Cognitive Pattern Recognition Models for Computational Musicology

Xiaoquan Li

In the fulfilment of the requirement for the degree of

Doctor of Philosophy

Centre for Excellence in Signal and Image Processing
Department of Electronic and Electrical Engineering
University of Strathclyde

Supervised by
Professor Jinchang Ren
Professor Stephan Weiss
Professor John Soraghan

Declaration

This thesis is the result 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.

Xiaoquan Li

Abstract

Music Information Retrieval (MIR) is essential for comprehending and analysing music, and it has various applications in music education, music creation, music recommendation, and other related areas. Conventional music processing techniques heavily depend on human derived characteristics and regulations, which hinders the comprehensive exploration of the abundant information embedded in music. This thesis aims to utilise artificial intelligence approaches, specifically modelling methods rooted in music knowledge and cognition, to address three objectives: automatic music transcription, predominant instrument detection, and music shape evaluation. Automatic music transcription (AMT) is the process of effectively identifying notes from audio signals. Predominant musical instrument recognition (PMIR) involves determining the dominant instrument in a musical section. Music shape evaluation (MSE) shows performance qualities and styles. This thesis introduces a cognition-guided framework for AMT, achieving F-measures of 76.3% on the MAPS dataset (an 8% improvement over the baseline), 80.17% on the BACH10 dataset (second-best performance), and 67.63% on the TRIOS dataset (leading performance). For PMIR, an innovative HHT-DCNN framework is proposed, achieving an 84% F-measure on the IRMAS dataset, which represents a 6% improvement over state-of-the-art methods. Finally, a new dataset is created for the MSE task, and a novel S-ResNN architecture is introduced, achieving an average accuracy of 93.78% across different training ratios. The experimental findings indicate that the suggested approaches may greatly improve current technical standards and achieve outstanding performance. Moreover, the findings of this thesis have the potential to be applied in several aspects of music education, such as the creation of curriculum, the development of interactive learning tools, and the design of personalised music training programmes. This thesis focuses on computational music comprehension and offers substantial contributions to automatic music transcription, instrument recognition, and performance analysis. It

highlights the importance and potential applications of research in computational musicology.

Acknowledgements

In the process of completing this doctoral thesis, I have received help and support from many people, and I would like to express my sincerest gratitude to them.

First, I would like to thank my previous first supervisor, Professor Jinchang Ren. In the early stages of my doctoral research, he provided me with meticulous guidance and selfless help. His extensive knowledge, keen insight, and rigorous academic attitude have deeply influenced me. He not only gave me many valuable suggestions in academic research but also gave me care and encouragement in life. Although he later left the University of Strathclyde, he laid a solid foundation for my research.

Next, I would like to thank my current first supervisor, Professor Stephan Weiss. After Professor Ren's departure, Professor Weiss took over my supervision without reservation. His rich experience, unique insights, and selfless dedication have benefited me greatly. He was always able to analyze problems from different perspectives and inspire me to explore new ideas. At the same time, he also gave me a great deal of research freedom, allowing me to try out different ideas. His careful guidance has played a vital role in my research.

I also want to express special thanks to my second supervisor, Professor John Soraghan. He has a wealth of knowledge and has given me many pertinent suggestions for my research. He was always able to communicate with me in an approachable manner and patiently answer my questions. His encouragement and support have allowed me to persevere when faced with difficulties.

Secondly, I would like to thank my colleagues and friends in the laboratory. Thank you to Mr. Yinhe Li for their help in data collection and processing; thank you to Dr. Yijun Yan for their suggestions on algorithm implementation and debugging. It is because of everyone's collaboration and mutual encouragement that I have been able to make continuous progress.

I also want to thank my family, especially my parents. Thank you for your nurturing care over the years and for always supporting me in pursuing my dreams. It is their

understanding and encouragement that have allowed me to devote myself wholeheartedly to scientific research.

Once again, I express my sincerest gratitude to all those who have helped me! Without you, this thesis would not have been completed.

Contents

List of Figures

List of Tables

List of Abbreviation

ABRSM - Associated Board of the Royal Schools of Music

AMT - Automatic Music Transcription

CNN – Convolutional Neural Network

CQT - Constant Q Transform

DCNN – Deep Convolutional Neural Network

DFT - Discrete Fourier Transform

DTW – Dynamic Time Wrapping

DL – Deep Learning

EM – Expectation-maximization

EMD - Empirical Mode Decomposition

ERB - Equivalent Rectangular Bandwidth

FFT – Fast Fourier Transform

GRU - Gated Recurrent Unit

HMM – Hidden Markov Model

HHT - Hilbert–Huang transform

HSD – Harmonic Structure Detection

IMFs - Intrinsic Mode Functions

MFCC - Mel Frequency Cepstral Coefficient

MFCCs - Mel Frequency Cepstral Coefficients

MFE – Multiple fundamental Frequency Estimation

MIDI - Musical Instrument Digital Interface

MIR – Music Information Retrieval

MIREX - Music Information Retrieval Evaluation eXchange

MPA - Music Performance Assessment

MPE – Music Pitch Estimation

MS – Music Shape

MSs – Music Shapes

MSE – Music Shape Evaluation

MSED - Music Shape Evaluation Dataset

MusicXML - Music Extensible Markup Language

NMF – Nonnegative Matrix Factorisation

PLCA – Probabilistic Latent Component Analysis

PMIR - Predominant Musical Instrument Recognition

SI-PLCA - Shift-Invariant probabilistic latent component analysis

S-ResNN - Siamese residual neural network

STFT - Short-time Fourier Transform

WT - Wavelet Transform

List of Variables

N - the number of data point

φ - the frequency index

p – pitch

f_0 - fundamental frequency

N_L – frame length in STFT

N_S - frame shift in STFT

μ - mean value

σ - standard deviation

$x(t)$ - original signal

$x(n)$ – the n^{th} sample in the time domain

$x_k(n)$ – the n^{th} sample in the k^{th} frame

$x_{k,windowed}(n)$ – the n^{th} sample in the k^{th} windowed frame

$X_k(f)$ – the frequency-domain representation of the k^{th} frame

X_{lf} - log frequency spectrum

$C_f(b)$ – the chromatogram of X_{lf}

b - the index of the sound level

$\tilde{x}(t)$ - normalized signal

$w(n)$ – window function

$S(k,f)$ – the spectrogram of FFT

B_o - the number of bins per octave

f_j – the j^{th} fundamental frequency

f_{min} - the lowest frequency

f_s – sampling rate

Q – the constant value

Δf – the ratio of the sampling rate f_s to the window size N

$N(j)$ – the length of window at f_j

$\frac{2\pi Q}{N(j)}$ – the digital frequency of the j^{th} component for the Constant-Q transform

f_c – the center frequency

$m(t)$ – mean of the upper and lower envelops

$h(t)$ – the first IMF candidate

$c_i(t)$ – the i^{th} IMF

$r_1(t)$ – the first residual

$r_n(t)$ – the final residual

$H[\cdot]$ – Hilbert transform

$a_i(t)$ – the instantaneous amplitude of the i^{th} IMF

$\theta_i(t)$ – the instantaneous phase of the i^{th} IMF

$\omega_i(t)$ – the instantaneous frequency of the i^{th} IMF

$P(i, j)$ – the probability of observing feature i and observation j

z_k – the k^{th} latent component

$P(z_k)$ – the prior probability of latent component k

$P(f_i|z_k)$ – the probability of observing feature i given latent component k

$P(c_j|z_k)$ – the probability of observing observation j given latent component k.

f – pitch-shifting parameter

s – instrument source

$P(t)$ – the energy distribution of the spectrogram

$P(z - f|s, d)$ – the spectral templates for a given pitch p and instrument source s with f pitch shifting across the log-frequency

$P(z|s, d)$ – the spectral templates for a given pitch p and instrument source s across the log-frequency

$P_t(f|d)$ – the log-frequency shift for each pitch on the time frame t

$P_t(s|d)$ – the instrumentation contribution for the pitch in the time frame t

$P_t(d)$ – the pitch contribution which can be considered as transcription matrix on the time frame t

$P(d, t)$ – the pianoroll transcription

N_0 – the number of non-zero fundamental-pitch

N_H – the number of harmonic-pitch

N_I – the number of instruments in the music piece

m – vector of pianoroll

PN_t – the normalised $\boldsymbol{P}(p,t)$ at time t

PF_t – the filtered result of PN_t

$PD(n)$ – non-zero notes index in n^{th} frame

$PCH(n,h)$ – the h^{th} harmonic pitch component of the pitch n

$PCP(n,h)$ – the h^{th} pitch of the harmonic n

$PHC(n,h)$ – the pitch value of $PCP(n,h)$

$ED(n)$ – the amplitude of $PD(n)$

$EDG(n,h)$ – the amplitude of fundamental pitch n and their corresponding harmonic

$EHC(n,h)$ – the amplitude of the h^{th} harmonic component presented in the pitch n

$EFF(n)$ – the final amplitude of fundamental frequency in pitch n

$W_i(h)$ – the guided weight of the h^{th} harmonic component for instrument i

$ED\_mono_t(1)$ – the first non-zero value of ED at frame t

$ED\_mono_t(h)$ – the rest non-zero value of ED at frame t

T_r – adaptive threshold

$R(n,t)$ – the rank value for each non-zero pitch n in each frame t

$\bar{R}$ – the averaged rank of the connected pitch group in $R(n,t)$

$A(\omega)$ – the Fourier spectrum

ω – the angular frequency

Chapter 1

1 Introduction

The major focus of this thesis is on using music cognition and pattern recognition to evaluate performance in music instruction. This chapter provides a description of the purpose and motivation of the study, which may be found in Section 1.1. Section 1.2 outlines the structure of the thesis, while Section 1.3 provides a comprehensive explanation of the primary contributions made by this research. The publications pertaining to the thesis are referenced in Section 1.4.

The music education industry has experienced substantial expansion in recent years, with an annual growth rate ranging from 6% to 8%. The online music education business is seeing significant expansion, with an annual growth rate of over 10%, highlighting its immense commercial potential. Astute Analytica predicts that the worldwide online I examined how

music education industry will develop at a compound annual growth rate (CAGR) of more than 18% between 2021 and 2027 [16]. This substantial expansion highlights the growing trend towards digital learning platforms in the field of music education.

Traditional music education faces several cognitive and practical challenges. Unlike other academic subjects, music education heavily relies on developing sophisticated cognitive processes, including auditory perception, pattern recognition, and musical memory. Students typically practice independently between lessons, where they must engage these cognitive skills without expert guidance. This cognitive development process requires extensive time for both execution and comprehension, resulting in high human resource costs. Moreover, the development of refined auditory cognition necessary for perceiving subtle musical nuances demands extensive training, further increasing educational expenses.

A significant challenge lies in the lack of cognitive support during students' practice sessions. Without guidance, students may develop incorrect cognitive patterns or fail to properly engage the multiple cognitive processes involved in musical learning. As per a research by the Associated Board of the Royal Schools of Music (ABRSM) [17], almost 50% of children discontinue their participation in music programmes, while over 80% of adults regret discontinuing their musical education in childhood, suggesting a need for better cognitive support during crucial learning periods.

Advanced Music Information Retrieval (MIR) technology provides solutions to these difficulties. The integration of capabilities such as audio visualisation, automated music transcription, musical performance assessment, and instrument classification has the potential to bring about a transformative impact on music education. Technologies such as Automatic Music Transcription (AMT) show potential for further advancement, particularly in the area of identifying intricate compositions. The exponential expansion in online music education, along with swift progressions in artificial intelligence, necessitates elevated benchmarks for Music Information Retrieval (MIR) technology in the field of music education.

1.1 Motivation and Aim

Music evaluation plays a pivotal role in music education's feedback system. While automated assessment systems promise to enhance engagement and provide immediate feedback, developing accurate and comprehensive evaluation tools remains challenging. Recent developments in Music Information Retrieval (MIR) technology offer promising solutions to these challenges. However, current technologies still face significant obstacles, particularly in automation levels and evaluation accuracy, due to lack of reasonable music cognition guidance. Additionally, the absence of comprehensive databases poses a substantial barrier to advancement in this field. Therefore, when evaluating musical performance, existing commercial software often prioritizes pitch accuracy above all else. In practice, musicians interpret pieces differently, resulting in varied musical shapes. This diversity becomes even more complex in ensemble performances, where both pitch and shape interactions between instruments significantly impact the overall musical expression. To achieve a more comprehensive and accurate assessment of musical performance, different instruments and evaluate musical shape must be identified in a meaningful way.

This thesis focuses on addressing these challenges through cognition-guided pattern recognition models in computational musicology. My research specifically targets three crucial applications:

1. Automatic Music Transcription (AMT): Focuses on pitch detection and melody extraction, addressing the fundamental elements of pitch and timing in musical performance.
2. Predominant Musical Instrument Recognition (PMIR): Enables accurate music classification and instrument identification, contributing to the understanding of tonal aspects in performance.
3. Musical Shape Evaluation (MSE): Extends beyond traditional analysis to encompass music structure analysis and emotion recognition, enabling comprehensive assessment of musical shape and overall performance quality.

By improving these three areas, I work toward establishing fair evaluation standards and consistent support mechanisms in music education. The following chapters will detail our innovative approaches in each of these areas, demonstrating how cognition-guided pattern recognition models can enhance music education through improved automation and accuracy in performance evaluation.

1.2 Thesis Structure

Chapter 2 offers a thorough examination of the essential components of music and their contribution to the advancement of music recognition.

Chapter 3 provides an overview of the related work for music signal pre-processing, machine learning and the related work for MIR.

Chapter 4 introduces a new pitch estimation method. This involves converting the audio to a time-frequency representation (e.g., Constant-Q transform), extracting features using shift-invariant probabilistic latent component analysis (SI-PLCA) and proposed harmonic structure detection module. The proposed framework demonstrates superior performance in multi-pitch estimation. This is evident in experiments using three widely used datasets (MAPS, BACH10, and TRIOS), where the framework achieves the highest F-measure (F1) scores.

Chapter 5 presents a musical instrument recognition method composed by Hilbert-Huang transform and deep convolutional neural network (DCNN). Through doing comprehensive ablation experiments, the best parameter selection for DCNN has been determined. In the benchmarking experiments, the proposed model compares against several state-of-the-art methods on 6705 musical pieces including 11 different instruments. And the experimental results show that the proposed method can produce reliable performance according to objective and subjective assessment.

Chapter 6 proposes a Siamese residual neural network (S-ResNN) to automatically classify the music expressiveness in piano pieces into different musical shape categories. In addition, a new musical shape evaluation dataset was created, which contains 4116 recordings with 28 different musical shapes generated from 147 piano notes. The proposed S-ResNN method is benchmarked with several state-of-the-art techniques on my proposed database. From the analysis, the proposed method yields the best performance in terms of precision, recall and F-measure.

Chapter 7 briefly summarize the contributions of this thesis and discuss some further improvement of the proposed methods in the future.

1.3 Contributions

In this thesis, three new methods for three different AI-driven music education applications are proposed and evaluated. Generally, these methods aim to extract more effective patterns for better performance evaluation in music education. A detailed summary of these contributions is highlighted in the following:

1. The methodology proposed for harmonic structure detection aims to extract multiple fundamental frequencies, and effective note tracking strategy aims to connects individual pitches across time frames to form coherent note tracks, better representing the music transcription[164].

2. A new method for recognizing the dominant musical instrument in the music pieces is proposed. It combines two powerful techniques: the Hilbert-Huang Transform (HHT) and a Deep Convolutional Neural Network (DCNN). The classification accuracy of the proposed method outperforms other benchmarking methods [152].
3. A promising new approach for automatically evaluating the musical shape of piano performances along with a new dataset called MSED-4K are proposed. This has the potential to be a valuable tool for both piano teachers and students [168].

The technologies proposed in this thesis collectively form a comprehensive framework for advanced music analysis, each enhancing and refining the capabilities of the others to provide a robust and detailed understanding of musical content, leading to enhanced music teaching, study and education. AMT can convert audio recordings into musical notation automatically, helping students analyze their own performances, compare their playing with reference recordings and study different interpretations of the same piece. It can also assist teachers in documenting student performances, providing detailed feedback on pitch and rhythm accuracy, creating teaching materials from recordings. PMIR enhances music learning through automatically identifying instruments in ensemble recordings and analyzing instrumental timbres and techniques, making it a valuable tool for orchestration study. This technology supports students in understanding instrumental roles within ensembles, facilitates focused listening exercises, and supports instrument-specific pedagogy. MSE advances musical interpretation by providing objective feedback on expressive elements, analyzing dynamics and temporal variations and comparing different interpretative approaches. It supports pedagogical development through assessment of musical expression, guidance on phrasing and articulation, and analysis of performance style. While significant progress has been made, further research and improvements are needed to enhance accuracy, handle complex polyphonic music, and achieve real-time high-precision analysis.

1.4 Publications

This thesis covers work from October 2016 and November 2022 at the University of Strathclyde in Glasgow, UK. Work on music shape analysis (detailed in Chapter 5) was performed in Robert Gordon University as a visitor researcher in 2022. The majority of my work has been published in international peer-reviewed conferences and journals:

Journal Papers

1) Li, X., Yan, Y., Soraghan, J., Wang, Z., & Ren, J, "A music cognition–guided framework for multi-pitch estimation," *Cognitive computation*, vol. 15, no. 1, pp. 23-35, 2023.

Conference Papers

1) Li, X., Weiss, S., et al. "S-ResNN: siamese residual neural network for musical shape evaluation in piano performance assessment." in *31st European Signal Processing Conference, EUSIPCO 2023 (pp. 216-220)*. Aalto, Finland, September, 2023.
2) Li, X., Wang, K., et al. "Fusion of Hilbert-Huang transform and deep convolutional neural network for predominant musical instruments recognition." in *Proceedings of the 9th International Conference on Computational Intelligence in Music, Sound, Art and Design (Part of EvoStar),* pp. 80-89, Seville, Spain, April, 2020.
3) Li, X., Yan, Y., et al. "Knowledge based fundamental and harmonic frequency detection in polyphonic music analysis." In *International Conference in Communications, Signal Processing, and Systems* (pp. 591-599), Harbin, China, July 2017.

In addition to the work described in the above papers, I have engaged in several other research projects at the intersection of deep learning and signal processing. My contributions to these endeavors have been substantial, leveraging my expertise in the wider field. These works are not reported in this thesis, but reflected in the following published papers:

1) Geng, J., Ma, L.E., Li, X., Zhang, X. and Yan, Y., “Printed Texture Guided Color Feature Fusion for Impressionism Style Rendering of Oil Paintings,” Mathematics, 10(19), p.3700, 2022.

2) Geng, J., Zhang, X., Yan, Y., Sun, M., Zhang, H., Assaad, M., Ren, J. and Li, X., “MCCFNet: multi-channel color fusion network for cognitive classification of traditional chinese paintings,” Cognitive Computation, 15(6), pp.2050-2061, 2023.

3) Gong, M., Soraghan, J., Di Caterina, G., Li, X. and Grose, D., “A boundary optimization scheme for liver tumors from CT images.” In 31st European Signal Processing Conference (EUSIPCO) (pp. 1135-1139). Helsinki, Finland, September, 2023.

Chapter 2

2 Music Cognition

This thesis aims to enhance music recognition by specifically targeting the essential components of music, including pitch, timbre, shape, and music notation tools. Music signals, known as audio signals produced by vibrations, possess a distinctive quality in that they are subject to both human-imposed regulations and inherent laws of nature. This research seeks to contribute to the underexplored topic of automated notation in the realm of Music Information Retrieval (MIR) by investigating advanced approaches related to key musical characteristics. The thesis will explore the technical components of music knowledge and their role in supporting the study aims in sections 2.1 to 2.4. These sections will offer a thorough examination of the essential components of music and their contribution to the advancement of music recognition. Section 2.1 will specifically examine the notion of basic frequency and the theories that support it. Section 2.2 will examine how various voice patterns affect timbre and explore techniques for identifying these structures from music signals. Section 2.3 will explore the concepts of rhythm and beat detection, which are crucial for comprehending the time-related elements of music. Section 2.4 will address dynamics, which pertain to the volume or strength of music.

2.1 Pitch Perception

The Standard pitch, a new regulation introduced in 1939 [18], marks a significant milestone in the long-term development of music. This regulation has greatly benefited music research by establishing a standardized correspondence between pitch p and fundamental frequency f_0 as $f_0 = 440 \times 2^{(p-69)/12}$. Therefore, its inverse can be expressed as $p = 69 + 12 \times log_2(\frac{f_0}{440})$.

The pitch-fundamental frequency table, which can be found in Appendix. A, prescribes A4 as 440 Hz, serving as a reference point for calculating the frequencies of other pitches. For music composed before the introduction of the Standard pitch regulation, researchers can still apply the above formulas by adjusting the frequency parameter for A4 based on the specific musical reference.

When a sound or pitch is produced, it generates a fundamental frequency accompanied by a series of harmonics or overtones. These harmonics are integer multiples of the fundamental frequency. For example, if the fundamental frequency is 100 Hz, the higher harmonics will be 200 Hz, 300 Hz, 400 Hz, 500 Hz, and so on. Similarly, if the fundamental frequency is 220 Hz, the harmonics would be 440 Hz, 660 Hz, 880 Hz, and so on. In terms of intervals on the musical scale, the harmonics correspond to specific notes relative to the base tone. The first harmonic is an octave above the base tone, followed by a perfect fifth above the octave, then a note two octaves up from the base tone, a major third above that, and so on. For instance, if the starting pitch is middle C (C4; 261 Hz), the overtones would be C5 (523 Hz), G5 (764 Hz), C6 (1046 Hz), E6 (1318 Hz), G6 (1568 Hz), B♭6 (1865 Hz), and so on. It is worth noting that there is a slight difference between 'equal temperament' and 'just intonation' when tuning an instrument. The Standard pitch equations are based on the 12-tone equal temperament system, which divides the octave into 12 equal parts, allowing for consistent tuning across different keys.

Harmonics are an integral part of the sound produced by a human voice or a musical instrument, contributing to the richness and complexity of the sound. However, human beings usually don't perceive harmonics as separate tones because their amplitude decreases as they increase in frequency. The presence of harmonics is what gives a voice or an instrument its unique character and timbre. Without harmonics, a voice would sound thin and uninteresting.

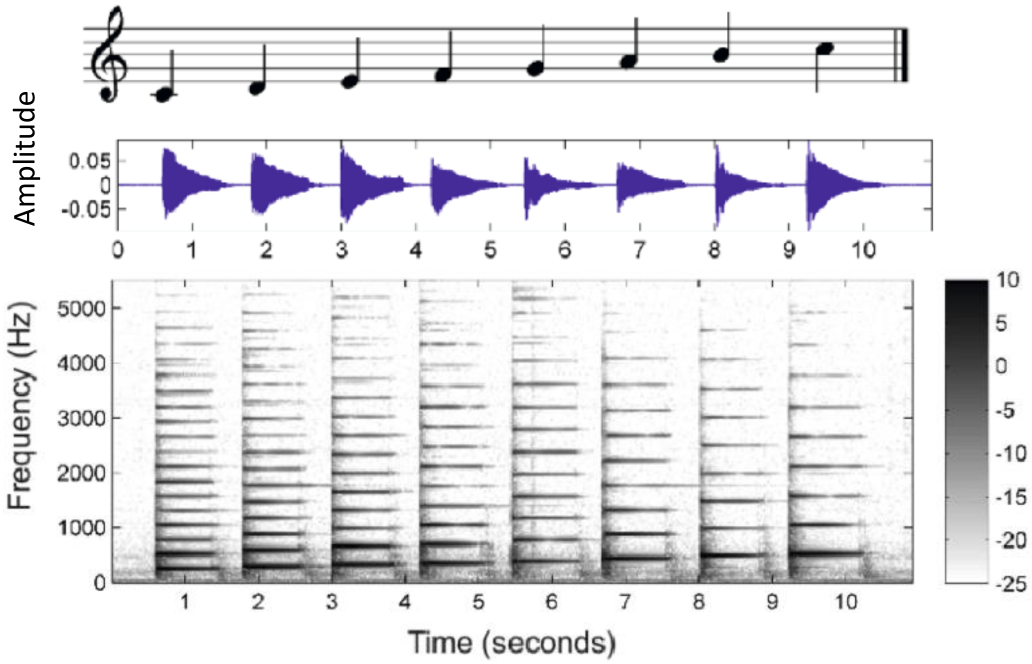


Fig. 2-1. Detection of single pitch for C major scale on a piano[1].

To understand how harmonics are produced, an example of a vibrating string is shown in Fig. 2-1. When a string vibrates, it creates a wave that moves from right to left. The total vibration is a sum of the right and left moving waves, which is known as a standing wave. Every standing wave has nodes (locations of minimum amplitude) and anti-nodes (locations of maximum amplitude). The same pitch will produce different energy levels for harmonics depending on the voice source.

