τ_{op}, that is, the frame where $\text{SSIM}(\mathbf{a}_n^{\text{best}}, \mathbf{b}) \leq \tau_{op}$ is the first frame of occlusion, and its frame number is denoted by $n_{\text{start}}^{\text{OP}}$. The SSIM exhaustive search algorithm continues to process the subsequent frames until $\text{SSIM}(\mathbf{a}_n^{\text{best}}, \mathbf{b}) > \tau_{op}$, which indicates that the hip marker has appeared again after occlusion, and its frame number is denoted by $n_{\text{end}}^{\text{OP}}$. Next, nonlinear interpolation, based on the centre coordinates of the hip marker and the distances between the hip and knee markers in $\text{F}_{n_{\text{start}}^{\text{OP}}}$ and $\text{F}_{n_{\text{end}}^{\text{OP}}}$, is performed to estimate the centre coordinates of the hip marker, denoted as $\{\hat{\mathbf{u}}\}_{\text{OP}}$, within the occluded frames $\{\text{F}_{n_{\text{start}}^{\text{OP}}}, \ldots, \text{F}_{n_{\text{end}}^{\text{OP}}}\}$. The overall marker tracking procedure for each marker is summarized in Algorithm 2.

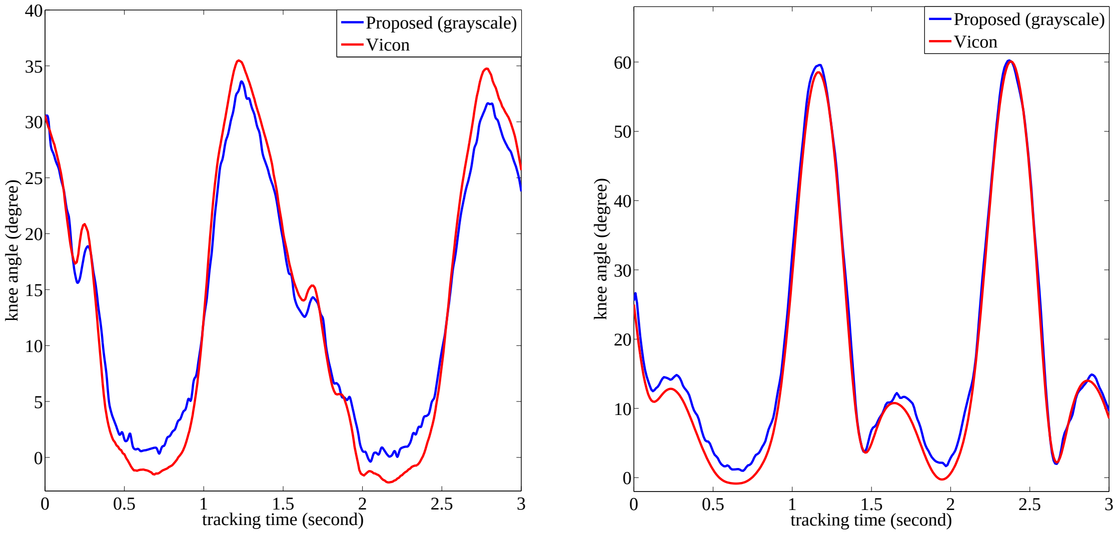
Algorithm 2: Marker tracking for a sample LtR walking trial.

```
Input: $\mathbf{D}$, $n^{\text{LtR}}_{\text{first}}$, $n^{\text{LtR}}_{\text{last}}$, $\mathbf{b}$, $\mathbf{R}$, $\mathbf{B}$, $\mathbf{Q}$, $\mathbf{E}$, $\tau_{op}$, $t$.
Output: $\{\hat{\mathbf{u}}\}$.
initialize $\hat{\mathbf{s}}_{n-1}, \hat{\mathbf{u}}_{n-1}$, $n = n^{\text{LtR}}_{\text{first}} + 1$, occlusion state (OS) $= 0$,
occlusion phase (OP) $= 0$;
for $n \leq n^{\text{LtR}}_{\text{last}}$ do
    $\text{F}_n = \mathbf{D}(n)$;
    if OS $= 0$ then
        DKF: $\mathbf{R}$, $\mathbf{B}$, $\mathbf{Q}$, $\mathbf{E}$, $\hat{\mathbf{s}}_{n-1}, \hat{\mathbf{u}}_{n-1}$, Equation (3.2) $\Rightarrow \hat{\mathbf{s}}^{-}_{n-1}$, $\mathbf{S}_n$;
        SSIM: $\mathbf{b}$, $\text{F}_n$, $\mathbf{S}_n \Rightarrow \mathbf{a}^{\text{best}}_n$;
        if SSIM$(\mathbf{a}^{\text{best}}_n, \mathbf{b}) < \tau_{op}$ then
            OS $= 1$, OP $=$ OP $+ 1$, $n^{\text{OP}}_{\text{start}} = n$;
            $\overline{v_c} = (\hat{\mathbf{u}}_{n-1}(1) - \hat{\mathbf{u}}_{n^{\text{LtR}}_{\text{first}}}(1))/(n - 1 - n^{\text{LtR}}_{\text{first}})$,
            $\overline{v_r} = (\hat{\mathbf{u}}_{n-1}(2) - \hat{\mathbf{u}}_{n^{\text{LtR}}_{\text{first}}}(2))/(n - 1 - n^{\text{LtR}}_{\text{first}})$;
            $\mathbf{S}^{'\text{CC}}_n = \hat{\mathbf{u}}_{n-1}(1)$, $\mathbf{S}^{'\text{RC}}_n = \hat{\mathbf{u}}_{n-1}(2)$;
        else
            $\mathbf{a}^{\text{best}}_n \Rightarrow \hat{\mathbf{u}}_n$;
    if OS $= 1$ then
        SSIM: $\mathbf{b}$, $\text{F}_n$, $\mathbf{S}'_n \Rightarrow \mathbf{a}^{\text{best}}_n$;
        if SSIM$(\mathbf{a}^{\text{best}}_n, \mathbf{b}) \geq \tau_{op}$ then
            $\mathbf{a}^{\text{best}}_n \Rightarrow \hat{\mathbf{u}}_n$;
            $n^{\text{OP}}_{\text{end}} = n - 1$, OS $= 0$;
            $\{\text{F}_{n^{\text{OP}}_{\text{start}}}, ..., \text{F}_{n^{\text{OP}}_{\text{end}}}\}$, $\Rightarrow \{\hat{\mathbf{u}}\}_{\text{OP}}$;
            DKF: $\mathbf{R}$, $\mathbf{B}$, $\mathbf{Q}$, $\mathbf{E}$, $\hat{\mathbf{s}}_{n^{\text{OP}}_{\text{start}}-1}, \{\hat{\mathbf{u}}\}_{\text{OP}}$, Equation (3.2) $\Rightarrow \{\hat{\mathbf{s}}^{-}\}_{\text{OP}}$, $\{\mathbf{S}\}_{\text{OP}}$;
        else
            $\mathbf{S}^{'\text{CC}}_n = \mathbf{S}^{'\text{CC}}_n + \overline{v_c}t$;
            $\mathbf{S}^{'\text{RC}}_n = \mathbf{S}^{'\text{RC}}_n + \overline{v_r}t$;
```

Since each video frame contains three channels, (R)ed, (G)reen, and (B)lue, we perform marker tracking in the three channels independently, and then calculate the mean values of the centre coordinates of the tracked marker, that is, $\{\hat{\mathbf{u}}\}^{\text{RGB}} = \frac{1}{3}(\{\hat{\mathbf{u}}\}^{\text{R}} + \{\hat{\mathbf{u}}\}^{\text{G}} + \{\hat{\mathbf{u}}\}^{\text{B}})$, given the fact that state-of-the-art trackers such as TLD and STR are only performed in grayscale images. In another approach, the grayscale scheme, we convert each frame into a single grayscale channel before marker tracking, and perform the marker tracking once, getting the centre coordinates of the tracked marker $\{\hat{\mathbf{u}}\}^{\text{grayscale}}$.

The knee angle is automatically calculated during RGB and grayscale marker tracking. Figures 3.5(a) and (b) show the sample knee angle plots of a stroke patient and a healthy volunteer, respectively, during a walking trial using grayscale marker tracking.

The marker trajectories are visualized by mapping the centre coordinates of all tracked markers into a single frame, as shown in the "Marker trajectories" figure of the "Result" panel of the GUI in Figure 3.4. The knee angle is shown in the "Knee angle" figure of the same panel.



(a) (b)

Fig. 3.5 (a) Sample knee angle of a stroke patient using grayscale marker tracking; (b) Sample knee angle of a healthy volunteer using grayscale marker tracking.

3.2.5 Autonomous gait event detection

Locating the gait events in each gait cycle is essential for gait analysis [94]. To the best of our knowledge, current kinetics-based gait event detection methods rely on adequate forceplate strikes [94], as done in conventional optical motion analysis systems such as VICON, where the limited area of the forceplate is impractical for gait event detection within multiple consecutive gait cycles. Similarly to kinematic-based gait event detection algorithms such as [95] and [96], these methods are limited to the detection of Initial Contact (heel strike) and Terminal Contact (toe-off), *i.e.*, only two gait events / phases.

In this section we discuss how we perform autonomous gait event detection which detects all six gait events / phases in each gait cycle, including Initial Contact (IC), Foot Flat (FF), Mid-Stance (MST), Heel Raise (HR), Terminal Contact (TC), and Mid-Swing (MSW) [94], based on processing the marker tracking result.

First, without loss of generality, we denote the (R)ow and (C)olumn coordinates of the (H)ip, (K)nee, and (A)nkle markers on the camera-facing leg in Frame T as $\mathrm{H}_T^{\mathrm{R}}$, $\mathrm{H}_T^{\mathrm{C}}$, $\mathrm{K}_T^{\mathrm{R}}$, $\mathrm{K}_T^{\mathrm{C}}$, $\mathrm{A}_T^{\mathrm{R}}$, and $\mathrm{A}_T^{\mathrm{C}}$, respectively. The motivation of creating the following classification rules is to distinguish each medically defined gait events based on marker locations. This is done by manually watching some of the captured videos and manually summarizing each gait event in terms of the relative positions of neighbouring markers both in the current frame and in the forward-backward frames. These classification rules are used as hand-crafted features for autonomous gait event detection. In particular, we formulate the autonomous gait event detection task as identification of frames where the following holds:

$$
\begin{aligned}
&\mathrm{IC}: \begin{cases} \left|\mathrm{A}^{\mathrm{C}}_{T} - \mathrm{A}^{\mathrm{C}}_{T-\lambda}\right| \geq \epsilon; \\ \max\{\mathrm{A}^{\mathrm{C}}_{T}, ..., \mathrm{A}^{\mathrm{C}}_{T+\lambda}\} - \min\{\mathrm{A}^{\mathrm{C}}_{T}, ..., \mathrm{A}^{\mathrm{C}}_{T+\lambda}\} \leq 3\epsilon; \end{cases} \\
&\mathrm{FF}: \begin{cases} \left|\mathrm{A}^{\mathrm{C}}_{T} - \mathrm{K}^{\mathrm{C}}_{T}\right| \leq 2\epsilon; \\ \max\{\mathrm{A}^{\mathrm{C}}_{T-\lambda}, ..., \mathrm{A}^{\mathrm{C}}_{T+\lambda}\} - \min\{\mathrm{A}^{\mathrm{C}}_{T-\lambda}, ..., \mathrm{A}^{\mathrm{C}}_{T+\lambda}\} \leq 3\epsilon; \end{cases} \\
&\mathrm{MST}: \begin{cases} \left|\mathrm{A}^{\mathrm{C}}_{T} - \mathrm{H}^{\mathrm{C}}_{T}\right| \leq 2\epsilon; \\ \max\{\mathrm{A}^{\mathrm{C}}_{T-\lambda}, ..., \mathrm{A}^{\mathrm{C}}_{T+\lambda}\} - \min\{\mathrm{A}^{\mathrm{C}}_{T-\lambda}, ..., \mathrm{A}^{\mathrm{C}}_{T+\lambda}\} \leq 3\epsilon; \end{cases} \\
&\mathrm{HR}: \begin{cases} \left|\mathrm{A}^{\mathrm{R}}_{T} - \mathrm{A}^{\mathrm{R}}_{T+\lambda}\right| \geq \epsilon; \\ \max\{\mathrm{A}^{\mathrm{C}}_{T-\lambda}, ..., \mathrm{A}^{\mathrm{C}}_{T}\} - \min\{\mathrm{A}^{\mathrm{C}}_{T-\lambda}, ..., \mathrm{A}^{\mathrm{C}}_{T}\} \leq 3\epsilon; \end{cases} \\
&\mathrm{TC}: \begin{cases} \left|\mathrm{A}^{\mathrm{R}}_{T} - \mathrm{A}^{\mathrm{R}}_{T+\lambda}\right| \geq \epsilon; \\ \left|\mathrm{A}^{\mathrm{C}}_{T} - \mathrm{A}^{\mathrm{C}}_{T+\lambda}\right| \geq 2\epsilon; \end{cases} \\
&\mathrm{MSW}: \begin{cases} \left|\mathrm{A}^{\mathrm{C}}_{T} - \mathrm{H}^{\mathrm{C}}_{T}\right| \leq 5\epsilon; \\ \left|\mathrm{A}^{\mathrm{C}}_{T+\lambda} - \mathrm{A}^{\mathrm{C}}_{T-\lambda}\right| \geq \epsilon. \end{cases}
\end{aligned}
$$

where λ and ϵ are two scaling factors. We follow the above heuristically set rules frame by frame, and visualize the autonomous gait event detection result by labelling “X” marks on both marker trajectories and knee angle plot, with a designated colour for each gait event / phase: IC-black, FF-green, MST-red, HR-blue, TC-magenta, and MSW-yellow. For evaluation, we first manually label the most representative frame for each gait event / phase by closely following [94], using a vertical-line in the knee angle plot with the same colourization scheme as for the “X” marks, which is assumed as the ground truth. That is, we use hand-labeled ground truth for all six gait events / phases, since again a conventional forceplate approach can only detect IC and TC [94]. Then we determine if detection is valid by comparing the identified frame by the proposed system with the corresponding ground truth, that is, if the difference between the frame number of a ground truth and that of its nearest same-gait-event detection is less than τ frames, this detection, along with its neighbouring same-gait-event detections, is designated as a single valid detection. Otherwise, these detections are designated as a single invalid detection. Figure 3.6 shows the visualized autonomous gait event detection result in a sample right-to-left walking trial. In this example, the difference between the frame number of the ground truth for the first IC, labelled as a black vertical-line, and that of its nearest same-gait-event detection, labelled as a black "X" mark, is less than τ, thus this detection, along with its neighbouring same-colour detections, is determined as a single valid detection. In practice, we set $\tau = 5$. We describe how to find the optimal λ and ϵ in Sec. 3.3.5.

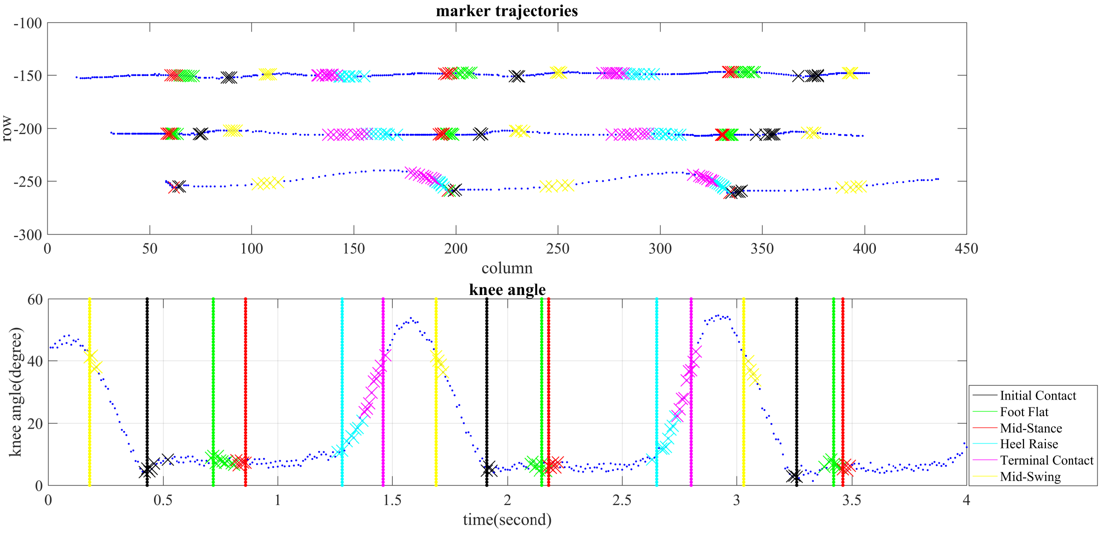


Fig. 3.6 Visualization of the proposed autonomous gait event detection on marker trajectories and knee angle plot with ground truth vertical-line labels in a sample right-to-left trial.

For each one-direction trial, we sum the number of valid detections and ground truth labels, and calculate the detection rate as the evaluation metric:

$$\text{detection rate} = \frac{\text{number of valid detections}}{\text{number of ground truth labels}} \times 100\%$$

3.2.6 System GUI

The system GUI provides all control options, and shows the visualization result on marker trajectories, knee angles, and gait event detection, as shown in Figure 4.3. Via the GUI, one can choose the video to be processed and select (reselect if required, erasing the previous selection) the marker-templates of the hip, knee, and ankle markers by mouse-click in the first frame of the video shown in the “Current frame” panel. The selected marker-templates are shown in the “Template” panel. Marker tracking is launched by clicking “Start tracking” in the “Initialization” panel, followed by showing the tracked marker blocks in the “Tracking” panel, and result visualization in the first two figures of the “Result” panel. Autonomous gait event detection is launched by clicking “Gait event detection” in the “Initialization” panel, followed by showing the visualized result in the subsequent two figures of the “Result” panel. The “Benchmark” panel is used for knee angle validation against VICON, showing the knee angle data from VICON side-by-side with that from the single-camera system in the second figure of the “Result” panel.

3.3 Results

The system is validated on 15 participants, including 10 stroke patients recruited between June 2011 and July 2012 from 4 UK hospitals, and 5 healthy volunteers recruited during May 2014 from the University of Strathclyde staff. Each participant performs two pairs of LtR and RtL walking trials; each trial includes at least 2 consecutive gait cycles. Thus, the test dataset includes 40 trials for stroke patients and 20 trials for healthy volunteers. The knee angle data is down-sampled from 210fps to 100fps, for a fair comparison against VICON (100fps). The data processing module is implemented in MATLAB R2014b on a laptop running Windows 8.1, with Core i7 2820QM 2.3GHz processor and 16GB RAM. In this section, we show the result of the evaluation of frames-of-interest detection, knee angle validation against VICON, accuracy investigation of RGB and grayscale marker tracking, and gait event detection. In all our experiments, the size of each marker template is always $q \times q = 11 \times 11$ pixels, which was heuristically found for optimal appearance representation of each marker that results in best tracking accuracy without sacrificing much computation cost; we set SSIM threshold $\tau_{op} = 0.4$, which gives the best result.

3.3.1 Evaluation of autonomous frames-of-interest detection

The proposed autonomous frames-of-interest detection scheme accurately detects the first and last tracking frames when all markers are present for the marker tracking process in all videos with a mean FDR (see Equation (3.1)) value 0.2%, that is, given a 1000-frame video, the difference between n_{detected} and n_{labelled} is only 2 on average. Moreover, the average execution time for each detection process is only 6.26s.

3.3.2 Comparison with state-of-the-art

In this section, we aim to compare our proposed DKF-SSIM marker tracking scheme with state-of-the-art, while showing the influence of adding Search Area (SA) constraint (see Section 3.2.4) to the competing schemes. We first randomly choose 2 trials, and select bullseye marker templates from the first frame of the corresponding video clip. Next, for each marker, we manually label the marker blocks in all frames of the video clip, with the same size as the marker template, as the ground truth (GT) to assess the bullseye marker tracking performance of the following methods: JCTH [42], JCTH with SA, TLD [43], TLD with SA, and proposed DKF-SSIM. In the JCTH and TLD approaches, for each marker, we fix the size of SA at $\lfloor 1.4q \rfloor \times \lfloor 1.4q \rfloor$ and let the centre

Table 3.1 Bullseye marker tracking in Trial 1.

Method	Precision	Recall	PMR
JCTH [42]	0.438	0.432	17.3%
JCTH [42] with SA	0.438	0.432	17.3%
TLD [43]	0.752	0.726	53.6%
TLD [43] with SA	0.884	0.743	59.7%
DKF-SSIM	0.992	0.994	98.6%

Table 3.2 Bullseye marker tracking in Trial 2.

Method	Precision	Recall	PMR
JCTH [42]	0.274	0.317	5.6%
JCTH [42] with SA	0.274	0.317	5.6%
TLD [43]	0.926	0.965	65.3%
TLD [43] with SA	0.977	0.981	66.8%
DKF-SSIM	1.00	0.990	96.1%

coordinate of the SA in the current frame be equal to the coordinate of the centre of the same marker detected in the previous frame.

We assess the performance by assigning True Positive (TP) if the detected marker block overlaps no less than 40% of the corresponding GT, and assigning False Positive (FP) otherwise. Furthermore, we define that a Perfectly Detected Marker (PDM) is assigned if the detected marker block overlaps no less than 90% of the corresponding GT. Let Q be the total number of frames. Then, we define the following metrics:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{3.7}$$

$$\text{Recall} = \frac{\text{TP}}{Q}, \tag{3.8}$$

$$\text{Perfect Marker Rate (PMR)} = \frac{\text{total number of PDMs}}{Q}, \tag{3.9}$$

where Precision and Recall indicate time proportion a tracking algorithm tracks the targeted marker; PMR indicates the accuracy of detecting the centre coordinate of the marker block.

Tables 3.1 and 3.2 show the performance of the five tracking algorithms for bullseye marker tracking in 1 trial respectively. JCTH [42] cannot recover from tracking failure caused by the *object-on-object* problem. TLD [43] updates the marker model to help recover from the tracking failure, resulting in much higher scores than JCTH [42]. The proposed DKF-SSIM tracking-by-detection scheme is best suited for bullseye marker tracking due to its ability to incorporate dynamic and measurement models during tracking and combining the luminance, contrast, and structure features of the marker for detection. Since the position of the centre coordinate of the detected marker block has significant influence on the accuracy of the joint angle calculation, none of the four benchmark tracking methods are suited for autonomous joint angle calculation due to their resulting low PMR.

3.3.3 Validation against VICON

Since the gait kinematics abnormalities of stroke patients make marker tracking very challenging, we separately validate the results for the stroke patients and healthy volunteers. We group the measurements of all 40 trials for stroke patients (20 trials for healthy volunteers) together forming a vector U_x, $x \in \{\text{(P)roposed}, \text{(V)icon}\}$. We then calculate the R-squared value, max difference, and root mean square difference between U_P and U_V; we adopt a Bland-Altman plot between U_P and U_V, and calculate the mean difference, 95% confidence interval, and a linear fit, based on the constructed Bland-Altman plot shown in Figure 3.7. The Bland-Altman plot is a typical clinical measurement scheme to evaluate a new measurement system based on an established one. In our experiment, for each value $U_\mathrm{P}(i) \in U_\mathrm{P}$ and corresponding $U_\mathrm{V}(i) \in U_\mathrm{V}$, Bland-Altman plot is constructed by assigning $[U_\mathrm{P}(i) + U_\mathrm{V}(i)]/2$ as the abscissa value, and $U_\mathrm{P}(i) - U_\mathrm{V}(i)$ as the ordinate value. Tables 3.3 and 3.4 show the knee angle validation result based on both RGB and grayscale marker tracking schemes on stroke patients and healthy volunteers, respectively.

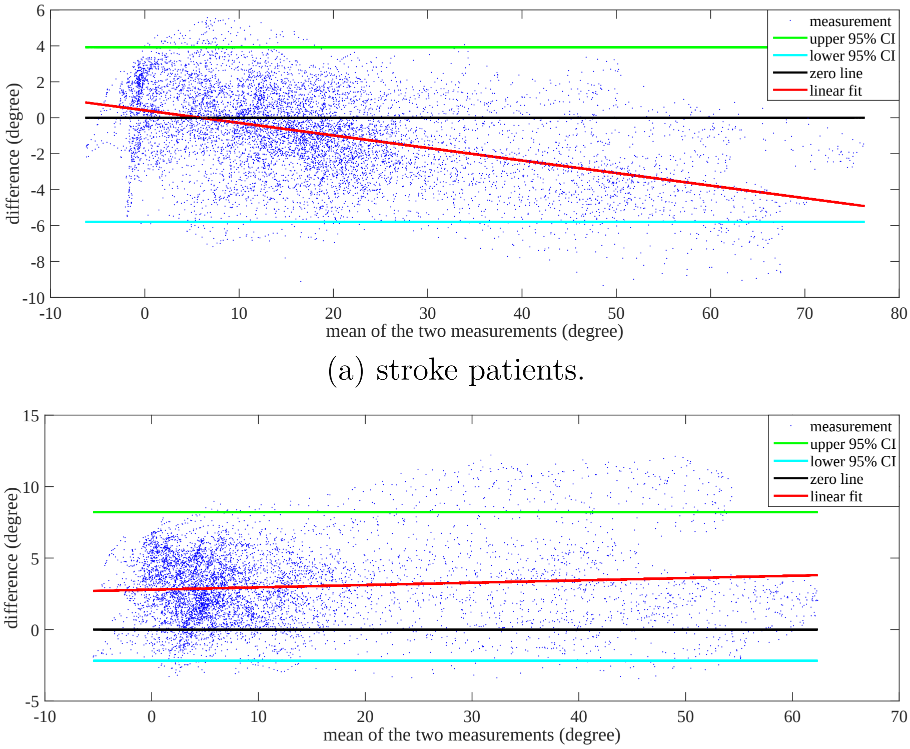
In general, RGB and grayscale schemes achieve similar accuracy, where the grayscale scheme performs almost 50% faster than the RGB scheme. In particular, with the grayscale scheme, the R-squared value is 0.982 for stroke patients and 0.971 for healthy volunteers, the maximum error is -9.34 degrees for stroke patients and 12.2 degrees for healthy volunteers, the root mean square error is less than 5 degrees for both groups; Bland-Altman plots show that the mean difference is less than 4 degrees for both groups, 95% confidence intervals are around 10 degrees where the interval is 0.7 degree smaller for stroke patients (9.72 degrees) compared to healthy volunteers (10.4 degrees), and linear fit is nearly horizontal with small intercepts for both groups.

3.3.4 Comparison of RGB and grayscale marker tracking

We compare the performance of the RGB and grayscale marker tracking using comparison metrics shown in Table 3.5. In particular, we find the best performing method (grayscale or RGB) by comparing the errors ϕ obtained by these two methods normalized by the error obtained by the grayscale method, shown in the second column of Table 3.5. If $\phi > 0$, we conclude that the RGB marker tracking performs better, except for the R-squared value where $\phi > 0$ means that the grayscale method is better (see column 3 in Table 3.5). The comparison results shown in Table 3.6 indicate that the grayscale scheme has significant processing speed advantage over the RGB scheme with negligible data accuracy loss.

3.3.5 Evaluation of the autonomous gait event detection

To evaluate our proposed autonomous gait event detection scheme in Sec. 3.2.5, we test 1) the sensitivity of the parameters (*i.e.*, scaling factors λ and ϵ) given different proportion of the training data, and 2) the classification accuracy for the six gait events. We do this by 1) performing a greedy search on the optimal sets of λ (with the range



(a) stroke patients.

(b) healthy volunteers.

Fig. 3.7 Bland-Altman plots for knee angle validation against VICON. CI: confidence interval.