While the detection technology for a single pitch is well-established, it is important to consider that the sound produced by musical instruments is not strictly harmonic due to the different nature of the sound source. The common assumption is that the sound is quasi-periodic. In some cases, partials may not be integer multiples of the fundamental frequency, such as in the case of Marimba or vibraphone. Additionally, vibrato can cause periodic amplitude modulation, which is often heard in the sound of a violin, flute, or human voice.

When detecting pitch, it is crucial to note that the notes emitted by an instrument can usually be decomposed into several stages: the attack stage, followed by decay, sustain, and release stages. This decomposition is important for accurate pitch detection and analysis. It should be noted that the experimental objects in this chapter are definite pitch instruments. The percussion instruments or indefinite pitch instruments are not considered.

The concept of pitch and its relationship to fundamental frequency has numerous applications in music research, analysis, and technology. Automatic music transcription (AMT) is one of the main applications since it seeks to translate musical auditory signals into symbolic representations, including musical scores or MIDI files [19]. AMT systems must have accurate pitch detection if they are to recognize the notes being performed in a musical composition.

Pitch is also essential in the development of music recommendation systems. By understanding the pitch-related features of musical pieces, such as melody, harmony, and key, recommendation algorithms can suggest songs or artists that share similar pitch-based characteristics with a user's preferences. Pitch information combined with other musical elements has been demonstrated to increase the accuracy and variety of song recommendations [20].

In the field of music education, pitch plays a vital role in ear training and sight-singing exercises [21]. Students learn to detect and imitate specific melodies, intervals, and harmonies, which is necessary for their musical development. Pitch detection algorithms are frequently integrated into computer-assisted music education software to offer students feedback and guidance during their practice sessions. Recent research has focused on how to use interactive technologies and gamification to improve pitch-based music instruction [22].

2.2 Timbre

Timbre is a critical component of sound that allows listeners to distinguish between two noises that have the same intonation and loudness. The American National Standards Institute defines it as a sensory attribute of sound that enables this distinction. The vibrations of the sound-emitting object and its components determine the timbre of a sound. When an object vibrates to generate sound, it emanates a fundamental tone that is accompanied by composite vibrations from the remaining components. The object's distinctive timbre is the result of the harmonics produced by these composite vibrations. The perceived timbre is substantially influenced by the proportion of mid-bass and mid-high tones in the overtone component [23]. A warm, gentle, full, and rich sound is produced by a higher proportion of mid-bass harmonics, whereas a frigid, tough, narrow, and sharp sound is produced by a larger proportion of mid-high tones.

Timbre is also influenced by time envelope parameters, including attack (the duration from silence to the initial peak of the sound), decay (the duration from the initial peak to a stable level), sustain (the duration of the sound at a stable level), and release (the duration from the stable level to silence) [24]. These parameters can be adjusted by a sampler to alter the timbre of a sound. For instance, the attack time parameter in a piano or trumpet signal can be increased, which can make it challenging for listeners to identify the instrument's tone [25]. This is because the initial transient sounds, such as the hammer striking the strings in a piano or the breath contacting the mouthpiece in a trumpet, are essential features for identifying the timbre of musical instruments.

When analyzing timbre, it is important to consider both the overtone changes in each frame and the time envelope of the entire sound production process. The time

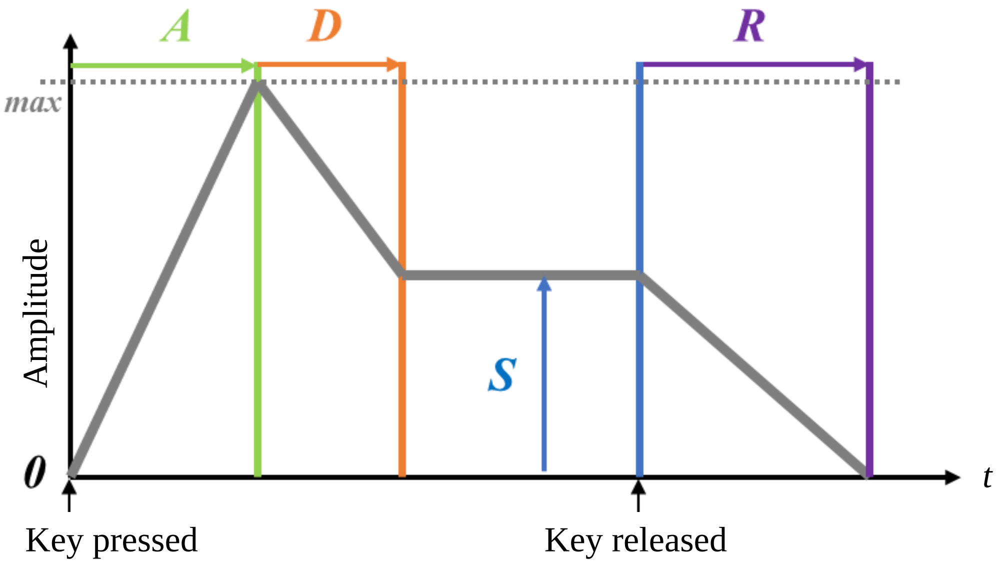


Fig. 2-2. The time envelope model ADSR (attack, decay, sustain and release).

envelope model, as shown in the Fig. 2-2 [24], illustrates the various stages of a sound's evolution over time.

The timbre of a sound, whether it originates from a human voice, musical instrument, natural source, or artificial creation, plays a crucial role in conveying emotions and shaping the overall character of a musical piece. In the context of musical instruments, each timbre possesses its own unique qualities and associations. For instance, when composing a march that aims to portray the image of a masculine hero, a composer is unlikely to choose soft-sounding instruments like the violin, flute, or oboe. Instead, they would likely opt for powerful and resonant brass instruments, such as the trumpet and trombone, to evoke a sense of strength and heroism. Similarly, when creating a piece that expresses lingering love, a composer might employ the warm, rich tones of the cello or the mellow, soulful sound of the saxophone to convey the desired emotional depth. In contrast, when depicting the bulky, lumbering nature of an elephant, the low, rough-sounding tones of the double bass or the booming strikes of the timpani would be more appropriate choices. The emotional expressions used to describe timbre heavily rely on the composer's intentional use of different instrument sounds to enhance the melody, harmony, rhythm, and dynamics of a piece, ultimately creating a distinct and impactful musical experience.

Mathematically, timbre can be represented using various techniques, such as the Fourier transform, which decomposes a complex sound wave into its constituent frequencies and amplitudes [26]. The resulting frequency spectrum provides insights into the harmonic content and overtone structure of the sound, which are essential factors in determining its timbre. Other mathematical tools, such as wavelet analysis and principal component analysis, can also be employed to extract and quantify timbral features from audio signals [27].

In music education, the concept of timbre is often introduced through listening exercises and instrument identification tasks. For example, students might be asked to distinguish between the sounds of different instruments playing the same melody, or to identify the primary instrument featured in a musical excerpt. By developing an ear for timbre and an understanding of how different instruments contribute to the overall sound of a piece, students can gain a deeper appreciation for the expressive possibilities of music and make more informed choices when composing or arranging their own works [28].

2.3 Shape

Emotions have shapes, and musical emotions mirror those shapes. Any music piece can be phrased in different ways rather like words in a sentence can be delivered differently, sometimes making the opposite sense from that which the composer wanted. So, playing the effectiveness and clarity of phrases in the style of the composer is an importance evaluate criteria of music performance.

The concept of shape is widely used by musicians, yet the mechanism that afford links between music and shape are little understood. Due to its abstraction and multisensory perception, the musicology study has progressed slowly. In 1963, Langer [29] proposed a famous theory of 'sound the way moods feel' where musical shape is considered as a fundamental unit in music's intrinsic properties. In 2004, Stern[30] [31] sees dynamics of experience as characterized by a sequence of present moments, each no more than a few seconds in length, which are shaped by feeling responses to incoming perceptions, and which group together to form dynamically shaped mini dramas, sensed as a gestalt, through which one lives. In 2006, Eitan and Granot [32] discussed the associations of dynamics, pitch, time, and articulation to musical shape by comparing the music perception with musicians and non-musicians. The result found that time patterns are highly related to the musical shapes. Pitch and dynamics patterns are potentially linked with musical shapes, which required further investigation. To tackle this issue, Küssner and Leech-Wikinson [33] carried out an extensive study in 2014, musicians(those with musical training) and nonmusicians(those without) match higher percent of pitch with motion, but they got more varies while matching loudness with motion and discovered that pitch contour does not closely associate with musical shapes though that too is important for musicians. In 2017, Daniel [34] defined that music shape refers to the small dynamics changes in music that can represent feeling and movement states, or any tiny changes varying with time, giving life-like qualities to music.

Generally, shape information does not reflect directly in the music scores. The Zygonic theory seeks to structure and form abstract narratives in sound in the absence of semantic content [35]. Using ‘zygonic’ theory, showing how different forms of mapping between the two may logically occur in cognition. Fig. 2-3 shows a small piece of oboe and cor anglaise duet from the third movement of Vaughan Williams’ Fifth Symphony.

The icon, index and symbols are used in this model to show the relationships between perspective values. The primary and secondary zygotic relationships can be seen in this figure. ‘Z’ represents the main zygotic relationship, and ‘I’ represents the "imperfect" zygotic relationship in which the generated values differ slightly from those mimicked, indicated by half arrows. Secondary zygons can be seen as connecting primary inter-perspective relationships, where one is thought to mimic the other. A single letter such as P for pitch, O for onset and T for timbre is used to denote a relationship type. As shown in Fig. 2-4, one shape is considered to mimic another

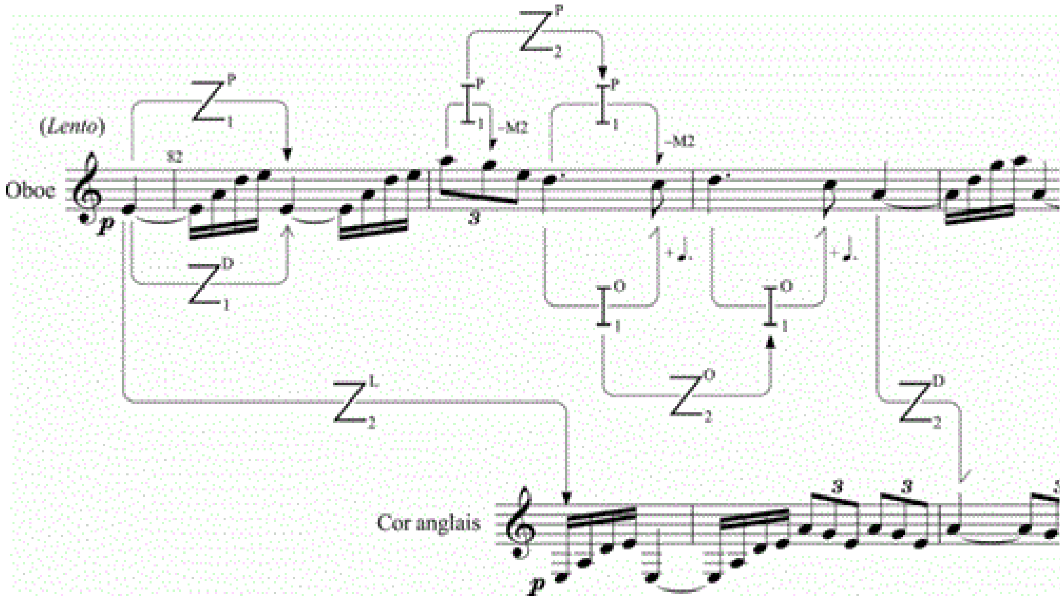


Fig. 2-3. Structure analysis of oboe and cor anglaise duet from the third movement of Vaughan Williams’ Fifth Symphony by using Zygonic [35].

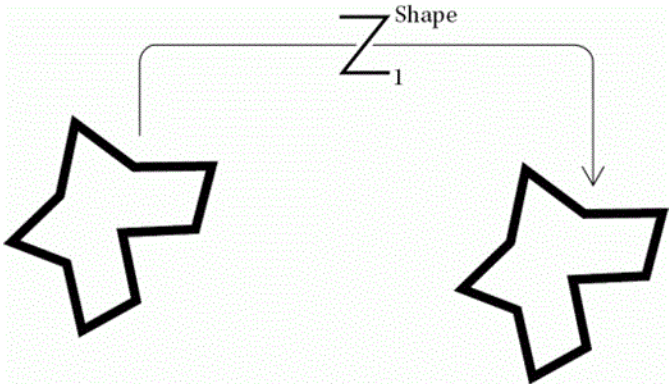


Fig. 2-4. One shape deemed in imitation of another

shape, which can also be used in the model such as the shape of the beginning of oboe and cor anglaise.

Adam Ockelford [36] proposed there are four types of cross-domain mapping between musical sounds and visual images relationship: ‘regular’ ‘indirect ‘arbitrary’, and ‘synaesthetic’ based on zygonic theory in which qualities of musical sounds and shapes (or their tactile equivalents) may be related systematically in cognition. They are imitated by ways of iconic, indexical, symbolic.

It is closely related to other concepts involving real or imagined movement through space (including gestures and trajectories) or across terrain (landscapes, silhouettes). On a more general level, it conceptualizes change over time. But fundamentally, in all of these discourses, shape is the modeling of changing sensations, and it is the mapping between the dynamics of musical sound and the dynamics of sensation that makes shapes so effective, as A way of thinking and talking about musical expressiveness. But the underlying mechanisms need to be teased out through other types of research.

2.4 Music Notation Tools

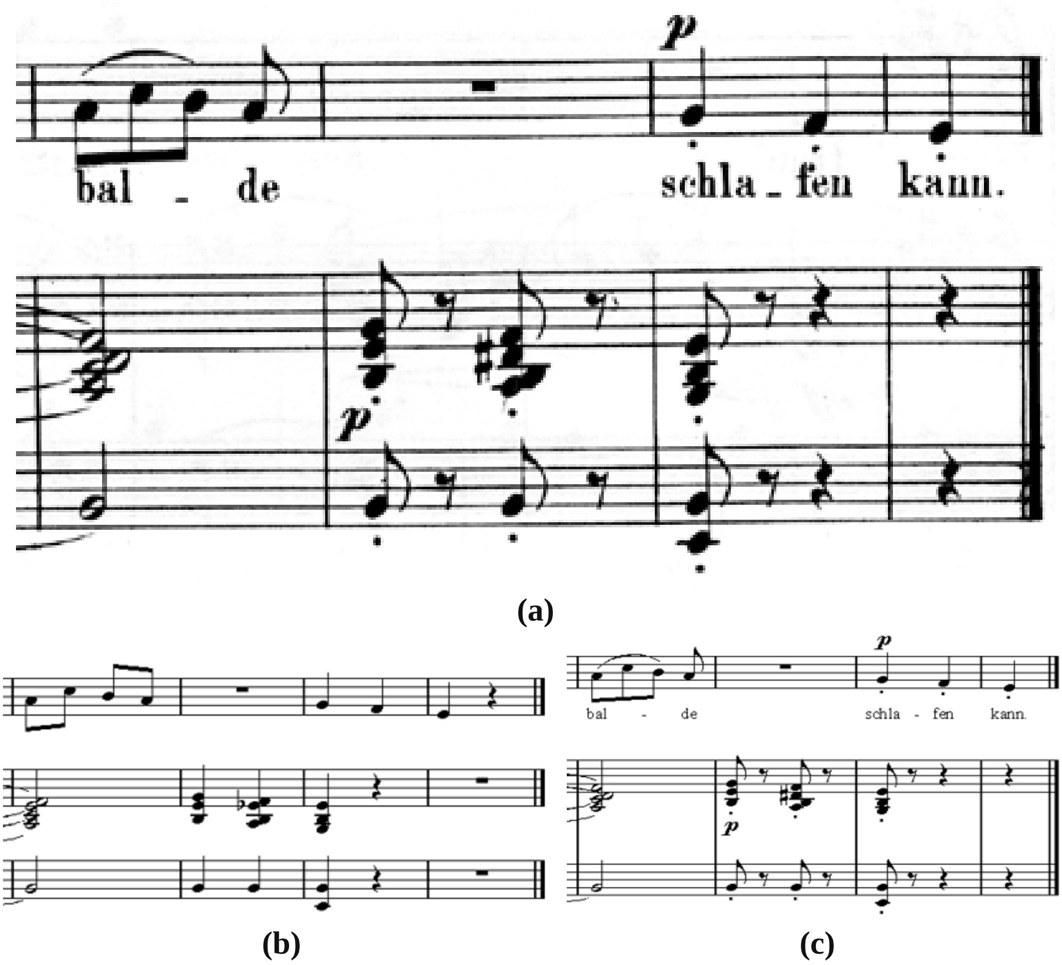
Musical scores can be digitized in various formats, but the most prevalent frameworks in computer-based music notation are the Musical Instrument Digital Interface (MIDI) and the Music Extensible Markup Language (MusicXML). MIDI was standardized in 1983, while MusicXML Version 1.0 was launched in 2004. However, it is important to note that both of these notation tools lack the capability to represent the 'shape' of music, which is a critical aspect in music perception and analysis. To address this limitation, these two tools are examined thoroughly and then endeavored to enhance them by integrating features that could effectively convey the shape information of music. MIDI is a crucial tool for musicians and producers, allowing various music gear to communicate using a common language. Rather than referring to any specific equipment or machine, "MIDI" is a specification and agreement between computer music equipment, a language between digital musical instruments, and a score that computers and electronic musical instruments can "understand." The major advantages of MIDI are summarized below:

1) MIDI files have small capacity, as they do not contain sound wave information but instead records music playback information, such as when and how long the music is played, with what tone, and at what pitch. Therefore, they are easy to process, even on the low specification computers.
2) MIDI files can be played back on various hardware sound sources, making them highly compatible and accessible across different platforms and devices.
3) MIDI files can accurately record the pitch and duration of notes. They are also easy to edited, making it a valuable tool for music composition and education

However, there are still a number of disadvantages of MIDI.

1) MIDI has a limited ability to capture the subtleties and nuances of human performance, such as slight variations in timbre and playing style. Finer details of musical scores, such as instrumentation, expression, stem direction, and beaming might be missing.
2) MIDI files rely on specific hardware and software to generate sound. The same MIDI played by different sound sources may yield different results, leading to inconsistent quality.
3) Due to missing various musical elements, MIDI files have limitations in notation, editing and synchronizaiton.

To tackle with MIDI’s limitations, MusicXML was invented. MusicXML is an open XML-based file format for recording Western-style musical scores. It was designed as an exchange format for notation information, especially between different notation software. MIDI files primarily describe musical notes, while MusicXML files capture the actual notation and layout. This means you don't have to do all the editing work on MIDI files. The creator of MusicXML, Michael Good, explains that MusicXML is designed to be useful over a range of music notation applications. Fig. 2-5(a) shows the original score of the last 4 bars of Schumann's Op.24, No. 4. Fig. 2-5(b) and (c) represent the MIDI version and MusicXML version of the same music piece, respectively. MIDI, primarily focused on recording playback information such as timing, tone, and pitch, offers a compact and easily distributable format for music sharing and sequence processing. While MIDI files also carry some level of expressive



(a)

(b)

(c)

Fig. 2-5. (a) the original piece (b) the midi piece and (c) the MusicXML piece of Schumann’s Op.24, No. 4.

details like volume changes and control changes to simulate dynamics, they still fall short in capturing the full complexity of musical scores, such as instrumentation, expression, stem direction, and beaming. On the other hand, MusicXML, provides a much more detailed representation of musical scores, thanks to its XML-based structure. It includes elements like pitch, duration, time signature, key signature, and lyrics, as well as complex score formatting and voice interactions. But MusicXML cannot completely capture the expressive subtleties and special performance qualities necessary for in-depth knowledge and study of musical performances. This limitation is common in standard notation formats, which struggle to convey the personalized expressions of performers.

In conclusion, while MIDI and MusicXML have played pivotal roles in the digitalization and dissemination of music, they both lack the capability to fully capture and convey the 'shape' of music. This limitation underscores an ongoing challenge in digital music notation: bridging the gap between the precision of written scores and the expressive depth of live performances. Addressing this gap remains a crucial area for future development in music technology, aiming to more accurately represent the full spectrum of musical expression and performance nuances.

2.5 Chapter Summary

In this chapter, four fundamental musical elements are explored: pitch, timbre, shape and notation tools. The chapter began with pitch perception, introducing the Standard pitch regulation and its mathematical relationship with fundamental frequency. The ways in which harmonics contribute to sound production and the challenges in pitch detection, particularly in the context of different musical instruments and their sound stages (attack, decay, sustain, and release), are studied and evaluated. Then how timbre is influenced by both overtone composition and time envelope parameters, making it essential for instrument recognition and emotional expression in music, is explored. Next, the development of musical shape theory is introduced. Finally, modern music notation tools are examined, specifically MIDI and Music XML. While Chapter 2 established the theoretical foundations of essential musical elements critical to music recognition and analysis. Chapter 3 examines how these elements are processed and analyzed using various signal processing techniques and machine learning approaches. Chapter 3 also delves into specific methodologies for automatic music transcription, instrument recognition, and shape evaluation, demonstrating how the theoretical understanding of pitch, timbre, and shape guides the development of practical MIR solutions.

Chapter 3

3 Related work

In order to motivate the present thesis, related works in terms of conventional feature extraction, machine learning methods and three research topics are reviewed in this chapter. Section 3.1 describes the background of music cognition, wherein the pitch, timbre, shape and natation tools are highlighted. Section 3.2 surveys the music signal pre-processing methods which includes time-frequency representation, representative features, and matrix factorization. Section 3.3 describes the machine learning used in music information retrieval. Section 3.4 review the related work of three research topics. Finally, a brief summary of evaluation metrics is given in Section 3.5.

3.1 Music Signal Pre-Processing

Preprocessing is a multifaceted procedure in the field of music detection that prepares audio signals for subsequent analysis and classification. The procedure includes several critical steps, each contributing to the overall effectiveness of music detection algorithms. These steps include signal acquisition, noise reduction and signal normalization, time-frequency transformation, feature extraction, feature dimensionality reduction, and output.

Signal acquisition: The first step involves capturing audio signals using sophisticated equipment and converting them into a digital format. This conversion process requires selecting appropriate sampling rates and bit depths to ensure a faithful digital representation of the audio. The selection of these parameters is critical since they directly affect the quality and accuracy of the subsequent preprocessing stages.

Noise reduction and signal normalization: After digitising the audio stream, noise reduction techniques like spectral gating or Wiener filtering are used to remove background noise and improve signal clarity. These techniques help to isolate the desired musical components from unwanted artifacts. Following noise reduction, signal normalization is applied to adjust the amplitude of the signal to a consistent range. Mathematically, this can be expressed as $\tilde{x}(t) = \frac{x(t)-\mu}{\sigma}$, where $\tilde{x}(t)$ is the normalized signal, $x(t)$ the original signal, μ the mean, and σ the standard deviation.

Time-frequency transformation: To analyze the frequency content of the signal over time, a time-frequency transformation is performed. The most common technique is the Fourier Transform, which converts the signal from the time domain to the frequency domain, revealing its spectral characteristics. Additionally, the Wavelet Transform is employed to provide a time-frequency representation that captures non-stationary aspects of the audio signal.

Feature extraction is the process of condensing the diverse characteristics of the audio signal into a format that is appropriate for analysis. For example, Mel-Frequency Cepstral Coefficients (MFCCs) are computed to mirror the human auditory system's response [169].

Feature dimensionality reduction is a critical stage in the process of simplifying the feature set while retaining its most informative aspects. This procedure is indispensable for optimising the efficacy and functionality of machine learning algorithms. Two other notable techniques employed in this stage are Probabilistic Latent Component Analysis (PLCA) and Non-negative Matrix Factorization (NMF), in addition to Principal Component Analysis (PCA).

By employing these techniques, the unprocessed music signals can be converted into a format that is more conducive to the processing of machine learning models, thereby obtaining feature vectors that accurately represent the music's content. In the subsequent sections, a comprehensive overview of these preprocessing methods will be presented, then how to employ them to extract a variety of musical features, thereby establishing a strong foundation for the efficient and precise recognition of music signals will be demonstrated.

3.1.1 Time-Frequency Representation

The Discrete Fourier Transform (DFT) is a technique used to convert a signal from the time domain to the frequency domain. However, it encounters limitations when applied to signals. The DFT offers a representation of the domain, across the entire signal duration assuming that the signal is stationary and that the frequency content remains constant. In reality audio signals are dynamic with varying sounds emerging and fading over time and their intensity fluctuating continuously. The Fourier Transform by itself cannot capture any such transient.

The Short Time Fourier Transform (STFT) operates on the assumption that a signals frequency domain remains stable over time intervals. It segments the signal into frames and conducts DFT on each frame individually. This approach yields a time frequency representation with time resolution dictated by the frame length. The STFT offers reduced computation cost compared to time frequency representations due to the efficiency of its DFT algorithm, which reduces time complexity from O(N^2) to O($Nlog(N)$), where N is the number of DFT point(that is, FFT size).