[1, 30] and a 1 increment) and ϵ (with the same range [1, 30] and a 1 increment) using randomly selected training data (same proportion for both healthy volunteers and stroke patients), and 2) classifying the gait events using the testing data. We randomly split the training/testing data with the same proportion, repeat 100 times, and present the mode of both λ's and ϵ's in Fig. 3.8 and average classification accuracy of all 100 runs in Figs. 3.9 and 3.10. It is clear in Fig. 3.8 that λ and ϵ always converge at 3 and 1, respectively, *i.e.*, both are not sensitive to different training/testing splits. As shown in Fig. 3.9, for intra-subject evaluation in healthy volunteers, the average of all six-gait event detection accuracy is above 85%, where the gait event Foot Flat has the lowest accuracy that is above 70%. Furthermore, Fig. 3.10 shows that, for intra-subject evaluation in stroke patients, the average of all six-gait event detection accuracy is also

Table 3.3 Knee angle validation on stroke patients.

metric			RGB	grayscale
R-squared value			0.982	0.982
max error (deg)			-9.15	-9.34
root mean square error (deg)			2.64	2.65
Bland-Altman plot	mean difference (deg)		-0.937	-0.935
	95% confidence interval		(-5.78, 3.91)	(-5.80, 3.93)
	linear fit	slope	-0.0697	-0.0698
		intercept	0.407	0.411
average execution time (s)			1301	724

Table 3.4 Knee angle validation on healthy volunteers.

metric			RGB	grayscale
R-squared value			0.971	0.971
max error (deg)			12.2	12.2
root mean square error (deg)			3.99	4.01
Bland-Altman plot	mean difference (deg)		2.98	3.01
	95% confidence interval		(-2.23, 8.18)	(-2.18, 8.21)
	linear fit	slope	0.0149	0.0162
		intercept	2.78	2.79
average execution time (s)			1282	697

above 85%, where the gait event Initial Contact has the lowest accuracy that is above 58%.

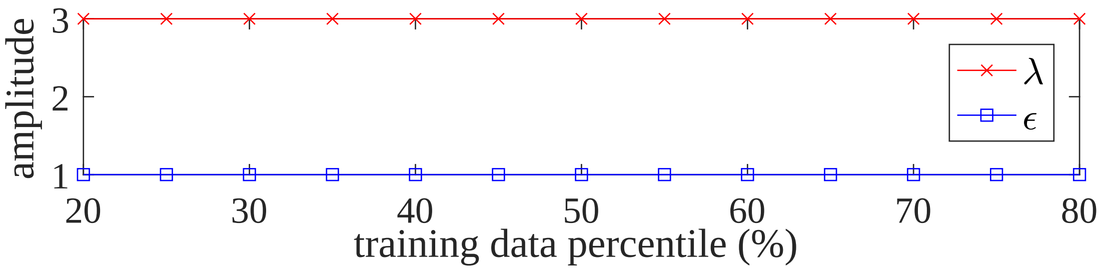


Fig. 3.8 Modes of both λ's and ϵ's for each 100 training/testing splits.

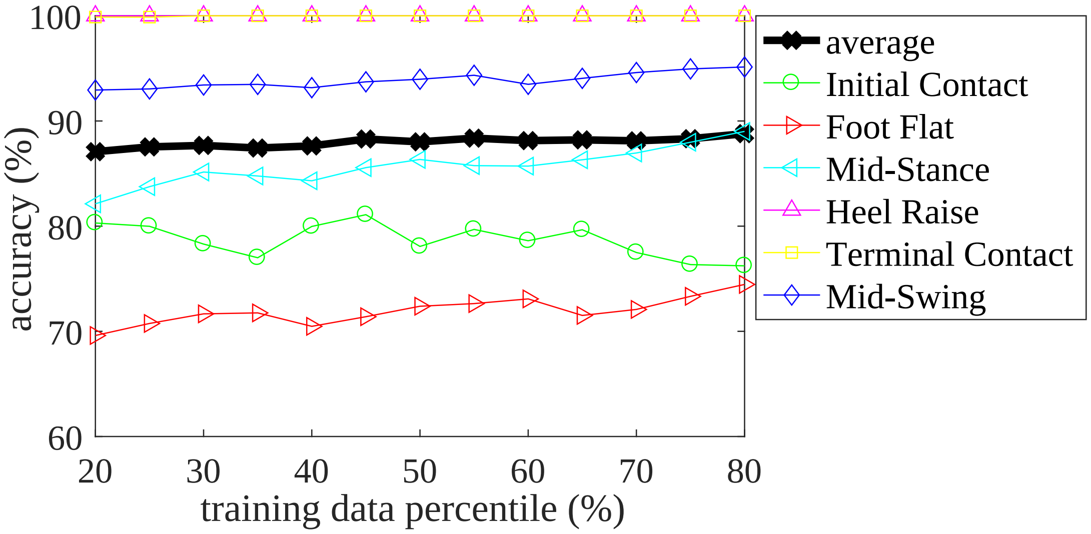


Fig. 3.9 Intra-subject gait event detection accuracy in healthy volunteers.

3.4 Discussion

In this section, we discuss the system operation speed, data accuracy, potential applications and improvements.

The average system operation time is 30 minutes for each participant, which includes 5 minutes for camera and tripod assembly, 2 minutes for adjustment of the camera height and distance to the participant, 2 minutes for marker attachment, 5 minutes for video recording, and the rest for data processing. The data processing module starts with the autonomous frames-of-interest detection, followed by marker-template selection via a mouse-click, grayscale marker tracking, knee angle calculation, autonomous gait event detection, and result visualization. The autonomous frame-of-interest detection scheme shows negligible FDR, *i.e.*, the proposed frame-of-interest detection method successfully recognizes both the first and last frames when all three markers are present for the marker tracking process – thus it can be used to replace manual labelling. The proposed autonomous gait event detection scheme detected each gait event / phase with high detection rate for most of the gait events by comparing with the ground truth frames, and thus the proposed gait event detection scheme can replace manual

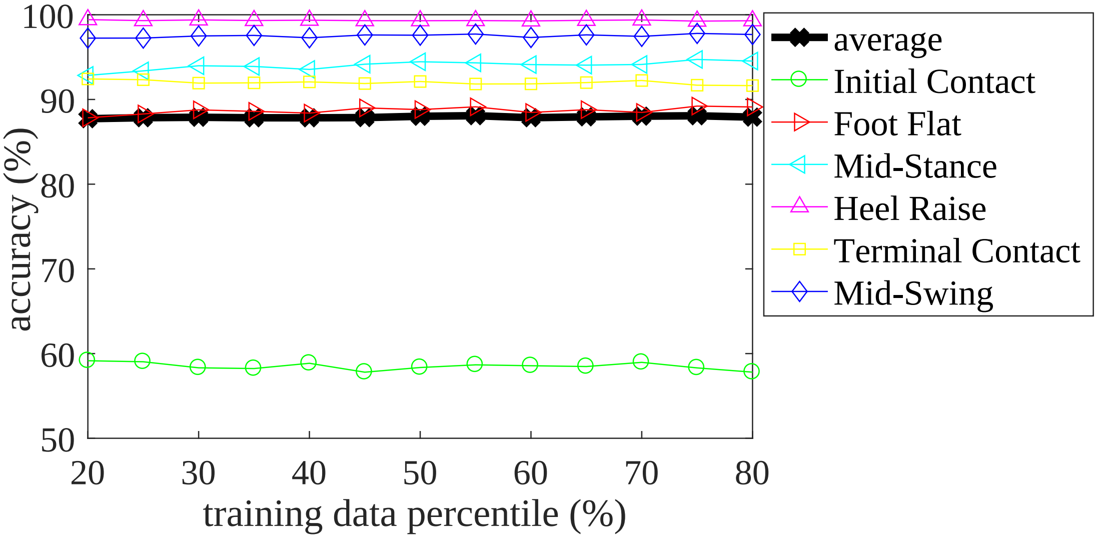


Fig. 3.10 Intra-subject gait event detection accuracy in stroke patients.

labeling effort that is performed in current medical studies. Given the fact that stroke patients are more likely to have transverse plane abnormalities, *e.g.*, internal rotation due to weak hip musculature or external rotation due to compensatory movements, experimental results indicate the system robustness to gait kinematics abnormalities for stroke patients and the potential for clinical gait kinematics analysis.

The proposed marker tracking scheme is more reliable than the state-of-the-art object tracking methods of [42] and [43], as shown in Section 3.3.2; the knee angle validation against VICON shows good agreement for all stroke patients and healthy volunteers. Thus the proposed system is robust to stochastic and sudden movements of stroke patients. The efficient performance of the grayscale marker tracking scheme indicates that it is sufficient to convert all RGB frames into grayscale, and then perform tracking and processing on grayscale frames.

There are two types of errors. The first type of error is caused by the deviation between the knee angle plane and the camera scene plane. The second type of error originates from the fundamental difference in defining the hip joint centre (HJC) between the gold standard VICON 3D system and our proposed 2D system. That

Table 3.5 Performance comparison scheme between RGB and grayscale marker tracking.

metric			comparison expression ϕ	winning scheme	
				$\phi > 0$	$\phi < 0$
RSV			$\phi_{\mathrm{RSV}} = \frac{\mathrm{RSV_G} - \mathrm{RSV_{RGB}}}{\mathrm{RSV_G}}$	G	RGB
MO			$\phi_{\mathrm{ME}} = \frac{\lvert \mathrm{ME_G} \rvert - \lvert \mathrm{ME_{RGB}} \rvert}{\lvert \mathrm{ME_G} \rvert}$	RGB	G
RMSE			$\phi_{\mathrm{RMSE}} = \frac{\lvert \mathrm{RMSE_G} \rvert - \lvert \mathrm{RMSE_{RGB}} \rvert}{\lvert \mathrm{RMSE_G} \rvert}$		
BA	MD		$\phi_{\mathrm{MD}} = \frac{\lvert \mathrm{MD_G} \rvert - \lvert \mathrm{MD_{RGB}} \rvert}{\lvert \mathrm{MD_G} \rvert}$		
	95% CI		$\phi_{\mathrm{UCI}} = \frac{(\mathrm{UCI_G} - \mathrm{LCI_G}) - (\mathrm{UCI_{RGB}} - \mathrm{LCI_{RGB}})}{\mathrm{UCI_G} - \mathrm{LCI_G}}$		
	LF	slope	$\phi_{\mathrm{S}} = \frac{\lvert \mathrm{S_G} \rvert - \lvert \mathrm{S_{RGB}} \rvert}{\lvert \mathrm{S_G} \rvert}$		
		I	$\phi_{\mathrm{I}} = \frac{\lvert \mathrm{I_G} \rvert - \lvert \mathrm{I_{RGB}} \rvert}{\lvert \mathrm{I_G} \rvert}$		
AET			$\phi_{\mathrm{ET}} = \frac{\lvert \mathrm{ET_G} \rvert - \lvert \mathrm{ET_{RGB}} \rvert}{\lvert \mathrm{ET_G} \rvert}$		

RSV: R-squared value; MO: max error; RMSE: root mean square error; BA: Bland-Altman plot; MD: mean difference; CI: confidence interval; LF: linear fit; I: intercept; AET: average execution time; UCI: upper 95% confidence interval; LCI: lower 95% confidence interval; G: grayscale.

is, the former uses the Harrington et al.'s hip regression equation [97] to calculate the HJC location for 3D kinematics [98], whereas the latter places the marker on the head of the greater trochanter for HJC calculation. Note that the Bland-Altman plot mean difference of stroke patients (Table 3.3) is approximately 2 degrees smaller in amplitude compared to healthy volunteers (Table 3.4), with a 95% confidence interval approximately 0.7 degree smaller. This is due to: (1) the knee range of motion in healthy volunteers being greater than stroke patients given the fact that stroke patients generally perform synergistic gait pattern during walking [99, 100] while healthy individuals perform selective joint movements [101]. This difference in knee range of motion occurs as a result of the deviation between the knee angle plane and the camera scene plane for healthy volunteers being larger than that for stroke patients; (2) both groups have small sample size, and the healthy group was half the size of the patient group. We stress that our proposed 2D system is much more cost-effective and less time consuming, albeit at the small cost of sacrificing a modest amount of accuracy compared to traditional optical motion analysis systems. We achieve a 95% confidence

Table 3.6 Result of performance comparison between RGB and grayscale marker tracking.

metric			stroke patients		healthy volunteers	
			scheme value	winner	scheme value	winner
RSV			-0.0204%	RGB	0.0206%	G
MO			2.06%	RGB	-0.302%	G
RMSE			0.272%	RGB	0.493%	RGB
BA	MD		-0.214%	G	1.07%	RGB
	95% CI		0.344%	RGB	-0.248%	G
	LF	slope	0.143%	RGB	8.02%	RGB
		I	1.02%	RGB	0.502%	RGB
AET			-80.1%	G	-83.9%	G

RSV: R-squared value; MO: max error; RMSE: root mean square error; BA: Bland-Altman plot; MD: mean difference; CI: confidence interval; LF: linear fit; I: intercept; AET: average execution time; G: grayscale.

interval of about 10 degrees for both groups compared to the clinically acceptable level of error for 3D kinematics, which is 5 degrees and considers both the intra- and inter-assessor variability [102].

Overall, the system is simple to assemble, highly adjustable for camera view, cost effective, and transportable for efficient gait analysis at local clinics and homes. In addition, the gait analysis result from the proposed system can be immediately sent to physiatrists for clinical consultation, indicating the potential to facilitate tele-rehabilitation [26–30]. These are in contrast to laboratory-based optical motion analysis systems that require large laboratory space, operational expertise, and have lots of pieces of equipment to assemble. Furthermore, our proposed system is more practical than recent single-camera approaches that require either substantial manual effort for joint angles [32, 34, 35], or specific video capturing background, ambient light, and clothing [31].

Note that, the proposed system is also capable of automatically measuring global segment orientations in the sagittal plane, *e.g.*, shank-to-vertical and thigh-to-vertical angles, with added bulls-eye markers at the femur and tibia, showing potential to facilitate ankle-foot orthosis fitting and tuning [103–105].

Our proposed system is more suitable for gait analysis in rehabilitation context, providing users with feedback on kinematic changes. Given the above reported knee angle errors, however, highly-accurate 3D optical motion analysis systems may still be needed to inform clinical decision making, *e.g.*, before orthopaedic surgery.

3.5 Summary

Emerging 2D single-camera systems are cost effective and highly portable with adequate fidelity of gait parameters compared to laboratory-based motion analysis systems. We propose a portable single-camera gait analysis system with autonomous frames-of-interest detection and grayscale marker tracking functionality. The proposed system is robust to the room background and study participant's clothing colours, autonomously tracks markers and calculates the knee angle in contrast to current video analysis software such as Pro-Trainer and Siliconcoach. Experimental results show that the proposed system can accurately detect the frames-of-interest, measure the knee angle and detect gait events with high average accuracy, provide objective visual feedback to patients and physiatrists, and thus offer an alternative, inexpensive and convenient solution for clinical gait analysis that can be used on the ward or in the community, and potential tele-rehabilitation.

Chapter 4

Upper Body Motion Analysis in Colour Videos[1]

4.1 Introduction

Building on our proposed gait analysis system in DKF-SSIM marker tracking in Chapter 3, in this chapter, we propose an alternative to state-of-the-art optical motion analysis systems such as VICON, cost-effective and portable, post-stroke recovery level assessment system via upper limb motion analysis, using a single camera. The system relies on detecting the markers attached to subject's pelvis, cervical spine, shoulder, elbow, and wrist (see Fig. 4.1(a)), tracking the markers frame by frame and autonomously computing joint angles (see Fig. 4.1(b)), data analytics for calculating relevant rehabilitation parameters, visualization, and robust classification leveraging on recent advances in signal processing on graphs. Experimental results on post-stroke recovery level classification show that the proposed decision support system has the potential to offer stroke patients and clinicians an alternative, affordable,

[1] This chapter is largely based on the work that appeared in 2014 IEEE International Conference on Image Processing [13] and IEEE Access [14].

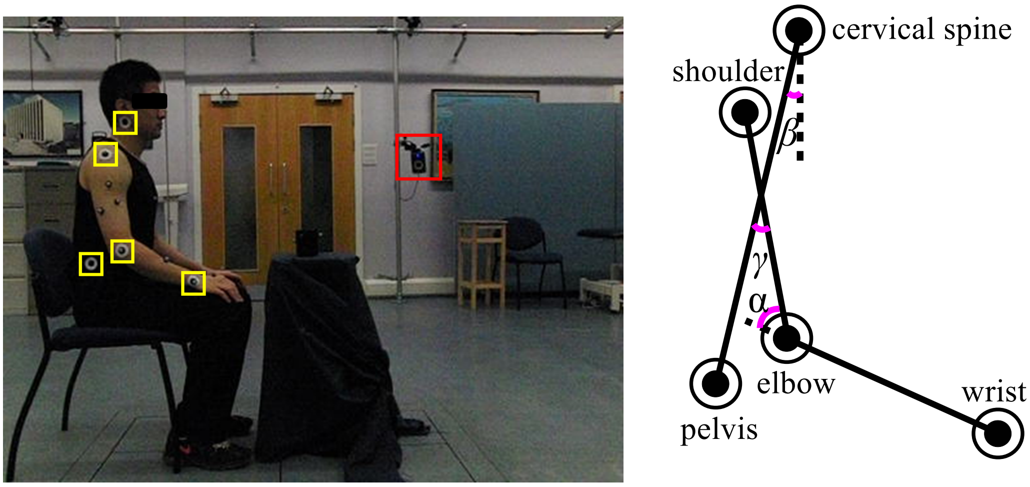
accurate and convenient impairment assessment option suitable for home healthcare and tele-rehabilitation.

We validate the proposed system with a standardized, multi-infrared-camera VICON system using a Bland-Altman plot [106], to evaluate the amount of agreement between the two systems. Experimental results show that the proposed system can capture upper limb motion patterns accurately, explicitly classify participants into a healthy group and different stroke groups with levels of impairment [51], provide visual and written feedback, and thus has potential to offer stroke patients and clinicians an alternative, affordable, accurate and convenient impairment assessment option.

In summary, the main contributions of this chapter are as follows:

1. Simultaneous body joint tracking in colour videos.
2. Novel multi-class and binary RGS classification methods for rehabilitation diagnostics.
3. Effective multimedia-based decision support tools for processing autonomously large RTG video datasets.
4. Overall plug-and-play cost-effective motion analysis system suitable for home use, including data capture, processing and visualisation blocks, tested on the patients and designed with the feedback from practitioners.

The remainder of this chapter is organized as follows. In the next section we discuss each component of the proposed system. In Section 4.3, we present the experimental results — tracking performance comparison with [42], [43], and [44, 107], angle accuracy validation with state-of-the-art motion analysis system VICON, and subject classification using RGS. We summarize this chapter in Section 4.4.



(a) Camera scene (b) Angles of interest

Fig. 4.1 Experimental setup for upper limb motion analysis.

4.2 Proposed upper limb motion analysis system

The aim of the proposed system is to autonomously assess the upper limb motor condition of the subject by accurately and simultaneously tracking the multiple bulls-eye markers adhered to the joints and provide visual and written feedback to stroke patients and clinicians.

Impairment of the upper limb following a stroke can be assessed in a number of ways [51], by measuring physical attributes such as range of motion, strength and co-ordination or more commonly by quantitatively assessing the ability to carry out a functional task such as the RTG movement [108], shown in Fig. 4.1(a), where the subject picks up a cup from the desk, carries it towards the mouth and puts it back on the desk. Three joint angles can be analysed during this activity, namely, (i) elbow movement defined by a supplementary angle to the shoulder-elbow-wrist angle denoted by α shown in Fig. 4.1(b); (ii) trunk-tilt defined by the pelvis-cervical spine-vertical angle β; and (iii) shoulder movement defined by an angle γ at the intersection of pelvis-cervical spine and shoulder-elbow lines.

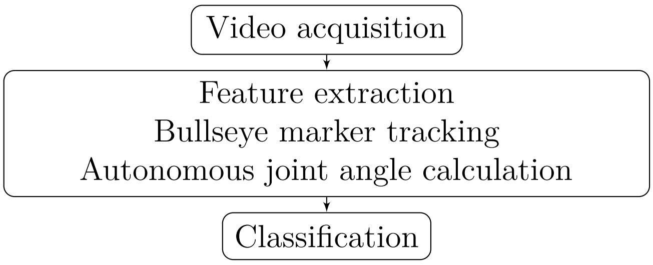


Fig. 4.2 Unit blocks of the proposed upper limb decision support system.

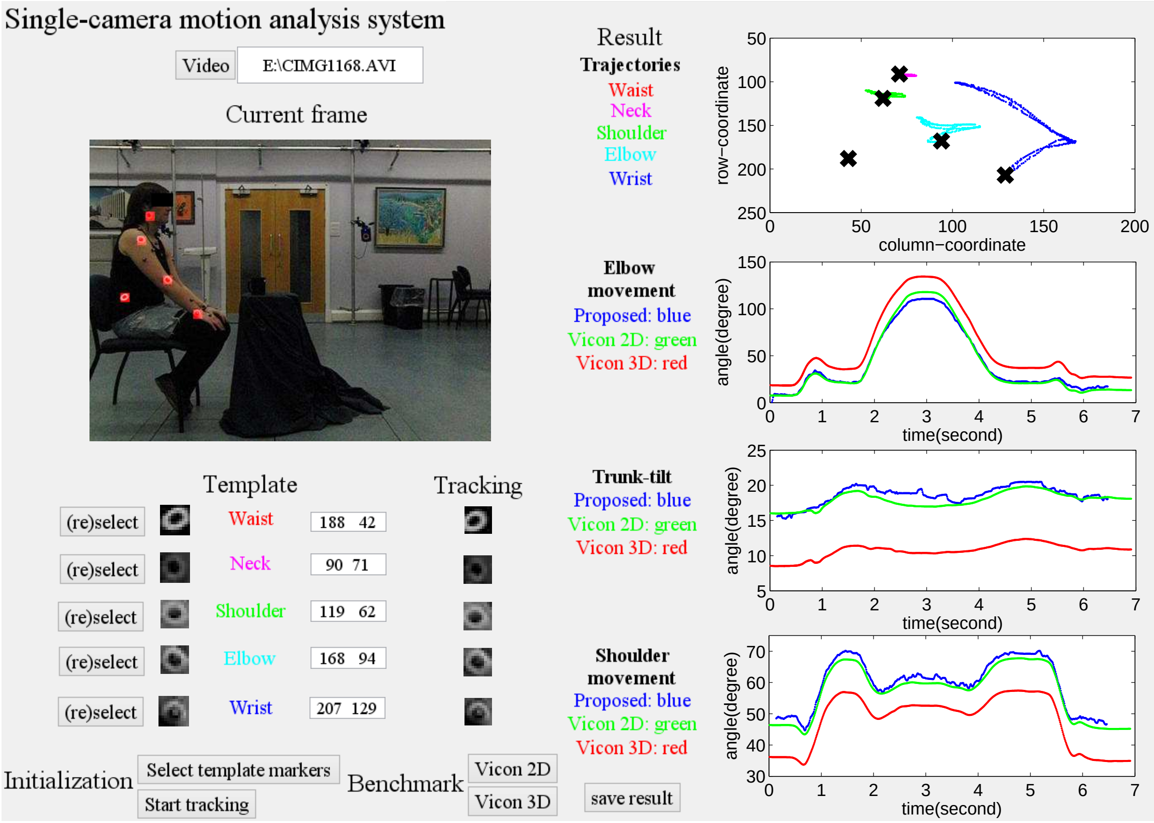
To calculate the relevant joint angles, we track, through the captured frames, five bulls-eye markers adhered to the skin overlying anatomical landmarks of the pelvis, cervical spine, shoulder, elbow, and wrist of the participant, highlighted by yellow squares in Fig. 4.1(a). The tracked motion patterns are then used to calculate the three angles in each frame, which are subsequently used for classification.

The main components of the proposed human upper limb motion analysis procedure to be described next are shown in Fig. 4.2.

4.2.1 Simultaneous multiple marker tracking

In the following, we describe the proposed object tracking method for the RTG dataset. First, as in Chapter 3, the centre coordinates of all bullseye marker templates are selected via mouse-click on our developed user interface in Frame 1 (see Fig. 4.3). Unlike individual marker tracking in Sec. 3.2.4, all markers are now tracked *simultaneously* using a discrete Kalman filter (DKF) [8, 9]. First the position and size of a rectangular Search Area (SA) for each marker is set in each frame based on the output of DKF. Then, for each marker, block matching is performed within the SA using structural-similarity (SSIM) [10] to identify a block most similar to the marker template.

The dynamic model in Frame i for tracking five markers (pelvis, cervical spine, shoulder, elbow, or wrist marker) simultaneously is given by $\hat{\mathbf{s}}_{si} = [\hat{\mathbf{s}}_i^1\ \hat{\mathbf{s}}_i^2\ \hat{\mathbf{s}}_i^3\ \hat{\mathbf{s}}_i^4\ \hat{\mathbf{s}}_i^5]^\top$. Similarly, the observation model is given by $\hat{\mathbf{u}}_{si} = [\hat{\mathbf{u}}_i^1\ \hat{\mathbf{u}}_i^2\ \hat{\mathbf{u}}_i^3\ \hat{\mathbf{u}}_i^4\ \hat{\mathbf{u}}_i^5]^\top$. $\mathbf{R}_s =$



Single-camera motion analysis system

Video: E:\CIMG1168.AVI

Current frame

	Template		Tracking
(re)select	Waist	188 42	
(re)select	Neck	90 71	
(re)select	Shoulder	119 62	
(re)select	Elbow	168 94	
(re)select	Wrist	207 129	

Fig. 4.3 GUI for the proposed upper limb motion analysis system.

$diag(\mathbf{R}^1, \mathbf{R}^2, \mathbf{R}^3, \mathbf{R}^4, \mathbf{R}^5)$ is the state transition matrix with $\mathbf{R}^j = \mathbf{R}$ for Marker j, $\mathbf{Q}_s = diag(\mathbf{Q}^1, \mathbf{Q}^2, \mathbf{Q}^3, \mathbf{Q}^4, \mathbf{Q}^5)$ is the observation matrix which translates $\hat{\mathbf{s}}_{si}$ to $\hat{\mathbf{u}}_{si}$, with $\mathbf{Q}^j = \mathbf{Q}$.

We use the same DKF initialization setup and occlusion handling scheme as in Sec. 3.2.4 to obtain the marker trajectories.

We note from the captured videos that only the waist marker can sometimes (rarely) be occluded, in which case we perform the same occlusion handling procedure as in Algorithm 2, Section 3.2.4. For the upper limb motion analysis, the centre coordinates of pelvis, cervical spine, shoulder, elbow, and wrist markers obtained by marker tracking are next used for visualization and autonomous joint angle calculation.

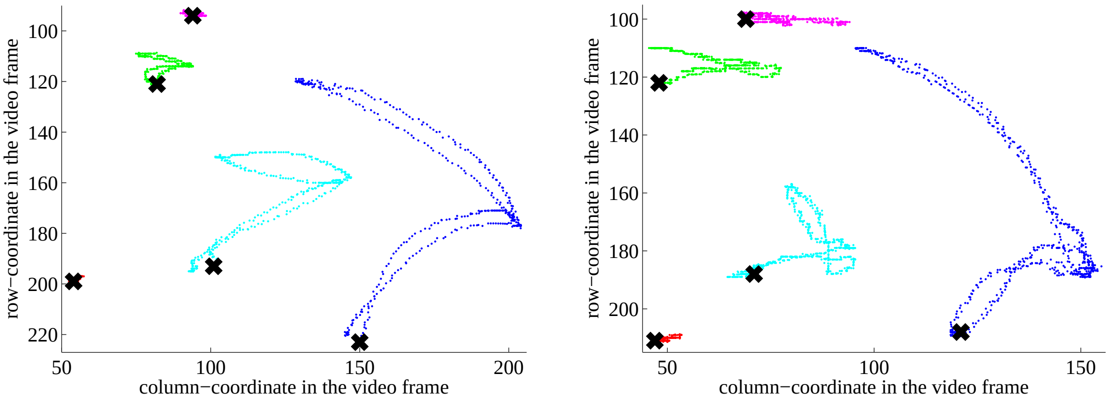
4.2.2 Autonomous joint angle calculation and visualization

During the tracking process, three joint angles - elbow movement α, trunk-tilt β, and shoulder movement γ, are, automatically and in real time, calculated on a frame-by-frame basis according to the centre coordinates of the detected markers. We record the marker trajectories by mapping the centre coordinates of all detected markers into a single frame. By working with practitioners and taking their feedback, we design a user interface in order to visualize all marker trajectories, and joint angles and check accuracy w.r.t benchmarks, as shown in Fig. 4.3. Via the interface, one can choose the video to be processed, and select (reselect if needed) the marker templates by mouse-click on the video frame shown in the "Current frame" panel. The "Template" panel then displays the appearance and centre coordinates of the marker templates. The marker tracking process begins by clicking "Start tracking", followed by showing appearance of the detected marker blocks in the "Tracking" panel and marker trajectories and joint angles, where VICON 3D is the original tracking result from the VICON system and VICON 2D projects the 3D result to one of the three orthogonal VICON system planes that is closely parallel to the plane of camera scene [11] in the "Result" panel.

Fig. 4.4 shows the marker trajectories of one trial from a healthy subject and one from a stroke patient. The corresponding joint angles for these examples shown in Fig. 4.5 indicate that the joint angle plots of the proposed method closely follow those of the benchmarks VICON 2D and 3D.

4.2.3 Subject classification

The aim of subject classification is to explicitly classify all participants into a healthy group and a patient group (binary classification) or a healthy group and several stroke groups with different levels of impairment [51] using the variations of the three tracked joint angles. Building on the principles of RGS [55, 58], we attempt to solve these



(a) healthy subject

(b) stroke patient

Fig. 4.4 Marker trajectories.

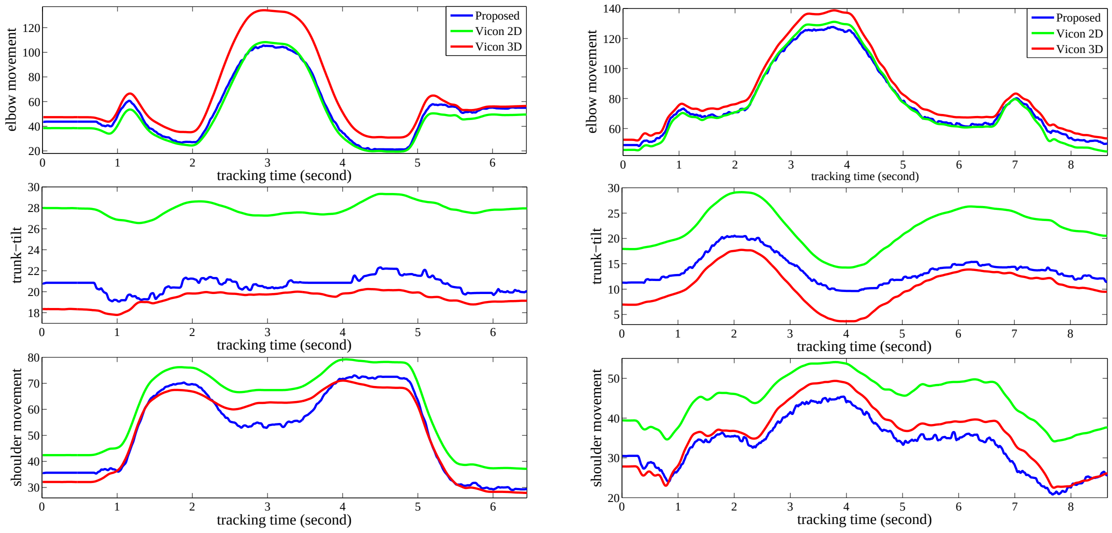
binary and multi-class classification problems. Since binary classification is a special case of the multi-class classification, in the following we describe only the proposed multi-class classification schemes.

As classification features we use the standard deviation as a heuristic hand-crafted feature that is able to quantify the variation of a joint angle during one trial.

We propose three RGS multi-class classification methods: “one-against-one” (OAO-RGS) — classify two classes at a time and next use the voting strategy, suggested in [109], to designate the final class for each sample, “one-against-all” (OAA-RGS) — consider one class at a time and group the other classes into a single class, and “once-for-all” (OFA-RGS) — classify all classes at once.

For OAO-RGS, we first design $f(f-1)/2$ binary classifiers, where $f > 2$ is the number of classes. Each classifier is trained using data from two of the f classes. In particular, given a set of data from Classes a and b:

$$\{\mathbf{x}_i^{ab}, y_i\}, y_i \in \{+1, 0, -1\}, \mathbf{x}_i^{ab} \in \mathbb{R}^V, i = 1, \ldots, D, \tag{4.1}$$



(a) healthy subject

(b) stroke patient

Fig. 4.5 Automatically calculated joint angles (in degrees). Top row: elbow movement α; middle row: trunk-tilt β; bottom row: shoulder movement γ.

where all data elements with known labels construct the set of two-class training data:

$$\{\mathbf{x}_i^{ab}, y_i\}, y_i \in \{+1, -1\}, \mathbf{x}_i^{ab} \in \mathbb{R}^V, i = 1, \ldots, N, N < D, \tag{4.2}$$

where D and N are the total number of samples and the number of training samples, respectively. For the classifier on data from Classes a and b, we define a connected, undirected, and weighted graph $G^{ab} = (\mathcal{X}^{ab},\zeta^{ab},\mathbf{J}^{ab})$, where $\mathcal{X}^{ab} = \{\mathcal{X}_1^{ab}, ..., \mathcal{X}_D^{ab}\}$ is a set of vertices corresponding to dataset $\mathbf{x}^{ab} = \{\mathbf{x}_i^{ab}, \ldots, \mathbf{x}_D^{ab}\}$, ζ^{ab} denotes a set of edges, and $\mathbf{J}^{ab}$ denotes a weighted adjacency matrix. In particular, the weight $\mathbf{J}_{i,j}^{ab}$ on edge $\zeta_{i,j}^{ab}$ indicates the graph similarity of vertices $\mathcal{X}_i^{ab}$ and $\mathcal{X}_j^{ab}$, and is modulated by a Gaussian kernel [52]:

$$\mathbf{J}_{i,j}^{ab} = \begin{cases} \exp\Big(- \frac{\left\|\mathbf{x}_i^{ab} - \mathbf{x}_j^{ab}\right\|_2^2}{2\theta^2}\Big) & \text{if } \left\|\mathbf{x}_i^{ab} - \mathbf{x}_j^{ab}\right\|_2^2 \leq \tau, \\ 0 & \text{otherwise}, \end{cases} \tag{4.3}$$

where θ denotes the Gaussian standard deviation, and τ is a threshold on the squared Euclidean distance of two vertices $\mathcal{X}_i^{ab}$ and $\mathcal{X}_j^{ab}$. Furthermore, we define a mapping of the graph G^{ab} as follows:

$$\mathbf{h}^{ab} : \mathcal{X}^{ab} \rightarrow \mathbb{R}, \mathcal{X}^{ab}_n \mapsto h^{ab}_n, \tag{4.4}$$

or

$$\mathbf{h}^{ab} = (h^{ab}_1, \ldots, h^{ab}_D)^\top \in \mathbb{R}^D, \tag{4.5}$$

where h^{ab}_i corresponds to vertex $\mathcal{X}^{ab}_i$ and data element $\mathbf{x}^{ab}_i$, and is given by: $h^{ab}_i = 1$ if $\mathcal{X}^{ab}_i$ belongs to Class a, -1 if $\mathcal{X}^{ab}_i$ belongs to Class b, and 0 if class is unknown.

Next, as in [55], we use the total variation on a graph (TV_G) to measure the total variation of G^{ab}:

$$\mathrm{TV}_{G^{ab}}(\mathbf{h}^{ab}) = \frac{1}{\|\mathbf{h}^{ab}\|_2^2} \left\| \mathbf{h}^{ab} - \frac{1}{|\eta^{ab}_{max}|} \mathbf{J}^{ab}\mathbf{h}^{ab} \right\|_2^2 \tag{4.6}$$

where the product $\tilde{\mathbf{h}}^{ab} = \mathbf{J}^{ab}\mathbf{h}^{ab}$ is the output of the *graph shift* [55], a nontrivial graph filter; η^{ab}_{max} is an eigenvalue of $\mathbf{J}^{ab}$ that has the largest amplitude with constraint $|\eta^{ab}_{max}| \geq |\eta^{ab}_i|, 1 \leq i \leq D$. The objective of the classification on $\mathrm{TV}_{G^{ab}}$ is to update all unknown labels within $\mathbf{h}^{ab}$ while fixing the labels of all training data samples to get the lowest total variation on a graph [58], that is, a minimum $\mathrm{TV}_{G^{ab}}(\mathbf{h}^{ab})$:

$$\mathbf{h}^{ab\prime} = \arg \min_{\mathbf{h}^{ab} \in \mathbb{R}^D} \mathrm{TV}_{G^{ab}}(\mathbf{h}^{ab}). \tag{4.7}$$

We apply the above OAO-RGS classification procedure using all $f(f-1)/2$ binary RGS-based classifiers and use the voting strategy of [109] to designate classes. In particular, if a data sample has the same number of votes for two or more classes, the class that firstly reached the maximum number of votes is designated as this sample's class.

For OAA-RGS, we design f binary classifiers. Each classifier is for data from one of the f classes and the group of remaining $f-1$ classes. In particular, we follow the procedure on graph construction as above, and define a graph G^{all} for data from all f classes. We then defined f different $\mathbf{h}$'s, *i.e.*, f different mappings of the same graph G^{all}, for data from each of the f classes, and minimize each corresponding $\mathrm{TV}_{G^{all}}(\mathbf{h})$ to designate the class labels for each set of testing samples.

For OFA-RGS, we adopt the same graph G^{all} as used in OAA-RGS. Instead of using the binary mapping $\mathbf{h}^{ab}$, we define a multi-class graph mapping $\mathbf{h}^{all}$ for G^{all} (see Section 4.3.3). We then minimize the total variation on G^{all}, that is, to get a minimum $\mathrm{TV}_{G^{all}}(\mathbf{h}^{all})$ and designate the class labels.

We discuss the multi-class classification process on the targeted upper limb motion analysis, and evaluate the performance of above three RGS methods, in Section 4.3.3.

4.3 Experimental results

In this section, we report the following experimental results:

- Comparison of bullseye marker tracking performance of the proposed DKF-SSIM tracking with four benchmark tracking methods JCTH [42], TLD [43], STR [44], and DKF-SSIM without the SA update (DKF-SSIM WSA).
- Separate validation of the proposed system with VICON 2D and VICON 3D (see Section 4.2.2) for the group of healthy subjects and the group of stroke patients since the stochastic movements of the stroke patients make tracking more challenging.
- Evaluation of binary and OAO-, OAA- and OFA-RGS multi-class classification methods (Section 4.2.3) for classifying all subjects into healthy and stroke groups.

Each video is captured using the same digital camera as in Chapter 3 with 360×480 resolution. We adapt the camera calibration method from [89], where the coefficients of the radial distortion are obtained by solving a nonlinear minimization problem with the Levenberg-Marquardt Algorithm [110], to correct lens distortion of the acquired video frames before marker tracking. For benchmarking and validation, as in Chapter 3, we simultaneously capture video with VICON (100fps), that is recognised as the state-of-the-art [111] and commonly used in clinical rehabilitation practice. Fig. 4.1(a) shows a sample frame, where one out of the 12 VICON infrared cameras is highlighted by a red square.

The proposed system is validated on 10 participants, including 5 healthy subjects and 5 stroke patients. Each of the 10 participants performed 5 RTG trials, *i.e.*, a total of 50 video clips are used, with a frame rate of 100fps for fair comparison with VICON. As in Section 3.3, the size of each marker template is always $q \times q = 11 \times 11$ pixels, which was heuristically found for optimal appearance representation of each marker that results in best tracking accuracy without sacrificing much computational cost.

4.3.1 Comparison with state-of-the-art

Similar to Section 3.3.2, in this section, we aim to compare our proposed DKF-SSIM marker tracking scheme with state-of-the-art, while showing the necessity of using SA constraint (see Section 4.2.1) in our DKF-SSIM scheme. We first randomly choose 1 of 5 trials for each participant, and select bullseye marker templates from the first frame of the corresponding video clip. Next, for each marker, we manually label the marker blocks in all frames of the video clip, with the same size as the marker template, as the ground truth (GT) to assess the bullseye marker tracking performance of all five methods: JCTH [42], TLD [43], STR [44], DKF-SSIM without the SA update (DKF-SSIM WSA), and DKF-SSIM. In the DKF-SSIM WSA approach, for each marker,

Table 4.1 Bullseye marker tracking on healthy subjects.

Method	Precision	Recall	PMR
JCTH [42]	0.581	0.581	27.5%
TLD [43]	0.958	0.922	64.6%
STR [44]	0.974	0.974	80.3%
DKF-SSIM WSA	0.852	0.852	82.8%
DKF-SSIM	0.998	0.998	97.3%

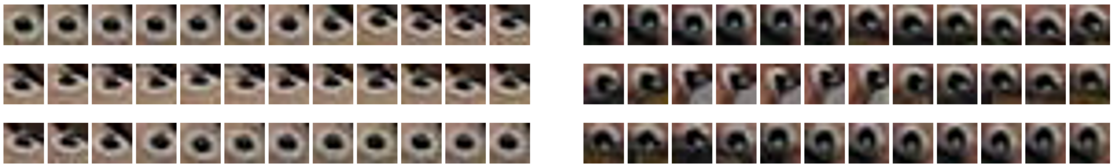
Table 4.2 Bullseye marker tracking on stroke patients.

Method	Precision	Recall	PMR
JCTH [42]	0.507	0.507	16.9%
TLD [43]	0.913	0.894	52.4%
STR [44]	0.955	0.955	81.7%
DKF-SSIM WSA	0.781	0.781	75.2%
DKF-SSIM	0.980	0.980	94.6%

we fix the size of SA at $\lfloor 1.4q \rfloor \times \lfloor 1.4q \rfloor$ and let the centre coordinate of the SA in the current frame be equal to the coordinate of the centre of the same marker detected in the previous frame.

We assess the performance by using the same metrics as in Section 3.3.2. Tables 4.1 and 4.2 show the performance of the five tracking algorithms for bullseye marker tracking on healthy subjects and stroke patients, respectively. Similar to Chapter 3, JCTH [42] cannot recover from tracking failure caused by the *object-on-object* problem. TLD [43] updates the marker model to help recover from the tracking failure, resulting in much higher scores than JCTH [42]. STR [44] outperforms TLD [43], but still cannot get the marker centre accurately during out-of-plane rotation which commonly occurs when performing the RTG movement (see Fig. 4.6 for an illustration of the hand-labelled groundtruth shoulder and wrist markers over one trial).

The results also show that the SA update in each frame brings a 15-20% improvement in PMR, at the cost of a higher tracking complexity. Indeed, the average tracking and processing time per frame was 35msec and 43msec, for DKF-SSIM WSA and the



(a) shoulder marker (b) wrist marker

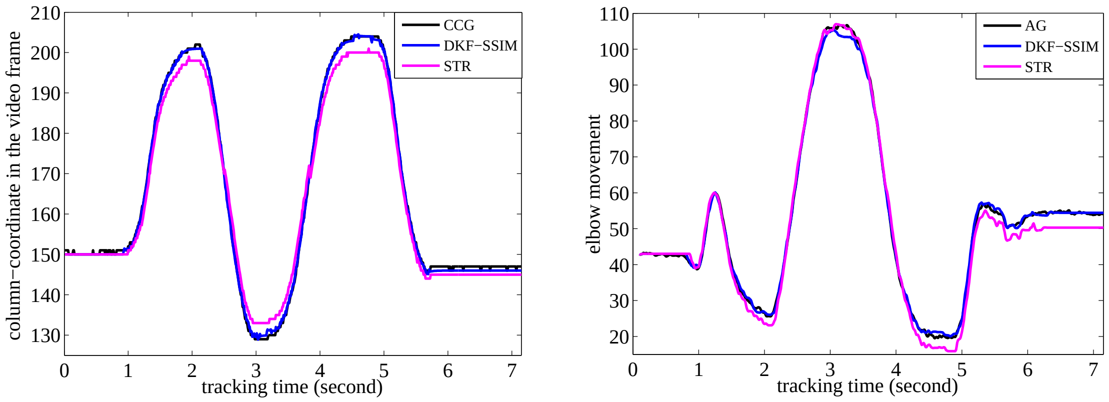
Fig. 4.6 Hand-labelled groundtruth shoulder and wrist markers.

proposed DKF-SSIM, respectively, measured in MATLAB R2014b on a laptop running Windows 8.1, with Core i7 2820QM 2.3GHz processor and 16GB RAM.

The proposed DKF-SSIM tracking-by-detection scheme is best suited for bullseye marker tracking due to its ability to incorporate dynamic and measurement models during tracking and combining the luminance, contrast, and structure features of the marker for detection. Since the position of the centre coordinate of the detected marker block has significant influence on the accuracy of the joint angle calculation, none of the four benchmark tracking methods are suited for autonomous joint angle calculation due to their resulting low PMR. To further demonstrate this, we show the tracking performance of the proposed DKF-SSIM and STR [44], the best benchmarking scheme among JCTH [42], TLD [43] and STR [44] according to Tables 4.1 and 4.2, on one trial of a healthy subject in Fig. 4.7, where Fig. 4.7(a) shows the column-coordinate of the wrist marker given the benchmarking hand-labelled column-coordinate groundtruth, and Fig. 4.7(b) shows the corresponding elbow movement angle (degree) given the benchmarking angle groundtruth calculated from the hand-labelled groundtruth shoulder, elbow, and wrist markers. The corresponding error is shown in Table 4.3.

Table 4.3 Tracking error in Fig. 4.7. CC=column-coordinate.

	wrist marker CC (pixel)		elbow movement (degree)	
	mean error	max error	mean error	max error
STR [44]	2.60	5.00	2.13	5.43
DKF-SSIM	0.567	2.42	0.735	3.33



(a) wrist marker CC. (b) elbow movement (degree).

Fig. 4.7 Illustration of the tracking performance of the proposed DKF-SSIM and STR. CC=column-coordinate. CCG=column-coordinate groundtruth. AG=angle groundtruth.

4.3.2 Angle accuracy validation

We validate the proposed DKF-SSIM tracking with VICON 2D and 3D using Bland-Altman plot [106] (see Section 3.3.3) for evaluation of the limits of agreement. The dataset used contains 25 trials from healthy subjects and another 25 trials from stroke patients. We group all 25-trial results of healthy subjects (stroke patients) together forming three vectors $\mathbf{F}_{\mathrm{X}}^{\alpha}$, $\mathbf{F}_{\mathrm{X}}^{\beta}$, and $\mathbf{F}_{\mathrm{X}}^{\gamma}$, where $\mathrm{X} = \{\mathrm{P}, \mathrm{V2}, \mathrm{V3}\}$, denotes (P)roposed, VICON 2D (V2) or 3D (V3). We calculate the mean difference (MD) and the standard deviation of $\mathbf{F}_{\mathrm{P}}$ and $\mathbf{F}_{\mathrm{V2}}$, and $\mathbf{F}_{\mathrm{P}}$ and $\mathbf{F}_{\mathrm{V3}}$, followed by lower and upper 95% confidence interval (LCI, UCI) and a linear fit, all of which are based on the constructed Bland-Altman plot, for complete limits of agreement evaluation.

Figs 4.8 and 4.9 show the Bland-Altman plots based on above construction process for the healthy subjects and stroke patients, respectively. Table 4.4 shows the corresponding limits of agreement (LOA). Note that good LOA is indicated by small MD, narrow 95% CI, and a linear fit that is close to zero [106]. Since the deviation between the elbow movement α plane and camera scene plane (CSP) is more notable than that between the trunk-tilt β plane and CSP and that between the shoulder movement γ

plane and CSP, validation of P and V3 on α shows a relatively large MD and wide 95% CI. Otherwise, P and V3 show good LOA on β and γ; P and V2 show good LOA for all motion patterns. In general, 3D information is needed in diagnostic systems. However, the above validation incorporates loss of 3D information, indicating that 2D suffices for the targeted RTG sagittal movement analysis. This is in accordance to the prior literature [32].

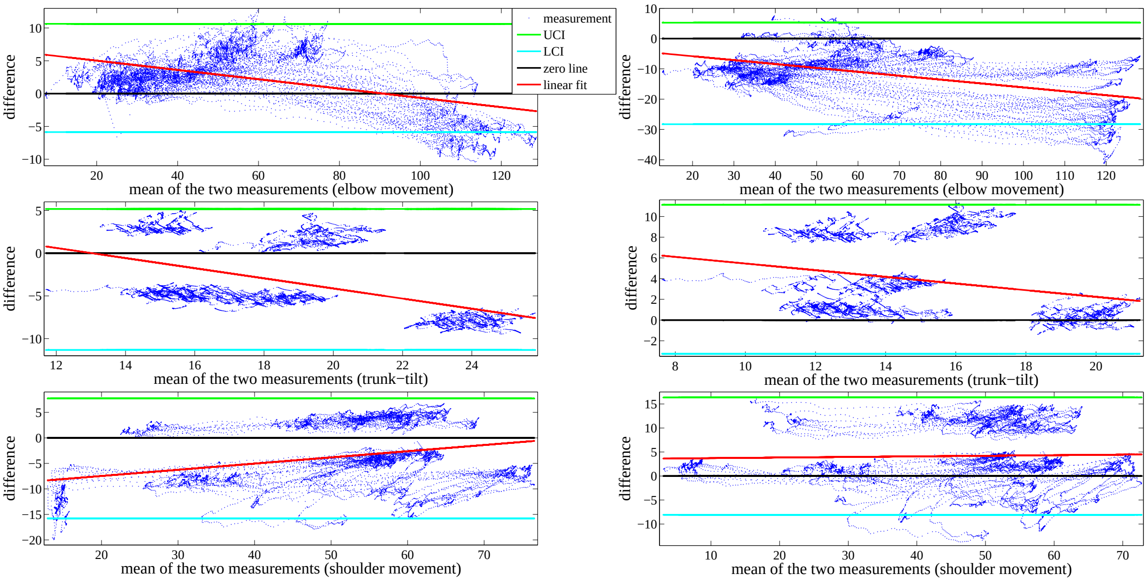


Fig. 4.8 Bland-Altman plots (in degrees) of all healthy subjects. Left column: P vs. V2. Right column: P vs V3. Top row: elbow movement α; middle row: trunk-tilt β; bottom row: shoulder movement γ.

Table 4.4 Limits of agreement (in degrees) between P and V2, and between P and V3 for all participants.

		Healthy subjects			Stroke patients		
		MD	LCI	UCI	MD	LCI	UCI
P vs V2	α	2.38	-5.86	10.6	7.72	-3.51	19.0
	β	-3.08	-11.3	5.16	-1.68	-10.8	7.39
	γ	-4.02	-15.8	7.72	-7.26	-20.9	6.37
P vs V3	α	**-11.5**	**-28.3**	**5.24**	**-18.8**	**-49.8**	**12.3**
	β	3.93	-3.24	11.1	7.01	0.22	13.8
	γ	4.13	-8.07	16.4	4.22	-11.6	20.1

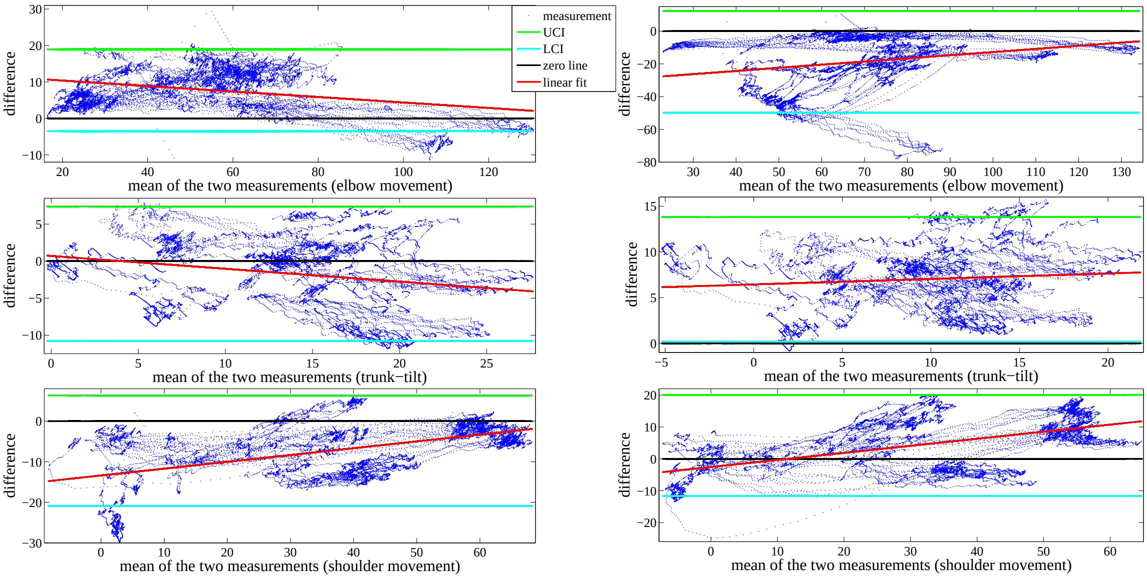


Fig. 4.9 Bland-Altman plots (in degrees) of all stroke patients.

4.3.3 Subject classification

As classification features we use the standard deviation of all three joint angles over one trial. That is, each data sample $(\sigma_{\alpha_i}, \sigma_{\beta_i}, \sigma_{\gamma_i})$ is a 3-dimensional feature vector that contains standard deviations of the joint angles α, β, and γ, where σ_{α_i}, σ_{β_i}, and σ_{γ_i} are the standard deviations during one trial of angles α, β, and γ, respectively. We evaluate the performance of the classification algorithms under different sizes of the training and testing data by using following metric:

$$\text{Classification Accuracy} = \frac{\text{Number of correctly classified samples}}{\text{Number of testing samples}}. \tag{4.8}$$

We clarify that the evaluation is intra-patient, due to the fact that: 1) only 5 health subjects and 5 stroke patients were recruited for the experimentation, 2) there are five classes for this classification task, namely, healthy, stroke recovery level 1, 2, 4, and 5, and at least one patient is at one of the four stroke recovery levels.

First, we perform binary classification, whose task is to group all subjects into two groups: healthy and stroke patients. We compare the proposed RGS binary classifier

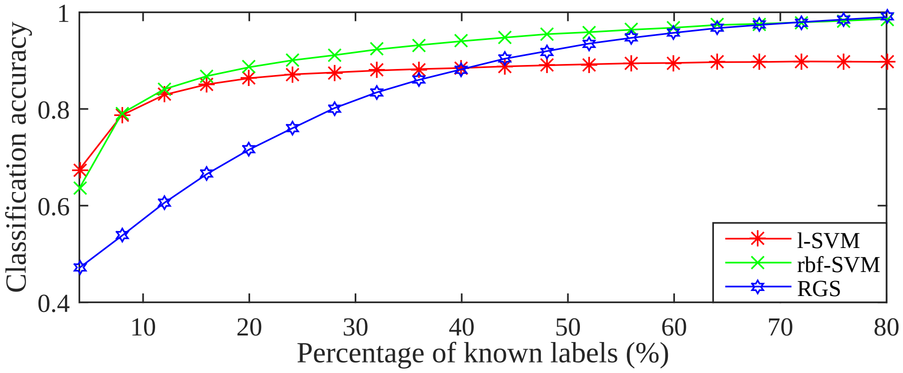


Fig. 4.10 Binary classification accuracy of testing data.

to that of linear and non-linear (we use a Gaussian Radial Basis Function (rbf) kernel with scaling factor $\rho = 1$) SVM binary classifiers, denoted as l-SVM and rbf-SVM, respectively. The results are given in Fig. 4.10 expressed as Classification Accuracy. In particular, we assume that between 4% and 80% of randomly selected labels are known for training, perform 10,000 tests, and then get the averaged result. It can be seen that RGS shows competitive performance with l-SVM when the percentage of known labels is above 40% at lower complexity.

Next, we turn to the multi-class classification, whose task is to classify further patients into different recovery levels. Table 4.5 shows the levels of upper limb impairment for 5 stroke participants, reported from a recruited rater, a biomechanics researcher with over ten years of experience in biomechanics data analysis, by observational assessment [51, 112]. Thus, we define $f = 5$ classes for all experimental data: Healthy, Stroke with ordinal scale 1 (OS 1), OS 2, OS 4, and OS 5, denoted as Class q, $q = 1, ..., 5$, respectively.

Table 4.5 Levels of impairment of stroke patients.

Stroke patient	SP 1	SP 2	SP 3	SP 4	SP 5
Ordinal scale	2	5	1	4	2

For OAO-RGS, we design $f(f-1)/2$ binary classifiers. For each classifier, we first define a graph for data from two of the f classes: a connected, undirected, and weighted graph $G = (\mathcal{X}, \zeta, \mathbf{J})$, with vertices $\mathcal{X} = \{\mathcal{X}_1, \ldots, \mathcal{X}_D\}$ correspond to the dataset $\mathbf{x} = \{\mathbf{x}_1, \ldots, \mathbf{x}_D\}$, edges ζ, and a weighted adjacency matrix $\mathbf{J}$ defined using (4.3), with $\theta = 1$ and $\tau = 100$ which balances the number of non-zero entries in $\mathbf{J}$ and computation time, where $\mathbf{x}_i = (\sigma_{\alpha_i}, \sigma_{\beta_i}, \sigma_{\gamma_i})$. Next, we define $\mathbf{h}$, *i.e.*, the mapping of the graph G, and minimized TV_G of G as defined in (4.6). Finally, we use the voting strategy [109] to designate groups for all testing data.