However, the STFT may suffer from spectral leakage when the frame size is not an integer multiple of the signal's period. This leakage introduces frequencies that should not be present in the calculated spectrum. To mitigate this issue, a window function is applied to each frame, which reduces the amplitude of the signal near the frame boundaries. Additionally, overlapping frames are used to ensure that all frequencies are analyzed evenly, as the windowing process may cause some frequencies to be ignored.

The choice of window size in STFT presents a trade-off between time and frequency resolution. A smaller window provides higher time resolution but lower frequency resolution, while a larger window offers higher frequency resolution but lower time resolution. Despite this trade-off, STFT has the advantage of being invertible, allowing the reconstruction of the original audio signal.

The Constant-Q transform (CQT) is a time-frequency representation that employs a filter bank with center frequencies distributed according to an exponential law. In contrast to the Fourier Transforms linear frequency axis, the Constant Q Transform (CQT) uses a scale with frequencies distributed exponentially. The bandwidth of filters varies with frequency. Their ratio of center frequency to bandwidth remains Q). CQT suits music signal processing well due to its alignment with scale on the axis. By computing the CQT spectrum of a music signal one can directly access amplitude values at each note frequency. This characteristic makes CQT more appropriate, for analysis compared to STFTs spaced frequencies.

The Mel spectrogram is a representation of signals optimized for auditory perception. The concept is based on transforming the STFT using an adjustment according to the Mel scale, which mirrors how the human ear perceives different frequencies Compared to the STFT spectrogram, the Mel spectrogram compresses the frequency axis, resulting in a more compact representation while preserving perceptually important information. However, the Mel spectrogram only contains amplitude (energy) information and cannot be inverted back to the original audio signal.

Other perceptually motivated frequency scales similar to the Mel scale include the Bark scale, Equivalent Rectangular Bandwidth (ERB), and Gammatone filters. These scales are based on the psychology of hearing and aim to provide representations that align with human auditory perception.

A chromatogram, also known as a pitch class profile, shows the distribution of energy over a range of pitch classes, typically the twelve-tone class of Western music (C C# D D# E E# F G G# A A# B). The chromatogram can be considered as a fold of CQT on the rating axis. Given a log frequency spectrum X_{lf} (e.g. CQT), the chromatogram is calculated by the following formula: $C_f(b) = \sum_{z=0}^{Z-1} \left| X_{lf}(b + z\beta) \right|$, where z represents the z^{th} octave, b is the index of the sound level, and β typically ranges from 0 to 11, encompassing all pitch classes within an octave. Like the Mel-frequency cepstral coefficients (MFCC), the chromatogram provides a deeper analysis compared to simpler characterization methods and can also be employed directly as a feature in various music information retrieval tasks.

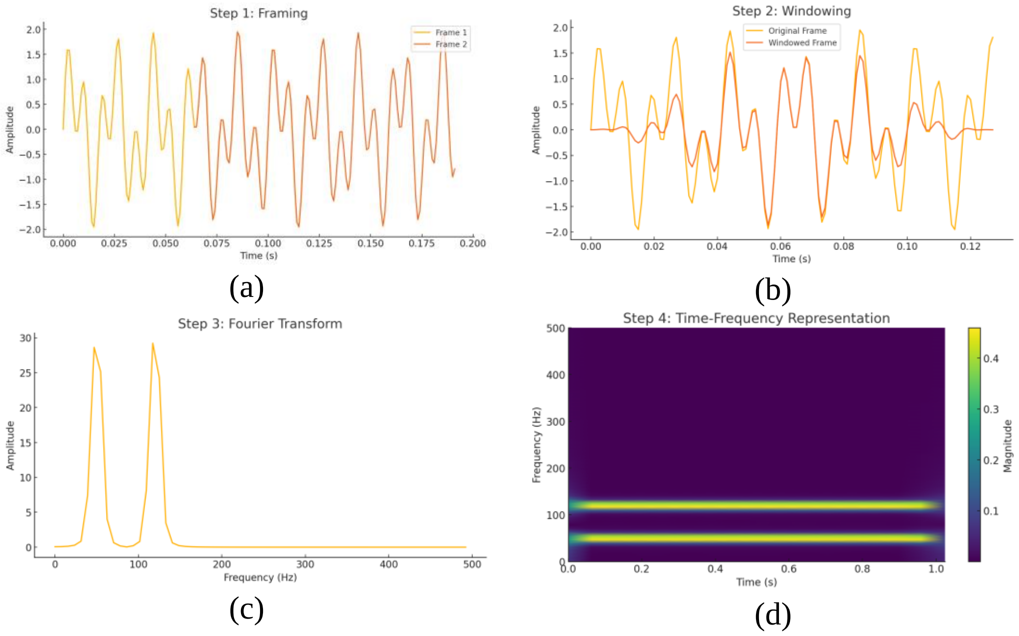
In contrast, CQT demonstrates unique advantages in music signal analysis tasks. CQT offers higher frequency resolution in the low-frequency region, which aligns with the human ear's higher sensitivity to frequency changes in low-pitched sounds. CQT also makes sure that the time quality is the same across all frequencies, enabling it to provide detailed information about the temporal evolution and rhythm of music. Unlike chromagrams that may lose pitch class data CQT retains all spectrum information while allowing for reconstruction of the signal through an inverse transformation, a process not possible with chromagrams. Moreover, CQT goes beyond pitch analysis. It can be applied to tasks such as chord recognition and pitch tracking, making it more versatile than chromagrams.

3.1.1.1 The Short-Time Fourier Transform (STFT)

Short Time Fourier Transform (STFT) is an important tool in signal processing because it shows how signals change over short periods of time. When working with sounds that change frequency over time, like speech or music, it works especially well. The spectrograms generated by STFT can show how the parts of a signal change over time. These can be used for analysis, speech processing, and sound creation. STFT plays a role, in fields including voice recognition, music production and acoustics research. Generally, STFT includes five steps, i.e., framing, windowing, Fourier transform, time-frequency representation and application. An example of first four steps is shown in Fig. 3-1 for better understanding.

1) Framing: Divide the long signal $x(t)$ into a series of short segments or frames. Given a frame length of N_L samples, and a frame shift of N_S samples (where N_S is typically less than N_L to allow overlap between frames), the k_{th} frame $x_k(n)$ can be described as: $x_k(n) = x(n + kN_S)$,$0 \leq n < N_L - 1$. This process involves dividing the signal into frames to facilitate the analysis of its time varying characteristics.

2) Windowing: Apply a window function $w(n)$ to each frame. The window function has the same length as the frame, and its values are maximum at the center of the frame



(a) (b)

(c) (d)

Fig. 3-1. Visualization of STFT steps: (a) illustrates a framing example where a long signal is divided into two frames; (b) compares the original frame with its Hamming-windowed version; (c) displays the frequency spectrum of the frame; (d) shows the signal's frequency content over time, with intensity indicating magnitude.

and gradually decrease to zero at the ends of the frame. The k^{th} windowed frame $x_{k,windowed}\ (n)$ can be represented as $x_{k,windowed}(n) = x_k\ (n) \cdot w(n)$. Different types of window functions have different characteristics:

- Rectangular Window: All values are one. It provides maximum time resolution but poor frequency resolution.
- Hann Window: Has a bell shape, which reduces the side lobes in the frequency domain and provides better frequency resolution.
- Hamming Window: Similar to the Hann window but with a slightly different shape, providing a compromise between time and frequency resolution.
- Blackman Window: Provides even better frequency resolution at the expense of time resolution.

3) Fourier Transform: Performing a Fast Fourier Transform (FFT) on each windowed frame to transform it from the time domain to the frequency domain will improve the efficiency for calculating the Discrete Fourier Transform (DFT). If an N-point FFT is

used, the frequency-domain representation $X_k(f)$ of the k^{th} frame can be represented as $X_k(f) = \sum_{n=0}^{N-1} x_k(n) \cdot e^{-j\frac{2\pi}{N}fn}$, where $x_k[n]$ is the windowed signal in the k^{th} frame, and f is the frequency index (f = 0, 1, ... , N-1). The resulting $X_k(f)$ represents the complex-valued spectrum of the frame, containing information about the magnitude and phase of the frequency components.

4) Constructing Time-Frequency Representation: The frequency-domain results obtained from the FFT for all frames are combined to form a time-frequency representation. The time-frequency representation, often referred to as a spectrogram, can be represented as a matrix $S(k,f) = |X_k(f)|^2$ or $S(k,f) = 20log_{10}(|X_k(f)|)$. The former equation represents the power spectrogram, where the magnitude squared of the complex spectrum is used, and the later equation represents the logarithmic spectrogram, where the magnitude of the complex spectrum is converted to a logarithmic scale (in dB) for better visualization. The resulting spectrogram is a visual representation of how the frequency content of the signal changes over time.

5) Analysis and Application: By observing the time-frequency representation, the time-frequency characteristics of the signal can be analyzed, different frequency components and their variations can be identified over time. This analysis is widely applied in audio processing, speech recognition, communication, and many other fields.

3.1.1.2 Constant-Q transform

To perform the time-frequency spectrogram in the MPE area, the Constant-Q transform (CQT) is the desirable technique, as it is more efficient in lower frequency and there are fewer frequencies required in given range, which testified its usefulness when the distribution of frequencies in several octaves is discrete. Meanwhile, its rate resolution will decrease with the increasing of the frequency bins, and this is suitable for auditory applications. It is linear when using the Fast Fourier Transform (FFT) to analyze the frequency. However, as the frequency of human ear perception is logarithmically distributed [37] the discrimination of relatively low frequency is relatively high. The CQT is based on the FFT principle but processes a logarithmic compression for closing to human's cochlea helical structure. The Fast Fourier transform (FFT) of a discrete signal $x(n)$ is defined in Eq.(3.1).

$$X(f) = \sum_{n=0}^{N-1} w(n)x(n)e^{-\frac{j2\pi\varphi n}{N}} \tag{3.1}$$

where $w(n)$ is the window function, $x(n)$ is the n^{th} sample in the time domain, φ is frequency index, the digital frequency is$\frac{2\pi\varphi}{N}$, and the period of samples is $\frac{N}{\varphi}$.

For music analysis, in order to make each bin in each octave equals to 12 semitones, the j^{th} frequency is defined by Eq.(3.2).

$$f_j = 2^{\frac{j-1}{B_o}} f_{min} \tag{3.2}$$

where B_o is the number of bins per octave, which equals to 60 as suggested in [38]. f_{min} is the lowest frequency, e.g. the minimum frequency on a piano is 27.5 Hz. The resolution or bandwidth Δf is the ratio of the sampling rate f_s to the window size N. To make the ratio of f_j to Δf to be a constant value Q, the window size must change over f_j. The constant value is given by $Q = \frac{f_j}{\Delta f} = \frac{1}{2^{\frac{1}{B_o}}-1}$ and the length of window at f_j is then defined by $N(j) = \frac{f_s}{\Delta f} = \frac{f_s \cdot Q}{f_j} = \frac{f_s}{f_j(2^{\frac{1}{B_o}}-1)}$.

Therefore, by taking account of the above constraints, the digital frequency of the j^{th} component for the Constant-Q transform is $\frac{2\pi Q}{N(j)}$. Since the window size is determined by $N(j)$, the window function is related to j as well as n. Meanwhile, as $N(j)$ varies with j, it is also used for normalization. Then Eq. (3.1) becomes:

$$X(j,n) = \frac{1}{N(j)} \sum_{n=0}^{N(j)-1} w(j,n)x(\mathrm{n})e^{-\frac{j2\pi Qn}{N(j)}} \tag{3.3}$$

Suggested by [39], the best window function for music signal processing is Blackman-Harris window which is defined in Eq.(3.4)

$$W(j,n) = \mathrm{a}_0 - \mathrm{a}_1 \cos\frac{2\pi n}{N(j)-1} + \mathrm{a}_2 \cos\frac{4\pi n}{N(j)-1} - \mathrm{a}_3 \cos\frac{6\pi n}{N(j)-1} \tag{3.4}$$

where $\mathrm{a}_0 = 0.35875;\ \mathrm{a}_1 = 0.48829;\ \mathrm{a}_2 = 0.14128;\ \mathrm{a}_3 = 0.01168$ and $0 \leq \mathrm{n} \leq N(j) - 1$

3.1.1.3 Equivalent Rectangular Bandwidth

The equivalent rectangular bandwidth (ERB) is a psychoacoustic measure that approximates the bandwidths of filters in the human auditory system. It simplifies the modeling of these filters as rectangular band-pass or band-stop filters, such as those used in tailor-made notched music training (TMNMT)[40].

For young listeners and moderate sound levels, the bandwidth of human auditory filters can be estimated using a polynomial equation (Eq.(3.5)) given by Moore and Glasberg [41].

$$ERB(f_c) = 6.23 \times f^2 + 93.39 \times f + 28.52 \tag{3.5}$$

where f_c represents the center frequency of the filter in kHz, and $ERB(f_c)$ is the bandwidth of the filter in Hz. This approximation is derived from the results of various simultaneous masking experiments and is valid for center frequencies between 0.1 and 6.5 kHz. An alternative approximation, also provided by Moore and Glasberg [42], is given by Eq.(3.6).

$$ERB(f_c) = 24.7 \times (4.37 \times f_c + 1) \tag{3.6}$$

This approximation is suitable for moderate sound levels and center frequencies between 0.1 and 10 kHz.

The ERB-rate scale, also known as the ERB-number scale, is a function $ERBs(f_c)$ that returns the number of equivalent rectangular bandwidths below a given frequency f_c. The units of this scale are called $ERBs$ or Cams, as suggested by Hartmann. The scale is constructed by solving the following differential system of equations (Eq.(3.7)).

$$\begin{cases} ERB(0) = 0 \\ \dfrac{df}{dERBs(f_c)} = ERB(f_c) \end{cases} \tag{3.7}$$

The solution for $ERBs(f_c)$ is obtained by integrating the reciprocal of $ERB(f_c)$ and setting the constant of integration such that $ERBs(0) = 0$. Using the second order polynomial approximation (Eq.(3.5)) for $ERB(f_c)$ yields:

$$ERBs(f_c) = 11.17 \times ln\left(\frac{f_c + 0.312}{f_c + 14.675}\right) + 43.0 \tag{3.8}$$

The VOICEBOX speech processing toolbox for MATLAB implements the conversion and its inverse can be presented as Eq.(3.9).

$$\begin{gathered} ERBs(f_c) = 11.17268 \times ln\left(1 + \frac{f_c \times 46.06538}{f_c + 14678.49}\right) \\ f_c = \frac{676170.4}{47.06538 - e^{0.0895 \times ERBs(f_c)}} - 14678.49 \end{gathered} \tag{3.9}$$

Using the linear approximation (Eq.(3.6)) for $ERB(f_c)$ yields:

$$ERBs(f_c) = 21.4 \times log_{10}(1 + 0.00437 \times f_c) \tag{3.10}$$

where the unit of f_c is Hz.

3.1.1.4 Hilbert-Huang Transform

Hilbert–Huang transform (HHT) is a signal processing technique that is applicable for nonstationary and nonlinear signals [43]. HHT is a combination of two methodologies [44], namely, empirical mode decomposition (EMD) and Hilbert transform. EMD is used to decompose the input signal into different components called intrinsic mode functions (IMFs), while the Hilbert spectrum is obtained by applying the Hilbert transform to the IMFs. Compared to other time-frequency analysis methods such as the wavelet transform (WT) and short-time Fourier transform (STFT), HHT offers adaptive decomposition of IMFs and enhanced time-frequency resolution [45, 46].

The EMD method is a crucial step in the HHT process, as it reduces the given data into a set of IMFs, which are suitable for Hilbert spectral analysis. IMFs represent simple oscillatory modes, similar to harmonic functions but with variable amplitude and frequency along the time axis.

The process of extracting an IMF is called sifting, which involves the following steps:

1. Identify all the local extrema (maxima and minima) in the original signal $x(t)$.
2. Connect all the local maxima by a cubic spline line as the upper envelope.
3. Repeat the procedure for the local minima to produce the lower envelope.
4. Calculate the mean of the upper and lower envelops, and denote the mean as $m(t)$
5. Subtract the mean from the original signal to obtain the first IMF candidate $h(t)$ using Eq. (3.11).

$$x(t) - m(t) = h(t) \tag{3.11}$$

6. Check if $h(t)$ satisfies the definition of an IMF, being symmetric and having all maxima positive and all minima negative. If not, repeat steps 1-5 with $h(t)$ as the new input signal until an IMF is obtained.
7. Once the first IMF, denoted as $c_1(t)$, is obtained, subtract it from the original signal to get the residual $r_1(t)$ by Eq. (3.12):

$$r_1(t) = x(t) - c_1(t) \tag{3.12}$$

8. Continue this process until the residue becomes a monotonic function or has at most one extremum.

At the end of the EMD process, the original signal $x(t)$ can be represented as the sum of the IMFs and the final residue as shown in Eq. (3.13).

$$x(t) = \sum_{i=1}^{n} c_i(t) + r_n(t) \tag{3.13}$$

where n is the total number of IMFs, $c_i(t)$ is the i^{th} IMF, and $r_n(t)$ is the final residue.

The Hilbert transform is applied to each IMF obtained from the EMD process. The Hilbert transform of a real-valued function $c_i(t)$ is given by the Cauchy principal value of the convolution with the function $1/\pi t$:

$$H[c_i(t)] = \left(\frac{1}{\pi}\right) P.V. \int_{-\infty}^{+\infty} (\frac{c_i(t)}{t - T}) dT \tag{3.14}$$

where $P.V.$ denotes the Cauchy principal value of integral.

The analytic signal $z_i(t)$ associated with each IMF $c_i(t)$ is defined by Eq. (3.15):

$$z_i(t) = c_i(t) + jH[c_i(t)] = a_i(t)e^{(j\theta_i(t))} \tag{3.15}$$

where $a_i(t)$ is the instantaneous amplitude, and $\theta_i(t)$ is the instantaneous phase, given by:

$$\begin{gathered} a_i(t) = \sqrt{(c_i{}^2(t) + H^2[c_i(t)])} \\ \theta_i(t) = \text{arctan}\ (H[c_i(t)]/c_i(t)) \end{gathered} \tag{3.16}$$

The instantaneous frequency $\omega_i(t)$ is then obtained by differentiating the instantaneous phase:

$$\omega_i(t) = d\theta_i(t)/dt \tag{3.17}$$

Finally, the Hilbert spectrum $H(\omega, t)$ is constructed by plotting the instantaneous amplitude and frequency of each IMF in the time-frequency plane:

$$H(\omega, t) = Real \sum_{i=1}^{n} a_i(t)e^{j\int w_i(t)dt} \tag{3.18}$$

The Hilbert-Huang transform provides a time-frequency representation of the original signal, allowing for the analysis of nonstationary and nonlinear data such as those music signals with varying amplitude, dynamic tempo changes, harmonic interactions between frequencies and complex resonance behaviors in instruments, etc. The EMD breaks down the signal into Intrinsic Mode Functions (IMFs) that synergize well with the Hilbert transform resulting in a time distribution referred to as the Hilbert spectrum.

The HHT has been successfully applied in fields such as geophysics, oceanography and biomedical engineering for examining nonlinear signals. Its adaptability and ability to offer time frequency resolution make it a valuable tool for signal processing and data analysis.

3.1.2 Matrix factorization

Matrix factorisations are mathematical techniques that decompose a matrix into a product of terms with specific properties, such as diagonality or orthogonality. The primary goal of matrix factorization is to uncover hidden structures or patterns within the data leading to an interpretable representation of the matrix. By deconstructing the matrix into its constituent parts, matrix factorization techniques can reveal concealed relationships, reduce dimensionality, and support signal processing tasks.

Non-negative matrix factorisation techniques are specifically applied in situations where the underlying data are inherently non-negative, such as magnitude or intensity values. These methods can decompose complex data into positive factors that often have clear physical interpretations. In the field of signal processing, non-negative matrices are commonly used to represent spectrograms of audio signals. A spectrogram is a visual representation of the spectrum of frequencies in a signal as it varies with time. Each element of the matrix corresponds to the energy or intensity of a specific frequency at a particular point in time. Methods such as non-negative matrix factorization (NMF) can effectively analyze these spectrograms by decomposing the non-negative matrix into two or more non-negative matrices. This decomposition process uncovers underlying spectral patterns and their temporal activations, providing a meaningful breakdown of the audio signal. The resulting factors often correspond to distinct sound sources or recurring acoustic patterns within the signal.

Moreover, non-negative matrix factorization techniques tend to produce sparse representations, meaning that most of the elements in the factorized matrices are zero or close to zero, with only a few significant non-zero values. Sparsity is desirable in many signal processing applications as it helps to highlight the most relevant or informative features, leading to more efficient storage, transmission, and processing of the data. Sparse representations can also enhance the interpretability of the results, as they focus on the essential components of the signal while suppressing noise or irrelevant information.

Furthermore, non-negative matrix factorization techniques have been successfully used in signal processing tasks such, as source separation, music transcription, speech enhancement and image denoising. By leveraging the non-negative and sparse properties of the factorized matrices, these techniques can effectively extract important elements from intricate signals enabling, in depth analysis, categorization and rebuilding of the information.

The following sections examine two prominent matrix decomposition methods in signal processing: Non-negative Matrix Factorization (NMF) and Probabilistic Latent Component Analysis (PLCA). These techniques have been extensively implemented across domains showcasing their efficacy in revealing hidden patterns, isolating sources, and offering understandable representations of non-negative data matrices.

3.1.2.1 Non-Negative Matrix Factorization (NMF)

Non-negative Matrix Factorization (NMF) is an approach for decomposing a non-negative matrix into a product of two non-negative matrices. It was first introduced by Lee and Seung [47] as a tool for estimating the underlying structure of non-negative data. The primary goal of NMF is to represent a non-negative matrix X as a product of two non-negative matrices W and H, while minimizing errors in reconstruction.

Mathematically, NMF can be formulated as follows:

$$X \approx WH \tag{3.19}$$

where X is an $m \times n$ non-negative matrix, W is an $m \times k$ non-negative matrix, and H is a $k \times n$ non-negative matrix where $k < min(m, n)$. The goal is to find the optimal matrices W and H that minimize the difference between X and the product WH, subject to the non-negativity constraints on W and H.

NMF has found significant applications in the field of music transcription, where X typically represents the spectral information of an audio signal, W corresponds to the weightings or basis functions, and H represents the time-varying activations or spectral templates. In this context, NMF can be used to decompose the audio spectrogram into a set of spectral templates and their corresponding time-varying intensities. This decomposition is particularly useful for analyzing and transcribing instruments with distinct spectral profiles, such as piano sounds [47].

To minimize the reconstruction error and find the optimal matrices *W* and *H*, various cost functions can be employed. One common approach is to use a cost function that promotes sparsity in the activations matrix *H*. Sparsity constraints encourage the decomposition to represent the data using a minimal number of active basis functions, leading to a more compact and interpretable representation. Smaragdis and Brown [48] and Bertin et al. [49] have successfully applied NMF with sparsity constraints for music transcription tasks.

NMF shares some similarities with Independent Component Analysis (ICA), another popular technique for blind source separation. ICA, introduced by Comon [50], aims to express a signal model as $x = As$, where x and s are n-dimensional vectors, and A is a non-singular mixing matrix. ICA seeks to identify the underlying sources by finding the latent signals that are maximally independent. The main difference between ICA and NMF lies in the constraints imposed on the factorizing matrices. In ICA, the rows of the mixing matrix A are required to be maximally statistically independent, while in NMF, both matrices W and H are constrained to be non-negative.

Abdallah and Plumbley [51] conducted a study comparing ICA and NMF for polyphonic music transcription. They found that NMF typically delivers superior separation results. Virtanen [52] also evaluated the performance of NMF and ICA for audio source separation and concluded that NMF outperformed ICA in terms of separation quality.

The success of NMF in music transcription and audio source separation can be attributed to its ability to capture the negative structure of audio spectrograms. By imposing non-negativity constraints on the basis functions and activations, NMF can identify meaningful spectral templates and their time varying contributions resulting in a more intuitive and understandable breakdown of the audio signal.

3.1.2.2 Probabilistic Latent Component Analysis (PLCA)

PLCA is a technique used for analyzing non-negative data especially in the realms of audio signal processing and music information retrieval. It operates on a model that represents observed data as a mixture of latent components. The aim of PLCA is to unveil the structure within the data by estimating the parameters of this model.

PLCA was initially introduced by Smaragdis et al. [53] as a probabilistic extension of Non-negative Matrix Factorization (NMF). Unlike NMF, which focuses on the factorization of a non-negative matrix into two non-negative matrices, PLCA models the data as a probability distribution and aims to estimate the latent components and their corresponding weights.