For OAA-RGS, we design f binary classifiers. For each classifier, we first define a graph G^{all} for data from all f classes with the same parameter setting $\theta = 1$ and $\tau = 100$, for $\mathbf{J}$, then defined $\mathbf{h}$ for G^{all}, followed by minimization of each corresponding $\mathrm{TV}_{G^{all}}(\mathbf{h})$ and the voting strategy [109] to designate the class labels for each set of testing samples.

To the best of the author's knowledge, most of the state-of-the-art multi-class classification methods are still based on an OAA approach [113]. There is no vigorous formulation on class labelling for an OFA approach. Thus, for OFA-RGS, we apply the same graph G^{all} as used in OAA-RGS, and heuristically defined a multi-class graph mapping $\mathbf{h}^{all}$ of G^{all} as follows: $h_i^{all} = -7 + 2q$ if $\mathcal{X}_i$ belongs to Class $q, q = 1, ..., 5$, and 0 if class is unknown.

We then perform $\mathbf{h}^{all'} = \arg \min_{\mathbf{h}^{all} \in \mathbb{R}^D} \mathrm{TV}_{G^{all}}(\mathbf{h}^{all})$ for class labels of all testing samples.

For benchmarking, we adopt "one-against-one" multi-class SVM classification [109, 114, 115], a competitive approach among five multi-class SVM classification methods compared in [116]. We first train $f(f-1)/2$ binary linear / non-linear (we use rbf kernels with scaling factor $\rho = 1$ which gives best classification results without overfitting) SVM classifiers, and then classify all testing data by using voting strategy in [109], denoted as OAO-l-SVM and OAO-rbf-SVM, respectively.

We evaluate the above 5 multi-class classification methods using k-fold cross-validation [117]. In particular, we set $k = 5$, *i.e.*, 4 folds are used for training and the last fold is used for evaluation. We repeat this process k times, leaving one different

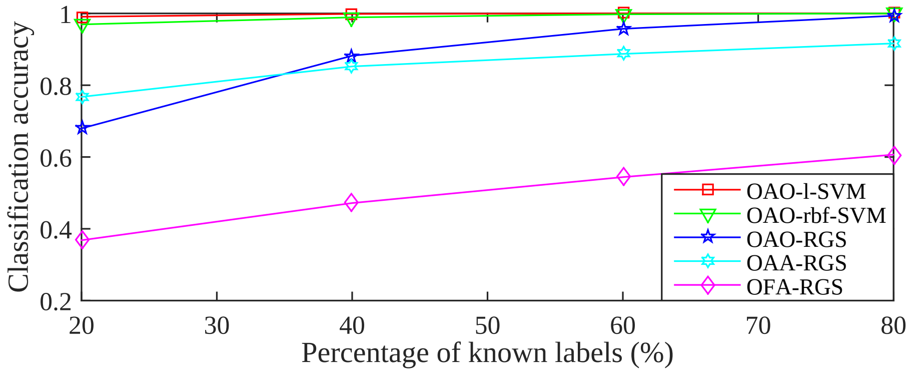


Fig. 4.11 Multi-class classification accuracy of testing data.

fold for evaluation each time. The ith process outputs a confusion matrix of data counts, denoted as

$$\mathbf{C}_c^i = \begin{bmatrix} c_{11}^i & \dots & c_{1l}^i \\ \vdots & \ddots & \vdots \\ c_{l1}^i & \dots & c_{ll}^i \end{bmatrix}, \tag{4.9}$$

whose columns represent the classifier prediction, and rows represent the true classes, *e.g.*, the value of index c_{ij}^i in $\mathbf{C}_c^i$ increases by 1 if a data sample that belongs to Class i is classified as Class j. k-fold cross-validation finally combines all $\mathbf{C}_c^i$'s into a single confusion matrix of *data counts* $\mathbf{C}_c$ with indices $c_{ij} = \sum_{i=1}^k c_{ij}^i$, and outputs the corresponding accuracy (**acc**) given by:

$$\mathbf{acc} = \frac{\sum_{i=1}^k c_{ii}}{\sum_{i=1}^k \sum_{j=1}^k c_{ij}}. \tag{4.10}$$

Note that $\mathbf{C}_c$ can be alternatively represented as a confusion matrix of *recognition rates*, denoted as

$$\mathbf{C}_r = \begin{bmatrix} \mathbf{C}_c(1,:)/\sum \mathbf{C}_c(1,:) \\ \vdots \\ \mathbf{C}_c(k,:)/\sum \mathbf{C}_c(k,:) \end{bmatrix}. \tag{4.11}$$

Next, we show the evaluation result of the above 5 multi-class classification methods using Accuracy in Fig. 4.11 (averaged over 10,000 runs based on the assumption that

Table 4.6 k-fold cross-validation result.

Method	$\mathbf{C}_r$	**acc**	$\overline{t_r}$ (ms)	$\overline{t_e}$ (ms)
OAO-l-SVM	$\mathbf{I}_5$	**1**	58.3	3.9
OAO-rbf-SVM	$\mathbf{I}_5$	**1**	53.1	4.6
OAO-RGS	$\mathbf{I}_5$	**1**	22.5	3.3
OAA-RGS	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0.1 & 0 & 0.9 & 0 & 0 \\ 0.2 & 0 & 0 & 0.8 & 0 \\ 0.2 & 0 & 0 & 0 & 0.8 \end{bmatrix}$	**0.94**	6.5	3.7
OFA-RGS	$\begin{bmatrix} 0.68 & 0.08 & 0.24 & 0 & 0 \\ 0 & 0.4 & 0.6 & 0 & 0 \\ 0 & 0 & 0.9 & 0.1 & 0 \\ 0 & 0 & 0.2 & 0.8 & 0 \\ 0 & 0 & 0.2 & 0.8 & 0 \end{bmatrix}$	**0.64**	7	1.5

between 20% and 80% of randomly selected labels are known for training) and k-fold cross-validation in Table 4.6, where $\overline{t_r}$ and $\overline{t_e}$ denote the average execution time for training and testing during the ith process of k-fold cross-validation, respectively. OFA-RGS is not competitive with any of above 4 methods. The performance of OAO-RGS is between SVM methods and OAA-RGS when the percentage of known labels is above 40%. SVM methods and OAO-RGS achieve the highest **acc**, where OAO-RGS performs faster than both SVM methods. Indeed, OAO-RGS performs over 100% and 15% faster, for training and testing, respectively, than the SVM methods. The overall performance of OAO-RGS indicate that our decision support system has the potential to accurately classify participants into a healthy group and different stroke groups with the aid of levels of impairment [51].

We note that the above multi-class analysis is provided to demonstrate the potential of the proposed methods, since the amount of data is insufficient to make firm conclusions.

4.4 Summary

Currently available optical motion analysis systems are expensive and require multiple infrared cameras, large laboratory space, and operational expertise to assess motor impairment of a stroke patient. In this chapter, we propose and evaluate an alternative, portable, and cheap, single-camera decision support system with the following components: simultaneous multiple bullseye marker tracking, autonomous joint angle calculation, visualization, and subject classification. Validation of the proposed tracking method with the current state-of-the-art VICON optical motion analysis system shows overall good limits of agreement on the upper limb motion analysis. In addition, we designed three RGS binary and multi-class classification methods, of which OAO-RGS has strong potential to explicitly classify participants into a healthy group and different stroke groups with the aid of levels of impairment. In practice, for a 10-second trial, a patient can get his/her upper limb kinematics assessed in under 2 minutes, given the average processing time (see Section 4.3.1) per video frame. Experimental results show that the proposed decision support system can track the markers with high accuracy, capture the upper limb motion explicitly, and give stroke patients and clinicians visual and written feedback based on classification with the aid of impairment levels.

Chapter 5

Abnormal Respiratory Event Detection during Sleep[1]

5.1 Introduction

Obstructive sleep apnea, characterized by repetitive obstruction in the upper airway during sleep, is a common sleep disorder that could significantly compromise sleep quality and quality of life in general. The obstructive respiratory events can be detected by attended in-laboratory or unattended ambulatory sleep studies. Such studies require many attachments to a patient's body to track respiratory and physiological changes, which can be uncomfortable and compromise the patient's sleep quality. In this chapter, we propose to record depth video and audio of a patient using a Microsoft Kinect camera during his/her sleep, and extract relevant features to correlate with obstructive respiratory events scored manually by a scientific officer based on data collected by Philips system Alice6 LDxS that is commonly used in sleep clinics. Specifically, we first propose a video recording scheme for H.264 video encoding. At the decoder,

[1] This chapter is largely based on the work that appeared in 2014 IEEE International Workshop on Hot Topics in 3D [15], 2014 IEEE International Workshop on Multimedia Signal Processing [16], and the work to appear in IEEE Transactions on Multimedia [17].

the uncoded 3 bits in each frame can be recovered via block-based search. Next, we perform temporal denoising on the decoded depth video, so that undesirable flickering can be removed without blurring sharp edges. Given the denoised depth video, we track a patient's chest and abdominal movements, extract ellipse model features and audio features, and insert them as input to a classifier to detect abnormal respiratory events. Experimental results show first that our depth video compression scheme outperforms a competitor that records only the 8 most significant bits. Second, we show that our graph-based temporal denoising scheme reduces the flickering effect without over-smoothing. Third, we show that using our extracted depth video and audio features, our trained classifiers can deduce respiratory events scored manually based on data collected by system Alice6 LDxS with high accuracy.

5.2 System overview

We first overview our proposed sleep monitoring system that employs an MS Kinect sensor to capture depth video and audio of a sleep patient. A potential usage of our system is as follows. When a patient stays overnight in a sleep clinic for initial testing, in addition to in-hospital system's sensors, we deploy also a Kinect sensor to capture depth video and audio for respiratory event classifier training. In subsequent nights at the patient's home, our proposed system that replicates the same Kinect sensor setup is activated to collect depth video and audio data non-intrusively for respiratory event classification. Without the body-attached sensors, this would mean a significant improvement in sleep comfort for the patient when at home.

Specifically, we employ a first-generation MS Kinect depth camera for depth video and audio processing and respiratory event classification. As shown in Fig. 5.1, the camera is set up at a higher elevation above and away from the head of the patient lying down. This camera location gives an unobstructed view of the patient's torso for depth video capture and analysis. The Kinect camera captures depth images of resolution 640 × 480 pixels with 11-bit pixel precision at 30 frames per second. The camera can also simultaneously capture audio at 16kHz, 16-bit sample precision with a PCM S16 LE audio codec [118]. Note that though Kinect camera has a 4-microphone array resulting in a 4-channel audio, we use only the first channel for recording.

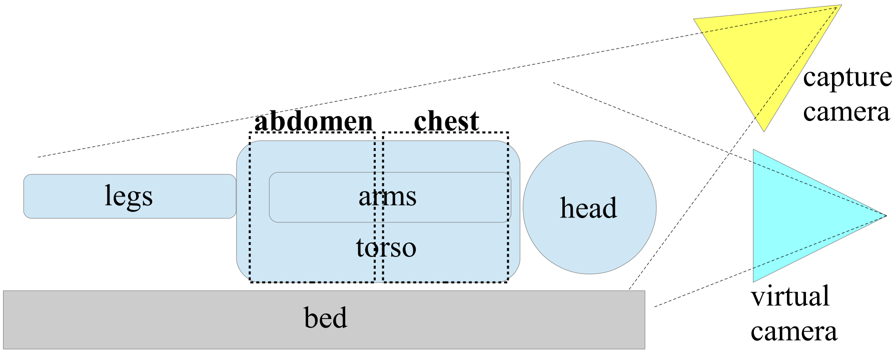


Fig. 5.1 Side view of sleep patient. Torso is divided into two cross sections, each modeled by an ellipse.

The first component of our system is the real-time capturing and compression of depth video (for transmission of captured video to a remote powerful server for storage and analysis) and recording of single-channel audio. We propose an efficient H.264 implementation of Kinect-captured video, where different 8 bits per pixel are extracted from 11 available bits of different temporal frames for encoding. At decoder, the uncoded 3 bits are recovered from neighboring frames via block motion search.

Second, we employ a graph-based temporal denoising algorithm to remove unwanted acquisition noise and flickers in recorded depth video. We show that the temporal flickers can be noticeably removed without over-smoothing and blurring of sharp edges typical in depth images.

Third, using the denoised depth video we track the chest and abdominal movements of the patient over time, as shown in Fig. 5.1. In a nutshell, we model the cross-sections of the patient's chest and abdomen as ellipses, and we derive ellipse parameters that best fit the observed depth pixels per frame. The changes of the ellipse parameters over time will reveal breathing cycles and patterns.

Finally, we perform wavelet packet transform (WPT) [119, 120] on the ellipse parameters to extract video features, and non-negative matrix factorization (NMF) [121–124] on the recorded audio to extract audio features. In particular, WPT decomposition adopts recursive splitting of vector spaces that is represented in a binary tree (see Fig. 8.1 in [120] for an example), which produces a redundant representation by using analysis filters for both high and low frequencies [125, 126]. NMF is commonly used for audio feature extraction. Indeed, NMF is frequently used in spectral data analysis [124], such as audio signals. By the virtue of nonnegativity [123], NMF is able to unsupervisedly learn parts representation of the signal, in contrast to other methods, such as Principal Component Analysis (PCA) and vector quantization, that learn holistic, distributed representations [122]. The extracted features are used to train an SVM classifier and a feed-forward neural network (NN) with four event classes: i) central apnea, ii) obstructive or mixed apnea, iii) hypopnea, and iv) all the other events that are available from the ground truth labels. Fig. 5.2 illustrates the overall proposed system.

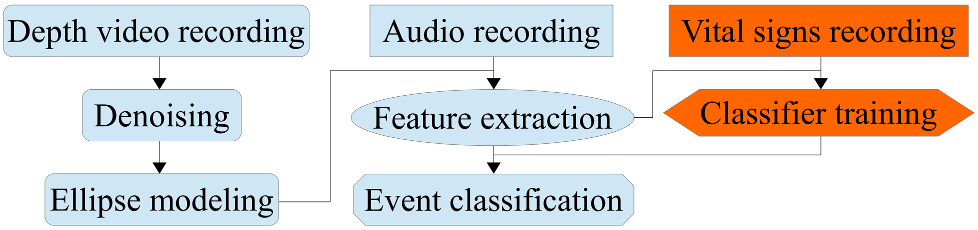
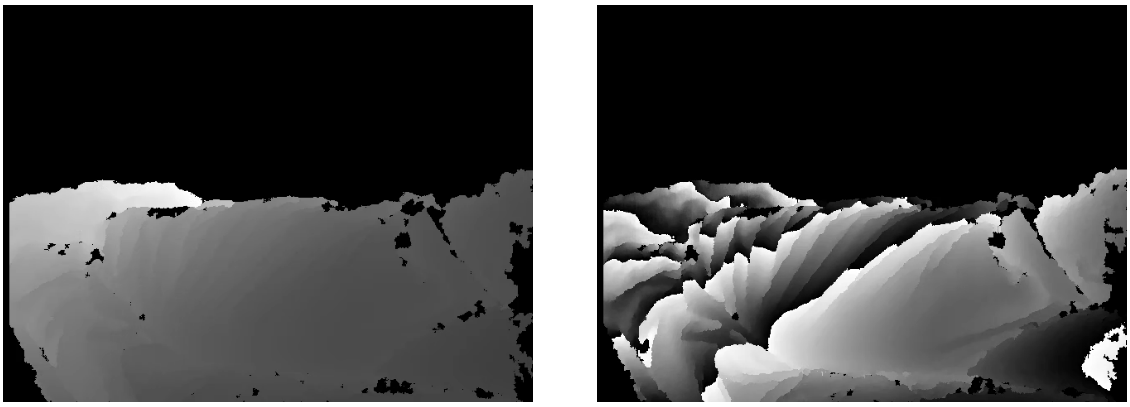


Fig. 5.2 System overview. ‘Vital Signs Recording’ is for ground truth; Orange: initial training components; Blue: regular usage components.

5.3 Depth video recording

We now describe our proposed coding algorithm to compress captured depth videos of sleeping patients. Each depth image captured by a first-generation MS Kinect sensor contains 11-bit precision pixels at spatial resolution 640 × 480. *Baseline profile* for video coding standard H.264 [127]—the most prevalent and optimized profile—supports only 8-bit precision, however[2]. Thus, we propose an alternating frame coding scheme to extract different 8 of 11 available bits in each captured pixel of different frames for encoding. At the decoder, we recover the uncoded 3 bits using our proposed recovery scheme. The reasons we can recover the uncoded 3 bits with high accuracy are: i) depth maps are known to be *piecewise smooth* (PWS), and ii) in a typical sleep video, only slow motion exists across frames. We discuss the encoding and decoding procedures next.

5.3.1 Encoder Selection of 8 Coding Bits



(a) MSB frame

(b) LSB frame

Fig. 5.3 Examples of MSB and LSB frames. In the MSB frame, the representation correctly shows the observations in different distances to the camera where brighter observations indicate further distances. However, due to the overflow fact, in the LSB frame, the representation incorrectly shows the observations in different distances, where brighter observations do not indicate further distances.

[2] Only High 4:4:4 Profile, that leads to high encoding complexity, supports 11 to 14 bits precision.

The encoder selects different 8 bits for each depth frame $\mathbf{Z}_t$ of time instant t for encoding as follows. Denote by M the *reference picture selection* (RPS) parameter used during H.264 video encoding [127]; *i.e.*, a P-frame $\mathbf{Z}_t$ can choose any one of the previous M frames $\mathbf{Z}_{t-1}, \dots, \mathbf{Z}_{t-M}$ as predictor for differential coding. If $t \bmod M = 0$, then we select the 8 *most significant bits* (MSB) of 11 captured bits in each captured depth pixel in target frame $\mathbf{Z}_t$ for encoding. Otherwise, we select the 8 *least significant bits* (LSB) of 11 available bits in each pixel for encoding. MSB frames and LSB frames are very different; MSB frames are very smooth with missing details (contained in lost LSBs), while LSB frames suffer from overflow due to missing MSBs. See Fig. 5.3 for an illustration. In the MSB frame, the representation correctly shows the observations in different distances to the camera where brighter observations indicate further distances. However, due to the overflow fact, in the LSB frame, the representation incorrectly shows the observations in different distances, where brighter observations do not indicate further distances. However, our proposed encoding scheme ensures that each MSB or LSB frame $\mathbf{Z}_t$ can find a similar previous frame $\mathbf{Z}_{t-i}$ in predictor frame set $\{\mathbf{Z}_{t-1}, \dots, \mathbf{Z}_{t-M}\}$ for differential coding thanks to RPS in H.264, thus achieving good coding efficiency (demonstrated in Sec. 5.7.2.1).

5.3.2 Decoder recovery of full 11 bits

At the decoder, we recover the uncoded 3 MSBs in an LSB frame as follows. We first segment an LSB frame into *smooth regions*, *i.e.*, spatial regions where adjacent pixels do not differ by more than a pre-defined threshold δ. Pixels in the same smooth region will share the same to-be-recovered 3 MSBs.

Next, we identify potential *overflow* pixels in an LSB frame due to encoding of LSBs only—pixels that were similar to adjacent pixels before removal of 3 MSBs. Specifically, given smooth region boundary pixel location $\mathbf{p}$ in LSB frame $\mathbf{Z}_t$ where its pixel value is close to zero, *i.e.*, $\mathbf{Z}_t(\mathbf{p}) \leq \delta$, we check if adding one significant bit 2^8 would bring it closer to within δ of one of its neighbors, *i.e.*,:

$$\min_{\mathbf{q} \in \mathcal{N}_\mathbf{p}} \left| \mathbf{Z}_t(\mathbf{p}) + 2^8 - \mathbf{Z}_t(\mathbf{q}) \right|, \leq \delta \tag{5.1}$$

where $\mathcal{N}_\mathbf{p}$ is the set of adjacent pixels to $\mathbf{p}$. If this is the case, then $\mathbf{p}$ is a potential overflow pixel. To check if $\mathbf{p}$ is an overflow pixel (or simply an object boundary), we perform *motion estimation* (ME) [128] using the most recent MSB frame $\mathbf{Z}_\tau$. Specifically, given an $R \times R$ block $B_\mathbf{p}$ with center at $\mathbf{p}$ of the current frame $\mathbf{Z}_t$ as target, we compute:

$$\min_{\mathbf{v}} \left| \mathbf{Z}_\tau(B_{\mathbf{p}+\mathbf{v}}) \bmod 2^5 - \left\lfloor \frac{\mathbf{Z}_t(B_\mathbf{p})}{2^3} \right\rfloor \right| + \mu |\mathbf{v}|, \tag{5.2}$$

where the 5 LSBs in block $B_{\mathbf{p}+\mathbf{v}}$ of $\mathbf{Z}_\tau$ and the 5 MSBs in block $B_\mathbf{p}$ of $\mathbf{Z}_t$ are compared—only 5 bits are common between MSB and LSB frames. Note that we add the magnitude of the *motion vector* (MV) $\mathbf{v}$ as a regularization term, because for PWS images, there can be multiple vectors $\mathbf{v}$ with very small block differences. $|\mathbf{v}|$ means we favor the smallest motion block in frame $\mathbf{Z}_\tau$, which is reasonable due to low level of motion in sleep videos. μ is a parameter that trades off the block differential and the regularization terms.

Given the best MV $\mathbf{v}_\mathbf{p}$ computed in (5.2), we then check if $B_{\mathbf{p}+\mathbf{v}_\mathbf{p}}$ is smooth in $\mathbf{Z}_\tau$. If so, then pixel $\mathbf{p}$ in $\mathbf{Z}_t$ is deemed an overflow bit, and we merge the smooth region of $\mathbf{p}$ with the corresponding neighboring smooth region; *i.e.*, the merged smooth region will share the same MSBs. If not, then this is actually an object boundary, and we copy the 3 MSBs in $B_{\mathbf{p}+\mathbf{v}_\mathbf{p}}$ of $\mathbf{Z}_\tau$ to *all* pixels in the smooth region containing $\mathbf{p}$. Fig. 5.4 illustrates the above procedure of decoder recovery of full 11 bits.

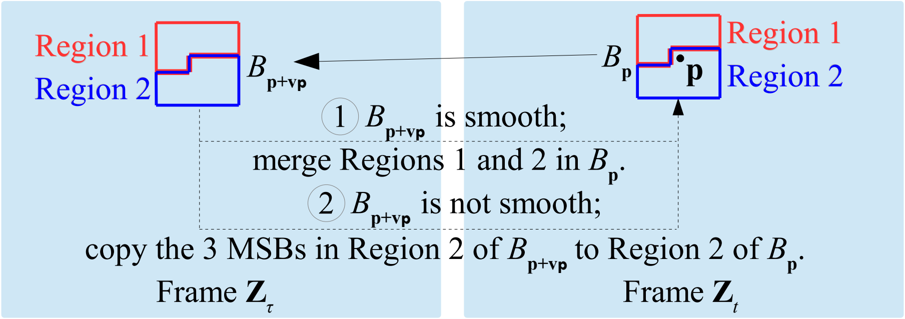

Region 1
Region 2
$B_{\mathbf{p+vp}}$
$B_{\mathbf{p}}$
p
Region 1
Region 2
1 $B_{\mathbf{p+vp}}$ is smooth;
merge Regions 1 and 2 in $B_{\mathbf{p}}$.
2 $B_{\mathbf{p+vp}}$ is not smooth;
copy the 3 MSBs in Region 2 of $B_{\mathbf{p+vp}}$ to Region 2 of $B_{\mathbf{p}}$.
Frame $\mathbf{Z}_\tau$
Frame $\mathbf{Z}_t$

Fig. 5.4 Decoder bits recovery given $\mathbf{p}$ is a potential overflow pixel.

5.4 Depth video denoising

Depth images captured by a Kinect camera are susceptible to acquisition noise and have missing pixel values especially around object boundaries, which can adversely affect the performance of subsequent sleep event classification. In this section, we propose a temporal denoising algorithm based on a graph-signal formulation. We show how a graph-signal smoothness prior can be used for temporal denoising in depth videos, which is more complex than spatial denoising [129] and involves the joint optimization of motion vectors (MV) and noise-corrupted pixels in the target frame.

We first formulate an optimization problem for the motion field in a frame t given previous frame $t-1$ and a motion smoothness prior. Then we discuss how the problem can be modified if frame t is corrupted by noise, and present an efficient algorithm to solve it.

5.4.1 Finding motion field

For simplicity, we assume first that neither target frame t nor previous frame $t-1$ is corrupted by noise. The goal is to find an accurate motion field for all $K \times K$ pixel blocks in frame t. Let $\mathcal{B}_{\mathbf{p}_i}(t)$ be the i-th $K \times K$ block in frame t, with upper-left pixel at $\mathbf{p}_i$. Let $\mathbf{v}_i = (x_i, y_i)$ be the MV of the i-th block. The MV field of all N blocks in the frame is expressed in vector form as $\mathbf{v} = [\mathbf{v}_1, \dots, \mathbf{v}_N]$.

We first assume a *spatial motion smoothness prior*: a block's MV will be similar to MVs of neighboring blocks if they belong to the same object; *i.e.*, the MV field is PWS. One way of expressing piecewise smoothness is through a graph [130–133]. We first construct a four-connected graph, where each node i represents a block $\mathcal{B}_{\mathbf{p}_i}(t)$ and is connected to nodes corresponding to neighboring blocks of $\mathcal{B}_{\mathbf{p}_i}(t)$. We compute the weight $w_{i,j}$ of an edge connecting two nodes (blocks) i and j as follows:

$$w_{i,j} = \exp\left\{ -\frac{\|\mathbf{v}_i - \mathbf{v}_j\|_2^2}{\sigma_v^2} \right\}, \tag{5.3}$$

where σ_v is a scaling factor. Given the constructed graph, we can define the *degree* and *adjacency* matrices, $\mathbf{P}$ and $\mathbf{A}$, correspondingly [52]. The *graph Laplacian* is defined as:

$$\mathbf{L} = \mathbf{P} - \mathbf{A}. \tag{5.4}$$

If the MV field is PWS, the *graph variation* term, $\|\mathbf{v}^\top \mathbf{L}\mathbf{v}\|_2^2$, is small:

$$\mathbf{v}^\top \mathbf{L}\, \mathbf{v} = \sum_{i,j} w_{i,j} \left(\mathbf{v}_i - \mathbf{v}_j\right)^2. \tag{5.5}$$

Note that because $\mathbf{v}_i$ contains x- and y-coordinates of the MV, $\|\mathbf{v}^\top \mathbf{L}\mathbf{v}\|_2^2$ means computing $\mathbf{v}^\top \mathbf{L}\mathbf{v}$ for the x- and y-coordinates, $\mathbf{v}(x)$ and $\mathbf{v}(y)$ of $\mathbf{v}$, separately, then computing the resulting vector magnitude square.

We can now define an optimal MV field as one that results in good block matches in the previous frame $t-1$ *and* is smooth with respect to the graph:

$$\min_{\mathbf{v}} \sum_i \|\mathcal{B}_{\mathbf{p}_i + \mathbf{v}_i}(t-1) - \mathcal{B}_{\mathbf{p}_i}(t)\|_2^2 \; + \; \lambda \, \|\mathbf{v}^\top \mathbf{L}\mathbf{v}\|_2^2, \tag{5.6}$$

where λ is a chosen weighting parameter that trades off the ME term (first term) and the MV smoothness term (second term).

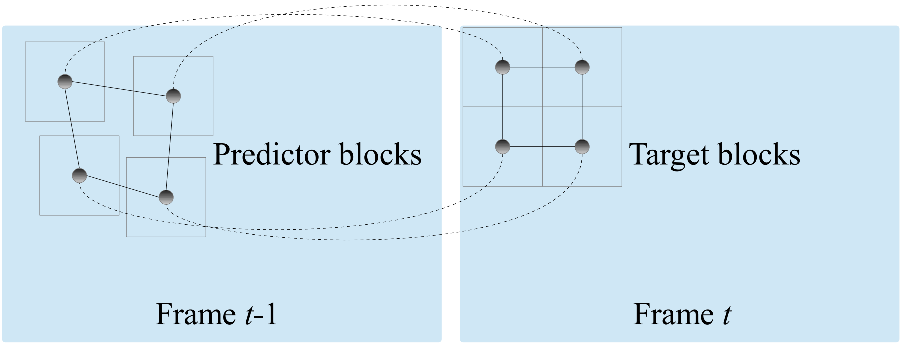


Fig. 5.5 Example graph construction given four blocks in target frame t and four corresponding predictor blocks in previous frame $t-1$.