The generative model of PLCA assumes that each observed data point is generated by a mixture of latent components. The probability of observing a particular data point is expressed as a weighted sum of the probabilities of the latent components. Mathematically, given a non-negative data matrix V of size m × n, where m is the number of features and n is the number of observations, PLCA decomposes V into a set of latent components and their corresponding weights:

$$P(i,j) \approx \sum k P(z_k) P(f_i \mid z_k) P(o_j \mid z_k) \tag{3.20}$$

where $P(i,j)$ is the probability of observing feature i and observation j, z_k represents the k-th latent component, $P(z_k)$ is the prior probability of latent component k, $P(f_i \mid z_k)$ is the probability of observing feature i given latent component k, $P(o_j \mid z_k)$ is the probability of observing observation j given latent component k.

The parameters of the PLCA model are estimated using the Expectation-Maximization (EM) algorithm [54]. This algorithm iteratively computes the probabilities of the hidden components (E step). Adjusts the model parameters to maximize the log likelihood of the observed data (M step). The EM algorithm switches between these two steps until convergence, as determined by some appropriate criterion.

PLCA is better than other matrix factorization methods in some ways. Firstly, its probabilistic nature allows for an approach, to handling uncertainties and incorporating knowledge into the model. Secondly, it's easy to add limits or regularization terms like sparsity or temporal continuity to PLCA [55]. Third, PLCA has been shown to be useful for signal processing jobs like separating audio sources [55], music transcription [56], and musical instrument recognition [57].

3.1.3 Other features

To process the data more effectively, it is essential to explore features that capture the focus of attention. Audio Content Analysis (ACA) is a well-known system [58] that utilizes digital signal processing and machine learning techniques to analyze audio signals, extracting useful information and features from audio files for further processing and analysis. Here is a detailed introduction to ACA systems:

The primary aim of ACA systems is to recognize, categorize and process content within audio signals. It has been used across various fields, including music information retrieval, speech recognition, environmental sound analysis, and multimedia content management. The main functions of ACA systems include but not limited to:

1. Audio Feature Extraction includes various features such as spectral features, rhythm features, and pitch features.
2. Audio Classification and Recognition involved sorting signals into categories based on identified characteristics or recognizing specific content within the audio.
3. Music Analysis focuses on analysing the structure of music signals, identifying elements such as rhythm, tonality, and harmony of tracks.
4. Speech Processing covers tasks such as speech recognition, emotion analysis and speaker identification.
5. Environmental Sound Analysis is the study of finding and classifying sounds in the environment, such as traffic noises and crowd noises.

ACA system is a powerful tool that can automatically process and analyze audio signals, extract valuable information. As technology improves, this system becomes more important in many fields because it help us understand and use audio material better. Besides, ACA system can also extract and analyze other important audio features, such as:

1. Mel-Frequency Cepstral Coefficients (MFCC): are widely used spectral features that simulate the non-linear auditory characteristics of the human ear and are highly effective in speech recognition and music analysis.
2. Chroma Features: are a representation of the pitch classes in a signal that are strongly associated with the tonality and harmony of music.
3. Zero-Crossing Rate (ZCR): is a measure of the frequency at which a signal passes the zero point. It is beneficial for differentiating between spoken and non-spoken language, as well as for analysing patterns of rhythm.
4. Spectral Centroid and Spectral Flux: are used to analyse the structure and variations in the spectrum. These attributes are often employed in segmenting audio and identifying music genres.
5. Pitch Features: are crucial for activities such as extracting melodies, tracking pitch, and aligning audio to MIDI.

The selection and combination of these features depend on the specific application scenario and task objectives. By thoroughly studying and utilizing these features, ACA systems can provide a more comprehensive and accurate analysis and understanding of audio content, supporting various fields such as music information retrieval, audio classification, and speech recognition. In the future, with the continuous development of new technologies like deep learning, ACA systems are expected to achieve higher levels of semantic understanding and interactive applications.

3.2 Machine Learning for Music Information Retrieval

In this section, typical machine learning methods encompassing both shallow and deep learning approaches are reviewed. Traditional shallow learning strategies such as artificial neural networks (ANNs), have demonstrate fundamental capabilities in music feature extraction and basic classification tasks. However, the emergence of deep learning, particularly convolutional neural networks (CNNs), enables more sophisticated feature learning and pattern recognition. Three typical CNN architectures (i.e., VGG, ResNEt and DenseNet) and reviewed due to their significant contributions to MIR. VGG demonstrates the effectiveness of deep, sequential architectures and small kernel sizes. ResNet introduces residual learning, crucial for training deeper networks. DenseNet provides insights into feature reuse and efficient parameter utilization. These fundamental concepts directly inform the design of music-specific networks.

3.2.1 Artificial Neural Network

Artificial Neural Networks (ANNs) are computational models designed to simulate the way the human brain processes information. They are widely used in the fields of machine learning and artificial intelligence [59]. ANNs consist of a large number of interconnected nodes (or "neurons") distributed across various layers, including an input layer, one or more hidden layers, and an output layer. The link between each layer has a weight that adjusts the strength of the signal passing through it. These weights can be adjusted by utilising training data in order to perform intricate nonlinear mappings, and thus enabling them to carry out tasks such as classification, regression, and clustering.

ANNs can be used to extract features from music signals, such as spectral features, rhythm features, and pitch features [60]. ANNs can be applied to music classification tasks, such as genre classification and emotion analysis [61]. They can also be used for music generation by training a network to learn the structure and style of music [62]. This often involves using more complex network structures such as Recurrent Neural Networks (RNNs) or Variational Autoencoders (VAEs) [63]. Additionally, ANNs are widely used in music recommendation systems to analyze a user's listening history and preferences [64].

3.2.2 CNN

Convolutional Neural Networks (CNNs) are a special type of ANN designed for processing data with a grid structure, such as images [9]. CNNs utilize convolutional layers to establish connections between neurons and distribute weights, resulting in a reduction of model parameters and enhancement of computing performance [10]. This design allows CNNs to effectively extract features and construct hierarchical structures by stacking multiple convolutional layers.

A conventional CNN model [65] shown in Fig. 3-2 typically consists of the following layers:

1. Input Layer: receives the raw input data, such as an image or a spectrogram.
2. Convolutional Layer: executes the convolution operation by moving filters across the input data. Each filter detects specific features or patterns in the input.
3. Activation Function: After each convolutional layer, an activation function is applied to introduce non-linearity into the model. Common activation functions include Rectified Linear Unit (ReLU), Hyperbolic Tangent Activation Function(Tanh) and Sigmoid Activation Function.
4. Pooling Layer: are used to downsample the feature maps obtained from the convolutional layers. They reduce the spatial dimensions of the feature maps while retaining the most important information. Common pooling operations include max pooling and average pooling.
5. Fully Connected Layer: After several convolutional and pooling layers, the extracted features are flattened and passed through one or more fully connected layers. These layers learn to combine the extracted features and make predictions or classifications based on the input data.
6. Output Layer: The final layer of the CNN produces the desired output, such as class probabilities for classification tasks or continuous values for regression tasks.

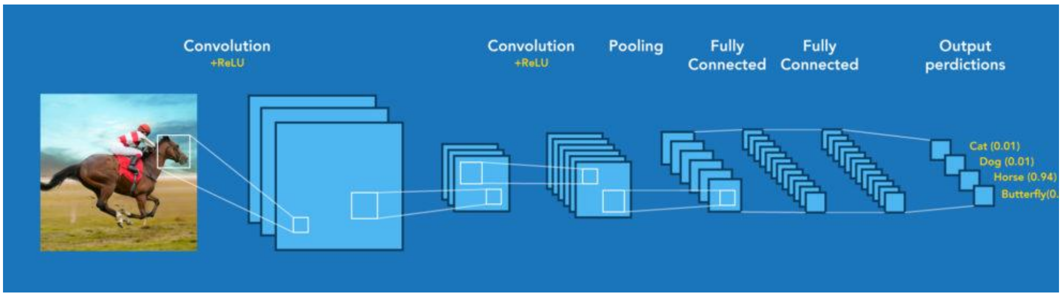


Fig. 3-2. Architecture of LeNet

3.2.3 VGG

The VGG architecture [66], a convolutional neural network (CNN) developed by the Visual Geometry Group at the University of Oxford, is renowned for its effectiveness in image recognition tasks. The model is celebrated for its simplicity and robust design, achieving impressive results in various image recognition challenges. Among the various versions of the VGG model, VGG16 is the most well-known, with the numbers indicating the total number of weighted layers.

As shown in Fig. 3-3, VGG16 consists of a total of 16 weighted layers, including 13 convolutional layers and 3 fully connected layers. The input to the model is an image with dimensions 224 x 224 x 3, indicating that it accepts RGB images of size 224 x 224 pixels. The convolutional layers are arranged in five blocks, with each block followed by a max-pooling layer for downsampling. The number of filters in each convolutional layer increases as the network goes deeper (64, 128, 256, 512, 512). All convolutional layers use small 3x3 filters, which is a key feature of the VGG architecture. By using multiple convolutional layers, the network is able to acquire a deeper understanding of intricate aspects. After the convolutional layers, there are three fully connected (FC) layers. The initial two fully connected (FC) layers consist of 4096 units each, whilst the last FC layer comprises 1000 units. These units correspond to the number of classes included in the ImageNet dataset, which was utilised to train the original VGG-16 model. After each convolutional layer and the first two fully connected layers, the rectified linear unit (ReLU) activation function is applied to introduce non-linearity into the model. The application of the softmax activation function to the last fully connected layer produces a probability distribution across the 1000 classes, hence yielding the ultimate output of the model.

VGG16-like models are frequently used for extracting and analysing features, expanding their usefulness beyond only image identification. Within the field of Music Information Retrieval (MIR), these models may be utilised for tasks such as identifying instruments, classifying music genres, and analysing emotions [67]. Within the realm of music performance analysis, researchers have investigated the use of VGG-style architectures for various purposes, including performance evaluation [68], and expressive performance modelling [69]. These studies have demonstrated the potential of deep learning techniques in capturing the nuances and complexities of musical performances.

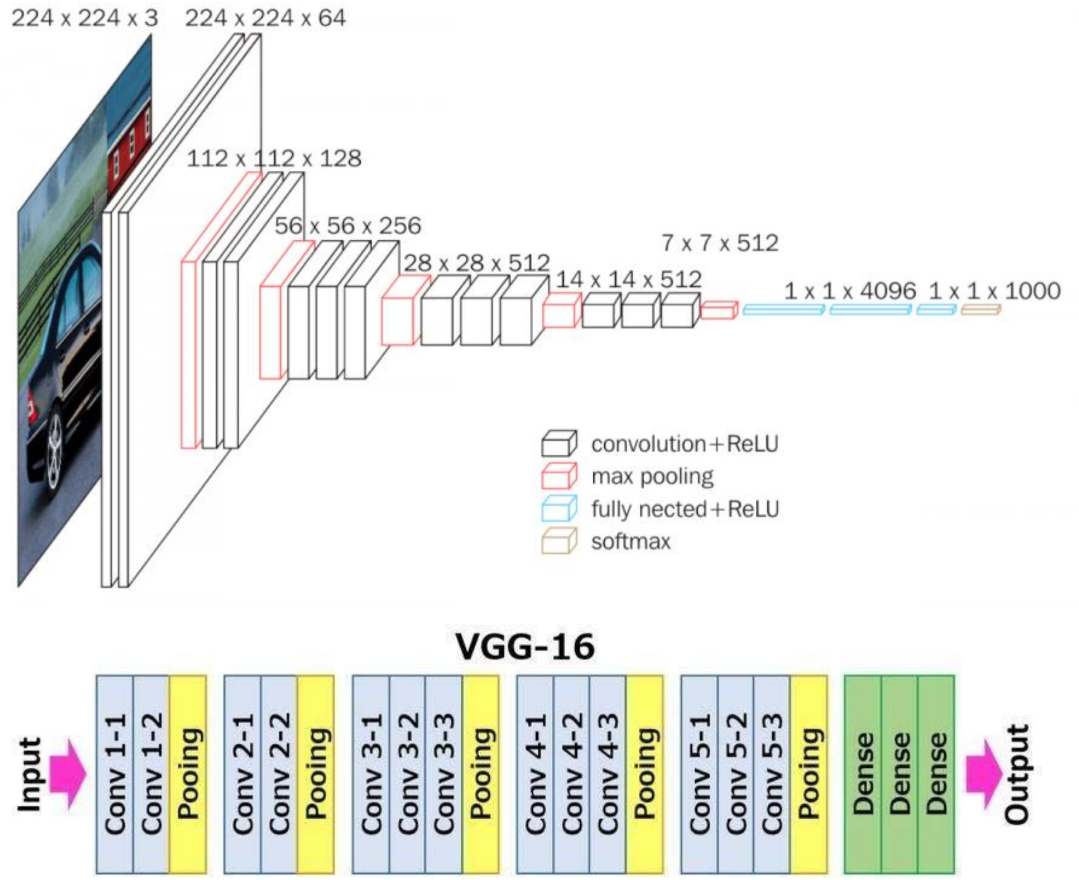


Fig. 3-3. Architecture of VGG-16

3.2.4 ResNet and DenseNet

ResNet (Residual Network) and DenseNet (Densely Connected Network) are both CNN architectures designed to enable the training of deeper neural networks than CNN. ResNet is characterized by its use of residual learning (via skip connections) to address the problem of vanishing gradients in deep models [70]. ResNet architectures typically consist of multiple stacked residual blocks, which allows the network to learn more complex features at different scales. ResNet models are known for their depth, with successful architectures ranging from 18 to 152 layers or even deeper. The skip connections enable the training of these deep networks without suffering from the vanishing gradient problem.

For deeper ResNet models, such as ResNet-50 and ResNet-101 [97], bottleneck blocks are used to improve computational efficiency. These blocks consist of three convolutional layers which are a 1x1 convolutional layer for reducing the number of channels, a 3x3 convolutional layer for learning spatial features, and another 1x1 convolutional layer for increasing the number of channels back to the original size. ResNet models frequently use global average pooling in place of fully linked layers at the end of the network. By doing so, overfitting is prevented and the number of parameters is decreased. After every convolutional layer, batch normalisation layers are frequently included in ResNet models. Batch normalization helps in stabilizing the training process, reducing the sensitivity to initialization, and allowing higher learning rates. The skip connections in ResNet models can be either identity mappings (when the input and output have the same dimensions) or projection mappings (when the dimensions differ). These shortcuts allow the network to learn the residual functions effectively.

ResNet is extensively used and has demonstrated state-of-the-art performance in a number of computer vision applications, such as semantic segmentation, object identification, and image classification. Its success has inspired the development of many subsequent architectures, such as DenseNet [71], Squeeze-and-excitation network [72], and ResNeXt [73].

DenseNet [71] is a convolutional neural network architecture that enhances layer connectivity. Unlike traditional CNNs, DenseNet connects each layer to every other layer, enabling direct access to feature maps from all preceding layers. This dense connectivity promotes feature reuse and improves gradient flow during backpropagation, which mitigates the vanishing gradient problem and results in more efficient and compact models. Small (3x3) convolutions are usually used in each layer of the architecture, together with batch normalisation and ReLU activation, to preserve stability and non-linearity.

The growth rate, which controls how many feature mappings each layer contributes to higher layers, is a crucial component of DenseNet. Bottleneck layers are used to save computation and memory without compromising accuracy by using 1x1 convolutions. Transition layers, comprising batch normalization, 1x1 convolution, and 2x2 average pooling, are used between dense blocks to downsample feature maps and control model complexity. Dense blocks, where layers are densely connected, facilitate efficient feature propagation and reuse, contributing to the network's compactness and learning efficiency.

Both ResNet and DenseNet have been applied to various MIR tasks, such as music genre classification [74], and music emotion recognition [75], etc. These architectures have demonstrated improved performance compared to traditional CNN models, particularly in scenarios where deep networks are required to capture complex musical features and patterns.

3.3 Related work for MIR

3.3.1 Automatic Music Transcription (AMT)

Estimation and tracking of multiple Fundamental Frequencies is one of the major tasks in Automatic Music Transcription (AMT) of polyphonic music analysis [19] and Music Information Retrieval (MIR) [76], which is referred to as a subtask in the Music Information Retrieval Evaluation eXchange (MIREX) [1]. Multiple fundamental

[1] http://www.music-ir.org/mirex/wiki/MIREX_HOME

Frequency Estimation (MFE), also namely Multiple Pitch Estimation (MPE), is challenging in processing simultaneous notes from multiple instruments in polyphonic music [10, 77]. There is often a trade-off between the robustness and efficiency of algorithms that focuses more on complexity rather than single-pitch estimation.

According to Benetos [78], the MPE approaches are categorised into three types, i.e. feature based, spectrogram-factorization based and statistical-model based methods. In feature based methods, signal processing techniques such as the pitch salience function [79] and pitch candidate set score function [80] are used. In spectrogram-factorization methods, both NMF and PLCA approaches have received a lot of attention in recent years [79], and numerous improved versions [11, 81, 82] based on both methods have been published and are recognised as leading spectrogram factorization-based methods in the MPE domain. The statistical model-based methods employ the maximum a posteriori [10] estimation, maximum likelihood or Bayesian theory [83] to detect the fundamental frequencies. It is worth noting that these three distinct types of MPE approaches can be joined or interacted with [79] for a variety of applications.

In recent years, many deep learning (DL) based supervised MPE approaches have also been developed. Cheuk, et al., [84] presented a DL model for AMT by combining the U-Net and Bidirectional Long Short-Term Memory (BiLSTM) neural network modules. He proposed ReconVAT, a semi-supervised AMT framework integrating a U-Net with self-attention, spectrogram reconstruction and virtual adversarial training, to leverage labeled and unlabeled data in the same year [85]. Mukherjee, et al. [86], used statistical characteristics and an extreme learning machine for musical instrument segregation, where LSTM and the recurrent neural network (RNN) [87] were combined to differentiate the musical chords for AMT. Fan [88] proposed a deep neural network to extract the singing voice, followed by a dynamic unbroken pitch determination algorithm to track pitches. Sigtia [13] developed a supervised approach for polyphonic piano music transcription that included a RNN and a probabilistic graphical model. Although DL approaches may provide adequate music transcriptions, they often require high performance computers and excellent graphic processing units

(GPU) to speed-up the lengthy training process [89]. CNN models are widely used in Automatic Music Transcription (AMT). Bittner, et al. [90] apply a CNN model, highlighting its instrument-agnostic capabilities. Kong, et al. [91] also employs a CNN-based model, specifically tailored for detailed piano transcription, including pedal effects. In [92], a comprehensive toolbox that includes various CNN-related models was proposed for transcribing a wide range of musical elements across different instruments. Thanks to the development of the Transformer framework in the NLP, many attention based models have been explored for AMT in the recent year [93-96]. In [93], Sony proposed an automatic piano transcription method that uses a two-level hierarchical time-frequency transformer architecture. In [94], a transformer model has been used to transcribe multiple instruments for a diverse range of styles and combinations of musical instruments. In [95] Google research combine a generic encoder-decoder transformer with a simply greedy decoding strategy for piano transcription. In [96], CNN and Transformer are fused together for AMT where the combination of convolutional blocks and Gated Recurrent Unit (GRU) is used to extract onset, offset and pitch, the combination of convolutional blocks and Transformer is used to extract the velocity. Furthermore, DL algorithms may suffer from inaccurately labelled data, and the performance may be susceptible to the training samples and the learning procedures used. For the AMT task, acquiring large-scale, high-quality labeled data for musical notes can be more challenging [19]. Consequently, an unsupervised method is emphasized, wherein prior cognitive theories and assumptions from previous studies [23, 97, 98] will be used to guide the fundamental pitch detection in polyphonic music pieces.

To distinguish the pitch using harmonic analysis, two types of statistical models are often used. One is the Expectation-maximization (EM)-based algorithms [54], and the other is Bayesian-based algorithms [99]. For EM-based methods, Emiya [100] et al., proposed a Maximum likelihood based method for multi-pitch estimation. Duan [101] proposed a three-stage music transcription system and applied Maximum likelihood for final note tracking. Ben and Amit [102] proposed an unsupervised music transcription framework using expectation maximization to iteratively align separatesourced musical scores with audio recordings and trained a transcription model using convolutional layer and LSTM layer. For Bayesian-based methods, Alvarado [103] et al., combined Gaussian processes and Bayesian models for multi-pitch estimation. Ryo [104] et al., integrate Hidden Markov Model and Bayesian inference together to precisely detect the vocal pitch. Those statistical models can be also considered as shallow learning methods, as data should first be observed to gain some prior knowledge, based on which the experiments should then be conducted. After constant addition of the information of the new samples into prior distribution, the posterior inference can be delivered along with the final results. Although the shallow learning approaches have been widely investigated [105], they still have much room to improve.

Apart from the aforementioned issues, most MPE methods are designed from the viewpoint of signal processing rather than music cognition, resulting in a lack of sufficient underpinning theory and inefficient modelling. To tackle this issue, in Chapter 4, a general framework will be proposed, where music cognitions are used to guide the entire process of MPE. In the pre-processing, inspired by cognitive neuroscience of music [98], the Constant-Q transform (CQT) [38] is employed to transfer the audio signal to Time-frequency spectrogram. The pianoroll transcription is then generated using a conventional matrix factorization approach, Shift-Invariant probabilistic latent component analysis (SI-PLCA) [81]. In the Harmonic structure detection (HSD) process, the cognitions of harmonic periodicity and instrument timbre [23] are used to guide the extraction of multiple pitch. The efficacy of the suggested methodologies has been fully validated by experiments on three publicly available datasets.

3.3.2 Predominant Musical Instrument Recognition (PMIR)

Music information retrieval (MIR) has drawn significant research attention in the last decade and has been used in many applications, such as music retrieval and automatic music transcription, etc. [106]. Instead of manually identifying the rhythm, genre and timbre by ear, MIR techniques automatically label audio data based on their time and frequency information. A sub-task of MIR is predominant musical instrument recognition (PMIR), which enables customers to search music by instruments, as well as making music transcription easier and more accurate [107]. However, PMIR is a very challenging topic and current PMIR approaches have yet to be commercialised due mainly to lack of robust performance. However, it is quite useful in some applications such as assisting automatic music transcription (AMT) detection, crude instrument classification, and instrument characterization.

The identification of a musical instrument depends primarily on its timbre [23]. From the point of view of physics, the timbre produced from an object is determined from its vibrational state, which characterises the object's waveform and harmonic properties. For a specific musical instrument, its spectrum change is very complicated. Due to different playing techniques, instrument condition and recording manner, the same type of musical instrument will have apparent changes in timbre. For timbre analysis, Pons [108] underscores the importance of focusing on timbre as a distinct characteristic of sound through a combination of weight decay regularization and the use of specific filter shapes. Hernandez [109] introduced the multi-head attention mechanism to adaptively highlights the unique and subtle timbral changes among various musical instruments in complex sound samples such as tremolo and pizzicato.

In general, there are two kinds of musical data: monophonic and polyphonic music. In monophonic music, the instrument is played independently. Most of the work on instrument recognition is done under the assumption of independent performance, which simplifies the recognition task. In the case of separate recordings, musical instrument digital interface (MIDI) can store each instrument in a channel, making it easier for a single instrument to be detected. Bhalke et al. [110] proposed a musical instruments classification method based on Mel Frequency Cepstral Coefficient (MFCC) features and a Counter Propagation Neural Network. Their method obtained an accuracy of 91.84% for recognising 19 instruments. Babak and Marcus [111] used frequency domain features with an Artificial Neural Network (ANN) to classify eight types of instruments and compared the results against five state-of-the-art methods. Anushka et al. [112] proposed an approach for string instrument recognition that gave an accuracy of 89.85% using a Support Vector Machine (SVM) and 100% with a

Random Forest classifier on the IRCAM dataset which contained only 4 string-family instruments. Hernandez [109] took mel spectrograms as the input and proposed a multi-head attention mechanism based deep learning model to effectively extract the timbral characteristics from the music recordings and classify musical instruments.