5.4.2 Temporal denoising

We now remove the earlier assumption that target frame t is noiseless, meaning we have to find MV field $\mathbf{v}$ *and* denoise blocks $\mathcal{B}_{\mathbf{p}_i}(t)$ simultaneously. Beyond spatial MV smoothness prior, we now assume further a *temporal MV smoothness prior*; *i.e.*, if the i-th block at position $\mathbf{p}_i$ of frame t has MV $\mathbf{v}_i$, then the predictor block at position $\mathbf{p}_i + \mathbf{v}_i$ of frame $t-1$ will have a MV $\mathbf{u}_{\mathbf{p}_i+\mathbf{v}_i}$ that is similar to $\mathbf{v}_i$. We can again express this notion of smoothness via a graph. In particular, in addition to the graph constructed for MV $\mathbf{v}_i$ in frame t, we create additional nodes to represent predictor blocks in frame $t-1$. We draw an edge between node representing block $\mathcal{B}_{\mathbf{p}_i}(t)$ in frame t and node representing corresponding predictor block $\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1)$ with weight computed by (5.3).

Furthermore, we draw an edge between two predictor blocks at locations $\mathbf{p}$ and $\mathbf{q}$ in frame $t-1$ if $\|\mathbf{p}-\mathbf{q}\|_2^2 \leq \Delta$, with edge weight computed as:

$$w_{i,j} = \exp\left\{-\frac{\|\mathbf{u}_\mathbf{p} - \mathbf{v}_\mathbf{q}\|_2^2}{\sigma_v^2}\right\} \exp\left\{-\frac{\|\mathbf{p}-\mathbf{q}\|_2^2}{\sigma_g^2}\right\}, \tag{5.7}$$

where σ_g is a scaling factor. This weight assignment is similar to the one done in *bilateral filtering* [134]. See Fig. 5.5 for an example of a graph constructed from four blocks in the target frame t and four corresponding predictor blocks in the previous frame $t-1$.

Without loss of generality, we define the combined motion vector ϖ to be a concatenation of MV $\mathbf{u}$ of predictor blocks of frame $t-1$ and MV $\mathbf{v}$ of target blocks of frame t, *i.e.*, $\varpi^\top = [\mathbf{u}^\top \ \mathbf{v}^\top]$. We can also define degree and adjacency matrices $\mathbf{P}$ and $\mathbf{A}$ as done previously for the larger graph. The resulting Laplacian $\mathbf{L}$ is again $\mathbf{L} = \mathbf{P} - \mathbf{A}$.

With these definitions, we can define the new objective to find MV $\mathbf{v}$ and denoised blocks $\mathcal{B}_{\mathbf{p}_i}(t)$ as a sum of three terms: i) ME error term, ii) MV smoothness term, and iii) fidelity term with respect to observed noisy blocks $\mathcal{B}^o_{\mathbf{p}_i}(t)$, *i.e.*,

$$\min_{\mathbf{v},\mathcal{B}(t)} \left\{ \begin{array}{l} \sum_i \|\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1) - \mathcal{B}_{\mathbf{p}_i}(t)\|_2^2 \ + \ \lambda \, \|\varpi^\top \mathbf{L} \varpi\|_2^2 \\ \ + \ \mu \, \sum_i \|\mathcal{B}_{\mathbf{p}_i}(t) - \mathcal{B}^o_{\mathbf{p}_i}(t)\|_2^2 \end{array} \right\}, \tag{5.8}$$

where μ is a weighting parameter for the fidelity term. Note that, an ME error term (the first term in (5.8)) is introduced so that similar blocks can be identified between the previous and current frames. A regularization term (the second term in (5.8)) is employed to constrain the search space in an under-determined inverse problem. Finally, a fidelity term (the third term in (5.8)) is used to ensure that the denoised block is closed to the observation. We discuss how we solve (5.8) next.

5.4.3 Optimization algorithm

(5.8) is difficult to solve as it involves many variables. Our strategy is to alternately solve one set of variables at a time while keeping the other set fixed, until convergence. Suppose first we initialize MV $\mathbf{v}$ using conventional ME [128], then fix $\mathbf{v}$ and solve for optimal blocks $\mathcal{B}_{\mathbf{p}_i}(t)$. The MV smoothness term is not affected by the selection of

$\mathcal{B}_{\mathbf{p}_i}(t)$, and so (5.8) reduces to:

$$\min_{\mathcal{B}(t)} \sum_i \|\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1) - \mathcal{B}_{\mathbf{p}_i}(t)\|_2^2 + \mu \sum_i \|\mathcal{B}_{\mathbf{p}_i}(t) - \mathcal{B}^o_{\mathbf{p}_i}(t)\|_2^2. \tag{5.9}$$

Let $\mathcal{B}_{\mathbf{p}_i}(t)$ be a convex combination of $\mathcal{B}_{\mathbf{p}_i-\mathbf{v}_i}(t-1)$ and $\mathcal{B}^o_{\mathbf{p}_i}(t)$, *i.e.*,

$$\mathcal{B}_{\mathbf{p}_i}(t) = \iota\, \mathcal{B}_{\mathbf{p}_i-\mathbf{v}_i}(t-1) + (1-\iota)\, \mathcal{B}^o_{\mathbf{p}_i}(t). \tag{5.10}$$

By substituting (5.10) into (5.9), taking the derivative with respect to ι and setting the equation to zero, we see that the optimal ι^* is:

$$\iota^* = \frac{1}{1+\mu}. \tag{5.11}$$

This agrees with intuition; if $\mu = 0$, then $\iota^* = 1$ and $\mathcal{B}_{\mathbf{p}_i}(t)$ is set to predictor block $\mathcal{B}_{\mathbf{p}_i-\mathbf{v}_i}(t-1)$, and if $\mu = 1$, then $\iota^* = 1/2$, and $\mathcal{B}_{\mathbf{p}_i}(t)$ is the average of predictor block $\mathcal{B}_{\mathbf{p}_i-\mathbf{v}_i}(t-1)$ and observed noisy block $\mathcal{B}^o_{\mathbf{p}_i}(t)$.

Now we fix blocks $\mathcal{B}_{\mathbf{p}_i}(t)$ and solve for the optimal MV $\mathbf{v}$. The fidelity term is not affected by MV $\mathbf{v}$, so (5.8) reduces to:

$$\min_{\mathbf{v}} \sum_i \|\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1) - \mathcal{B}_{\mathbf{p}_i}(t)\|_2^2 \;+\; \lambda\, \|\varpi^\top \mathbf{L} \varpi\|_2^2. \tag{5.12}$$

(5.12) is still difficult to solve, since each change in MV $\mathbf{v}_i$ induces a change in corresponding predictor block $\mathcal{B}_{\mathbf{p}_i+\mathbf{v}_i}(t-1)$, resulting in a different predictor MV $\mathbf{u}_{\mathbf{p}_i+\mathbf{v}_i}$ and a modified Laplacian $\mathbf{L}$. Our strategy then is to find first the optimal MV $\mathbf{v}^*$ that minimizes the smoothness term, then insert $\mathbf{v}_i^*$ into (5.12) to see if the objective is reduced.

Given ϖ is a concatenation of predictor MV $\mathbf{u}$ and target MV $\mathbf{v}$, we can rewrite the smoothness term as:

$$\begin{aligned} & \underbrace{\begin{bmatrix} \mathbf{u}^\top & \mathbf{v}^\top \end{bmatrix}}_{\varpi^\top} \underbrace{\begin{bmatrix} \mathbf{L_{uu}} & \mathbf{L_{uv}} \\ \mathbf{L_{vu}} & \mathbf{L_{vv}} \end{bmatrix}}_{\mathbf{L}} \underbrace{\begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}}_{\varpi} \\ = \; & \mathbf{u}^\top \mathbf{L_{uu}} \mathbf{u} + \mathbf{u}^\top \mathbf{L_{uv}} \mathbf{v} + \mathbf{v}^\top \mathbf{L_{vu}} \mathbf{u} + \mathbf{v}^\top \mathbf{L_{vv}} \mathbf{v}. \end{aligned} \tag{5.13}$$

The first term is a constant and not influenced by $\mathbf{v}$. Additionally,

$$\mathbf{u}^\top \mathbf{L_{uv}} \mathbf{v} = \mathbf{v}^\top \mathbf{L_{vu}} \mathbf{u}. \tag{5.14}$$

Thus to find $\mathbf{v}^*$ that minimizes the smoothness term, we write:

$$\min_{\mathbf{v}} \mathbf{v}^\top \mathbf{L_{vv}} \mathbf{v} + 2\mathbf{u}^\top \mathbf{L_{uv}} \mathbf{v}. \tag{5.15}$$

This is an unconstrained quadratic programming problem, with closed form solution [135]:

$$\mathbf{v}^* = \mathbf{L}^{\#}_{\mathbf{vv}} \left(-\mathbf{u}^\top \mathbf{L_{uv}} \right)^\top, \tag{5.16}$$

where $\mathbf{L}^{\#}_{\mathbf{vv}}$ is the pseudo-inverse of $\mathbf{L_{vv}}$.

Because $\mathbf{v}^*$ only minimizes the second term in objective (5.12), we perform the following greedy procedure using $\mathbf{v}^*$ to reduce the overall objective function value: we iteratively insert a maximally “beneficial” component of $\mathbf{v}^*$ (one that decreases the objective (5.12)) into the current vector $\mathbf{v}$. We stop when no more beneficial components in $\mathbf{v}^*$ exist.

Pixels in frame t, $\mathcal{B}(t)$, and MV $\mathbf{v}$ are alternately optimized using the two procedures described above, until the solution converges. Experimentation shows this only requires a few iterations in practice.

The proposed graph-based depth video temporal denoising scheme is summarized in Algorithm 3.

Algorithm 3: Graph-based depth video temporal denoising.

```
Input: Frames $t-1, t$;
Output: Denoised Frame $t$;
Initialise $\mathbf{u}$, $\mathbf{v}$;
while not converged do
    Optimise $\mathcal{B}_{\mathbf{p}_i}(t)$ in Frame $t$ by minimizing (5.9) given $\mathcal{B}^{o}_{\mathbf{p}_i}(t)$ and fixed $\mathbf{v}$;
    Optimise $\mathbf{v}$ by minimizing $\varepsilon \left\| \varpi^{\top} \mathbf{L} \varpi \right\|_2^2$ in (5.12) given $\mathbf{u}$ and $\mathcal{B}_{\mathbf{p}_i}(t)$;
    Further optimise $\mathbf{v}$ by iteratively inserting maximally "beneficial" component of
        $\mathbf{v}^*$ (to minimize (5.12)) into current $\mathbf{v}$ until no more beneficial components in $\mathbf{v}^*$
        exist;
end while
```

5.5 Ellipse modelling of human torso

In this section we discuss how we build our ellipse model in two steps using the denoised depth video. In the first step, each depth pixel from the captured camera view is mapped to a virtual camera view (*head-on view*) as illustrated in Fig. 5.1. To reduce the computation time, the region of interest is identified as a bounding box that contains only depth pixels of the patient, which is based on the difference of the depth images taken before and after the patient gets in bed. Each depth pixel with coordinate (u, v, d) in the virtual view is then classified into two different cross sections of the patient's torso—chest and abdomen—based on depth value d. See Appendix B.2 for details of the above view transformation.

In the second step, we model each cross section (chest or abdomen) as an ellipse; *i.e.*, we estimate a best-fitting ellipse based on the set of observations (u, v)'s classified to this cross section. During regular breathing, the patient's chest and abdomen will expand and contract, resulting in ellipse size changes over time. We estimate the major and minor radii of ellipses per frame given observed depth video to track the patient's breathing cycle over time. Our ellipse model can in addition detect the patient's body tilt during sleep (*e.g.*, sleeping on the side), resulting in rotated model ellipses about the origin. We describe how we formulate and solve the ellipse-fitting problem in detail next.

5.5.1 Problem formulation

Let $\mathbf{o}= \{\mathbf{o}_1, \ldots, \mathbf{o}_N\}$ be the set of N observations for construction of one ellipse, where $\mathbf{o}_n$ is (u_n, v_n)—the observation's location in the *u*-*v* image coordinate system as observed from the virtual view. The parametrization of an ellipse in a Cartesian *u*-*v* coordinate system is:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} c_u \\ c_v \end{pmatrix} + \begin{pmatrix} \cos\delta & -\sin\delta \\ \sin\delta & \cos\delta \end{pmatrix} \begin{pmatrix} a\ \cos\phi \\ b\ \sin\phi \end{pmatrix}, \phi \in [0, 2\pi], \tag{5.17}$$

where (c_u, c_v) denotes the center of the ellipse, a and b denote the major and minor radii, respectively, and δ denotes the ellipse tilt that models the patient's body tilt. For simplicity, we assume that the center of the ellipse is at the origin, *i.e.*, $c_u = c_v = 0$. An ellipse can thus be characterized by $\theta= (a, b, \delta)$. In practice, the middle point of the bed within the view-transformed depth image is designated as the centre of the ellipse.

Denote by $s_\theta(\mathbf{o}_n)$ the *minimum Euclidean distance* between observation $\mathbf{o}_n$'s location (u_n, v_n) and the ellipse with parameter θ. We formulate the following objective to find the best-fit ellipse parameters θ^* given observations $\mathbf{o}$:

$$\theta^* = \arg\min_{\theta} \sum_{n=1}^{N} s_\theta^2(\mathbf{o}_n). \tag{5.18}$$

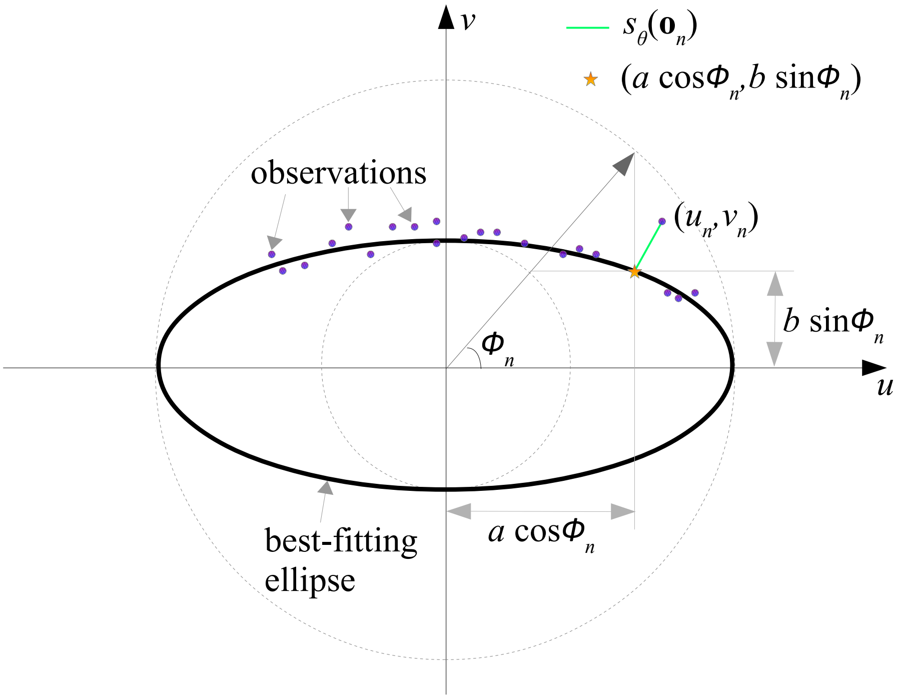


Fig. 5.6 Best-fitting ellipse from multiple depth observations of the cross section. The closest ellipse point to each observation is perpendicular to the tangent of ellipse at that point.

For example, for an ellipse with $\theta = (a, b, \delta)$, $s_\theta(\mathbf{o}_n) = \|(u_n, v_n) - (u_{\min}, v_{\min})\|_2$, where $(u_{\min}, v_{\min})$ is the closest point on the ellipse to (u_n, v_n); *i.e.*, the vector (u_n, v_n) to point $(u_{\min}, v_{\min})$ on the ellipse is orthogonal to the tangent of the ellipse at $(u_{\min}, v_{\min})$ [136, 137]. See Fig. 5.6 for an illustration of an ellipse with $\theta = (a, b)$.

5.5.2 Optimization algorithm

Conventionally, (5.18) can be computed via either geometric ellipse fitting in parametric form by solving an equivalent nonlinear least squares problem, or fast algebraic ellipse fitting with geometric distance weighting [138]. Neither of these two approaches require initial ellipse parameters θ. However, the former can be very inefficient when building Jacobian due to large number of $\mathbf{o}_n$'s, and the latter does not generally minimize the geometric distance.

Note that, computing each $s_{\theta}(\mathbf{o}_n)$ given initial ellipse parameters θ is a well-known *root-finding*problem, which can be solved by solving a quartic equation with four roots [136, 139]. The root that is closest to (u_n, v_n) is then chosen to determine ϕ_n. However, this is clearly inefficient given a large number of $\mathbf{o}_n$'s. Instead, we adopt the Bisection (BS) method [137, 140–142]. (See Appendix B.3.) We choose the BS method instead of Newton's Method as done in [15, 16], because the latter has numerical problems when v_n is nearly zero.

Since $\sum_{n=1}^{N} s_{\theta}^{2}(\mathbf{o}_n)$ is non-convex, we resort to a local numerical method—the Nelder-Mead (NM) simplex method [141, 143, 144]—to find the best ellipse parameters θ^{*} in (5.18), using $s_{\theta}(\mathbf{o}_n)$ found by the BS method explained above. See Appendix B.4 for details.

5.6 Feature extraction and classification

In this section, we describe how to extract relevant features from the depth video signal (*i.e.*, the four computed 1D signals—major and minor radii of the two fitted ellipses (chest and abdomen) as functions of time) and audio signal. We note that the time duration for each experimental data segment for feature extraction—both the computed 1D ellipse signal segments $\mathbf{x}$ and the audio signal segment $\mathbf{y}$—is set at 10 sec, which is the medically defined duration of a respiratory event [145]. The segment window is then shifted by 5 sec, so neighboring segments have a 5-sec overlap.

5.6.1 Depth video features

Unlike [15, 16] where we directly used the variances of the ellipses' major and minor radii in a time window to perform classification, in this chapter, we adopt wavelet analysis, namely, WPT [119, 120, 146] (see Sec. 5.2).

In particular, each sub-segment $\boldsymbol{a} \in \mathbb{R}$ (with the size defined in Sec. 5.7.2.3) of the 10-sec 1-D ellipse signal segment $\mathbf{x}$ (*i.e.*, the amplitude of the ellipse major/minor radius over time), is approximated at the scale 2^J, *i.e.*, at J levels (where $J \in [0, \log_2 N]$, with N being the number of samples in $\boldsymbol{a}$). Each level j contains N approximation and detail coefficients that are divided into 2^j tree-nodes, and each tree-node thus contains $N/2^j$ coefficients.

After this WPT signal decomposition, we concatenate the normalized logarithmic energy [146] of each coefficient in the increasing order of the tree-nodes resulting in the feature vector $\tilde{\mathbf{E}}_i$ for the i-th sub-segment $\boldsymbol{a}_i$ of the original 1D ellipse signal segment $\mathbf{x}$. Finally, we concatenate all $\tilde{\mathbf{E}}_i$ forming a feature vector

$$\tilde{\mathbf{E}} = \left[\tilde{\mathbf{E}}_1, \ldots, \tilde{\mathbf{E}}_P\right] \tag{5.19}$$

for $\mathbf{x}$, where P is the number of sub-segments $\boldsymbol{a}_i$ of $\mathbf{x}$.

5.6.2 Audio features

For audio feature extraction we resort to NMF[121, 123] (see Sec. 5.2). We perform NMF decomposition in the following way. We first apply short-time Fourier transform (STFT) on each sub-segment $\boldsymbol{b}$ (with the size defined in Sec. 5.7.2.3) of the 10-sec 1-D audio signal segment $\mathbf{y}$, resulting in a spectrogram matrix $\mathbf{Y}\in \mathbb{R}^{m\times n}$ as the magnitude of STFT. Then, we solve the NMF problem, *i.e.*, find a spectral-feature matrix $\mathbf{W}\in \mathbb{R}_{\geq 0}^{m\times k}$ and a temporal-activity matrix $\mathbf{H}\in \mathbb{R}_{\geq 0}^{k\times n}$ by minimizing the following cost function:

$$D(\mathbf{Y}|\ \mathbf{WH}) = ||\mathbf{Y} - \mathbf{WH}||^2, \tag{5.20}$$

where D indicates the distance between $\mathbf{Y}$ and $\mathbf{WH}$, the product $\mathbf{WH}$ is an approximate factorization of $\mathbf{Y}$ at rank k. We discuss how to choose an appropriate rank k in our experiments in Sec. 5.7.2.3.

An alternating least-square (ALS) update rule [124] is used to find the optimal matrices $\mathbf{W}$ and $\mathbf{H}$. Specifically, $\mathbf{W}$ is initialized as an $m \times k$ random dense matrix, then iteratively solve for $\mathbf{H}$ based on

$$\mathbf{W}^{\top}\mathbf{W}\mathbf{H} = \mathbf{W}^{\top}\mathbf{Y}, \tag{5.21}$$

followed by a projection step, *i.e.*, setting all negative elements within $\mathbf{H}$ to 0. Next, we solve for $\mathbf{W}$ based on

$$\mathbf{H}\mathbf{H}^{\top}\mathbf{W}^{\top} = \mathbf{H}\mathbf{Y}^{\top}, \tag{5.22}$$

followed by the same projection step on $\mathbf{W}$. The above ALS rule with projection steps aids sparsity, converges faster and performs more consistently comparing with multiplicative update rules [124]. To alleviate the uniqueness problem which can be easily seen by considering $\mathbf{W}\mathbf{X}\mathbf{X}^{-1}\mathbf{H}$ for any non-negative nonsingular matrix $\mathbf{X}$ [124], given $\mathbf{W}$ and $\mathbf{H}$ after each iteration, we first normalize them as

$$\hat{\mathbf{W}} = \mathbf{W}\mathbf{X}, \;\; \hat{\mathbf{H}} = \mathbf{X}^{-1}\mathbf{H}, \tag{5.23}$$

where

$$\mathbf{X} = \mathrm{diag}(\sqrt{\sum_{u=1}^{n} \mathbf{H}(1,u)^2}, ..., \sqrt{\sum_{u=1}^{n} \mathbf{H}(k,u)^2}). \tag{5.24}$$

Then, for obtaining a consistent permutation, we reorder the columns of $\hat{\mathbf{W}}$ as $\tilde{\mathbf{W}}$ by the index of the decreasing magnitude of the elements in:

$$\dot{\mathbf{W}} = \left[\textstyle\sum_{u=1}^{m} \hat{\mathbf{W}}(u,1)^2, ..., \sum_{u=1}^{m} \hat{\mathbf{W}}(u,k)^2\right], \tag{5.25}$$

followed by reordering the rows of $\hat{\mathbf{H}}$ as $\tilde{\mathbf{H}}$ accordingly.

We perform NMF decomposition on $\boldsymbol{b}_i$ using the designated rank k, reshape $\tilde{\mathbf{W}}$ as

$$\breve{\mathbf{W}} = \left[\tilde{\mathbf{W}}(:,1)^\top, \ldots, \tilde{\mathbf{W}}(:,k)^\top\right], \tag{5.26}$$

reshape $\tilde{\mathbf{H}}$ as

$$\breve{\mathbf{H}} = \left[\tilde{\mathbf{H}}(1,:), \ldots, \tilde{\mathbf{H}}(k,:)\right], \tag{5.27}$$

concatenate $\breve{\mathbf{W}}$ and $\breve{\mathbf{H}}$ as the feature vector $\tilde{\mathbf{U}}_i = [\breve{\mathbf{W}}, \breve{\mathbf{H}}]$ for the i-th sub-segment $\boldsymbol{b}_i$ of the original 1-D audio signal segment $\mathbf{y}$. Finally, we concatenate all $\tilde{\mathbf{U}}_i$ forming a feature vector

$$\tilde{\mathbf{U}} = \left[\tilde{\mathbf{U}}_1, \ldots, \tilde{\mathbf{U}}_Q\right] \tag{5.28}$$

for $\mathbf{y}$, where Q is the number of sub-segments $\boldsymbol{b}_i$ in $\mathbf{y}$.

5.6.3 Classification

Next, we train classifiers using $\tilde{\mathbf{E}}$ and $\tilde{\mathbf{U}}$, our extracted relevant depth video and audio features, respectively, for respiratory event classification. We train an SVM with a linear kernel, since given $\tilde{\mathbf{E}}, \tilde{\mathbf{U}} \in \mathbb{R}^{z\times 1}, z > 2000$, *i.e.*, the number of features is large, it is preferable to use linear kernel, *i.e.*, mapping data to a higher dimensional space does not improve the performance (see Appendix C in [147]). Since SVM does not include a feature selection process, we also train a feed-forward NN with sigmoid hidden neurons and softmax output neurons, to investigate if training a classifier that involves nested subset feature selection methods can improve classification performance and cost-effectiveness [148, 149] for our high-dimensional datasets. We present our classification results in Sec. 5.7.2.3.

5.7 Experimentation

5.7.1 Experimental configurations

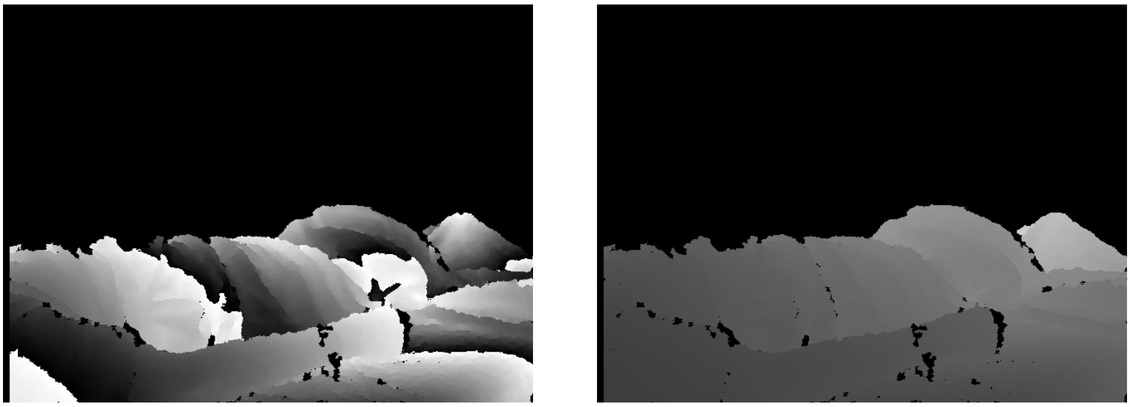
We captured a 480-minute depth video and audio for each patient with suspected sleep apnea at Concord Private Hospital in Sydney, Australia during January and February 2015. The data were collected from four consenting patients over a two-day period. The data used for training and testing SVM and NN classifiers is limited to sleep periods (including wake periods that occurred during sleep periods)—382 ± 37 minutes for each subject. Besides our depth video and audio capturing, each patient was connected to the Alice6 LDxS as used in the corresponding attended diagnostic sleep studies. The sleep studies were attended polysomnography, and the scientific officer who scored the sleep studies was blinded to our multimedia feature learning study. The data obtained from the system was manually scored according to the AASM 2007 manual [62] and the respiratory events were identified. These event labels are the ground truth data for our experiments. For a respiratory event that is of over 10-second length, we used the same segment window (10-second in length with a 5-sec overlap) as we used in the video and audio data to get data segments that have the same class as that event.

We present experimental results in the following order: depth video compression, depth video denoising, and respiratory event detection.

5.7.2 Experimental results

5.7.2.1 Depth video recording

We first validate our proposed block-based search procedure to recover the 3 uncoded MSBs in an LSB frame. We set block size to 8×8 (see Sec. 5.3). Fig. 5.7 shows an example of the decoded LSB frame and the recovered LSB frame. First, we see in Fig. 5.7(a) that due to overflows, there are discontinuities even within the same physical object. We see in the recovered LSB frame in Fig. 5.7(b) that the overflow problem is corrected, resulting in a much smoother and natural looking depth image.