However, music is more often polyphonic than monophonic, such as in a symphony orchestra or recording live scenes. Recognition of a single instrument in a polyphonic music recording is therefore much more difficult, and several attempts have been made for automatic recognition. Olga Slizovskaia et al. [113] extracted instrument features through a standard bag-of-features pipeline and achieved a 67% classification accuracy on IRMAS database [114], which includes 11 different instruments in the recordings. Han et al. [107] integrated MFCC and CNN together to get a classification accuracy of 63.3% on IRMAS dataset. Peter Li et al. [115] achieved 82.74% accuracy on the MedleyDB dataset [116] by applying a CNN model on the raw audio data. In 2018, Yun-Ning et al. [117] achieved an 81.7% accuracy by using the constant Q-transform (CQT) and skip connection methods on MedleyDB and other datasets. In 2019, Siddharth et al. [118] proposed an approach for handling weakly labeled data using an attention enabled deep learning model on the OpenMIC dataset that contains 20 instruments [119]. Their method achieved an average F1-score of approx. 81.03%. Gururani et al. [120] implements a Convolutional Neural Network (CNN) architecture to perform simultaneous tasks, including frame-level instrument identification and timbral feature extraction. In 2000, Racharla et al. [121] took Mel-Frequency Cepstral Coefficients (MFCC) for feature extraction and applies various machine learning models like Support Vector Machines (SVM) and Random Forest for the classification of predominant musical instruments in audio samples. Kratimenos et al., [122] proposed to combine data augmentation methods such as pitch shifting and time stretching with Convolutional Neural Networks (CNNs) together for advanced PMIR. In 2021, the method mentioned in [123] involves using hierarchical structures within a CNN model even with a limited amount of training data, effectively improving performance in few-shot learning scenarios. In 2022, Hsin-Hung at al. [124] combined U-net and CNN architectures together for initial data processing, incorporated noise reduction techniques, transformed data into an Abstract Syntax Tree (AST) format, and concluded with Fully Connected Layers (FCL) for final instrument classification.

Existing approaches for identifying instruments in monophonic music have achieved relatively satisfied performance. However, there is still a large gap for improving the accuracy of instrument recognition in polyphonic music pieces [114]. This is the primary motivation of proposing a new framework for PMIR which uses the Hilbert-Huang Transform (HHT) to generate a feature matrix from music recordings and input these features into a deep learning model for automatically learning instrument features from polyphonic music recordings. The implementation details will be presented in Chapter 5.

3.3.3 Musical Shape Evaluation (MSE)

As a sub-task of music information retrieval (MIR), music performance assessment (MPA) has drawn considerable attention [125] [126]. In practical music education, MPA helps to improve the candidates’ capability of performing and self-evaluation. However, it needs comprehensive music perception and cognition which are mostly built up via long-term practice and understanding. Currently, MPA for piano students is mostly guided by music trainers, resulting in extensive time and resources needed. An AI-driven MPA would be particularly useful for raising the efficacy of music education whilst reducing the cost (Fig. 3-4).

Alignment of score to audio performance is an essential preprocessing step for many music analysis tasks. It links descriptive score symbols to audio features derived from performance, enables studying performances in musical context, relating score semantics to acoustic realizations. In [127], it generated note-level alignments between scores (MusicXML files) and performances (audio recordings). A two-stage system including dynamic time wrapping (DTW)-based audio-to-score alignment and Hidden Markov Model (HMM)-based note transition modelling has been proposed. [128] Extracted beat-synchronous and measure-wise audio features aligned to musical score according to the sync-toolbox pipeline. [129] extracted the features such as chroma, and loudness using some off-the-shelf methods and developed a time-alignment

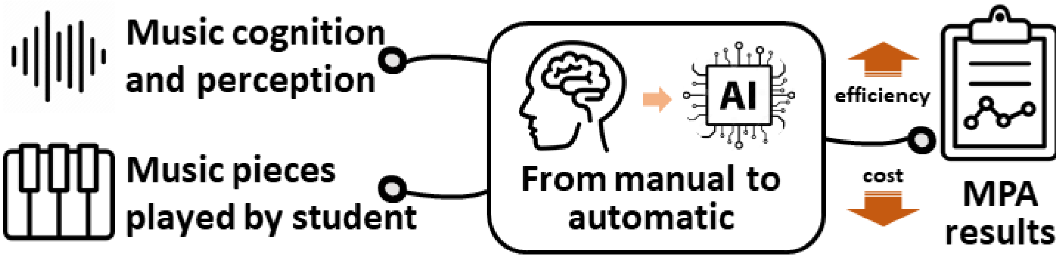

Music cognition and perception
Music pieces played by student
AI
From manual to automatic
efficiency
cost
MPA results

Fig. 3-4. Concept of current MPA in music education.

method to automatically compare and align multiple interpretations of classical piano works based on DTW. In [130], it first used DTW for alignment, and a convolutional neural network architecture was proposed to evaluate student instrumental performance compared to reference standard.

The marking criteria of music scores in some music educational exams, such as the Associated Board of the Royal Schools of Music (ABRSM), are determined by five elements, i.e., pitch, time, tone, shape and overall performance [17]. Many machine learning and deep learning models have been explored to evaluate the pitch, time, rhythm and tone in MPA [126]. Typical works include convolutional neural network (CNN) for local tempo and tempo stability assessment [131], support vector regressor for assessing rhythmic accuracy and tone quality [132], fusion of CNN and recurrent network for pitch extraction [133], and the integration of 2DCNN and 3DCNN for overall evaluation of piano skills [134], etc. In [135], a deep neural network was used to extract the pitch, onset/offset time of the transcription, followed by the HMM for alignment between MIDI data and musical score. In [136], convolutional layer and GRU are combined together in an deep neural network to estimate note-level MIDI velocity of piano performance. In [137], a contrastive-based network is proposed to improve the overall performance assessment where 1-D convolutional layers are stacked followed by contrastive loss.

However, musical shape evaluation (MSE) for MPA remains unaddressed to date. Musical shape (MS), as a unit to build a coherent narrative music environment, is one of the most evident characteristics, which offers a more ecologically valid way of understanding the feeling responses to the music than seeing music as expressive of particular emotional states [34]. Due to its abstraction and multisensory perception, the musicology study has progressed slowly. In 1963, Langer [29] proposed a famous theory of ‘sound the way moods feel’ where musical shape is considered as a fundamental unit in music’s intrinsic properties. In 2006, Eitan and Granot [32] discussed the associations of dynamics, pitch, time, and articulation to musical shape by comparing the music perception with musicians and non-musicians. The result found that time patterns are highly related to the musical shapes. Pitch and dynamics patterns are potentially linked with musical shapes, which required further investigation. To tackle this issue, Küssner and Leech-Wikinson [33] carried out an extensive study in 2014 and discovered that pitch contour does not closely associate with musical shapes though that too is important for musicians. In 2017, Daniel [34] defined that music shape refers to the small dynamics changes in music that can represent feeling and movement states, or any tiny changes varying with time, giving life-like qualities to music. Therefore, the core to MSE is to discriminate the time and dynamics patterns in music pieces.

Alexander Lerch presented a critical review of MPA [125] and pointed out that most MIR researchers neglect the difference between score-like and performance information. Some scholars analyze MPA by focusing on rhythm and timing [131, 138]. Basak[138] presents an automatic system for assessing students' rhythmic pattern imitation in music education. And this article [131] explored various methods for measuring tempo stability. The parameter of playing Techniques has been explored in studies like [139] and [140], these studies analyze the recognition and classification of different playing techniques such as bowing variations, plucking styles, fingerpicking patterns, and other instrumental techniques. Some scholars analyzed Music Performance Analysis (MPA) focusing on rhythm and dynamics. In [141], a recurrent neural network based model was proposed to classify music performers based on their interpretative styles by analyzing timing, dynamics and articulation. In [142], it introduced a new dataset where some basic analysis of loudness and timing variations across pianists and pieces are carried out for note-level performance analysis. [143] focused on the analysis of timing and dynamics as part of evaluating expressive performance in piano music. It specifically looked at how generative models handle these aspects and the perceptual impact of any discrepancies or variations in these parameters when compared to human performances. In [144], a conditional variational autoencoder framework is proposed to predict continuous controllers for dynamics and timing. Although many scholars have studied time and dynamics patterns with the corresponding datasets reported [125], they do not intend to evaluate the music's intrinsic properties. Thus, it brings a large barrier for using existing MIR technique to interpret the music's intrinsic properties.

To advance the field of Music Performance Analysis (MPA), it's imperative to incorporate a novel approach that bridges the gap between human perception and the inherent characteristics of music. For this purpose, a specific dataset is also needed. Consequently, Chapter 5 introduces an innovative MSE dataset, meticulously curated to serve this purpose. This dataset is unique in its composition, featuring performances from three experienced music trainers and ten young students. The conceptual foundation for this dataset, inspired by the illustration in Fig. 3-4, is the differentiation between 'normal music shape' and 'specific music shape'. Performances by the music trainers, informed by their deep understanding of music cognition and perception, are categorized as exhibiting a 'normal music shape.' Conversely, renditions by the students, interpreted through their learning and interpretative lens, are deemed to embody a 'specific music shape'.

On a different note, music shape (MS) is an example of how multisensory capacity is facilitated by the sensorimotor cortex in the brain [34], and the neocortex is organized in a manner to make the underlying processes as efficient as possible. This has motivated us to develop a deep neural model for MSE.

Siamese network is an architecture with two identical branches, where each head takes one input data and the weights and bias of any neural network in each branch are the same [145]. The advantage of this architecture is that it can learn semantic similarity from the two inputs and doesn't rely on massive data to perform well [146]. On the other hand, Residual network [70] has attracted much attention thanks to the strong feature representation of residual blocks. As a result, its variants have been widely used for image classification and produce impressive performance [147]. Taking the advance of both deep learning frameworks, a Siamese residual neural network (S-ResNN) is proposed in Chapter 6. In this model, the music piece played by music trainer and the corresponding one played by student are taken as the inputs which will be transformed into spectrum before sending to both branches. Then the proposed S-ResNN will extract the global spectral feature and identify the musical shapes in the piano pieces. Extensive experiments on the proposed MSE dataset have shown the superiority of S-ResNN when comparing with the combination of machine learning and conventional signal processing methods, and deep learning models such as VGG16 [66], ResNet50 [70] and DenseNet161 [71].

3.4 Evaluation metrics

Evaluation metrics play a crucial role in assessing the performance of predictive models. This section explores a comprehensive set of metrics, including the Area Under the ROC Curve (AUC), Mean Absolute Error (MAE), Maximum F-Measure (MaxF), Overall Accuracy (OA), Average Accuracy (AA), Kappa coefficient (KP), Precision, Recall (or Sensitivity), F-Measure (or F-Score, F1), Confusion Matrix, and Receiver Operating Characteristic (ROC) Curve. These metrics provide a multi-faceted view of model performance, allowing for a thorough understanding of their strengths and weaknesses.

1. AUC (Area Under the ROC Curve): AUC measures the two-dimensional area under the ROC curve, which plots the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The AUC value ranges from 0 to 1, with 1 indicating a perfect model and 0.5 suggesting a model no better than random guessing. The AUC can be calculated using the following equation:

$$AUC = \int_0^1 TPR(FPR) dFPR \tag{3.21}$$

where $TPR(FPR)$ is the true positive rate as a function of the false positive rate.

2. MAE (Mean Absolute Error): MAE measures the average magnitude of errors in a set of predictions, without considering their direction. It is calculated as the mean of the absolute differences between the predicted and actual values over the test sample:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}| \tag{3.22}$$

where n is the number of samples, y_i is the actual value, and $\widehat{y_i}$ is the predicted value.

3. Confusion Matrix: The confusion matrix provides an effective way to evaluate and analyze the performance of a multi-class classification model like the one proposed here. As shown in Fig. 3-5, the confusion matrix aggregates the predictions on the test set into a table that compares the true instrument labels to the predicted labels. The diagonal cells show results where the prediction matches the ground truth, divided into true positives (TP) where the model correctly predicts the positive class, and true negatives (TN) where the model correctly identifies negative examples. Off-diagonal elements are cases where the prediction was incorrect - false positives (FP) when the model mistakenly predicts positive, and false negatives (FN) when the model misses a positive example.
4. OA (Overall Accuracy): OA is the simplest evaluation metric, calculating the proportion of all true results (both true positives and true negatives) among the total number of cases examined:

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.23}$$

5. AA (Average Accuracy): AA computes the accuracy for each class individually and then takes the average. This metric is particularly important in imbalanced datasets where some classes have significantly fewer samples than others:

$$AA = \frac{1}{C}\sum\nolimits_{i=1}^{C} \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \tag{3.24}$$

where C is the number of classes, and TP_i, TN_i, FP_i, and FN_i are the true positives, true negatives, false positives, and false negatives for class i, respectively.

6. KP (Kappa coefficient): KP is to measure the interrater reliability that represents the degree of similarity between the change map and the ground truth defined as follows:

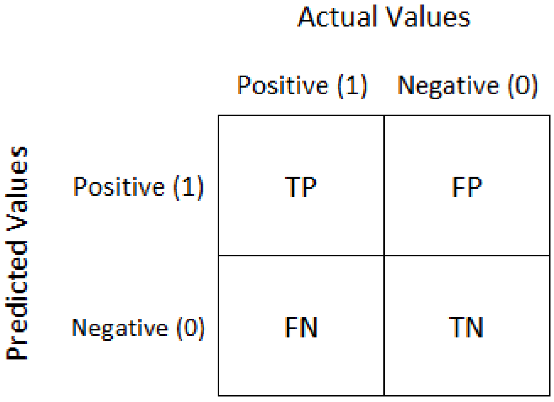
$$KP = \frac{OA - PRE}{1 - PRE}$$

$$PRE = \frac{(TP + FP)(TP + FN) + (FN + TN)(FP + TN)}{(TP + TN + FP + FN)^2} \tag{3.25}$$

7. Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positives:

$$Precision = \frac{TP}{TP + FP} \tag{3.26}$$

8. Recall (or Sensitivity): Recall, also known as sensitivity or true positive rate, is the ratio of correctly predicted positive observations to all observations in the actual class:



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Fig. 3-5. Tidy representation of confusion matrix

$$Recall = \frac{TP}{TP + FN} \tag{3.27}$$

9. F-Measure (or F-Score, F1): The F-Measure is the harmonic mean of precision and recall, providing a balanced measure of a test’s accuracy:

$$F - Measure = 2\frac{Precision \cdot Recall}{Precision + Recall} \tag{3.28}$$

The F-Measure ranges from 0 to 1, with 1 indicating perfect precision and recall.

10. MaxF (Maximum F-Measure): The MaxF is the maximum value of the F-Measure across all possible thresholds.
11. ROC Curve: The ROC Curve is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. It plots the true positive rate (TPR) against the false positive rate (FPR) at different threshold settings. The ROC Curve provides a visual representation of the trade-off between sensitivity and specificity.

3.5 Chapter Summary

This chapter presented a comprehensive review of the technical foundations and related work in music information retrieval. It begins with an examination of music signal pre-processing techniques, including time-frequency representations such as STFT, CQT, and the Hilbert-Huang Transform. These fundamental signal processing methods serve as the backbone for musical analysis.

The chapter then explored matrix factorization techniques, particularly focusing on Non-Negative Matrix Factorization (NMF) and Probabilistic Latent Component Analysis (PLCA), which have proven effective in decomposing complex musical signals. Additionally, the evolution of machine learning approaches in music information retrieval is discussed, tracing the progression from traditional artificial neural networks to advanced architectures such as CNN, VGG, ResNet, and DenseNet.

A significant portion of the chapter was dedicated to reviewing related work in three key research areas: Automatic Music Transcription (AMT), Predominant Musical

Instrument Recognition (PMIR), and Musical Shape Evaluation (MSE). The major challenges for three research areas are summarized below

1. Automatic Music Transcription (AMT):
    - Current methods focus primarily on signal processing rather than music cognition;
    - Lack of sufficient underpinning theory in existing approaches;
    - Limited accuracy in polyphonic music transcription.
2. Predominant Musical Instrument Recognition (PMIR):
    - Significant performance gap between monophonic (91.84%) and polyphonic (67%) recognition;
    - Limited ability to handle complex multi-instrument scenarios;
    - Need for more efficient feature extraction methods for polyphonic music.
3. Musical Shape Evaluation (MSE):
    - Absence of comprehensive frameworks for evaluating musical shape;
    - Limited understanding of the relationship between technical and expressive aspects;
    - Lack of standardized datasets for musical shape analysis.

To tackle those challenges, the following chapters will introduce the proposed models. Chapter 4 presents a novel cognitive-guided framework for Multiple Pitch Estimation (MPE), uniquely integrating a cognitive understanding of harmonic periodicity and instrument timbre to enhance multiple pitch extraction.

Chapter 4

4 A Music Cognition–guided Framework for Multi-pitch Estimation

4.1 Introduction

This chapter addresses an effective system for multi-pitch estimation (MPE) of polyphonic music wherein a novel harmonic structure detection (HSD) method is presented. The proposed unsupervised MPE system is based on the Constant-Q transform and a state-of-the-art matrix factorization method called Probabilistic Latent Component Analysis (PLCA) that resolves the polyphonic short-time magnitude log-spectra for multiple F0 estimation and source-specific feature extraction. The proposed HSD method detects the pitches by analyzing the characteristics of contiguous notes and the relationship between fundamental frequency and harmonic frequencies, where correlative music knowledge and probability model are used to guide the key process. In addition, the performance of this MPE system is compared to a number of existing state-of-the-art approaches (seven unsupervised and four supervised) on three widely used datasets (i.e. MAPs, BACH10, and TRIOS) in terms of F-measure values. The experimental results show that the new MPE method provides the best overall performance compared to other existing methods.

The major contributions of this chapter may be highlighted as follows. First, a new HSD model that incorporates music cognition for multiple fundamental frequency extraction is proposed. Second, a new note tracking method guided by music connectivity and multi-pitch model is proposed. By combining conventional pianoroll transcription approaches and the proposed HSD model, a new music cognition guided optimization framework is introduced for MPE. Experimental results on three datasets have demonstrated the merits of the proposed approach, when benchmarked with 11 state-of-the-art methods.

The rest of the chapter is structured as follows: Section 4.2 describes the implementation of the proposed harmonic structure detection method. Section 4.3 presents the experimental results and performance analysis. Finally, a thorough conclusion is drawn in Section 4.4.

4.2 Cognition guided multiple pitch estimation

4.2.1 System Overview

The objective of this work is to detect the multiple pitch from music pieces of mixed instruments, where an MPE system is proposed, which contains three key modules, i.e., pre-processing, harmonic structure detection and note tracking. Preprocessing covers a standard procedure, in which an input music signal needs to go through a time-frequency (TF) representation and matrix factorization for feature extraction. The overall diagram of the MPE framework is illustrated in Fig. 4-1, where the implementation details are presented in the rest sub-sections.

4.2.2 Pre-Processing

According to the cognitive neuroscience of music [98, 148], before selectively stimulating the auditory cortex, different frequencies within the music need be first filtered by the human cochlea. As the frequency of human auditory perception is logarithmically distributed [38], there is a greater discrimination when hearing relatively lower frequencies. The Constant-Q transform (CQT) [39], based on the FFT principle, can process a logarithmic compression similar to that of the human’s cochlea helical structure [39]. Therefore, the CQT is employed as the TF representation module to derive the TF spectrogram, as it is efficient at lower frequencies. There are fewer frequencies required in a given range, which has testified its usefulness when the frequency distribution in several octaves is discrete.

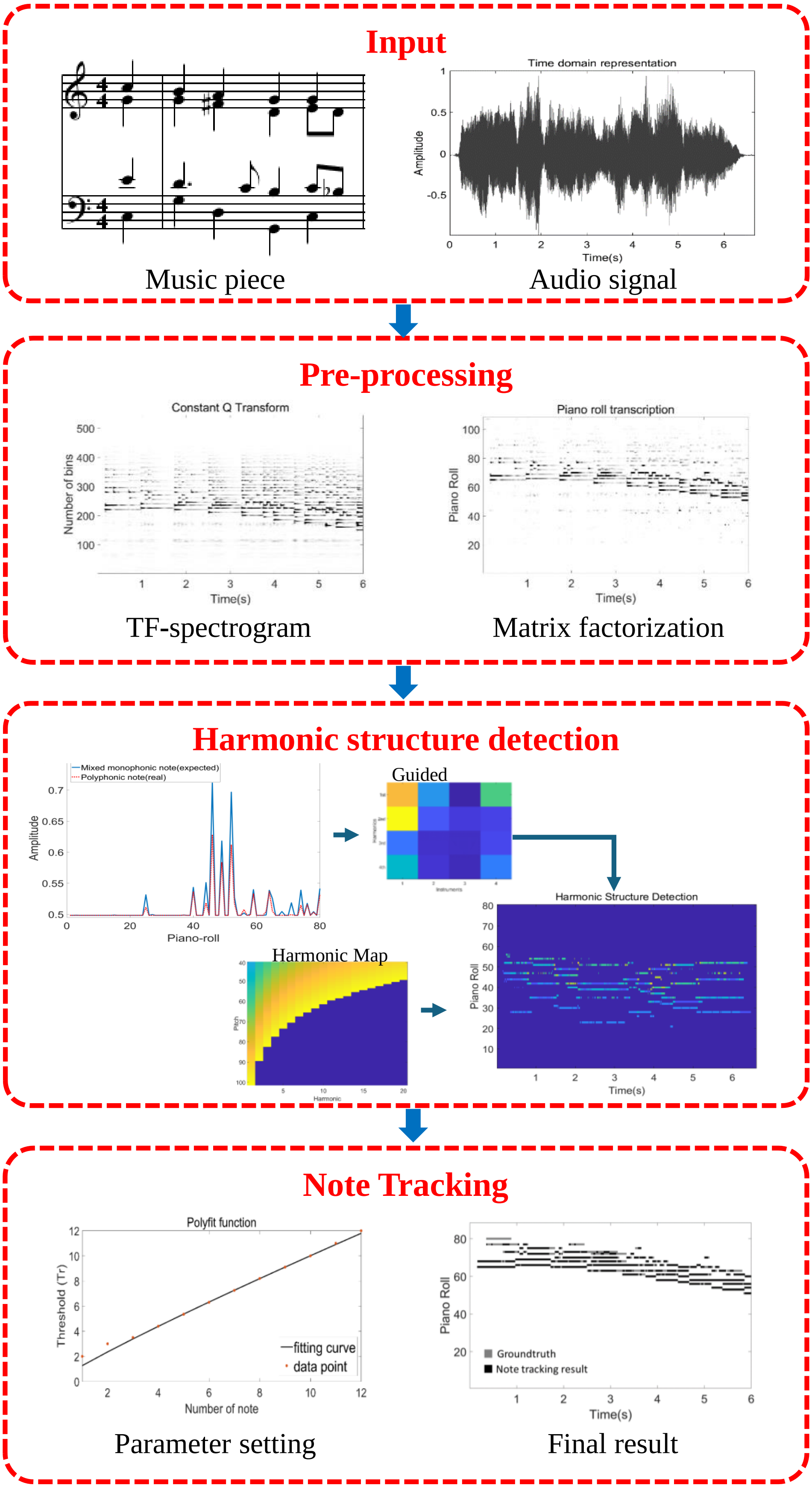


Fig. 4-1. The overall MPE system.

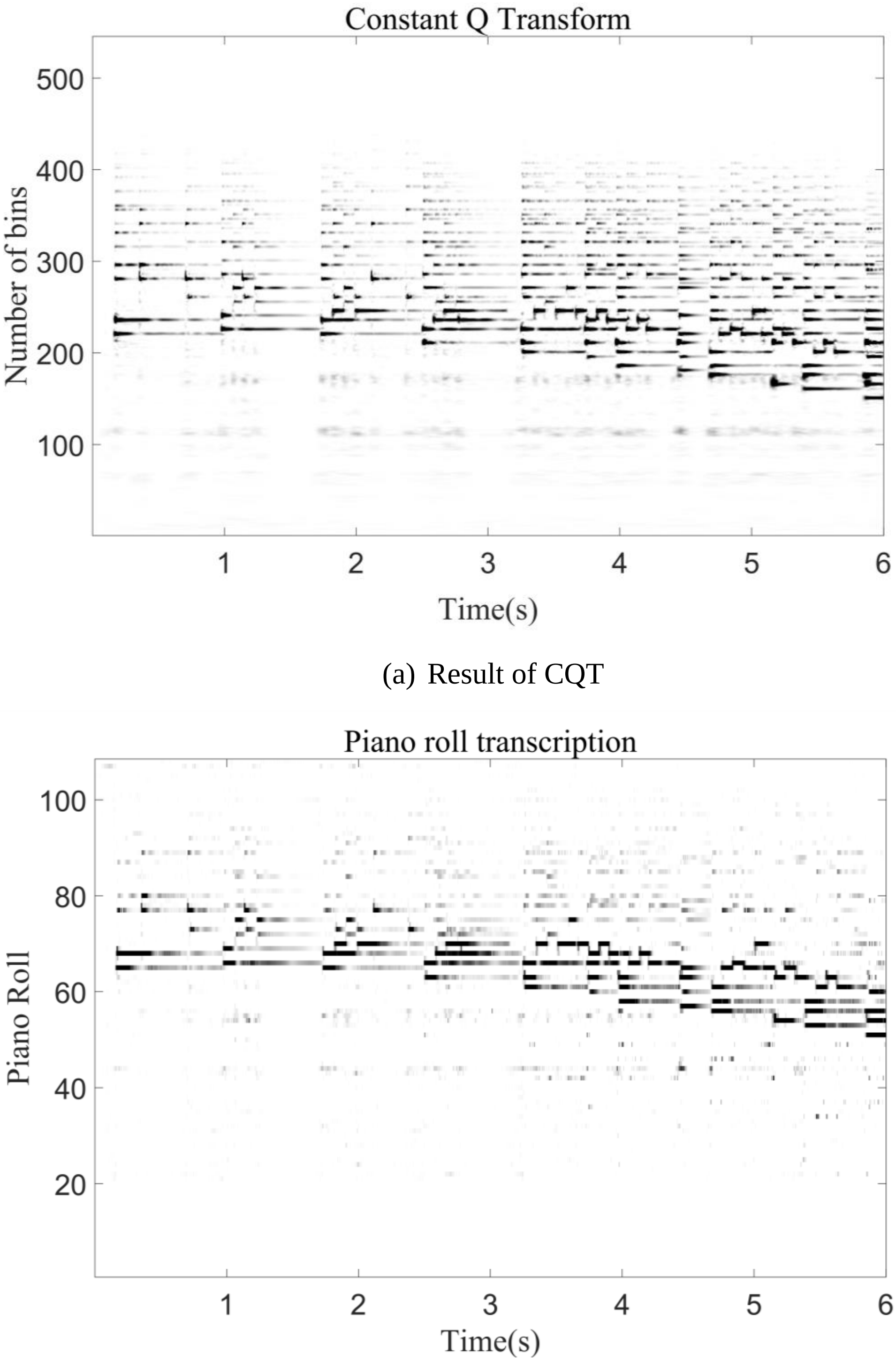
Meanwhile, an increased frequency bins correlates to a decrease in the temporal resolution rate, making it suitable for auditory applications. A spectral resolution of 60 bins per octave is used as suggested by Brown [38]. The outputs from the TF transformation are linear when using the Fast Fourier Transform (FFT) to analyse the frequency (Fig. 4-2 (a)).