(a) original LSB frame (b) recovered LSB frame

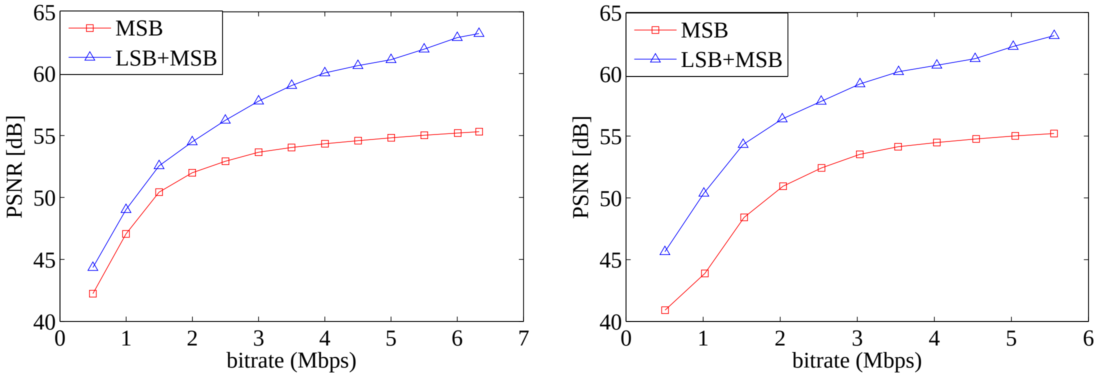
Fig. 5.7 Examples of decoded LSB frame and recovered LSB frame.

Next, we compare compression performance of our LSB-MSB coding scheme with RPS parameter $M = 5$ to the scheme that compresses only the 8 MSBs of each depth frame using the same H.264 implementation—AVC part 10 codec [127]. As a performance metric we used PSNR, calculated as:

$$\mathrm{PSNR} = 10 \log_{10} \frac{(2^{11}-1)^2 \cdot X \cdot Y}{\sum_{i=1}^{X} \sum_{j=1}^{Y} [\mathcal{X}(i,j) - \mathcal{Y}(i,j)]^2}, \tag{5.29}$$

where $\mathcal{X}$ and $\mathcal{Y}$ are two $X \times Y$ (640×480 in experiment) pixel 11-bit depth images. Uncompressed 11-bit depth images were used as ground truth, and for the 8-MSB coding scheme, three zero bits were appended to the decompressed 8-bit values.

Fig. 5.8 shows the coding performance as PSNR averaged over all frames of the two coding schemes for two sleep video sequences. The results indicate that our LSB-MSB coding scheme outperforms 8-MSB coding scheme for up to 8dB.



(a) Video sequence 1. (b) Video sequence 2.

Fig. 5.8 Compression performance for two sleep video sequences.

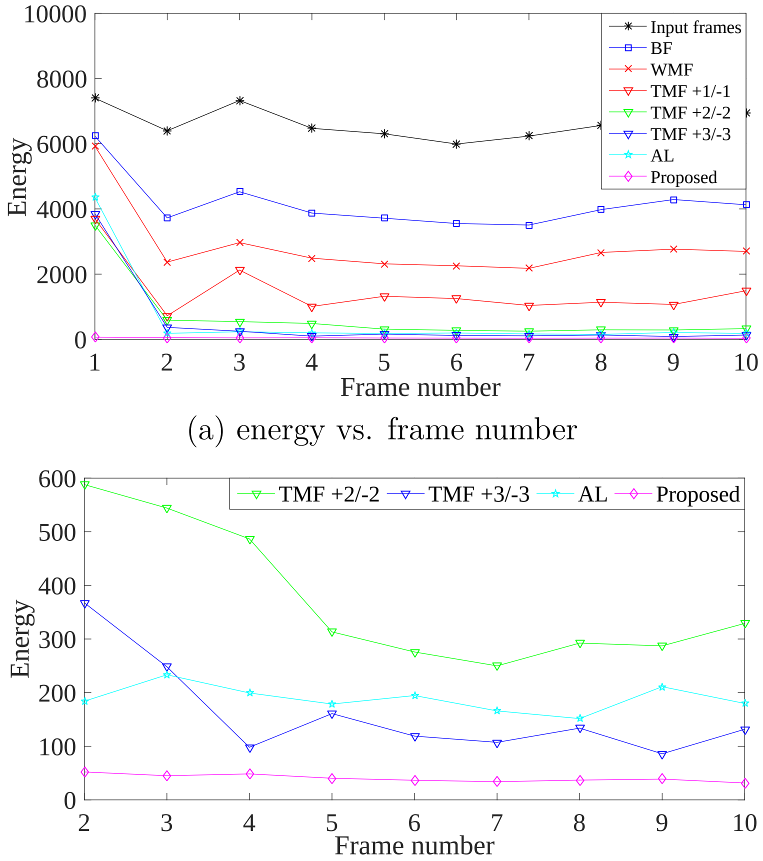
5.7.2.2 Depth video denoising

We next evaluate the performance of our proposed graph-based temporal denoising scheme in terms of flickering reduction. Table 5.1 lists the parameter settings for our denoising scheme (see Sec. 5.4). For comparison, we used the following as competing schemes. The first scheme is bilateral filtering (BF) [134] that performs spatial filtering using local neighboring pixels. We also implemented an algorithm that performs motion estimation and temporal median denoising (TMF) separately, similar to existing works such as [150]. Additionally, we performed weighted mode filtering (WMF) [151] and tested an augmented Lagrangian-based (AL) video denoising algorithm [152].

Table 5.1 Parameter settings of the proposed graph-based temporal video denoising scheme.

sign	parameter	setting
S	block size in pixels	8
Δ	thresholding for predictor-block edge	5
σ_v	target-block edge weight scaling	1
σ_g	predictor-block edge weight scaling	1
μ	weight for the fidelity term	0.1
λ	weight for the MV smoothness term	1

Fig. 5.9 shows the energy of the difference between two consecutive frames for our scheme and the competing schemes for the first 10 frames of an acquired sleep video



(a) energy vs. frame number

(b) zoomed version of (a)

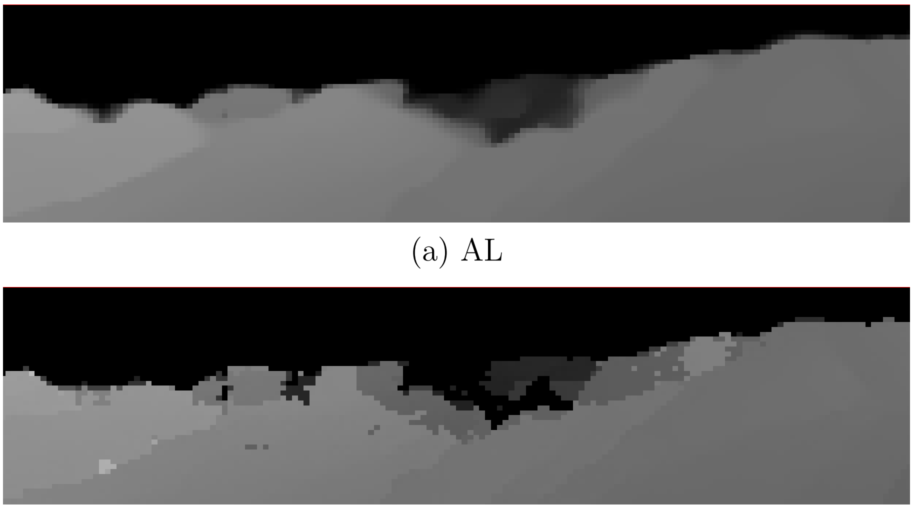
Fig. 5.9 Energy of the difference between two consecutive frames, where $+i/-i$ denotes the number of future and previous depth images used for TMF. Our scheme is lowest in frame-difference energy for each of the tested consecutive frames.

sequence. We observe that our scheme is lowest in frame-difference energy for each of the tested consecutive frames.

Fig. 5.10 shows an example of a zoomed segment of a denoised depth frame using AL [152] and our proposed denoising scheme. We observe that our scheme preserves sharp edges without over-smoothing.

5.7.2.3 Respiratory event detection

Since only 4 subjects were recruited for the experimentation, we performed intra-patient evaluation. We first performed four-class classification—i) central apnea, ii)



(a) AL

(b) proposed

Fig. 5.10 Sample segments of denoised frames by using AL and proposed scheme. Our scheme preserves sharp edges without over-smoothing.

obstructive / mixed apnea, iii) hypopnea, and iv) all the other events—using depth video features extracted from the 1-D signals based on our dual-ellipse model. For each 10-sec segment $\mathbf{x}$, we used a sub-segment size of 5-sec with 0.5-sec increments and performed WPT at $J = 5$ levels on each sub-segment. To train a four-class SVM classifier, we adopted one-against-one strategy by training six binary SVM classifiers, a competitive approach among five multi-class SVM classification methods compared in [153]. We trained a two-layer feed-forward NN with 10 sigmoid hidden neurons and 4 softmax output neurons as a competing classifier. The two-layer feed-forward neural network was trained using scaled conjugate gradient backpropagation with a Neural Pattern Recognition tool in MATLAB R2015b. We also used the variance (VAR) of the ellipse major/minor radius as hand-crafted depth video features [15, 16] for training the same classifiers.

Fig. 5.11 shows the classification error rates of inverse 5-fold cross-validation (CV) (each time using 1-fold for training and the remaining 4-folds for testing), inverse 3-fold CV, 3-fold CV, and 5-fold CV based on video features only. We see that the classifiers with WPT features significantly outperform the heuristically hand-crafted VAR features in [15, 16]. Fig. 5.12 demonstrates a 300-minute sample of a sleep patient showing the major/minor radius and the tilt of the chest/abdomen ellipse, with groundtruth-sleeping-poses marked side-by-side. One can see that our system can robustly track the patient's respiratory patterns regardless of the sleeping pose, and $\delta_{abdomen}$ shows strong correlation with the actual sleeping pose. Fig. 5.13 shows the successfully detected respiratory events using WPT depth video features during the sideway sleep period that is highlighted in Fig. 5.12. In particular, the colourised bars at the top of the figures denote the manually scored events by a scientific officer based on data collected by system Alice6 LDxS[3]; the plotted lines denote the major and minor radii of the fitted ellipses for the patient's chest and abdominal cross sections, and the colours on the plotted lines are the detected respiratory events by our learned classifier.

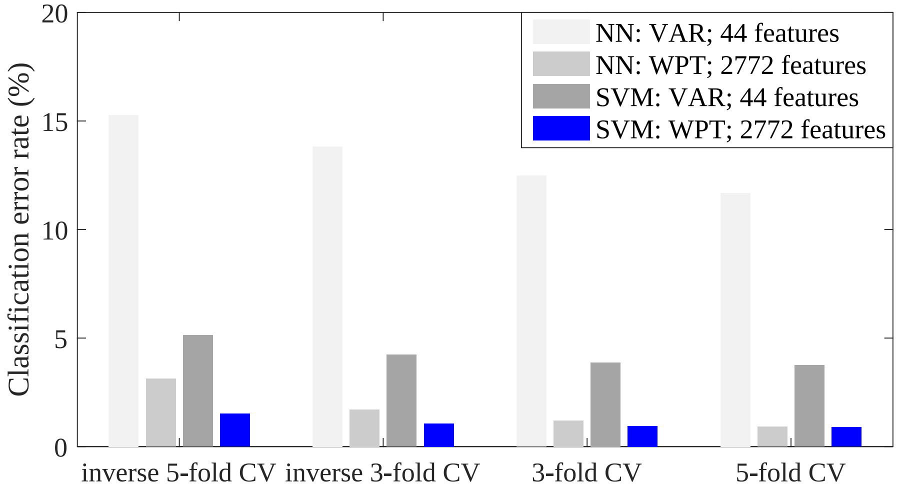


Fig. 5.11 Error rates of classification based on depth video features.

[3] In our experiments, an apneic event containing periods which fulfill hypopnea rules and do not fulfill apnea rules are treated as multiple individual events, *e.g.*, we treat an apneic event that begins with a period that fulfills hypopnea rules followed by an immediate following period that fulfill apnea rules as two individual events - a hypopnea event with an immediate following apnea event. After the individual respiratory events are correctly classified, it is straightforward to automatically combine a hypopnea event with an immediate following apnea into a single apnea event, as specified in AASM recommendations [62].

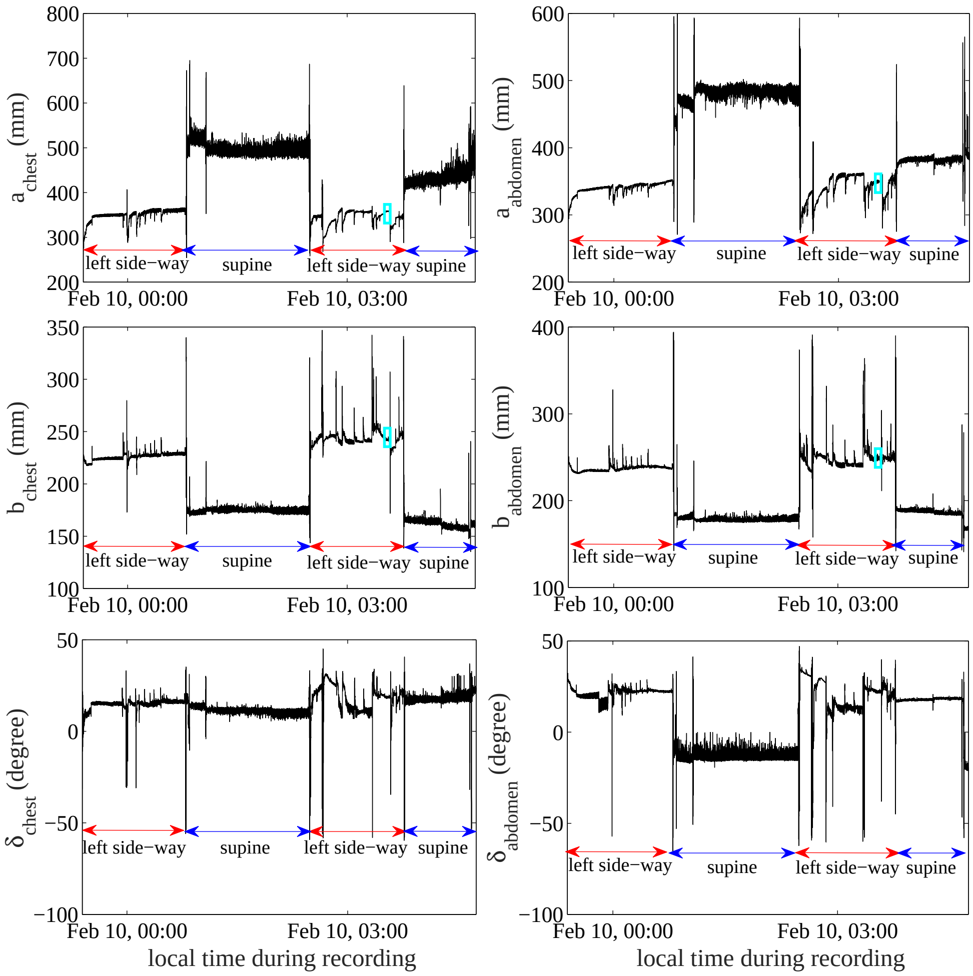


Fig. 5.12 300-minute sample of a sleep patient showing six ellipse parameters over time. a_{chest} and b_{chest} are the major and minor radii of the chest-ellipse, respectively; $a_{abdomen}$ and $b_{abdomen}$ are the major and minor radii of the abdomen-ellipse, respectively; $\delta_{abdomen}$ and δ_{chest} are the tilts of the abdomen-ellipse and the chest-ellipse, respectively.

For four-class respiratory event classification with audio features, we heuristically set the rank $k = 3$ for NMF feature extraction (see our discussion in Section 5.7.3.2). As competing feature sets we use the following two sets: i) We apply WPT at $J = 7$ levels (each 10-sec 1-D audio signal segment $\mathbf{y}$ has much more elements than $\mathbf{x}$) on each segment of $\mathbf{y}$'s and training classifiers since such biomedical audio signals also contain different types of time-frequency structures [120]. ii) We concatenate the following conventional audio features as a MIX audio feature vector and train

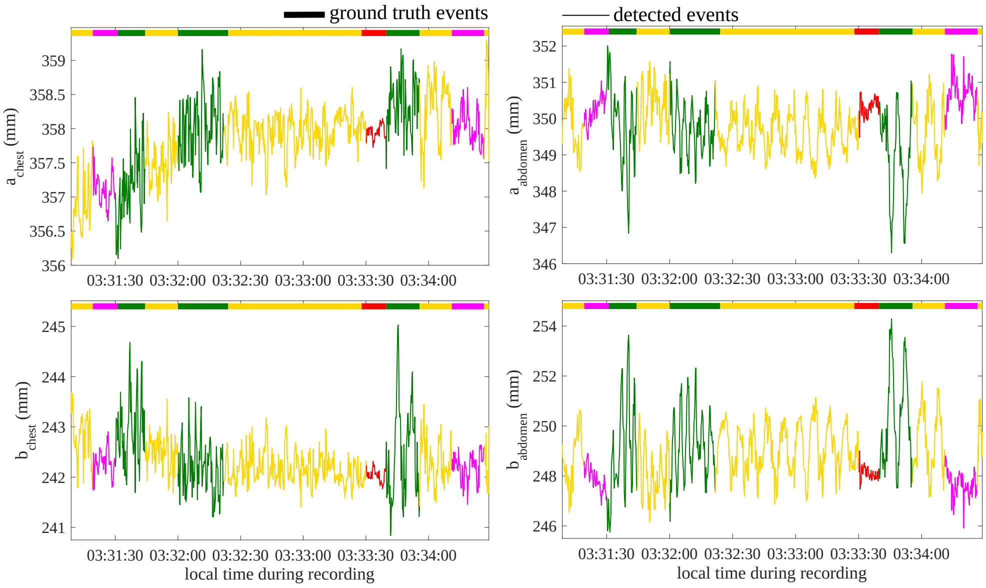


Fig. 5.13 Successfully detected events based on WPT depth video features showing $a_{chest}, b_{chest}, a_{abdomen}$ and $b_{abdomen}$ during the sideway sleep period that is highlighted in Fig. 5.12. Red: central apnea; Magenta: obstructive and mixed apnea; Yellow: hypopnea; Green: other events.

classifiers, namely, energy, energy entropy, harmonic ratio, fundamental frequency, spectral centroid, spectral entropy, spectral rolloff, spectral flux, zero crossing rate, Mel-frequency cepstral coefficients and chroma vectors [154]. Fig. 5.14 shows the classification error rates based on audio features only. Both SVM and NN classifiers trained by using NMF features show their best performance.

Finally, we train SVM and NN classifiers by combining both depth video and audio features used above. One can see in Fig. 5.15 that both classifiers perform better than using the features extracted from either of the two media, where the combination WPT+NMF shows the best performance with inverse 5-fold CV error rates of only 0.4% and 1.67%, for SVM and NN classifiers, respectively. Table 5.2 shows the inverse 5-fold CV error rates of SVM classification based on the above three sets of features: WPT depth video feature, NMF audio feature, and WPT video+NMF audio feature.

Additionally, for each class we compute *sensitivity* and *specificity* of SVM classification based on WPT video+NMF audio feature with cross-validation, which are defined

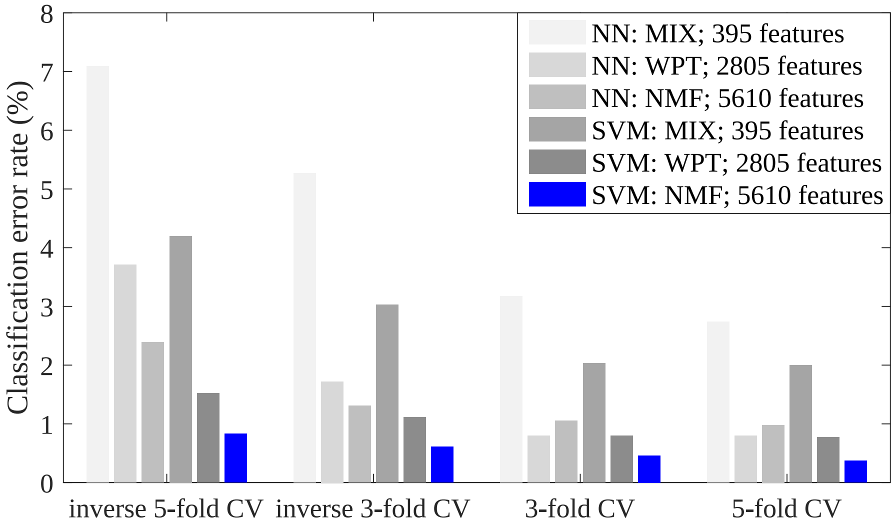


Fig. 5.14 Error rates of classification based on audio features.

as follows:

$$\text{sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}}, \tag{5.30}$$

where true positive (TP) denotes that a central apnea (resp. obstructive or mixed apnea, hypopnea and all the other events) testing sample is correctly classified, false positive (FP) denotes that a non-central apnea testing sample is incorrectly classified as central apnea, true negative (TN) denotes that a non-central apnea testing sample is correctly classified as non-central apnea, and false negative (FN) denotes that a central apnea testing sample is incorrectly classified as non-central apnea. The results are shown in Figs. 5.16 and 5.17 based on inverse 5-fold CV, inverse 3-fold CV, 3-fold CV, and 5-fold CV, respectively. The minimum sensitivity of the trained classifier is 98.2% for central apnea in inverse 5-fold CV and minimum specificity 99.76% for all the other events in 3-fold CV.

Table 5.2 Inverse 5-fold CV error rates of SVM classification based on WPT video features, NMF audio features, and the combination of them.

features	video	audio	video + audio
error rates	1.52%	0.83%	**0.4%**

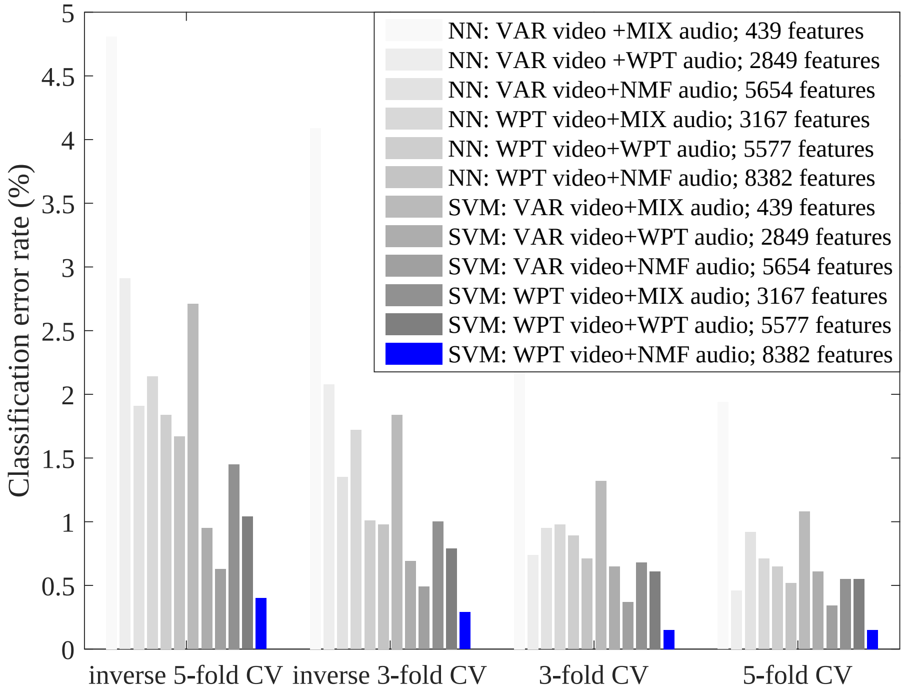


Fig. 5.15 Error rates of classification based on depth video+audio features.

5.7.3 Discussions

5.7.3.1 Depth video recording and denoising

First, our LSB-MSB depth video compression scheme outperforms 8-MSB coding scheme at the PSNR range of sufficient quality for respiratory event detection. Second, our graph-based temporal denoising scheme can more effectively reduce frame-difference energy, and thus flickering effects, over the competing schemes, even if fewer number of frames were used in the processing window than competing schemes; while our denoising scheme reduces the flickering effect, it does not over-smooth and preserves sharp edges well.

5.7.3.2 Respiratory event detection

For respiratory event detection with video features, we compared the performance of Newton's Method-based ellipse-fitting scheme [15, 16] and the proposed Bisection

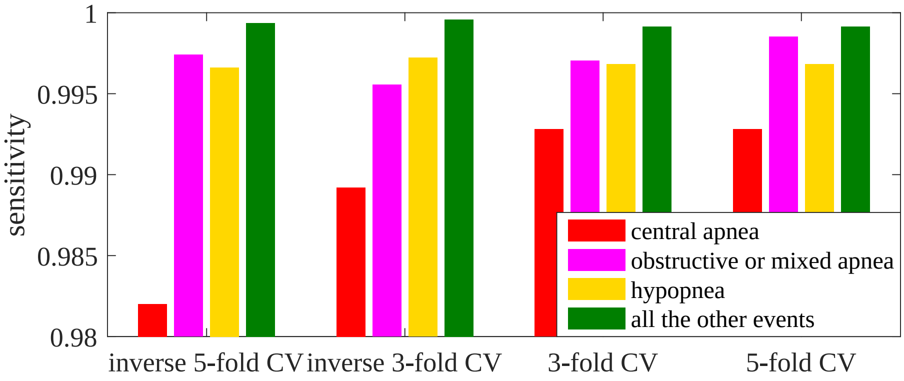


Fig. 5.16 The sensitivity of classifying different respiratory events using a trained SVM classifier based on WPT video+NMF audio feature with CV.

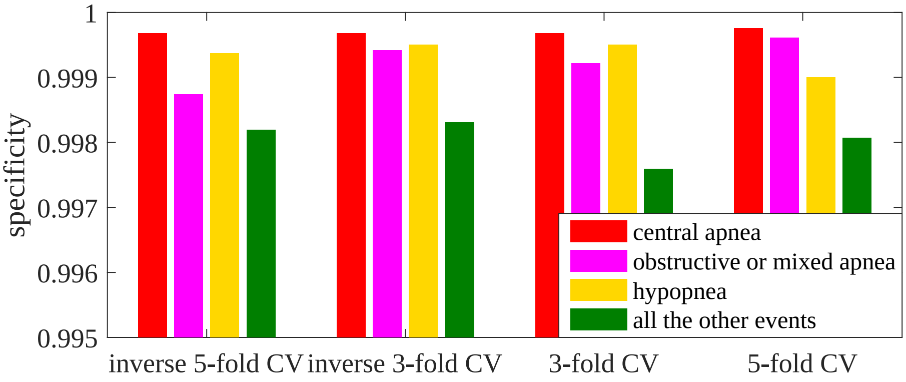


Fig. 5.17 The specificity of cross-validation classifying different respiratory events using a trained SVM classifier based on WPT video+NMF audio feature with CV.

method and Nelder-Mead simplex method-based (BSNM) scheme (Section 5.5.2) in terms of the computation speed. We ran both algorithms on 100 consecutive depth video frames in MATLAB R2014b on a Windows 10 laptop with Intel Core i7-4600U and 8GB RAM, and report that the average computation time per ellipse is 36.53s using [15, 16] and 8.68s using BSNM, *i.e.*, there is a 76% speed-up and also one can get ellipse-tilts in addition to major/minor radius, by using the new BSNM ellipse-fitting method.

Next, we built a competing dual-rectangle model and compared it to our dual-ellipse model in terms of the classification performance. Specifically, given observations $\mathbf{o}$, we found the best-fit rectangle ϱ^*, $\varrho = (\vartheta, \nu, \varphi)$, with ϑ, ν and φ denoting the length, width and the tilt that represents the body pose, using the objective that is similar to (5.18):

$$\varrho^{best} = \arg\min_{\varrho} \sum_{n=1}^{N} h_n^{-1} \left(s_{\varrho}(\mathbf{o}_n)\right)^2 . \tag{5.31}$$

We trained SVM classifiers using similar hand-crafted features on the same video clips as in [15, 16], *i.e.*, the variances of four ellipse major/minor radius for dual-ellipse model and those of four rectangle length/width for dual-rectangle model, for fair comparison. We performed binary classification (*i.e.*, Class 1: central / obstructive / mixed apnea / hypopnea; and Class 2: all the other events) with 50% data used for training and the remaining 50% for testing. The resulting confusion matrices (in the following format: [true positive, false positive; false negative, true negative]), $[50\%, 0\%; 0\%, 50\%]$ and $[10\%, 6\%; 40\%, 44\%]$, for the dual-ellipse and dual-rectangle model, respectively, show significant performance advantage of using our dual-ellipse model.

For respiratory event detection with audio features, we justify how we set the rank k for NMF feature extraction. For each 10-sec segment $\mathbf{y}$, we used the same sub-segment size (5-sec with 0.5-sec increments). For the i-th 5-sec sub-segment $\boldsymbol{b}_i$, we computed its spectrogram $\mathbf{Y}$ by STFT with 25ms STFT-window and 12.5ms increments. We first applied singular value decomposition (SVD) on all $\mathbf{Y}$'s. Fig. 5.18 shows the mean singular values of all $\mathbf{Y}$'s. One can see that the majority of the singular values are small.

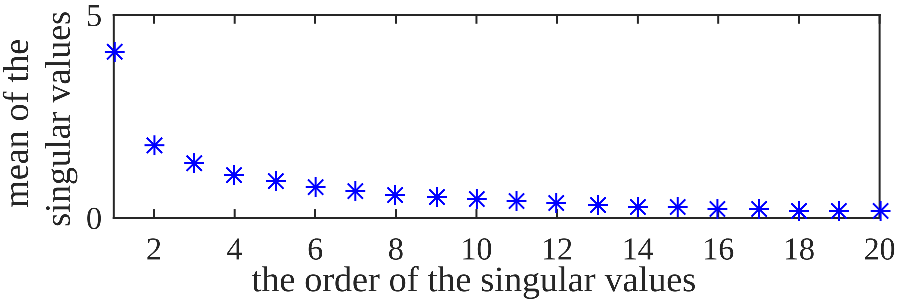


Fig. 5.18 The first 20 of the mean singular values of all $\mathbf{Y}$'s.

Table 5.3 5-fold CV error rates of SVM classification based on NMF audio features with $k = 2, 3, 4$.

sample handling	features	$k = 2$	$k = 3$	$k = 4$
with sub-segments	5610	0.46%	**0.34%**	0.4%
no sub-segments	630	0.77%	**0.65%**	1.01%

Since there is no clear dropoff between these singular values, in Table 5.3, we present the 5-fold CV error rates of SVM classification based on NMF audio features using $k = 2$, 3 and 4. Specifically, we extracted NMF features from $\mathbf{Y}$'s, trained SVM classifiers, and show the classification error rates in the row "with sub-segments" in Table 5.3; we also extracted NMF features from the spectrograms that were generated by performing STFT on each complete 10-sec $\mathbf{y}$'s and trained SVM classifiers, with the classification error rates shown in the row "no sub-segments". Given the fact that classifier always performs best at $k = 3$, we set $k = 3$ for our subsequent classification experiments. This is consistent with our initial hypothesis that the audio contains: i) background noise, ii) machine sound (*e.g.*, the cooling module of the system), and iii) human sound.

The trained classifiers with WPT video features outperforms the hand-crafted VAR features in our prior work. The classification with NMF audio features indicates that when the captured depth video is obstructed, one can still use the audio signal to detect respiratory events. Finally, the result of sensitivity and specificity for SVM classification with video-audio features reported in Figs. 5.16 and 5.17 indicates that our trained classifier has good ability to both correctly identify a central apnea (resp. obstructive or mixed apnea, hypopnea and all the other events) and correctly identify a non-central apnea, with 20% or more training data.

5.8 Summary

Existing sleep monitoring systems are expensive and intrusive enough that they negatively affect the quality of a patient's sleep. In this chapter, we propose to record audio and depth video of a patient using a Microsoft Kinect camera during his/her sleep, so that relevant features can be extracted non-intrusively for detection of different respiratory events. Our proposal contains three parts. First, we propose an efficient H.264 video coding scheme, where the captured 11-bit video can be reliably recovered at the decoder even though the compressed video is first converted to 8-bit. Second, we propose a graph-based depth video denoising algorithm, so that undesirable flicker can be removed without over-smoothing. Third, we propose a dual ellipse model to track the patient's chest and abdominal movements given captured depth pixels. When ellipse features are combined with audio features, different respiratory events, as scored manually based in data collected by a medical sleep monitoring device, can reliably be detected.

Chapter 6

Conclusions

Emerging home-use healthcare techniques greatly help patients self-manage their health through easy-to-use condition assessment applications in-home. However, most of current home-use healthcare systems still require significant amount of manual effort or lack benchmarks to demonstrate their clinical effectiveness. In this thesis, we advance home-use healthcare applications in three aspects via: 1) autonomous gait analysis with gait event detection, 2) autonomous post-stroke recovery level assessment via upper limb motion analysis, and 3) non-intrusive sleep monitoring.

Specifically, the following new application-driven algorithms are proposed to achieve this. The first targeted application on gait analysis in Chapter 2 consists of: 1) histogram-based algorithm for autonomous frame-of-interest detection, DKF-SSIM lower-limb-joint individual-marker tracking (*i.e.*, separately tracking the required hip, knee and ankle joints) and gait event detection, and 2) performance comparison of the proposed tracking scheme in colour and grayscale video sequences.

Next, the second targeted application on post-stroke recovery level classification in Chapter 3 includes: 1) simultaneous DKF-SSIM upper-limb-joint tracking, and 2) classification of the stroke recovery level by minimization of graph total variation with graph-based signal processing, where the ground truth labels were marked by a biomechanics expert.

Then, the third targeted application on abnormal respiratory event detection during sleep in Chapter 4 consists of 1) an alternating-frame video coding scheme for H.264 video coding. 2) temporal denoising on the decoded depth video using a motion vector graph smoothness prior in order to remove undesirable flickering while retaining sharp edges, 3) tracking patient's chest and abdominal movements based on a dual-ellipse model, and 4) extracting ellipse model features via a WPT and audio features via NMF for abnormal respiratory event classification.

However, there still remains several technical challenges in portable home-use health monitoring applications:

- comprehensive limb motion analysis via body joint tracking without parallax error.

To tackle this problem, the future work will be focused on further improvement of data accuracy and measurement capability of more limb motion parameters using a stereo 2D-camera system or a single depth sensing device [20, 59, 155–158], without sacrificing portability, to remove the parallax error, and leverage the 3D information for quantifying a larger number of limb motion parameters such as hip, knee, and ankle angles in both the sagittal and frontal planes, and pelvis tilt, calculating temporal-spatial parameters, and measuring sagittal/frontal plane knee motion, step length and width, gait speed and step length symmetry, as well as spinal elongation/shrinkage, but at an increased processing complexity.

- completely non-intrusive and computationally efficient body joint tracking for maximal-comfort limb motion analysis, *i.e.*, marker-less body joint tracking.

Most of state of the art body joint tracking algorithms still lack tracking stability or require large dataset for body joint classifier training [7]. Future work will investigate signal processing-based body joint estimation methods, *e.g.*, articulated Gaussian Kernel correlation [159], as candidate schemes, with a focus on improvement of accuracy of body joint estimation and tracking stability. The outcome can be used not only in gait and upper limb motion analysis, but also remote full body motion analysis during quiet stance [160] that has been used for an individual's functional independence measure [161].

- computationally efficient respiratory movement tracking for sleep monitoring.

We note that, our sleep monitoring system requires large storage for data recording, it is relatively slow in fitting of the dual-ellipse respiratory model and person-specific classifier training for each human subject. Using large amount of collected data, future work would focus on developing more efficient and less complex model fitting methods and feature extraction for training classifiers that are generally applicable to different subjects.

- comprehensive, contact-less sleep quality assessment.

Despite the proposed, clinically validated sleep monitoring system is able to detect abnormal respiratory events during a person's sleep, this system does not reveal anything about sleep stages or dream types [162] which are important for sleep quality assessment and sleep disorder diagnostics. It has been shown that the patterns in Electroencephalography (EEG) signal[1] during a person's sleep are correlated with sleep stages [163]. In order to retain the 'non-intrusive' feature for an EEG-aided sleep analysis system, the EEG signals would be collected by installing EEG sensors onto the pillow, which brings the following challenges: 1) potential significant acquisition noise, and 2) incomplete observations of brain activities due to head movement of a sleep subject throughout the night. Future work would first investigate graph signal processing (applied in Section 4.2.3, Chapters 3 and Section 5.4, Chapter 4 for classification and image denoising, respectively) as an emerging tool for signal processing, as well as a robust alternative to established machine learning-based

[1] EEG indicates brain electrical activity.

classification approaches. The developed generic methods will then be tailored and applied to address these challenges.

In summary, this thesis aims to attract more research interest in cost-effective home-use healthcare, with the objective of not only monitoring people's health condition, but also providing feedback and helping to improve people's physical and psychological well-being.

Bibliography

[1] C. Perera, C. H. Liu, and S. Jayawardena, "The emerging internet of things marketplace from an industrial perspective: A survey," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 4, pp. 585–598, Dec 2015.

[2] F. Erden, S. Velipasalar, A. Z. Alkar, and A. E. Cetin, "Sensors in assisted living: A survey of signal and image processing methods," *IEEE Signal Processing Magazine*, vol. 33, no. 2, pp. 36–44, 2016.

[3] M. F. Romano, M. V. Sardella, and F. Alboni, "Web health monitoring survey: A new approach to enhance the effectiveness of telemedicine systems," *JMIR Research Protocols*, vol. 5, no. 2, p. e101, Jun. 2016.

[4] J. G. Richards, "The measurement of human motion: A comparison of commercially available systems," *Human Movement Science*, vol. 18, no. 5, pp. 589–602, 1999.

[5] A. Buke, F. Gaoli, W. Yongcai, S. Lei, and Y. Zhiqi, "Healthcare algorithms by wearable inertial sensors: a survey," *China Communications*, vol. 12, no. 4, pp. 1–12, April 2015.

[6] C.-C. Yang and Y.-L. Hsu, "A review of accelerometry-based wearable motion detectors for physical activity monitoring," *Sensors*, vol. 10, no. 8, pp. 7772–7788, 2010.

[7] F. Han, B. Reily, W. Hoff, and H. Zhang, "Space-time representation of people based on 3d skeletal data: A review," *CoRR*, vol. abs/1601.01006, 2016. [Online]. Available: http://arxiv.org/abs/1601.01006

[8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering-Transactions of the ASME*, vol. 82, no. 1, pp. 35–45, 1960.

[9] G. Welch and G. Bishop, "An introduction to the kalman filter," Chapel Hill, NC, USA, Tech. Rep., 1995.

[10] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004.

[11] C. Yang, U. C. Ugbolue, B. Carse, V. Stankovic, L. Stankovic, and P. J. Rowe, "Multiple marker tracking in a single-camera system for gait analysis," in *IEEE International Conference on Image Processing*, September 2013, pp. 3128–3131.

[12] C. Yang, U. C. Ugbolue, A. Kerr, V. Stankovic, L. Stankovic, B. Carse, K. T. Kaliarntas, and P. J. Rowe, “Autonomous gait event detection with portable single-camera gait kinematics analysis system,” *Journal of Sensors*, vol. 2016, Jan. 2016.

[13] C. Yang, A. Kerr, V. Stankovic, L. Stankovic, and P. Rowe, “Upper limb movement analysis via marker tracking with a single-camera system,” in *IEEE International Conference on Image Processing*, Paris, France, Oct. 2014.

[14] C. Yang, A. Kerr, V. Stankovic, L. Stankovic, P. Rowe, and S. Cheng, “Human upper limb motion analysis for post-stroke impairment assessment using video analytics,” *IEEE Access*, vol. 4, pp. 650–659, Jan. 2016.

[15] C. Yang, G. Cheung, K. Chan, and V. Stankovic, “Sleep monitoring via depth video recording & analysis,” in *IEEE International Workshop on Hot Topics in 3D*, Chengdu, China, Jul. 2014.

[16] C. Yang, Y. Mao, G. Cheung, V. Stankovic, and K. Chan, “Graph-based depth video denoising and event detection for sleep monitoring,” in *IEEE International Workshop on Multimedia Signal Processing*, Jakarta, Indonesia, Sept. 2014.

[17] C. Yang, G. Cheung, V. Stankovic, K. Chan, and N. Ono, “Sleep apnea detection via depth video & audio feature learning,” *IEEE Transactions on Multimedia*, in press.

[18] R. L. Sacco et al., “An updated definition of stroke for the 21st century: A statement for healthcare professionals from the american heart association/american stroke association,” *Stroke*, vol. 44, no. 7, pp. 2064–2089, 2013.

[19] P. Langhorne, F. Coupar, and A. Pollock, “Motor recovery after stroke: a systematic review,” *The Lancet Neurology*, vol. 8, no. 8, pp. 741–754, 2009.

[20] A. Muro-de-la Herran, B. Garcia-Zapirain, and A. Mendez-Zorrilla, “Gait analysis methods: An overview of wearable and non-wearable systems, highlighting clinical applications,” *Sensors*, vol. 14, no. 2, pp. 3362–3394, 2014.

[21] B. Toro, C. Nester, and P. Farren, “A review of observational gait assessment in clinical practice,” *Physiotherapy Theory and Practice*, vol. 19, pp. 137–149, 2003.

[22] F. Ferrarello, V. A. M. Bianchi, M. Baccini, G. Rubbieri, E. Mossello, M. C. Cavallini, N. Marchionni, and M. D. Bari, “Tools for observational gait analysis in patients with stroke: a systematic review,” *Physical Therapy*, vol. 93, no. 12, pp. 1673–1685, 2013.

[23] G. Li, T. Liu, J. Yi, H. Wang, J. Li, and Y. Inoue, “The lower limbs kinematics analysis by wearable sensor shoes,” *IEEE Sensors Journal*, vol. 16, no. 8, pp. 2627–2638, April 2016.

[24] M. U. B. Altaf, T. Butko, and B. H. Juang, “Acoustic gaits: Gait analysis with footstep sounds,” *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 8, pp. 2001–2011, Aug 2015.

[25] S. Hagler, D. Austin, T. L. Hayes, J. Kaye, and M. Pavel, “Unobtrusive and ubiquitous in-home monitoring: a methodology for continuous assessment of gait velocity in elders,” *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 4, pp. 813–820, 2010.

[26] T. Johansson and C. Wild, “Telerehabilitation in stroke care — a systematic review,” *Journal of Telemedicine and Telecare*, vol. 17, no. 1, pp. 1–6, 2011.

[27] K. E. Laver, D. Schoene, M. Crotty, S. George, N. A. Lannin, and C. Sherrington, “Telerehabilitation services for stroke,” *Cochrane Database of Systematic Reviews*, vol. 12, p. CD010255, 2013.

[28] D. Theodoros and T. Russell, “Telerehabilitation: Current perspectives,” *Studies in Health Technology and Informatics*, vol. 131, pp. 191–209, 2008.

[29] D. M. Brennan, S. Mawson, and S. Brownsell, “Telerehabilitation: Enabling the remote delivery of healthcare, rehabilitation, and self management,” *Studies in Health Technology and Informatics*, vol. 145, pp. 231–248, 2009.

[30] P. Gregory, J. Alexander, and J. Satinsky, “Clinical telerehabilitation: Applications for physiatrists,” *PM&R: The journal of injury, function, and rehabilitation*, vol. 3, no. 7, pp. 647–656, 2011.

[31] P. Soda, A. Carta, D. Formica, and E. Guglielmelli, “A low-cost video-based tool for clinical gait analysis,” in *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, Minneapolis, MN, Sept 2009, pp. 3979–3982.

[32] U. C. Ugbolue, E. Papi, K. T. Kaliarntas, A. Kerr, L. Earl, V. M. Pomeroy, and P. J. Rowe, “The evaluation of an inexpensive, 2D, video based gait assessment system for clinical use,” *Gait & Posture*, vol. 38, no. 3, pp. 483–489, 2013.

[33] J. C. Wall, J. Devlin, R. Khirchof, and B. Lackey, “Measurement of step widths and step lengths: a comparison of measurements made directly from a grid with those made from a video recording,” *Journal of Orthopaedic & Sports Physical Therapy*, vol. 30, no. 7, pp. 410–417, 2000.

[34] D. A. McDonald, J. Q. Delgadillo, M. Fredericson, J. McConnell, M. Hodgins, and T. F. Besier, “Reliability and accuracy of a video analysis protocol to assess core ability,” *PM&R: The journal of injury, function, and rehabilitation*, vol. 3, no. 3, pp. 204–211, 2011.

[35] S. Richardson, A. Cooper, G. Alghamdi, M. Alghamdi, and A. Altowaijri, “Assessing knee hyperextension in patients after stroke: comparing clinical observation and Siliconcoach software,” *International Journal of Therapy and Rehabilitation*, vol. 19, no. 3, pp. 163–168, 2012.

[36] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” *ACM Computing Surveys*, vol. 38, no. 4, p. 13, 2006.

[37] Y. Wu, J. Lim, and M.-H. Yang, “Online object tracking: A benchmark,” in *IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, June 2013, pp. 2411–2418.

[38] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, “A survey of appearance models in visual object tracking,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 4, no. 4, p. 58, 2013.

[39] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, “Visual tracking: An experimental survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, 2014.

[40] T. J. Broida and R. Chellappa, “Estimation of object motion parameters from noisy images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 1, pp. 90–99, 1986.

[41] A. Yilmaz, X. Li, and M. Shah, “Contour-based object tracking with occlusion handling in video acquired using mobile cameras,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1531–1536, 2004.

[42] J. Ning, L. Zhang, D. Zhang, and C. Wu, “Robust object tracking using joint color-texture histogram,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 7, pp. 1245–1263, 2009.

[43] Z. Kalal, K. Mikolajczyk, and J. Matas, “Tracking-learning-detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[44] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M. M. Cheng, S. L. Hicks, and P. H. S. Torr, “Struck: Structured output tracking with kernels,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2096–2109, Oct. 2016.

[45] S. Zhang, X. Yu, Y. Sui, S. Zhao, and L. Zhang, “Object tracking with multi-view support vector machines,” *IEEE Transactions on Multimedia*, vol. 17, no. 3, pp. 265–278, 2015.

[46] Y. Yuan, H. Yang, Y. Fang, and W. Lin, “Visual object tracking by structure complexity coefficients,” *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1125–1136, 2015.

[47] X. Zhang, W. Hu, W. Qu, and S. Maybank, “Multiple object tracking via species-based particle swarm optimization,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 11, pp. 1590–1602, 2010.

[48] Z. H. Khan, I. Y.-H. Gu, and A. G. Backhouse, “Robust visual object tracking using multi-mode anisotropic mean shift and particle filters,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 1, pp. 74–87, 2011.

[49] B. H. Dobkin, “Rehabilitation after stroke,” *New England Journal of Medicine*, vol. 352, no. 16, pp. 1677–1684, 2005.

[50] H. M. Clayton and H. C. Schamhardt, “Measurement techniques for gait analysis,” *Equine locomotion*, pp. 55–76, 2001.

[51] M. Kelly-Hayes, J. T. Robertson, J. P. Broderick, P. W. Duncan, L. A. Hershey, E. J. Roth, W. H. Thies, C. A. Trombly *et al.*, “The american heart association stroke outcome classification: executive summary,” *Circulation*, vol. 97, no. 24, pp. 2474–2478, 1998.

[52] D. I. Shuman et al., “The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains,” in *IEEE Signal Processing Magazine*, vol. 30, no. 3, May 2013, pp. 83–98.

[53] J. Xu, V. Jagadeesh, Z. Ni, S. Sunderrajan, and B. Manjunath, “Graph-based topic-focused retrieval in distributed camera network,” *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 2046–2057, 2013.

[54] B. Macchiavello, C. Dorea, E. M. Hung, G. Cheung, and W.-T. Tan, “Loss-resilient coding of texture and depth for free-viewpoint video conferencing,” *IEEE Transactions on Multimedia*, vol. 16, no. 3, pp. 711–725, 2014.

[55] A. Sandryhaila and J. M. Moura, “Classification via regularization on graphs.” in *GlobalSIP*, 2013, pp. 495–498.

[56] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[57] C. J. Burges, “A tutorial on support vector machines for pattern recognition,” *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[58] A. Sandryhaila and J. M. Moura, “Discrete signal processing on graphs: Frequency analysis,” *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3042–3054, 2014.

[59] M. Ye, C. Yang, V. Stankovic, L. Stankovic, and A. Kerr, “A depth camera motion analysis framework for tele-rehabilitation: Motion capture and person-centric kinematics analysis,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 5, pp. 877–887, Aug. 2016.

[60] A. Malhotra and D. P. White, “Obstructive sleep apnoea,” *The Lancet*, vol. 360, no. 9328, pp. 237–245, Jul. 2002.

[61] P. Peppard et al., “Prospective study of the association between sleep-disordered breathing and hypertension,” *The New England Journal of Medicine*, vol. 342, no. 19, pp. 1378–1384, May 2000.

[62] C. Iber et al., *The AASM Manual for the Scoring of Sleep and Associated Events.* American Academy of Sleep Medicine, 2007.

[63] J. Behar et al., “A review of current sleep screening applications for smartphones,” *Physiological Measurement*, vol. 34, no. 7, pp. R29–R46, Jun. 2013.

[64] D. S. Avalur, "Human breath detection using a microphone," Master's thesis, Faculty of Mathematics and Natural Sciences, University of Groningen, Aug. 2013.

[65] Z. Chen et al., "Unobtrusive sleep monitoring using smartphones," in *International Conference on Pervasive Computing Technologies for Healthcare and Workshops*, Venice, Italy, May 2013.

[66] N. Oliver and F. Flores-Mangas, "Healthgear: Automatic sleep apnea detection and monitoring with a mobile phone," *Journal of Communications*, vol. 2, no. 2, Mar. 2007.

[67] J. Behar et al., "SleepAp: An automated obstructive sleep apnoea screening application for smartphones," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 325–331, Jan. 2015.

[68] L. Jiang et al., "Automatic sleep monitoring system for home healthcare," in *IEEE-EMBS International Conference on Biomedical and Health Informatics*, Jan. 2012.

[69] D. C. Mack et al., "Development and preliminary validation of heart rate and breathing rate detection using a passive, ballistocardiography-based sleep monitoring system," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 1, pp. 111–120, Jan. 2009.

[70] K. Malakuti and A. Albu, "Towards an intelligent bed sensor: Non-intrusive monitoring of sleep irregularities with computer vision techniques," in *International Conference on Pattern Recognition*, Istanbul, Turkey, Aug. 2010.

[71] J. Paalasmaa et al., "Unobtrusive online monitoring of sleep at home," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug. 2012.

[72] N. Patwari et al., "Monitoring breathing via signal strength in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 8, pp. 1774–1786, Aug. 2014.

[73] M. Martinez et al., "Breath rate monitoring during sleep using near-IR imagery and PCA," in *International Conference on Pattern Recognition*, Tsukuba, Japan, Nov. 2012.

[74] A. Loblaw et al., "Remote respiratory sensing with an infrared camera using the Kinect^{TM} infrared projector," in *World Congress in Computer Science, Computer Engineering, & Applied Computing*, 2013.

[75] B. Krüger et al., "Sleep detection using de-identified depth data," *Journal of Mobile Multimedia*, vol. 10, no. 3&4, pp. 327–342, Dec. 2014.

[76] D. Falie et al., "Respiratory motion visualization and the sleep apnea diagnosis with the time of flight (ToF) camera," in *WSEAS International Conference on Visualization, Imaging and Simulation*, Bucharest, Romania, Nov. 2008.

[77] M.-C. Yu et al., “Multiparameter sleep monitoring using a depth camera,” in *Biomedical Engineering Systems and Technologies*, J. Gabriel et al., Ed. Springer, 2013, vol. 357, pp. 311–325.

[78] C.-W. Wang et al, “Unconstrained video monitoring of breathing behavior and application to diagnosis of sleep apnea,” *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 2, pp. 396–404, Feb. 2014.

[79] M. W. Lee and R. Nevatia, “Body part detection for human pose estimation and tracking,” in *IEEE Workshop on Motion and Video Computing*, Austin, TX, Feb. 2007.

[80] J. Shotton et al., “Real-time human pose recognition in parts from single depth images,” in *IEEE Conference on Computer Vision and Pattern Recognition*, Collorado Springs, CO, Jun. 2011.

[81] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, and A. Blake, “Efficient human pose estimation from single depth images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2821–2840, Dec. 2013.

[82] J. Yu, Y. Rui, Y. Y. Tang, and D. Tao, “High-order distance-based multiview stochastic learning in image classification,” *IEEE Transactions on Cybernetics*, vol. 44, no. 12, pp. 2431–2442, Dec. 2014.

[83] J. Yu, D. Tao, J. Li, and J. Cheng, “Semantic preserving distance metric learning and applications,” *Information Sciences*, vol. 281, pp. 674 – 686, Oct. 2014.

[84] M. Madadi et al., “Multi-part body segmentation based on depth maps for soft biometry analysis,” *Pattern Recognition Letters*, vol. 56, pp. 14–21, Apr. 2015.

[85] V. Metsis et al., “Non-invasive analysis of sleep patterns via multimodal sensor input,” *Personal and Ubiquitous Computing*, vol. 18, no. 1, pp. 19–26, Jan. 2014.

[86] L.-C.-L. Chen et al., “A sleep monitoring system based on audio, video and depth information for detecting sleep events,” in *IEEE International Conference on Multimedia & Expo*, Chengdu, China, Jul. 2014.

[87] J. Lee et al, “Sleep monitoring system using kinect sensor,” *International Journal of Distributed Sensor Networks*, vol. 2015, Apr. 2015.

[88] F. Centonze et al., “Feature extraction using ms kinect and data fusion in analysis of sleep disorders,” in *International Workshop on Computational Intelligence for Multimedia Understanding*, Prague, Czech Republic, Oct. 2015.

[89] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, Nov 2000.

[90] V. M. Pomeroy, P. J. Rowe, J.-C. Baron, A. Clark, R. Sealy, U. C. Ugbolue, A. Kerr, and S. C. Investigators, "The SWIFT Cast trial protocol: A randomized controlled evaluation of the efficacy of an ankle-foot cast on walking recovery early after stroke and the neural-biomechanical correlates of response," *International Journal of Stroke*, vol. 7, no. 1, pp. 86–93, 2012.

[91] E. Papi, U. C. Ugbolue, S. Solomonidis, and P. J. Rowe, "Comparative study of a newly cluster based method for gait analysis and plug-in gait protocol," *Gait & Posture*, vol. 39, no. Supplement 1, pp. S9–S10, 2014.

[92] R. B. Davis III, S. Ounpuu, D. Tyburski, and J. R. Gage, "A gait analysis data collection and reduction technique," *Human Movement Science*, vol. 10, no. 5, pp. 575–587, 1991.

[93] A. S. Rahmathullah, R. Selvan, and L. Svensson, "A batch algorithm for estimating trajectories of point targets using expectation maximization," *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4792–4804, Sept 2016.

[94] D. C. Kerrigan, M. Schaufele, and M. N. Wen, "Gait analysis," in *Rehabilitation Medicine: Principles and Practice. 3rd ed.*, J. A. Delisa and B. M. Gans, Eds. Philadelphia: Lippincott-Raven Publishers, 1998, pp. 167–187.

[95] C. M. O'Connor, S. K. Thorpe, M. J. O'Malley, and C. L. Vaughan, "Automatic detection of gait events using kinematic data," *Gait & Posture*, vol. 25, no. 3, pp. 469–474, 2007.

[96] J. A. Zeni Jr., J. G. Richards, and J. S. Higginson, "Two simple methods for determining gait events during treadmill and overground walking using kinematic data," *Gait & Posture*, vol. 27, no. 4, pp. 710–714, 2008.

[97] M. E. Harrington, A. B. Zavatsky, S. E. M. Lawson, Z. Yuan, and T. N. Theologis, "Prediction of the hip joint centre in adults, children, and patients with cerebral palsy based on magnetic resonance imaging," *Journal of Biomechanics*, vol. 40, no. 3, pp. 595–602, 2007.

[98] M. Sangeux, H. Pillet, and W. Skalli, "Which method of hip joint centre localisation should be used in gait analysis?" *Gait & Posture*, vol. 40, no. 1, pp. 20–25, 2014.

[99] B. F. Mazuquin, Batista JP Jr., L. M. Pereira, J. M. Dias, M. F. Silva, R. L. Carregaro, P. R. Lucareli, F. A. Moura, and J. R. Cardoso, "Kinematic gait analysis using inertial sensors with subjects after stroke in two different arteries," *Journal of Physical Therapy Science*, vol. 26, no. 8, pp. 1307–1311, August 2014.

[100] C. L. Chen, H. C. Chen, S. F. Tang, C. Y. Wu, P. T. Cheng, and W. H. Hong, "Gait performance with compensatory adaptations in stroke patients with different degrees of motor recovery," *American Journal of Physical Medicine & Rehabilitation*, vol. 82, no. 12, pp. 925–35, December 2003.

[101] J. Chaler, B. Müller, A. Maiques, and E. Pujol, "Suspected feigned knee extensor weakness: Usefulness of 3d gait analysis. case report," *Gait & Posture*, vol. 32, no. 3, pp. 354 – 357, 2010.

[102] J. L. McGinley, R. Baker, R. Wolfe, and M. E. Morris, “The reliability of three-dimensional kinematic gait measurements: a systematic review,” *Gait & Posture*, vol. 29, no. 3, pp. 360–369, 2009.