In the matrix factorization module, the CQT spectrogram results are used as the input, approximately modelled as a bivariate probability distribution $\boldsymbol{P}(p,t)$. The output of this module is a 2-dimensional non-binary representation of pianoroll transcription (a pitch vs. time matrix shown in Fig. 4-2 (b)). In this chapter, the fast Shift-Invariant probabilistic latent component analysis (SI-PLCA) [48] is used for automatic transcription of polyphonic music, as it is extremely useful for log-frequency spectrogram, due to the same inter-harmonic spacing for all periodic sounds [5]. Given an input signal $\boldsymbol{X}_t$, the output of CQT is a log-frequency spectrogram $\boldsymbol{V}_{z,t}$ that can be considered as a joint time-frequency distribution $\boldsymbol{P}(z,t)$ where z and t denote the frequency and time, respectively. After applying the SI-PLCA, $\boldsymbol{P}(z,t)$ can be further decomposed into several components by [48]:

$$\boldsymbol{V}_{z,t} = \boldsymbol{P}(z,t) = \boldsymbol{P}(t)\sum_{d,f,s}\boldsymbol{P}(z-f|s,d)\boldsymbol{P}_t(f|d)\boldsymbol{P}_t(s|d)\boldsymbol{P}_t(d) \tag{4.1}$$

where d,f,s are latent variables which denote respectively the pitch index, pitch-shifting parameter, and instrument source. In Eq. (4.1), $\boldsymbol{P}(t)$ is the energy distribution of the spectrogram, which is known from the input signal. $\boldsymbol{P}(z-f|s,d)$ denotes the spectral templates for a given pitch p and instrument source s with f pitch shifting across the log-frequency. $\boldsymbol{P}_t(f|d)$ is the log-frequency shift for each pitch on the time frame t, $\boldsymbol{P}_t(s|d)$ represents instrumentation contribution for the pitch in the time frame t, and $\boldsymbol{P}_t(d)$ is the pitch contribution which can be considered as transcription matrix on the time frame t. Since there are latent variables in this model, the expectation maximization (EM) algorithm [54] is often used to iteratively estimate the corresponding unknown variables.

In the Expectation step, the Bayes's theorem is adopted to estimate the contribution of the latent variables *d*, *f*, *s* for reconstruction of the model:



(a) Result of CQT

(b) Result of PLCA

Fig. 4-2. Illustration of input music signal TF representation module and pianoroll transcription module, the range from 200-300 bins in (a) are probably corresponding to 40-60 pitches in (b).

$$\boldsymbol{P}_t(d,f,s|z) = \frac{\boldsymbol{P}(z-f|s,d)\boldsymbol{P}_t(f|d)\boldsymbol{P}_t(s|d)\boldsymbol{P}_t(d)}{\sum_{d,f,s}\boldsymbol{P}(z-f|s,d)\boldsymbol{P}_t(f|d)\boldsymbol{P}_t(s|d)\boldsymbol{P}_t(d)} \tag{4.2}$$

In the Maximization step, the posterior of Eq. (4.2) is used to maximise the log-likelihood function in Eq. (4.3), which leads to the update of Eqs. (4.4)-(4.7). As suggested in [48], this step can converge after 15-20 iterations. Finally, the pianoroll transcription $\boldsymbol{P}(d,t) = \boldsymbol{P}(t)\boldsymbol{P}_t(d)$ is calculated by the following equations:

$$\mathcal{L} = \sum_{z,t} \boldsymbol{V}_{z,t} \log(\boldsymbol{P}(z,t)) \tag{4.3}$$

$$\boldsymbol{P}_t(z|s,d) = \frac{\sum_{f,t}\boldsymbol{P}_t(d,f,s|z+f)\boldsymbol{P}(z+f,t)}{\sum_{f,w,t}\boldsymbol{P}_t(d,f,s|z+f)\boldsymbol{P}(z+f,t)} \tag{4.4}$$

$$\boldsymbol{P}_t(f|d) = \frac{\sum_{z,s}\boldsymbol{P}_t(d,f,s|z)\boldsymbol{P}(z,t)}{\sum_{f,z,s}\boldsymbol{P}_t(d,f,s|z)\boldsymbol{P}(z,t)} \tag{4.5}$$

$$\boldsymbol{P}_t(s|d) = \frac{\sum_{z,f}\boldsymbol{P}_t(d,f,s|z)\boldsymbol{P}(z,t)}{\sum_{s,z,f}\boldsymbol{P}_t(d,f,s|z)\boldsymbol{P}(z,t)} \tag{4.6}$$

$$\boldsymbol{P}_t(d) = \frac{\sum_{z,f,s}\boldsymbol{P}_t(d,f,s|z)\boldsymbol{P}(z,t)}{\sum_{d,z,f,s}\boldsymbol{P}_t(d,f,s|z)\boldsymbol{P}(z,t)} \tag{4.7}$$

4.2.3 Harmonic Structure Detection

This section is the core of the proposed MPE system where music theories in terms of the pattern of beat length and assumption of equal energy between mixed monophonic and polyphonic music pieces are used to guide the model for the extraction of the multiple fundamental frequencies from a mixture of music sources.

For a given piece of music, the time domain representation is illustrated in the input module in Fig. 4-1. The results of CQT and SI-PLCA are given in Fig. 4-2 (a) and Fig. 4-2 (b), respectively. Upon observing Fig. 4-2 (b), the fundamental pitch and its harmonics have been highlighted by the shaded black and grey strips. However, there is considerable noise and redundant information represented by small and grey dots which may be misconstrued for pitches at lower frequencies. Furthermore, the white gaps in the black and grey strips indicate frequency information that has been lost in the analysis. This suggests that the consistency of fundamental pitch is insufficient if considered frame by frame (each frame was set to 10ms). To address these inconsistencies, the HSD method is proposed followed by a note tracking process (Fig. 4-1).

The proposed HSD includes two main stages. In the first stage, the pianoroll transcription $\boldsymbol{P}(p,t)$ is normalised into [0,1] by using the following max-mean sigmoid activation function [149]:

$$\boldsymbol{PN} = \frac{1}{1+e^{-\boldsymbol{z}}} \text{ where } \boldsymbol{z} = \frac{\boldsymbol{P}(p,t) - mean(\boldsymbol{P}(p,t))}{\max(\boldsymbol{P}(p,t)) - \min(\boldsymbol{P}(p,t))} \tag{4.8}$$

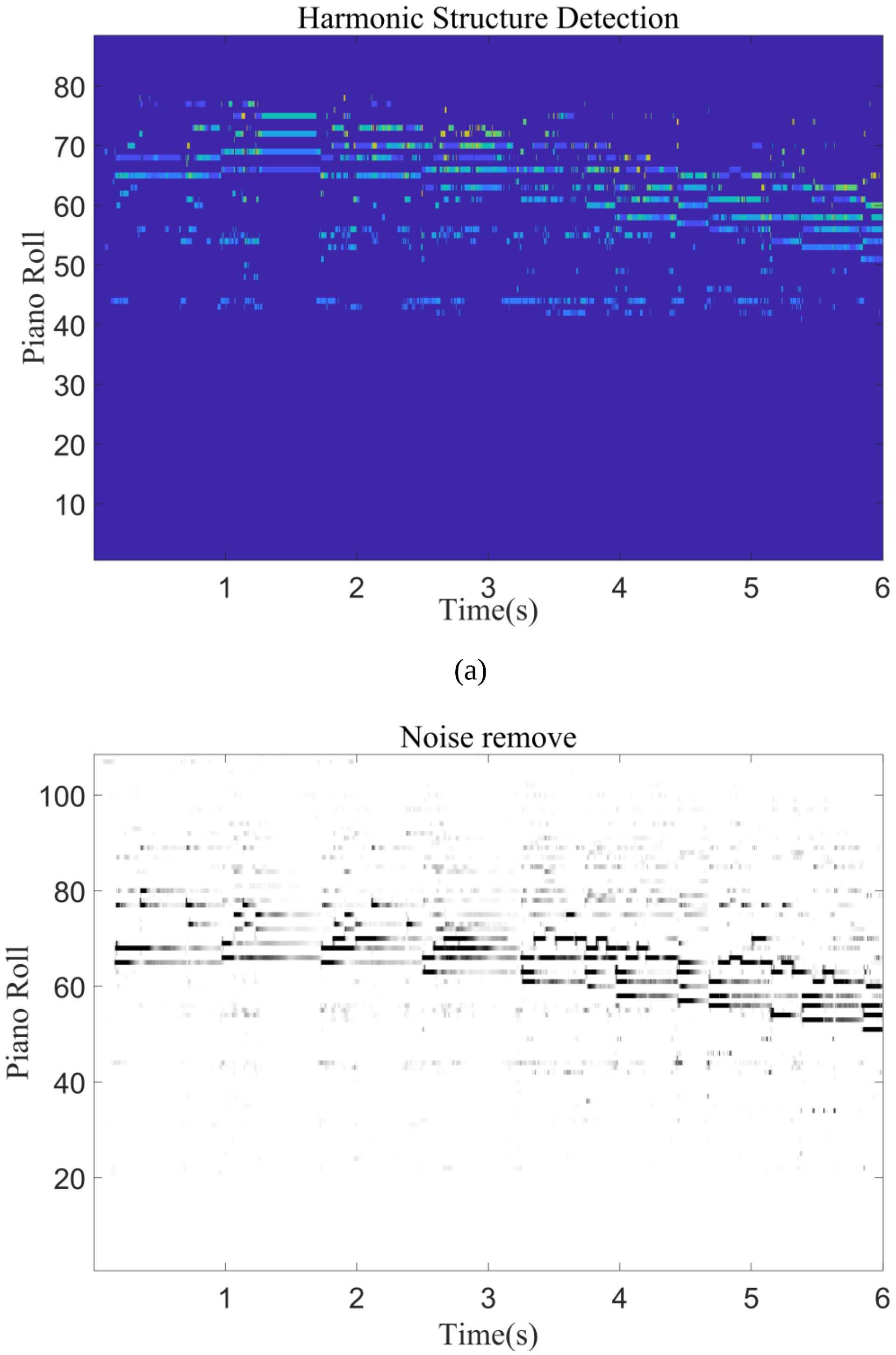
where $\boldsymbol{PN}$ represents the normalised $\boldsymbol{P}(p,t)$. By applying a mean filter in Eq. (4.8), the spectrogram can be smoothed. For extreme values which are too large or too small than expected, they can also be rationalised. For any $\boldsymbol{PN}$, the value of $\boldsymbol{PN}_t$ at time *t* can be expressed by Eq. (4.9).

$$\boldsymbol{PN}_t = (\boldsymbol{PN}_{t-1} + \boldsymbol{PN}_t + \boldsymbol{PN}_{t+1})/3 \tag{4.9}$$

$$\boldsymbol{PF}_t = \boldsymbol{PN}_t * \delta; \quad \delta = \begin{cases} 1, & if\ \boldsymbol{PN} > TH_1 \\ 0, & otherwise \end{cases} \tag{4.10}$$

Inspired by the music theory that most high-order harmonic components are in the high frequency range with low amplitude [97], a two-step hard constrain is used to remove most of the high frequency components, noise and redundancy. First, a fixed threshold TH_1 is applied in Eq. (4.10) to remove small values. Based on the characteristic of sigmoid function (Eq.(4.8)), TH_1 is set to 0.5. Finally, the filtered result $\boldsymbol{PF}$ of the whole frames is obtained and shown in Fig. 4-3(a).

In the second step, the statistics of the beat length is used to guide the removal of noise and redundant information. According to the cognition of music perception, most notes in musical rhythms have a large number of crotchets and quavers, but fewer numbers of semiquavers and demisemiquavers [150]. The rate of occurrence of different notes in the BACH10 database were observed and measured according to the ground truth. A plot was generated of time vs. rate of occurrence in Fig. 4-4, with the labelled fractions (i.e. $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}$) denoting minim, crotchet, quaver, semiquaver and demisemiquaver, respectively. Fig. 4-4 illustrates that the rate of occurrence of



(a)

(b)

Fig. 4-3 Results from the first step (a) and second step (b) in HSD module.

crotchets and quavers is larger than that of the demisemiquavers, semiquavers and minims. Especially, the number of demisemiquavers and semiquavers are extremely low. Furthermore, if the length of a semibreve is defined as τ, the length of a

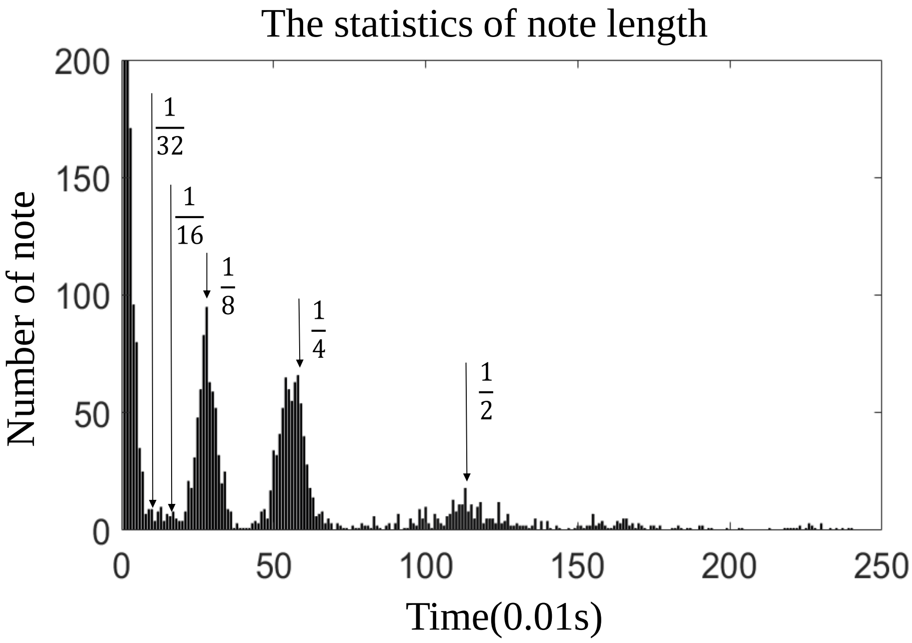


Fig. 4-4. The relationship between time (note type) and note appearance number extracted from the BACH10 database.

demisemiquaver is $\tau/32$. Any notes shorter than a demisemiquaver will be removed in $\boldsymbol{PF}$ before any further processing in the second stage.

In Fig. 4-4, a peak value is identified at the initial time steps of the simulation and this may be due to two reasons. Firstly, manually played music may contain some timing errors, for example holding a note for its precise duration for every note in the piece may be impossible. Secondly, ornaments such as vibrato and glissando may be mistakenly performed despite not being present on the music score. The length of such vibrato and glissando are equal to a demisemiquaver or lower [151]. To extract more of the main body of multiple pitch, factors such as human playing habits or ornaments are ignored in the proposed work. Relevant results given in section 3 demonstrate that the multiple pitch is highlighted whilst removing most of the unwanted noise.

After filtering the amplitudes from PLCA, the HSD framework was proposed to detect the fundamental pitch in the second stage. The flowchart in Fig. 4-5 outlines the process of HSD, and Table 4-I lists the description of each parameter. As described in the flowchart in Fig. 4-5, the output from previous steps will be analysed in two domains i.e., pitch domain $\boldsymbol{PD}$ and energy domain $\boldsymbol{ED}$. In this context, each frame of $\boldsymbol{PF}$ is split into two vectors, $\boldsymbol{PD}(n)$ and $\boldsymbol{ED}(n)$. $\boldsymbol{PD}(n)\epsilon\mathbb{R}^{N_0\times 1}$ is non-zero notes

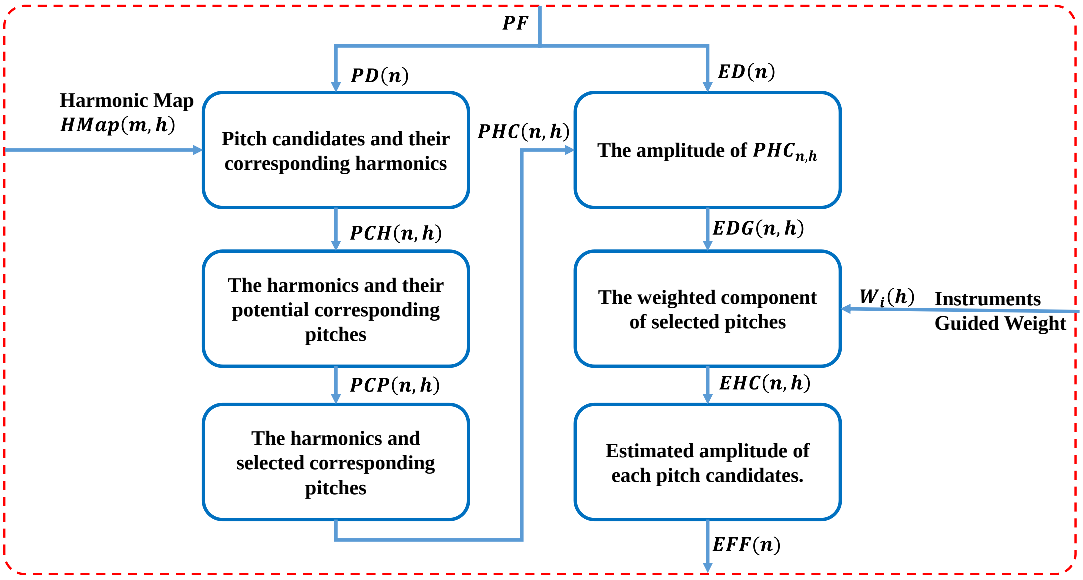

PF
$PD(n)$
$ED(n)$
Harmonic Map $HMap(m,h)$
Pitch candidates and their corresponding harmonics
$PHC(n,h)$
The amplitude of $PHC_{n,h}$
$PCH(n,h)$
$EDG(n,h)$
The harmonics and their potential corresponding pitches
The weighted component of selected pitches
$W_i(h)$
Instruments Guided Weight
$PCP(n,h)$
$EHC(n,h)$
The harmonics and selected corresponding pitches
Estimated amplitude of each pitch candidates.
$EFF(n)$

Fig. 4-5. Flowchart of the proposed HSD

index in each frame, $\boldsymbol{ED}(n)\epsilon\mathbb{R}^{N_0\times 1}$ is the amplitude of $\boldsymbol{PD}(n)$, $N_0$is the number of non-zero notes. As seen, the process is only applied once on the non-zero notes rather than the whole frame, because there is no need to analyse those zero-value notes for efficiency. $\boldsymbol{PCH}(n,h)$ establishes potential relationships between harmonics and their corresponding pitches. $\boldsymbol{PCP}(n,h)$ refines the relationships by selecting the most relevant corresponding pitches. $\boldsymbol{EDG}(n,h)$ computes weighted components of the selected pitches using instrument-guided weights $\boldsymbol{W}_i(h)$. $\boldsymbol{EHC}(n,h)$ estimates the amplitude of each pitch candidate using the weighted components. $\boldsymbol{EFF}(n)$ Provides the final estimated amplitude of the pitch candidates. The details of pitch domain analysis and energy domain analysis are described in Section 4.2.4 and 4.2.5, respectively.

4.2.4 Pitch Domain Analysis

After that, a matrix of pitch candidates and their corresponding harmonics $\boldsymbol{PCH}\epsilon\mathbb{R}^{N_0\times N_H}$ can be extended from $\boldsymbol{PD}(n)$. The first column of this matrix is non-zero pitch values and the rest columns have the associated harmonic pitches of each non-zero pitch, where the harmonic pitch is the corresponding pitch value of the harmonic frequency. A harmonic map $\boldsymbol{HMap}\epsilon\mathbb{R}^{M\times N_H}$ is employed here to guide the extension process, which includes the pianoroll number (m) of the fundamental

Table 4-I Description of parameters

Parameters	Definition	Index/Dimension
$\boldsymbol{N_0}$	The number of non-zero fundamental-pitch;	$n\epsilon[1, N_0]$
$\boldsymbol{N_H}$	The number of harmonic-pitch; default is 5	$h\epsilon[1, N_H]$
$\boldsymbol{N_I}$	The number of the instruments in the music piece.	$i\epsilon[1, N_I]$
$\boldsymbol{m}$	Vector of pianoroll	$\mathbb{R}^{N_0\times 1}$
PF	Spectrogram of SI-PLCA after filtering	$\mathbb{R}^{88\times Time}$
PD	Pitch value of *PF*	$\mathbb{R}^{N_0\times 1}$
PCH	Value of pitch candidates and their corresponding harmonics	$\mathbb{R}^{N_0\times N_H}$
PCP	Value of harmonics and their potential corresponding pitches	$\mathbb{R}^{N_0\times N_H}$
PHC	Value of harmonics and selected pitches	$\mathbb{R}^{N_0\times N_H}$
ED	Energy value of *PF*	$\mathbb{R}^{N_0\times 1}$
EDG	Amplitude of fundamental pitch and their corresponding harmonic	$\mathbb{R}^{N_0\times N_H}$
EHC	Amplitude of harmonic components presented in the pitch *n*	$\mathbb{R}^{N_0\times N_H}$
EFF	Final result of pitch amplitude	$\mathbb{R}^{N_0\times 1}$

Table 4-II Example of calculating A4 in the $\boldsymbol{HMap}$

Attribute	Fundamental frequency F_0	Harmonic Frequency kF_0 (Hz）			
		$2F_0$	$3F_0$	$4F_0$	$5F_0$
Frequency (Hz)	440	880	1320	1760	2200
Pianoroll	49	61	68	73	77
MIDI number	69	81	88	93	97
Letter name	A4	A5	E6	A7	C#7/Db7

frequency ($\boldsymbol{F_0}$) and the corresponding harmonic frequency for every note. Following the MIDI tuning standard, the n^{th} non-zero fundamental frequency is transferred to its corresponding pianoroll number using Eq. (4.11). Here, $\boldsymbol{PD}$ needs to be subtracted by 20 due to the difference between the pianoroll and the MIDI number.

$$\begin{gathered}\boldsymbol{PD}(n) = 69 + 12\log_2\left(\frac{\boldsymbol{F_0}(n)}{440Hz}\right) \\ \boldsymbol{m}(n) = \boldsymbol{PD}(n) - 20, \quad |\boldsymbol{m}|\ \epsilon\ [1{,}88]\end{gathered} \tag{4.11}$$

where 69 and 440 are the values of the MIDI number and frequency for the standard A, respectively. 12 is the number of notes in one octave. Given a frequency of the input signal, its harmonic frequencies are multiples of the fundamental frequency. Note that concert A is not always the standard A with 440Hz, it depends on the transposing instruments such as clarinet and horn, etc. In this study, 440Hz is set to the standard A for easy implementation. An example of calculating MIDI number of harmonic frequency in $\boldsymbol{HMap}$ is given in Table 4-II

$\boldsymbol{PCH}(n, h)$ is the h^{th} harmonic pitch component of the pitch *n* where *n* lies within [1, N] and h is within [1, N_H]. N_H is set to 5 in the experiment, N_0 is the number of non-zero value in each frame.

$$\boldsymbol{PCH}(n, h) = \boldsymbol{HMap}(\boldsymbol{m}(n), h), \qquad \boldsymbol{PCH} \in \mathbb{R}^{N_0 \times N_H} \tag{4.12}$$

Let $\boldsymbol{PCP}$ be a matrix of the harmonics and their potential corresponding pitches, which contains the harmonic components and their associated pitches being calculated from the original pitch at a specific value of h as follows:

$$\delta(x - y) = \begin{cases} 1, & if\ x = y \\ 0, & otherwise. \end{cases} \tag{4.13}$$

$$\boldsymbol{PCP}(n, h) = \boldsymbol{PCH}(n, h) \cdot \delta[\boldsymbol{PCH}(n, h) - \boldsymbol{PCH}(n, 1)],\ \boldsymbol{PCP} \in \mathbb{R}^{N_0 \times N_H} \tag{4.14}$$

where $\delta(x - y)$ is a function of the equivalence gate with two inputs. The output of the equivalence gate will be 1 if the two inputs equals (i.e., h=1). Otherwise, it will become zero. Using Eqs. (4.13-4.14), $\boldsymbol{PCP}(n, h)$ can be identified for each harmonic component.