[103] C. B. Meadows, “The influence of polypropylene ankle-foot orthoses on the gait of cerebral palsied children,” Ph.D. dissertation, University of Strathclyde, 1984.

[104] B. Carse, R. Bowers, B. C. Meadows, and P. Rowe, “The immediate effects of fitting and tuning solid ankle–foot orthoses in early stroke rehabilitation,” *Prosthetics and Orthotics International*, vol. 39, no. 6, pp. 454–462, December 2015.

[105] E. Owen, “The importance of being earnest about shank and thigh kinematics especially when using ankle-foot orthoses,” *Prosthetics and Orthotics International*, vol. 34, no. 3, pp. 254–269, 2010.

[106] J. M. Bland and D. G. Altman, “Statistical methods for assessing agreement between two methods of clinical measurement,” *The Lancet*, vol. 327, no. 8476, pp. 307–310, 1986.

[107] S. Hare, A. Saffari, and P. H. Torr, “Struck: Structured output tracking with kernels,” in *IEEE International Conference on Computer Vision*, Barcelona, Spain, Nov. 2011, pp. 263–270.

[108] A. J. Turton, P. Cunningham, E. Heron, F. van Wijck, C. Sackley, C. Rogers, K. Wheatley, S. Jowett, S. L. Wolf, and P. van Vliet, “Home-based reach-to-grasp training for people after stroke: study protocol for a feasibility randomized controlled trial,” *Trials*, vol. 14, no. 1, p. 1, 2013.

[109] J. H. Friedman, “Another approach to polychotomous classification,” Department of Statistics, Stanford University, Tech. Rep., 1996. [Online]. Available: http://www-stat.stanford.edu/~jhf/ftp/poly.ps.Z

[110] J. J. Moré, “The levenberg-marquardt algorithm: implementation and theory,” in *Numerical analysis.* Springer, 1978, pp. 105–116.

[111] J. G. Richards, “The measurement of human motion: A comparison of commercially available systems,” *Human Movement Science*, vol. 18, no. 5, pp. 589 – 602, 1999.

[112] J. Bernhardt, P. J. Bate, and T. A. Matyas, “Accuracy of observational kinematic assessment of upper-limb movements,” *Physical therapy*, vol. 78, no. 3, pp. 259–270, 1998.

[113] X. Zhu, Z. Ghahramani, and J. Lafferty.

[114] U. H.-G. Kreßel, “Pairwise classification and support vector machines,” in *Advances in kernel methods.* MIT Press, 1999, pp. 255–268.

[115] C.-C. Chang and C.-J. Lin, “Libsvm: a library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[116] C.-W. Hsu and C.-J. Lin, “A comparison of methods for multiclass support vector machines,” *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, 2002.

[117] R. Kohavi *et al.*, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *International Joint Conference on Artificial Intelligence*, vol. 14, no. 2, 1995, pp. 1137–1145.

[118] K. C. Pohlmann, *Principles of Digital Audio*, 6th ed. McGraw-Hill Professional, 2010.

[119] R. R. Coifman, Y. Meyer, and V. Wickerhauser, “Wavelet analysis and signal processing,” in *Wavelets and Their Applications*, M. B. Ruskai, Ed. Boston: Jones and Barlett, 1992, pp. 153–178.

[120] S. Mallat, *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way.* Academic Press, 2008.

[121] P. Paatero and U. Tapper, “Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values,” *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.

[122] D. D. Lee et al., “Learning the parts of objects by non-negative matrix factorization,” *Nature*, vol. 401, pp. 788–791, Oct. 1999.

[123] D. D. Lee et al., “Algorithms for non-negative matrix factorization,” in *Annual Conference on Neural Information Processing Systems*, T. K. Leen et al., Ed. MIT Press, 2001, pp. 556–562.

[124] M. W. Berry et al., “Algorithms and applications for approximate nonnegative matrix factorization,” *Computational Statistics and Data Analysis*, vol. 52, no. 1, pp. 155–173, 2007.

[125] I. Daubechies, *Ten Lectures on Wavelets.* Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1992.

[126] R. R. Coifman et al., “Entropy-based algorithms for best basis selection,” *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 713–718, Mar. 1992.

[127] T. Wiegand et al., “Overview of the H.264/AVC video coding standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[128] Y.-W. Huang et al., “Survey on block matching motion estimation algorithms and architectures with new results,” *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 42, pp. 297–320, Mar. 2006.

[129] W. Hu, X. Li, G. Cheung, and O. Au, “Depth map denoising using graph-based transform and group sparsity,” in *IEEE International Workshop on Multimedia Signal Processing*, Pula, Italy, Oct. 2013.

[130] W. Hu, G. Cheung, X. Li, and O. Au, “Depth map compression using multi-resolution graph-based transform for depth-image-based rendering,” in *IEEE International Conference on Image Processing*, Orlando, FL, Sept. 2012.

[131] W. Hu, G. Cheung, A. Ortega, and O. Au, “Multi-resolution graph Fourier transform for compression of piecewise smooth images,” in *IEEE Transactions on Image Processing*, vol. 24, no. 1, Jan. 2015, pp. 419–433.

[132] J. Pang, G. Cheung, W. Hu, and O. C. Au, “Redefining self-similarity in natural images for denoising using graph signal gradient,” in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, Siem Reap, Cambodia, Dec. 2014.

[133] J. Pang, G. Cheung, A. Ortega, and O. C. Au, “Optimal graph Laplacian regularization for natural image denoising,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Brisbane, Australia, Apr. 2015.

[134] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *IEEE International Conference on Computer Vision*, Bombay, India, 1998.

[135] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2004.

[136] R. Safaee-Rad et al., “Accurate parameter estimation of quadratic curves from grey-level images,” *CVGIP: Image Understanding*, vol. 54, no. 2, pp. 259–274, Sept. 1991.

[137] D. H. Eberly, “Distance from a point to an ellipse, an ellipsoid, or a hyperellipsoid,” Geometric Tools, LLC, Tech. Rep., 1998.

[138] W. Gander et al., “Least-square fitting of circles and ellipses,” *BIT Numerical Mathematics*, vol. 34, no. 4, pp. 558–578, Dec. 1994.

[139] P. Rosin, “Analysing error of fit functions for ellipses,” *Pattern Recognition Letters*, vol. 17, no. 14, pp. 1461–1470, 1996.

[140] R. L. Burden and J. D. Faires, *Numerical Analysis: 4th Edition.* Boston, MA, USA: PWS Publishing Co., 1989.

[141] W. H. Press et al., *Numerical Recipes 3rd Edition: The Art of Scientific Computing.* Cambridge University Press, 2007.

[142] D. H. Eberly, *3D Game Engine Design, Second Edition: A Practical Approach to Real-Time Computer Graphics.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2006.

[143] J. A. Nelder and R. Mead, “A simplex method for function minimization,” *The Computer Journal*, vol. 7, pp. 308–313, 1965.

[144] J. C. Lagarias et al., “Convergence properties of the nelder–mead simplex method in low dimensions,” *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 112–147, Dec. 1998.

[145] R. B. Berry et al., “Rules for scoring respiratory events in sleep: Update of the 2007 AASM manual for the scoring of sleep and associated events: Deliberations of the sleep apnea definitions task force of the american academy of sleep medicine,” *Journal of Clinical Sleep Medicine*, vol. 8, no. 5, pp. 597–619, Oct. 2012.

[146] R. N. Khushaba et al., “Driver drowsiness classification using fuzzy wavelet-packet-based feature-extraction algorithm,” *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 121–131, Jan. 2011.

[147] C.-W. Hsu et al., “A practical guide to support vector classification,” Department of Computer Science, National Taiwan University, Tech. Rep., 2003.

[148] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, Mar. 2003.

[149] R. Kohavi and G. H. John, “Wrappers for feature subset selection,” *Artif. Intell.*, vol. 97, no. 1-2, pp. 273–324, Dec. 1997.

[150] S. Matyunin et al., “Temporal filtering for depth maps generated by kinect depth camera,” in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, Antalya, Turkey, 2011.

[151] D. Min et al., “Depth video enhancement based on weighted mode filtering,” *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 1176–1190, Mar. 2012.

[152] S. H. Chan et al., “An augmented lagrangian method for total variation video restoration,” *IEEE Transactions on Image Processing*, vol. 20, no. 11, pp. 3097–3111, Nov. 2011.

[153] C.-W. Hsu and C.-J. Lin, “A comparison of methods for multiclass support vector machines,” *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, Mar. 2002.

[154] T. Giannakopoulos and A. Pikrakis, *Introduction to Audio Analysis, A MATLAB® Approach.* Academic Press, 2014.

[155] H. M. Hondori and M. Khademi, “A review on technical and clinical impact of microsoft kinect on physical therapy and rehabilitation,” *Journal of Medical Engineering*, vol. 2014, pp. 1–16, 2014.

[156] M. Ye, C. Yang, V. Stankovic, L. Stankovic, and A. Kerr, “Kinematics analysis multimedia system for rehabilitation,” in *International Conference on Image Analysis and Processing*, 2015, pp. 571–579.

[157] M. Ye, C. Yang, V. Stankovic, L. Stankovic, and A. Kerr, “Gait analysis using a single depth camera,” in *IEEE Global Conference on Signal and Information Processing*, December 2015, pp. 285–289.

[158] C. D. Lim, C. M. Wang, C. Y. Cheng, Y. Chao, S. H. Tseng, and L. C. Fu, “Sensory cues guided rehabilitation robotic walker realized by depth image-based gait analysis,” *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 171–180, Jan. 2016.

[159] M. Ding and G. Fan, "Articulated and generalized gaussian kernel correlation for human pose estimation," *IEEE Transactions on Image Processing*, vol. 25, no. 2, pp. 776–789, Feb. 2016.

[160] T. Kato, S.-i. Yamamoto, T. Miyoshi, K. Nakazawa, K. Masani, and D. Nozaki, "Anti-phase action between the angular accelerations of trunk and leg is reduced in the elderly," *Gait & posture*, vol. 40, no. 1, pp. 107–112, 2014.

[161] I. McDowell and C. Newell, *Measuring health: A guide to rating scales and questionnaires.* Oxford University Press, 1996.

[162] K. Šušmáková, "Human sleep and sleep eeg," *Measurement Science Review*, vol. 4, no. 2, pp. 59–74, 2004.

[163] H. T. Wu, R. Talmon, and Y. L. Lo, "Assess sleep stage by modern signal processing techniques," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 4, pp. 1159–1168, Apr. 2015.

[164] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2003.

[165] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, Feb. 2012.

Appendix A

A.1 Pilot study protocol for limb motion analysis

Research Group

Philip Rowe, Biomechanics Unit, Biomedical Engineering, University of Strathclyde;

Andrew Kerr, Biomechanics Unit, Biomedical Engineering, University of Strathclyde;

Vladimir Stankovic, Electronic and Electrical Engineering, University of Strathclyde;

Lina Stankovic, Electronic and Electrical Engineering, University of Strathclyde;

Shikha Sarkar, Electronic and Electrical Engineering, University of Strathclyde;

Cheng Yang, Electronic and Electrical Engineering, University of Strathclyde.

Protocol Title

Validation of a 2D single camera video system and a 3D single camera video system to measure limb motor control: A pilot study

Design

Pilot validation of a two and three dimensional video system to measure metrics of limb motor control. Concurrent comparison will be made with a state of the art motion analysis (VICON, Oxford, UK).

Sample

A pilot sample of 5-10 healthy subjects.

The small sample size (5-10) is considered sufficient to provide the mean and variance data for a power calculation of sample size for a future larger study as well a testing the feasibility of procedures.

Recruitment

Individuals will be recruited from the staff and student community of the University of Strathclyde. Interested individuals being given information sheets.

Data Collection Protocol

Interested individuals will be offered a two hour appointment in the large biomechanics laboratory (Wolfson Building, Biomedical Engineering, University of Strathclyde) with dates and times arranged at the convenience of the interested individual.

Taking Consent

On arrival at the biomechanics laboratory one of the research team will provide a short introduction to the biomechanics unit as well as the motion analysis and video camera systems. The objectives of the study will be explained again to the participants and they will be given an opportunity to ask any questions. If the individual is still happy to proceed they will be asked to sign a consent form. At this point they will be assigned a study number, *e.g.* PVT001, and referred to thereafter as study participant number PVT00[No.].

Participant Preparation

Each participant will then be helped to put on lycra clothing, if the clothes they are wearing are considered too loose fitting. To help construct the three dimensional model each participant will then have the following dimensions measured and recorded in their individual case report file in addition to their age and gender:

Height – cm;

Weight – Kg;

Arm length - cm;

Shoulder width – cm;

Elbow width – cm;

Wrist width – cm;

Leg length – cm;

Pelvis width – cm;

Knee width – cm;

Ankle width - cm;

Retroreflective markers will then be attached to the skin (or clothes) overlying the following anatomical points:

- Upper limb:

posterior superior iliac spine;

anterior superior iliac spine;

mid iliac crest;

spinous process of 7th cervical vertebra;

tragus of ear;

most lateral border of the acromion process;

mid humerus;

lateral and medial epicondyles of humerus;

radial styloid process;

head of ulna.

- Lower limb:

anterior superior iliac spine;

front waist;

back waist;

tip of the big toe;

outside of the thigh below hand swing;

outside of the knee joint;

outside of the lower leg;

bony prominence on the outside of the ankle;

back of the foot;

outside of the foot at the base of the little toe;

tip of the big toe.

For the 2D and 3D video analysis, circular sticky paper labels marked with a concentric black and white pattern will be attached to the wrist [radial styloid process], elbow [lateral epicondyles of humerus], shoulder [most lateral border of the acromion process], pelvis [mid iliac crest), and head (tragus of ear), for upper limb; pelvis, knee, and ankle, for lower limb. Where these landmarks already have the retroreflective markers attached the paper labels will be placed directly under the reflective marker which will be attached to the centre of the paper label. This has worked well in a previous validation study for the lower limb (Ugbolue et al 2012).

For upper limb movement analysis, participants will sit in a standard sized armless chair in front of an adjustable table in the middle of the laboratory with their affected arm resting on the table. For lower limb movement analysis, participants will walk on a scaled mat (6 meters long, 0.7 meter wide) in one direction, and the other way around.

For the 2D video analysis, a high speed camera (Camera A) will be mounted on a tripod and position approximately 4m from the participant in line with their elbow. For the 3D video analysis, a conventional camera with a distance sensor will be mounted above Camera A.

Movement Tasks

For upper limb movement analysis, when the starting position has been achieved and the participant is happy to continue they will be asked to perform the following movements three times each giving a total of 15 movements.

1. Reach forward to touch a plastic cup positioned on the table directly in front of them at a distance equivalent to 80;
2. Reach forward to lift the same plastic cup towards their mouth. Instruction: bring the cup to your mouth;
3. Reach forward to lift the same plastic cup and turn it. Instruction: reach and turn cup over;
4. Reach forward to touch a plastic cup positioned on the table toward their unaffected at a distance equivalent to 20;
5. Lift their hand to touch the back of their head.

For lower limb movement analysis, again, when the starting position has been achieved and the participant is happy to continue they will be asked to perform the following movements three times each giving a total of 6 movements.

1. Walk from one end (Point A) of the scaled mat to the other (Point B);
2. Walk from Point B to Point A.

Participants will be asked to perform each movement as naturally as they can.

Once the movements have all been attempted the participant will be thanked for their participation and markers removed from their body. This will conclude their participation in the study.

Data Storage

The case report file (using participants study number) will be stored in a locked cabinet for the duration of the study *i.e.* until September 2014. Video files will be stored on an encrypted external hard drive and locked in the same cabinet. After the study has ended the video files will be permanently deleted and the paper records of the data kept in storage (locked cabinet in the Biomedical Engineering, Department) for a period of 5 years. Processed results will be made available for publication and to inform a grant application for a larger study.

Statistical Analysis

Data from the three systems (3D motion analysis and 2D and 3D video analysis) will be compared for each movement using analysis of variance and intra class correlation coefficients. The variables for comparison will be:

1. Movement duration;
2. Maximum forward tilt of trunk;
3. Magnitude of angular displacement at shoulder and elbow;
4. Relative timing of trunk, shoulder and elbow movements;
5. Magnitude of angular displacement at knee;
6. Relative timing of hip, knee, and ankle movements.

This data will be used to inform a power calculation of sample size for a second larger study planned for next year which will investigate validity, reliability and responsiveness of the video system with a larger sample including stroke survivors as participants.

Appendix B

B.1 Structural-Similarity

SSIM [10] is an image quality measurement from an image formation point of view. This algorithm is a combination of luminance, contrast, and structure comparison between the test image and the reference image, *i.e.*, between the candidate block within the searching area and the template block, in our experiment.

Initially, the template block $\mathbf{b}$ and a searching area S for frame F_n are acquired. The detection process is to compare the SSIM value of each candidate block $\mathbf{a}_n$ in S with $\mathbf{b}$. SSIM consists of three components:

$$\text{Luminance comparison: } l(\mathbf{b}, \mathbf{a}_n) = \frac{2\mu_{\mathbf{b}}\mu_{\mathbf{a}_n} + C_l}{\mu_{\mathbf{b}}^2 + \mu_{\mathbf{a}_n}^2 + C_l} \tag{B.1}$$

$$\text{Contrast comparison: } c(\mathbf{b}, \mathbf{a}_n) = \frac{2\sigma_{\mathbf{b}}\sigma_{\mathbf{a}_n} + C_{\mathbf{a}_n}}{\sigma_{\mathbf{b}}^2 + \sigma_{\mathbf{a}_n}^2 + C_{\mathbf{a}_n}} \tag{B.2}$$

$$\text{Structure comparison: } s(\mathbf{b}, \mathbf{a}_n) = \frac{2\sigma_{\mathbf{b}\mathbf{a}_n} + C_S}{\sigma_{\mathbf{b}}\sigma_{\mathbf{a}_n} + C_S} \tag{B.3}$$

where $\mu_{\mathbf{b}}$ is the mean intensity of $\mathbf{b}$, $\mu_{\mathbf{a}_n}$ is the mean intensity of $\mathbf{a}_n$, $\sigma_{\mathbf{b}}$ is the standard deviation of $\mathbf{b}$, $\sigma_{\mathbf{b}}$ is the standard deviation of $\mathbf{a}_n$, $\sigma_{\mathbf{b}\mathbf{a}_n}$ is the standard deviation of $\mathbf{b} \odot \mathbf{a}_n$, and C_l, $C_{\mathbf{a}_n}$, and C_s are pre-defined constants.

The SSIM measurement is given by:

$$\mathrm{SSIM}(\mathbf{b}, \mathbf{a}_n) = [l(\mathbf{b}, \mathbf{a}_n)]^{\alpha}[c(\mathbf{b}, \mathbf{a}_n)]^{\beta}[s(\mathbf{b}, \mathbf{a}_n)]^{\gamma} \tag{B.4}$$

To simplify the problem, we set $\alpha = \beta = \gamma = 1$, and $C_S = 0.5C_{\mathbf{a}_n}$, thus this measurement is derived as:

$$\mathrm{SSIM}(\mathbf{b}, \mathbf{a}_n) = \frac{2\mu_{\mathbf{b}}\mu_{\mathbf{a}_n} + C_l}{\mu_{\mathbf{b}}^2 + \mu_{\mathbf{a}_n}^2 + C_l} \cdot \frac{2\sigma_{\mathbf{b}\mathbf{a}_n} + C_{\mathbf{a}_n}}{\sigma_{\mathbf{b}}^2 + \sigma_{\mathbf{a}_n}^2 + C_{\mathbf{a}_n}} \tag{B.5}$$

Note that the block which has the maximum SSIM value is selected as the marker.

B.2 View transformation

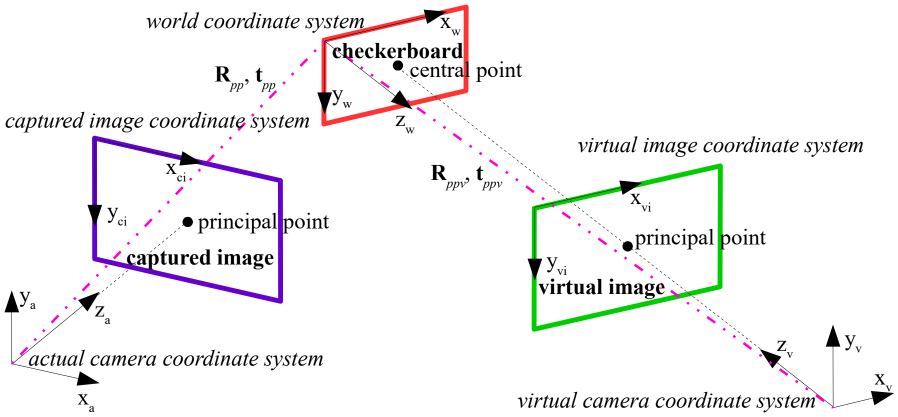

world coordinate system
x_w
checkerboard
central point
$\mathbf{R}_{pp}$, $\mathbf{t}_{pp}$
y_w
z_w
captured image coordinate system
virtual image coordinate system
x_{ci}
y_{ci}
principal point
captured image
$\mathbf{R}_{ppv}$, $\mathbf{t}_{ppv}$
x_{vi}
principal point
y_{vi}
virtual image
y_a
z_a
actual camera coordinate system
x_a
z_v
y_v
x_v
virtual camera coordinate system

Fig. B.1 View transformation setup.

As shown in Fig. B.1, we follow [89], set the origin of the *world coordinate system* to the upper-left feature point of a checkerboard, fix the actual camera - Kinect, and capture n infrared images and n corresponding depth images with different checkerboard orientations, including a pair of infrared and depth images showing that the checkerboard is closely perpendicular (pp) to the centerline of the virtual view, l mm away from the virtual camera, denoted as I_{pp} and D_{pp}, respectively. Each infrared image, denoted as I_i, is formed by projecting pixels in the *world coordinate system* into the *captured image coordinate system* using a perspective transformation [164]:

$$\lambda_i [u \ v \ 1]^\top = \mathbf{K}[\mathbf{R}_i | \mathbf{t}_i][X \ Y \ Z \ 1]^\top, \tag{B.6}$$

where $\mathbf{K}$, $\mathbf{R}_i$ and $\mathbf{t}_i$ are the intrinsic camera matrix, rotation matrix and translation vector respectively, (u, v) are the coordinates of a pixel in image I_i, (X, Y, Z) are the pixel coordinates of the point that is the backprojection of $I_i(u, v)$ into the *world coordinate system*, λ_i is a scaling factor of I_i.

We estimate $\mathbf{K}, \mathbf{R}_i$ and $\mathbf{t}_i$ with a closed-form solution [89], and minimize

$$\sum_{i=1}^{n} \sum_{j=1}^{L} \| \mathbf{m}_{i,j} - \hat{\mathbf{m}}(\mathbf{K}, \mathbf{R}_i, \mathbf{t}_i, \mathbf{M}_j) \|^2 \tag{B.7}$$

to refine them, where $\mathbf{m}_{i,j}$ is the intensity of a detected feature point in image I_i and $\hat{\mathbf{m}}$ is the projection of the world point $\mathbf{M}_j = [X_j \ Y_j \ Z_j]^\top$ in image I_i.

Given a checkerboard of size $(g \times w)$mm$\times(g \times h)$mm, and l mm away from the virtual camera in both I_{pp} and D_{pp}, the rotation matrix and translation vector of the virtual camera when 'capturing' the virtual *depth* pattern plane image D_{ppv}, denoted as $\mathbf{R}_{ppv}$ and $\mathbf{t}_{ppv}$ (as shown in Fig. B.1) respectively, are given by:

$$\mathbf{R}_{ppv} = \mathbf{I}_3, \quad \mathbf{t}_{ppv} = [\frac{g \times w}{2} \ \ \frac{g \times h}{2} \ \ l]^\top. \tag{B.8}$$

The virtual image coordinates function based on perspective transformation is given by:

$$\lambda_{ppv}[u_2 \ v_2 \ 1]^\top = \mathbf{K}\mathbf{R}_{pp}^{-1}\mathbf{K}^{-1}\lambda_{pp}[u_1 \ v_1 \ 1]^\top - \mathbf{K}\mathbf{R}_{pp}^{-1}\mathbf{t}_{pp} + \mathbf{K}\mathbf{t}_{ppv}, \tag{B.9}$$

where $\mathbf{R}_{pp}$ and $\mathbf{t}_{pp}$ are the rotation matrix and translation vector of the actual camera when capturing I_{pp}, $\lambda_{pp} = S_1/c_1$, c_1 is from $[a_1 \ b_1 \ c_1]^\top = \mathbf{K}^{-1}[u_1 \ v_1 \ 1]^\top$, and the relationship between the actual depth value (in mm) S_1 and the observed disparity $D_{pp}(u_1, v_1)$ in D_{pp} is given by (see [165]):

$$S_1 = \frac{1}{-2.85 \times 10^{-6} D_{pp}(u_1, v_1) + 0.003}. \tag{B.10}$$

Similarly we have

$$S_2 = \frac{1}{-2.85 \times 10^{-6} D_{ppv}(u_2, v_2) + 0.003}. \tag{B.11}$$

Finally, the observed disparity of the point in the virtual image is given by:

$$D_{ppv}(u_2, v_2) = d = \frac{0.003 S_2 - 1}{2.85 \times 10^{-6} S_2}. \tag{B.12}$$

B.3 Bisection method

Following [137], we use the implicit form of the ellipse

$$E(x_n, y_n) = (\frac{x_n}{a})^2 + (\frac{y_n}{b})^2 - 1 = 0, \tag{B.13}$$

and calculate half of the gradient of $E(x_n, y_n)$, *i.e.*, the normal vector to (x_n, y_n), *i.e.*,

$$(u_n^{'}, v_n^{'}) - (x_n, y_n) = q\nabla\frac{E(x_n, y_n)}{2} = q(\frac{x_n}{a^2}, \frac{y_n}{b^2}), \tag{B.14}$$

or

$$u_n^{'} = x_n(1 + \frac{q}{a^2}), v_n^{'} = y_n(1 + \frac{q}{b^2}), \tag{B.15}$$

where q is a scalar. Without loss of generality $a \geq b$. With exception of the following four special cases for $s_\theta(\mathbf{o}_n)$:

$$s_\theta(\mathbf{o}_n) = \begin{cases} |\sqrt{a^2 {u'_n}^2 + b^2 {v'_n}^2} - a|, \text{if } a = b \\ a, \text{if } |u'_n - a| < \varsigma, |v'_n - b| < \varsigma \\ |u'_n - a|, \text{if } |u'_n - a| \geq \varsigma, |v'_n - b| < \varsigma \\ |v'_n - b|, \text{if } |u'_n - a| < \varsigma, |v'_n - b| \geq \varsigma \end{cases} \tag{B.16}$$

where $\varsigma > 0$ is a small tolerance, (B.15) can be solved for x_n and y_n as:

$$x_n = \frac{a^2 u'_n}{q + a^2}, y_n = \frac{b^2 v'_n}{q + b^2}. \tag{B.17}$$

Thus, we have

$$E(q) = (\frac{a u'_n}{q + a^2})^2 + (\frac{b v'_n}{q + b^2})^2 - 1 = 0, \tag{B.18}$$

where $q \in [q_{\min}, q_{\max}], q_{\min} = -b^2 + b v'_n, q_{\max} = -b^2 + \sqrt{a^2 {u'_n}^2 + b^2 {v'_n}^2}, E(q_{\min}) > 0, E(q_{\max}) < 0$ [137]. BS first examines the sign of $E(\frac{q_{\min}+q_{\max}}{2})$, then replaces $q_{\min}$ ($q_{\max}$) with $\frac{q_{\min}+q_{\max}}{2}$ if $E(q_{\min})$ ($E(q_{\max})$) has the same sign as $E(\frac{q_{\min}+q_{\max}}{2})$. Let all the subsequent intervals of q's be $[q^*_{\min}, q^*_{\max}]$. BS stops at $|q^*_{\max} - q^*_{\min}| < \tau$, where $\tau > 0$ is a small tolerance. We use the above BS procedure to determine $s_\theta(\mathbf{o}_n)$.

B.4 Nelder-Mead simplex method

NM starts from $V = \{V_1, \ldots, V_K\}$, the $(K+1)$ points in K-dimensional space defining the initial simplex, for minimization of a function with k variables. Let $V_k = f_k(\theta)$. NM continuously updates V with three operations, naming, reflection, contraction, and expansion [143], until

$$\forall V_k, \sqrt{\frac{\left(V_k - (1/K)\sum_{k=1}^{K} V_k\right)^2}{K}} < \chi, \tag{B.19}$$

where χ is a small tolerance, *i.e.*, the minimum has been reached.

Index