Let $\boldsymbol{PHC}(n, 1)$ be a harmonic component and $\boldsymbol{PHC}(n, h)$ $(h = 2, \ldots, N_H)$ represents the relative associated pitches. $\boldsymbol{PHC}$ is the value that correlates to $\boldsymbol{PCP}$ in identifying potentially the original pitch values. The matrix for all of the potentially original pitch values is estimated below. If $\boldsymbol{PCP}(n, h) = \boldsymbol{PCP}(n, 1)$, an equivalence gate value of 1 is assigned and the output value from the square brackets becomes 1 in Eq. (4.15).

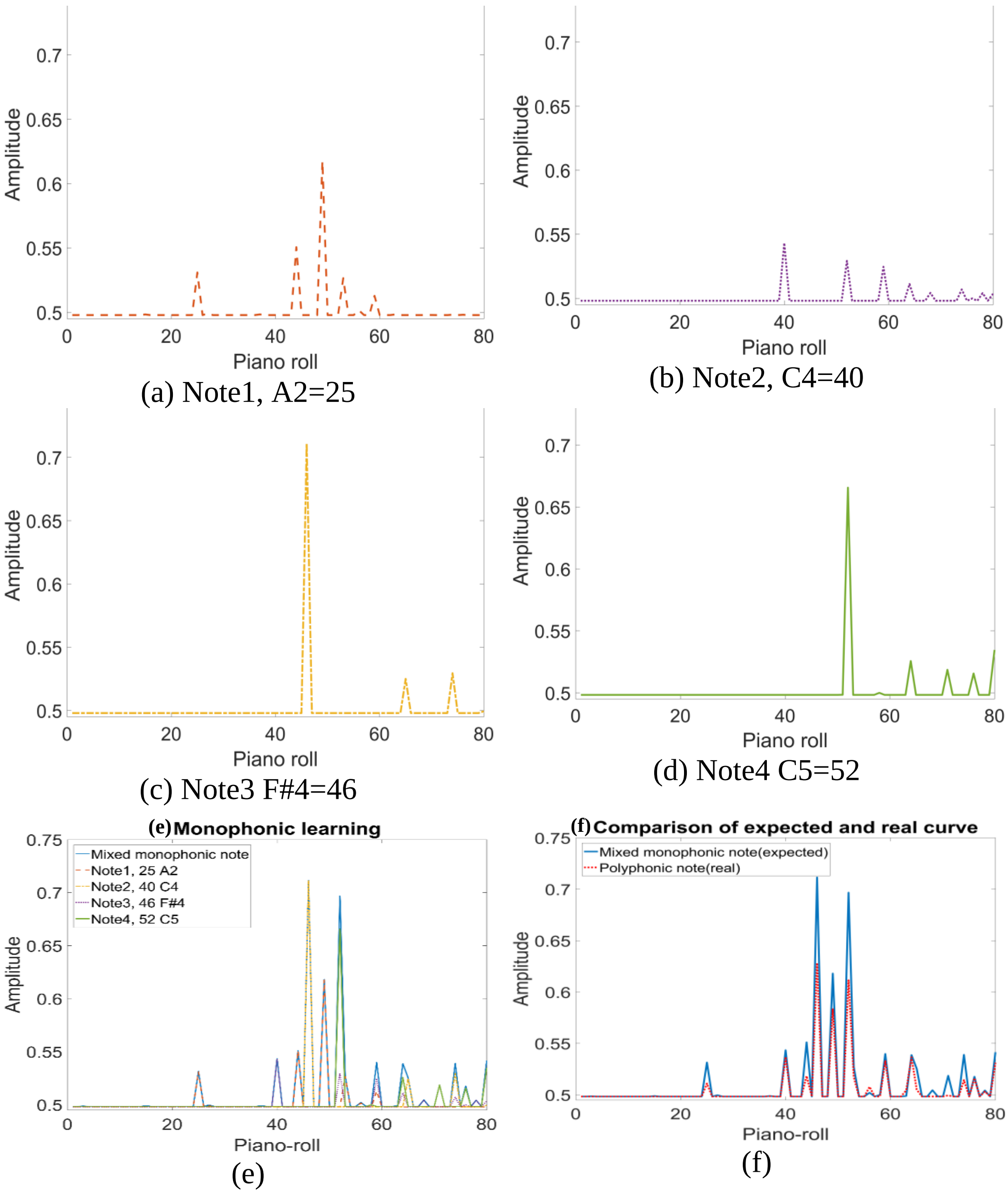
$$\begin{gathered} \boldsymbol{PHC}(n, h) = \boldsymbol{PCP}(n, 1) \cdot \delta[\boldsymbol{PCP}(n, h) - \boldsymbol{PCP}(n, 1)], \\ \boldsymbol{PHC} \in \mathbb{R}^{N_0 \times N_H}, \qquad n \in [1, N_0], h \in [1, N_H] \end{gathered} \tag{4.15}$$

4.2.5 Energy Domain Analysis

In the energy domain, $\boldsymbol{EDG}(n,h)$ is a value generated from $\boldsymbol{ED}\epsilon\mathbb{R}^{N_0\times N_H}$ and $\boldsymbol{PHC}(n,h)$ as defined below:

$$\boldsymbol{EDG}(n,h) = \boldsymbol{ED}(n)\cdot\delta[\boldsymbol{PHC}(n,h)-\boldsymbol{PHC}(n,1)], \qquad \boldsymbol{EDG}\ \epsilon\ \mathbb{R}^{N_0\times N_H} \qquad (4.16)$$

The following section describes two cognitive theories that have inspired the guided weight mechanism for fundamental frequency detection. First, according to the harmonic periodicity and instrument timbre theory [23], the harmonic periodicity of different instruments should be the same, although the sound of which varies by their timbres as reflected on the ratio of harmonic amplitude to the fundamental amplitude [152]. The instruments from different families will have a large ratio, and vice versa. For the instrument that produces a sound from strings such as piano, and violin (Fig. 4-6 (d)), their harmonic amplitudes generally decrease gradually. On a different note, for woodwind instruments such as clarinet (Fig. 4-6 (c)) and bassoon (Fig. 4-6 (a)), the amplitudes of their first harmonic would be lower than that of their second harmonic. Therefore, the energy ratio of the fundamental frequency and harmonic frequency energy (timbre) is unaffected by monophonic, or polyphonic textures, but unique in individual instruments. Second, according to the acoustic theory [153], when two or more sound waves occupy the same space, they move through rather than bounce off each other. For example, the result of any combination of sound waves is simply the addition of these waves. Theoretically, the energy of the mixed monophonic and polyphonic audio should be the same, though there is unavoidable difference in the real case. The results of a single frame after step 1 (section 4.2.3) of the Harmonic structure detection (HSD) are plotted as profile of pitch values as shown in Fig. 4-6. The profiles of four single music sources are shown in Fig. 4-6 (a-d). The profile of the mixed monophonic notes is given in Fig. 4-6 (e), which is composed of four single music sources, i.e., Notes #1-#4, and the profile of the polyphonic notes shown in Fig. 4-6 (f) is generated from one mixed channel. Considering that the profile of mixed monophonic notes is the ideal value, and the profile of the polyphonic notes is the predicted actual value. As seen in Fig. 4-6 (f), there is few amplitude difference



(a) Note1, A2=25

(b) Note2, C4=40

(c) Note3 F#4=46

(d) Note4 C5=52

(e)

(f)

Fig. 4-6. Profile of pitch values for monophonic and polyphonic analysis in a single frame. Single notes with its MIDI number for Bassoon (a), Saxophone (b), Clarinet (c), and Violin (d); (e) is the monophonic learning result when combining the four note, (f) is the comparison of real polyphonic value with expected mixed monophonic notes

between the profiles of the polyphonic and monophonic notes due to the resonance in the polyphonic notes and channel distortion during data recording and transmission, but the overall trend of the two profiles is very similar.

Motivated by these, the guided weight mechanism is proposed and denoted as Eq. (4.17) in the model for improving the detection of the fundamental frequency.

$$\boldsymbol{W}_i(h) = \frac{1}{T}\sum_{t=1}^{T}\frac{\boldsymbol{ED\_mono}_t(h)}{\boldsymbol{ED\_mono}_t(1)}, \qquad h \in [1, N_H], i \in [1, N_I] \tag{4.17}$$

where T is the number of time frames in the monophonic data, the first non-zero value of $\boldsymbol{ED\_mono}_t(1)$ is always the fundamental frequency, and the rest non-zero values $\boldsymbol{ED\_mono}_t(h)$ are the harmonic frequencies. The guiding weight is calculated by the averaged ratio of the amplitude of harmonic $\boldsymbol{ED\_mono}(h)$ and fundamental frequency $\boldsymbol{ED\_mono}(1)$ in the monophonic data, before being applied to the polyphonic data. The variableN_I is the number of known instruments that can be identified in the music piece.

In order to estimate the amplitude of the harmonic components (EHC), Eq. (4.18) is utilized to calculate how strong each harmonic is for the pitch n by multiplying the guided weight of selected instrument with $\boldsymbol{EDG}$.

$$\begin{gathered}\boldsymbol{EHC}_i(n,h) = \boldsymbol{EDG}(n,h) \cdot \boldsymbol{W}_i(h), \\ \boldsymbol{EHC}_i \in \mathbb{R}^{N_0 * N_H}, \qquad n \in [1, N_0], h \in [1, N_H]\end{gathered} \tag{4.18}$$

Theoretically, the amplitude of harmonic should be a portion to the amplitude of the fundamental frequencies. It is noted that the fundamental frequencies must occur at h = 1, then harmonic frequencies occur at h = 2: N_H.

Based on the $\boldsymbol{EHC}_i$ determined from Eq. (4.18), the amplitude of fundamental frequency in pitch *n* after subtracting the summed harmonic components' amplitude will be kept updating until the fundamental frequencies from all instruments are estimated as:

$$\boldsymbol{ED}(n) = \boldsymbol{EHC}_i(n,1) - \sum_{h=2}^{N_H} \boldsymbol{EHC}_i(n,h) \tag{4.19}$$

Eventually, the amplitude of fundamental frequency in pitch *n*, represented a $\boldsymbol{EFF}$, can be obtained by Eq. (4.20)

$$\boldsymbol{EFF}(n) = \boldsymbol{ED}(n), \qquad \boldsymbol{EFF} \in \mathbb{R}^{N_0 * 1} \tag{4.20}$$

For each non-zero pitch n in each frame t, it will have a rank value $\boldsymbol{R}(n)$ according to the $\boldsymbol{EFF}(n)$, then a 2D rank map $R(n,t)$ will be generated for the whole music piece, i.e. pitch/pianoroll vs. time frame as shown in Fig. 4-3 (b), which will be used to fully represent detected harmonic structure. A brief implementation of energy domain procedure is summarized in Algorithm 1.

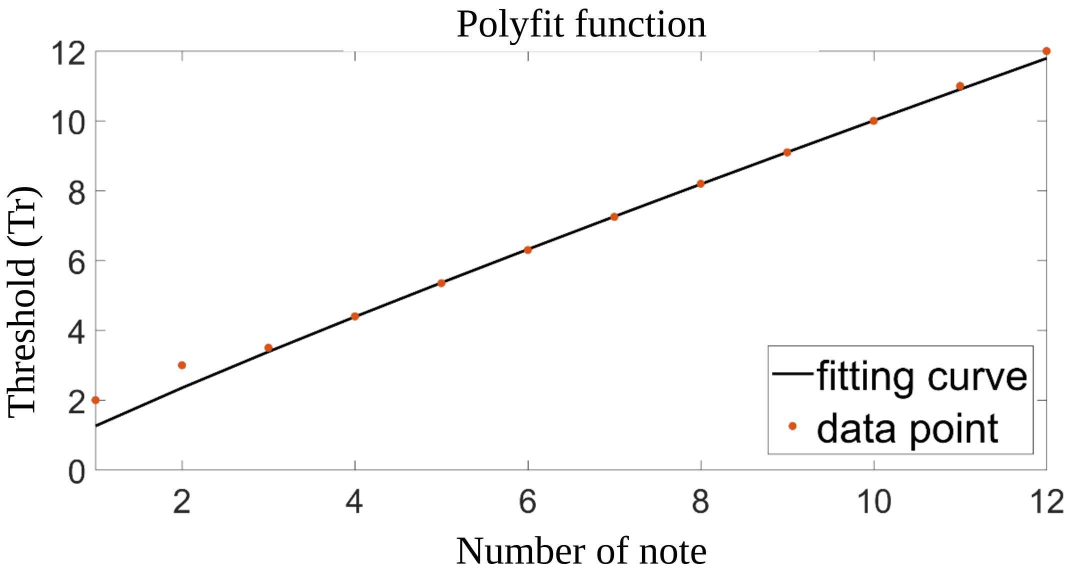
```
Algorithm 1
Inputs: $\boldsymbol{ED}(\boldsymbol{n})$
Step 1: Generate a matrix including the amplitude of fundamental pitch and their corresponding harmonic pitches using Eq. (4.16)
Step 2: Calculate the weight for each type of instrument using Eq. (4.17)
Step 3: Estimate the amplitude of harmonic components ($\boldsymbol{EHC}$) presented in the pitch n using Eq. (4.18)
Step 4: Update $\boldsymbol{ED}$ by Eq. (4.19)
Step 5: Repeat step 1-4 until the fundamental frequencies from all instruments are estimated;
    Obtain the final estimated amplitude of fundamental frequency in pitch n by Eq. (4.20)
```

4.2.6 Note Tracking

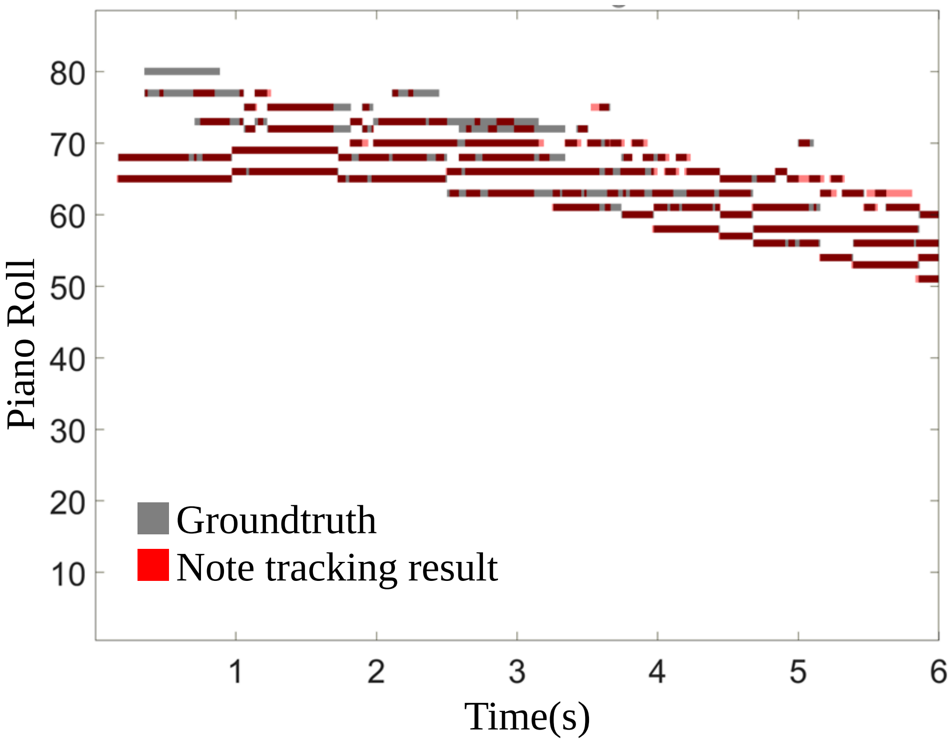
As seen in Fig. 4-3 (b), although most fundamental pitch have been extracted, the notes still show a poor consistency. To improve this, a note tracking method based on the music perception and multi-pitch probability weight was proposed. According to the music theory [150], the occurrence of demisemiquaver is generally quite low in music pieces. As a result, notes with a length shorter than demisemiquaver are filtered out. The averaged rank of the connected pitch group in the rank map is calculated and denoted as $\bar{R}$. If $\bar{R}$ is larger than an adaptive threshold T_r, the pitch group is considered a harmonic and will be skipped from the analysis. As the polyphonic music pitches vary over time, the T_r will also change accordingly. To account for this change, a fitting function was generated for T_r (Fig. 4-7 (a)), which is adaptive to the number of notes $x \in [1,12]$ for each frame, as given below.

$$T_r = 1.26x^{0.9} \tag{4.21}$$

The fitting curve of T_r is obtained by minimise the fitting error between grountruth and estimation. Fig. 4-7 (b) displays the note tracking results where most of the noise



(a)



(b)

Fig. 4-7 Poly function of threshold $\boldsymbol{T_r}$ (a) and Results from the note tracking in comparison to the ground truth (b).

and the inconsistencies have been filtered out. The result has also achieved a similar profile to that of the ground truth data.

4.3 Experimental Results

4.3.1 Experimental Settings

To validate the effectiveness of the proposed approach, the first dataset used for evaluation is the MIDI Aligned Piano Sounds (MAPs) [154], in which all music pieces are recorded in the MIDI format initially and then converted into '.wav' format. MAPS also has differently purposed subsets such as monophonic excerpts and chords. For this case, only one subset is used which includes polyphonic music pieces. In addition, there are several instruments and recording conditions in MAPs. The 'ENSTDkCI' is chosen as it has been widely used in many studies [3, 7] and the music is played using a real piano (i.e., Yamaha Disklavier Mark III (upright)) rather than an acoustic one, i.e. a virtual instrument, and recording occurs in soundproofed conditions. The second dataset is BACH10 [9], which contains 10 pieces using violin, clarinet, saxophone and bassoon from J.S. Bach chorales, where each piece lasts approximately 30 seconds. The third dataset is TRIOS [155], which is the most complex one among the three as it contains five multitrack chamber music trio pieces. The sampling rate for all music pieces is 44100 Hz.

For objective assessment, the most commonly used frame-based metric, F-measure (F1) [14, 15], is adopted. It combines the positive predictive value (*PPV*, also referred to as precision) and the true positive rate (*TPR*, also labelled recall) for a comprehensive evaluation as follows:

$$F1 = \frac{2 \cdot PPV \cdot TPR}{PPV + TPR} \tag{4.22}$$

where $TPR = \frac{T_p}{T_p+F_n}, PPV = \frac{T_p}{T_p+F_p}$, and T_p, F_p, and F_n refer respectively to the number of correctly detected F_0, incorrectly detected F_0, and missing detection of the F_0. Specifically, these three components can be calculated by comparing the binary masks of the detected MPE results and the ground truth.

4.3.2 Performance Evaluation

Table 4-III shows the quantitative assessment of 11 benchmarking methods on MAPS, BACH10, and TRIOS datasets. All benchmarking methods are divided into two categories: Shallow learning method and DL method. Shallow learning methods include a traditional machine learning model or a prior knowledge-based model whereas DL methods include deep neural networks and deep convolutional neural networks.

Many MPE approaches select a pair of methods from CQT, PLCA, Equivalent Rectangular Bandwidth (ERB) and NMF for pianoroll transcription. Therefore, two of the most representative methods, i.e. CQT+PLCA proposed by Benotos [5] and ERB+NMF proposed by Vincent [7], are chosen for benchmarking. In Table 4-III , Benetos [3] and Vincent [7] can produce the second best performance on the MAPs and TRIOS datasets, respectively, which validates the effectiveness of CQT+PLCA and ERB+NMF. However, due to the lack of efficient harmonic analysis, the performance of both methods is inferior to the proposed HSD method. Unlike the methods from Benetos and Vincent, other methods adopt different ideas for MPE. SONIC [12] proposed a connectionist approach where an adaptive oscillator network was used to track the partials in the music signal. However, without a matrix factorization process, its performance is limited on the three datasets. Su [11] proposed a combined frequency and periodicity (CFP) method to detect the pitch in both frequency domain and lag (frequency) domain. The CFP method in Table 4-III gives the best performance on the BACH10 dataset, but relatively poorer results on the other two datasets. The main reason here is possibly because the music pieces in the MAPS and TRIOS datasets have more short notes than those in the BACH10 dataset, and CFP has the limited ability for detecting the short notes but exhibit less errors for continuous long notes. Furthermore, the assumption of CFP does not hold for high-pitch notes of piano, as both MAPS and TRIOS have many piano music pieces. In addition, the music pieces in the MAPS database contain multiple notes in most frames, which has led to extra difficulty for polyphonic detection. However, the proposed method can still successfully solve this problem by effectively analysing the relationship of the position and energy between the fundamental frequency and harmonic frequencies for the notes. As a result, the performance of the proposed method on MAPS is the best, which is 8% higher than that of CFP. Klapuri [10] proposed an auditory model based F_0 estimator and Duan [9] proposed a maximum-likelihood approach for multiple F_0 estimation but both methods result in inferior performance compared to the results achieved by Benetos [3, 5], Vincent [7] or CFP [11]. Furthermore, Klapuri's [10] and Duan's [9] methods lack an effective pre-processing stage (i.e. TF representation and matrix factorization) or harmonic analysis, which is the main reason why their overall performance is less effective in comparison to ours.

The proposed method was also compared with four deep-learning based supervised approaches on MAPS dataset. Due to lack of publicly available source codes, only the data that was reported in the original paper was duplicated for comparison. The first two methods are proposed by Sigtia [13], which are mainly based on the music language models (MLMs). However, due to insufficiently labelled data in the existing polyphonic music databases for training, such limitations have affected further analysis of DL-based approaches. Furthermore, the MLM model is not robust to ambient noise, whereas music pieces in reality generally contain a lot of ambient noise. This has resulted in DL-based methods failing to fully analyse the inner structure of the music pieces. As a result, DL-based methods cannot achieve the same performance as the HSD method or some of the other unsupervised methods such as Benetos [3] on the MAPS dataset. Li [14] and Kelz [15] also proposed DL-based methods for AMT. Although better than [13], their performance is still not ideal as there is insufficient music knowledge support embedded. To this end, more music theories should be introduced for improved AMT.

In summary, referring to Table 4-III, the proposed method yields the best results on both the MAPs and TRIOS datasets, also the second-best in BACH10 according to F1 value, thanks to the guidance of music cognition. However, the method can still be improved, especially for reducing the computation cost. As it takes 2 minutes to process a 30-second music piece, this is longer than some other methods. In addition,

Table 4-III Frame-level performance of different methods on three datasets (top two methods in each column are highlighted in bold and italic respectively). Results marked with * are quoted from their original publications due to unavailability of source code.

Category	Methods	F1		
		MAPS	BACH10	TRIOS
Shallow learning	Benetos* [3]	64.17	68.40	*66.46*
	Benetos [5]	59.31	70.57	64.93
	Vincent* [7]	*72.35*	79.78	59.40
	Duan [9]	67.41	70.90	45.80
	Klapuri [10]	60.10	68.30	50.50
	CFP* [11]	68.67	**85.51**	64.64
	SONIC* [12]	63.60	66.49	56.65
	HSD(proposed)	**76.30**	*80.17*	**67.63**
Deep learning	ConvNet* [13]	64.14	--	--
	RNN* [13]	57.67	--	--
	Li* [14]	69.42	--	66.34
	INN* [15]	72.29	--	--

although the profile of the real polyphonic note is close to the expected mixed monophonic note, as shown in Fig. 4-6 (e-f), there are still some differences in the final values of the monophonic and polyphonic profiles which can be further improved.

According to the Table 4-III, the best F1 scores achieved for MAPS, BACH10 and TRIOS are 76.30%, 85.51% and 67.63% respectively. These scores are indeed far from perfect. The usefulness of these results depends heavily on the specific application.

For critical applications requiring high accuracy, such as automated music transcription for professional use, these scores might be insufficient. Many such applications may require F1 scores of 90% or higher to be considered practically viable.

However, for less critical applications, these results could still provide valuable information. For instance, music recommendation systems or preliminary analysis tools for musicians might benefit from this level of accuracy. In many music information retrieval tasks, even imperfect results can offer useful insights or serve as a starting point for further analysis.

4.3.3 Key Stage Analysis

In this section, the contribution of several major stages in the proposed MPE system is discussed, where the performance of each stage is evaluated on the MAPS dataset in terms of the precision, recall and F1. To calculate these three metrics, the result of each stage is normalised by using Eq. (8) and the results are binarized with a fixed threshold value of 0.5.

The proposed MPE system is divided into four key stages detailed as follows:

- Stage A: The transcription map from SI-PLCA and CQT.
- Stage B: The result after applying the first-step HSD.
- Stage C: The result after applying the second-step HSD.
- Stage D: The result after applying note tracking.

Table 4-IV illustrates the details of the system configurations. By combination of different key stages, the corresponding system is built up for evaluation. Each stage has specific components which are indispensable to the results of the system. Stage A shows the highest recall and lowest precision after applying CQT and SI-PLCA. The

Table 4-IV System configuration

Configurations	**Precision**	**Recall**	**F-measure**
A	0.408	0.879	0.545
A+B	0.438	0.876	0.571
A+B+C	0.747	0.718	0.725
A+B+C+D	0.753	0.773	**0.763**

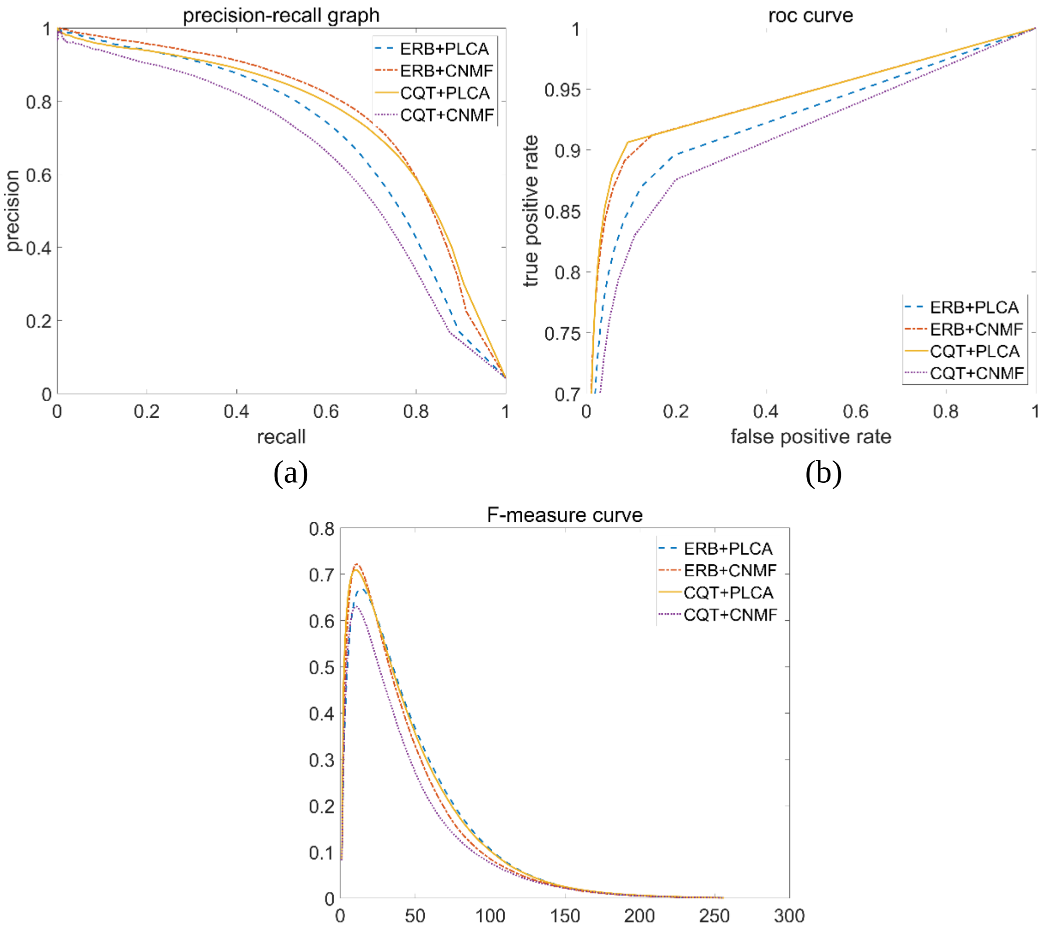
presence of F_0 and harmonics are all detected, however, many amplitudes are concentrated in higher frequency (harmonic) regions which inhibits the identification of F_0. After combining the stage B, the recall value decreases by 0.03%, but the precision value increases by almost 3%. This is mainly due to the removal of noise in HSD. In stage C, the core of the MPE system contributes to an increase of nearly 30% for precision and 15-18% for F1 compared to previous combinations. Finally, after applying the proposed note tracking step (Stage D), the recall value is further improved by 5.5% which leads to the final F1 value improved by 3.8% compared to the previous stage.

4.3.4 Assessment of CQT and ERB

In the proposed MPE system, CQT is employed to model human cochlea perception. However, cochlea perception is not always constant in Q. Therefore, apart from CQT, the Gammatone filter-bank technique is also widely used for time and frequency transform. Gammatone filter-bank is designed to model the human auditory system. It can decompose a signal by passing it through a bank of gammatone filters equally spaced on the equivalent rectangular bandwidth (ERB) scale. However, it might not be necessary for better performance on MPE problems. To validate this assumption, CQT [38], ERB [7], PLCA [3] and NMF [7] are integrated in a pair-wise manner. Four methods (i.e. CQT+PLCA, CQT+NMF, ERB+PLCA, ERB+NMF) are then analyzed in terms of precision-recall, ROC and F-measure curve (Fig. 4-8), AUC, MAE and maxF (Table 4-V). Here AUC is the are under the ROC curve, MAE is the mean average error, and maxF is the max value of F-measure curve. These three criteria have same importance. From Fig. 4-8(a,b), it can be seen that the ERB+NMF and CQT+PLCA show comparable results and better than other two pair-wise methods. From Table 4-V, although ERB+NMF gives the best maxF value, CQT+PLCA gives the best AUC value and lowest MAE value, which means it has a smaller false alarm. Therefore, CQT+PLCA is the best among these four pair-wise methods which are also the main reason why it is employed in the proposed MPE system.

Table 4-V Time-freqeuncy transform and piano-roll transcription comparison

Methods	AUC	MAE	maxF
ERB+PLCA	0.922	0.0403	0.6687
ERB+CNMF	0.939	0.0487	**0.7213**
CQT+PLCA	**0.942**	**0.0390**	0.7089
CQT+CNMF	0.906	0.0411	0.6296



(a)

(b)

(c)

Fig. 4-8. Precision-Recall (a), Receiver 0perating Characteristic (b) and F-measure (c) curve of four pair-wise methods.

4.4 Chapter Summary

In this chapter, a harmonic analysis method is proposed for MPE system, inspired by music cognition and perception. CQT and SI-PLCA are employed in the pre-processing stage for pianoroll transcription in mixture music audio signal, from which the proposed HSD is used to extract the multi-pitch pianorolles. The proposed MPE system is not limited by the number of instruments. For multi-instrument cases (i.e. symphony in BACH10 and TRIOS datasets), the mixture characteristics of each instrument can be extracted for adaptively detection of the fundamental frequencies. From the experiment results, the proposed MPE system yields the best performance on the MAPS and TRIOS datasets, and the second-best on the BACH10 dataset. Through investigation of the performance of key components, the HSD provided the greatest contribution to the system, which validates the value of adding an efficient harmonic analysis model for improving significantly the performance of the MPE system. Furthermore, adding note tracking can further improve the efficacy of the MPE system.

Moving into Chapter 5, another critical technology: predominant musical instrument recognition (PMIR) will be discussed. This technology aims to accurately identify and classify the sounds of the main instruments within complex musical audio. It plays a significant role in automated music annotation and classification, music education, and recommendation systems. Predominant instrument recognition enhances the accuracy of AMT transcriptions through detailed instrument analysis, while AMT provides clear note and timing references for instrument identification. However, this technology still needs to address challenges such as distinguishing between timbrally similar instruments and achieving high accuracy in complex, multi-instrument settings. Together, these technologies drive the progress and development of music technology, laying the foundation for more intelligent and precise music analysis.

Chapter 6 delves into musical shape evaluation, which builds upon and improves the results of Chapters 4 and 5. Musical shape evaluation focuses on identifying and comparing various types of musical shapes, providing insights into their compositional elements and stylistic features. This technology is crucial for music composition, education, and automated music generation, offering a deeper understanding of the underlying patterns and structures of music. By leveraging the precise note and instrument information provided by AMT and instrument recognition, musical shape evaluation addresses the current limitations in understanding complex musical structures and diverse styles.

Chapter 5

5 Fusion of Hilbert-Huang Transform and Deep Convolutional Neural Network for Predominant Musical Instruments Recognition

5.1 Introduction

In the third chapter, automatic music transcription (AMT) and predominant instrument recognition techniques were explored. While these technologies have shown excellent performance in many applications, challenges remain in handling complex polyphonic audio and improving recognition accuracy. This chapter continues to advance these technologies, with a particular focus on enhancing the precision and efficiency of PMIR.

As a subset of music information retrieval (MIR), predominant musical instruments recognition (PMIR) has attracted substantial interest in recent years due to its uniqueness and high commercial value in key areas of music analysis research, such as music retrieval and automatic music transcription. While traditionally PMIR has been associated with timbre analysis, the approach presented in this chapter goes beyond purely timbral characteristics to analyze comprehensive spectral-temporal patterns that uniquely identify different instruments. In this chapter, the Hilbert-Huang Transform (HHT) is employed to map one-dimensional audio data into a two-dimensional matrix format, capturing multiple acoustic properties such as spectral energy distribution, time-varying amplitude, inter-harmonic relationships and

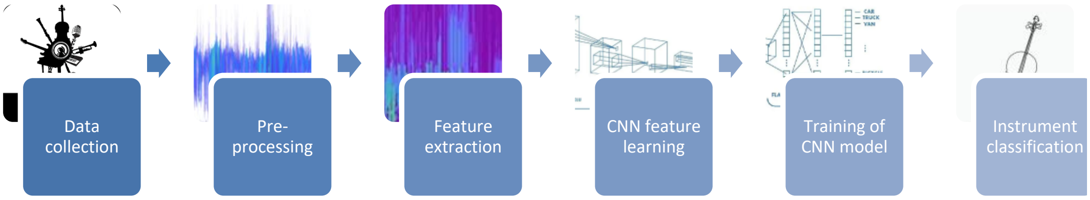


Fig. 5-1. Workflow of the PMIR system

overtone structures, etc. Subsequently, a deep convolutional neural network (DCNN) is developed to learn rich and effective features for PMIR.

To validate the efficacy of the proposed approach, an experiment is conducted using 6705 audio pieces, including 11 musical instruments. The results are compared to four benchmarking methods and show significant improvements in terms of precision, recall, and F-measures.

5.2 System Overview

The proposed PMIR system (as shown in Fig. 5-1), designed for MIR, starts by selecting the IRMAS database, a mixed collection featuring 11 types of musical instruments. The first step involves audio preprocessing, where Hilbert spectrogram sampling is employed to analyze the audio data. Next, these preprocessed spectrograms are fed into a Convolutional Neural Network (CNN) model for training. This training phase is crucial for teaching the system to accurately classify different musical instruments. The comprehensive workflow encapsulates the entire journey of an MIR system, from the initial database selection and audio processing stages to the sophisticated instrument classification via deep learning. This approach showcases the integration of advanced audio processing techniques with CNN models to achieve effective instrument recognition.

5.2.1 Hilbert-Huang Transform

Hilbert-Huang Transform (HHT), introduced by Huang from NASA in 1998 [43], has become a powerful tool for analyzing signals from nonlinear systems and nonstationary processes, making it particularly suitable for musical instrument classification. HHT has seen successful applications across various fields, including geophysics and biomedicine, where it has delivered outstanding results [156, 157]. The method comprises two primary components: Empirical Mode Decomposition (EMD) and Hilbert Spectral Analysis (HSA).

EMD decomposes given signals into several Intrinsic Mode Functions (IMFs), each representing different frequency components. The adaptive nature of EMD allows for the precise extraction of these intrinsic modes, making it especially useful for analyzing the harmonics of music signals. By capturing the complex features of musical instruments, EMD provides a detailed and accurate representation of the signal's components.

Once the signal is decomposed into IMFs, the Hilbert transform is applied to each IMF to obtain the corresponding Hilbert spectrum. The Hilbert spectrum of the original signal is then derived by summing all the Hilbert spectra of the IMFs. The resulting Hilbert spectra for different instruments—such as cello, clarinet, flute, acoustic guitar, electric guitar, organ, piano, saxophone, trumpet, violin, and voice—exhibit distinct characteristics, as illustrated in Fig. 5-2. These variations highlight HHT's ability to capture the unique features of each instrument.

HHT was first used in PMIR in 2018 by Daeyeol et al. [8], who proposed using HHT as a replacement for the Short-Time Fourier Transform (STFT) and Mel-Frequency Cepstral Coefficients (MFCC). Their performance results demonstrated that HHT outperformed these two traditional methods, establishing its effectiveness in the PMIR domain.

The HHT can be expressed as:

$$H(\omega, t) = Real \sum_{i=1}^{n} a_i(t) e^{j \int w_i(t) dt} \tag{5.1}$$

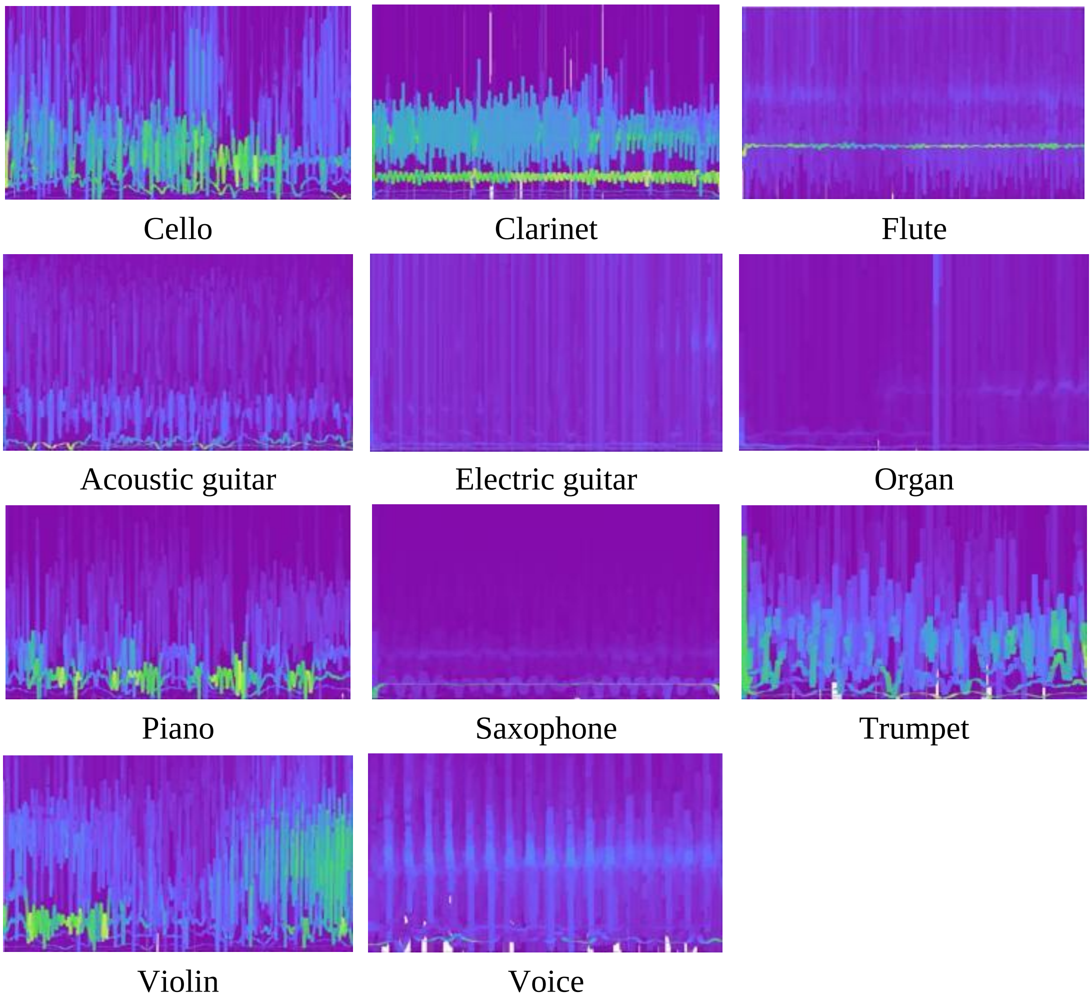
where $H(\omega, t)$ represents the Hilbert spectrum, $a_i(t)$ is the instantaneous amplitude of the i^{th} Intrinsic Mode Function (IMF), $w_i(t)$ is the instantaneous frequency of the i^{th} IMF, and n is the number of IMFs. The subscript i in $a_i(t)$ denotes the i^{th} IMF obtained from the EMD process.

In contrast, the Fourier transform can be represented as:

$$A(\omega) = \sum_{t=-\infty}^{\infty} x(t)e^{-j\omega t} \tag{5.2}$$

where $A(\omega)$ is the Fourier spectrum, $x(t)$ is the time-domain signal, and ω is the angular frequency.

A key difference between the two transforms is that HHT can be thought of as a phase shift converter, while traditional Fourier analysis uses a series of triangular basis functions to perform orthogonal operations on signals [158]. The resulting Fourier spectrum provides only the weighted mean of the frequencies over a certain period and cannot accurately describe time-frequency changes. In contrast, HHT's ability to



Cello

Clarinet

Flute

Acoustic guitar

Electric guitar

Organ

Piano

Saxophone

Trumpet

Violin

Voice

Fig. 5-2. Example Hilbert spectra for each of the 11 instruments, where the x-axis represents time (s) and the y-axis represents instantaneous frequency (Hz).

define the instantaneous frequency enables it to handle more complex signals, such as polyphonic music pieces. It's worth noting that the Short-Time Fourier Transform (STFT) addresses some limitations of the standard Fourier transform by applying the transform to short segments of the signal, allowing for time-localized frequency analysis. However, the HHT still offers advantages in terms of adaptive decomposition and higher time-frequency resolution, particularly for complex signals like music.

HHT's unique advantages in handling music signals lie in its high time-frequency resolution, which is essential for identifying the intricate features of musical instruments. The adaptive nature of HHT allows for the precise extraction of intrinsic modes, representing various frequency components, and effectively captures the transient sounds produced by musical instruments. These qualities are crucial for accurate classification and significantly enhance feature discrimination, leading to more precise identification of different instrument sounds.

5.2.2 Architecture of DCNN

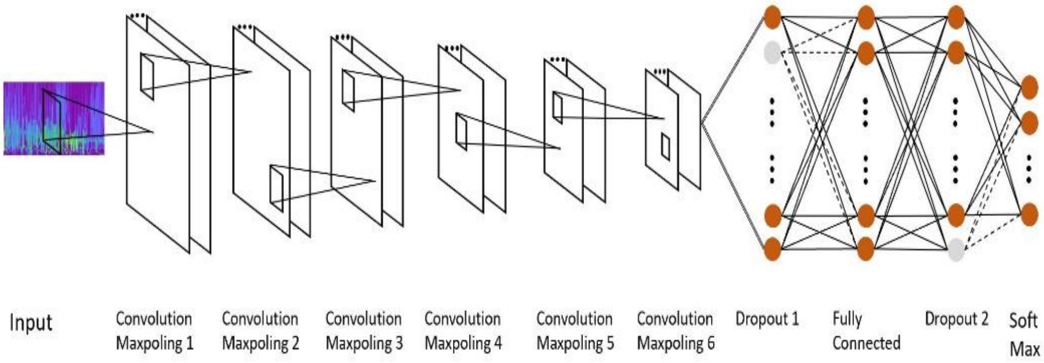


Fig. 5-3. Flowchart of the proposed DCNN

In the field of PMIR, the choice of a deep convolutional neural network (DCNN) is motivated by the ability of CNNs to effectively capture local patterns and features from the spectrogram as it can reflect the melody, harmony and timbre difference across different musical instruments. Traditional approaches to PMIR often rely on hand-crafted features extracted from the time or frequency domain independently, such as zero-crossing rate, spectral centroid, or mel-frequency cepstral coefficients

Table 5-I. Proposed DCNN structure.

Layers	Output size	Description
HHT (Input)	$135 \times 240 \times 3$	Feature matrix from Hilbert spectrum
Convolution 1	$68 \times 120 \times 32$	Filter size: 7 × 7; Stride size: 2 × 2;
Max pooling 1	$34 \times 60 \times 32$	Pool size : 2 × 2 ; Stride size: 2 × 2;
Convolution 2	$34 \times 60 \times 64$	Filter size: 5 × 5; Stride size: 1 × 1;
Max pooling 2	$17 \times 30 \times 64$	Pool size : 2 × 2 ; Stride size: 2 × 2
Convolution 3	$17 \times 30 \times 128$	Filter size: 3 × 3; Stride size: 1 × 1;
Max pooling 3	$8 \times 15 \times 128$	Pool size : 2 × 2 ; Stride size: 2 × 2;
Convolution 4	$8 \times 15 \times 256$	Filter size: 3 × 3; Stride size: 1 × 1;
Max pooling 4	$4 \times 7 \times 256$	Pool size : 2 × 2 ; Stride size: 2 × 2
Convolution 5	$4 \times 7 \times 512$	Filter size: 3 × 3; Stride size: 1 × 1;
Max pooling 5	$2 \times 3 \times 512$	Pool size : 2 × 2 ; Stride size: 2 × 2;
Convolution 6	$2 \times 3 \times 1024$	Filter size: 3 × 3; Stride size: 1 × 1;
Max pooling 6	$1 \times 1 \times 1024$	Pool size : 2 × 2 ; Stride size: 2 × 2
Dropout 1	1024	Dropout fact : 0.25
Fully connected	1024	Output size :11
Dropout 2	1024	Dropout fact : 0.25
Softmax	11	Softmax function

(MFCCs). However, these features may not fully capture the complex interactions and dependencies between time and frequency components in music signals. In contrast, DCNNs have the ability to learn and extract features that jointly consider both time and frequency information, making them well-suited for PMIR tasks.

In Fig. 5-3, the architecture of the proposed DCNN is illustrated. The specific details of the architecture are provided in Table 5-I. The proposed DCNN was inspired by the VGG-16 model [66], which contains 9 hidden layers (6 convolutional and 3 fully connected). Each convolutional layer is followed by a batch normalization layer, an activation layer, and a max pooling layer. The feature matrix produced by the Hilbert spectrum is size $135 \times 240 \times 3$, where 3 means RGB channels, the size of each channel is 135×240. A rectified linear unit (ReLU) is selected as the activation function for each activation layer due to its popularity and ability to increase the learning speed. In the first convolutional layer the stride size is 2×2, which is changed to 1×1 for the rest of the convolutional layers. Both pool size and stride size of each max pooling layer are set to 2 × 2. In addition, the input for each convolutional layer is same-padded in order to preserve the spatial resolution. The number of filters for each convolution layer is twice that of the previous layer, increasing from 32 to 1024 at the last layer. After the final max pooling layer, a dropout layer is added before and after the fully connected layer to avoid overfitting. At the end of the network, the Softmax function is used to classify 11 types of instruments.

As seen the proposed model has fewer layers compared to VGG16, resulting in less parameters. This means that it requires less computational power and memory to train and deploy. In scenarios where resources are limited or real-time processing is needed, a lightweight model can be more practical and efficient. Due to their smaller size and fewer parameters, lightweight models typically have shorter training times and faster inference speeds. This is particularly important when dealing with large datasets or when the model needs to make predictions quickly, such as in real-time music analysis or live performance settings.

5.3 Experimental Results

5.3.1 Dataset description

The IRMAS dataset [114] is used in this work to evaluate predominant musical instrument recognition. This dataset contains 6705 annotated musical audio excerpts with labels indicating the predominant instrument in each excerpt. The dataset was originally compiled for the task of automatic musical instrument recognition.

Each audio clip is a 16-bit stereo WAV file sampled at 44.1 kHz and 3 seconds in length. As shown in Table 5-II, there are 11 predominant instrument categories including string instruments like cello and violin, woodwind instruments like clarinet, flute and saxophone, brass instrument like trumpet, keyboard instruments like organ and piano, plucked string instruments like acoustic guitar and electric guitar, and human singing voice. The second column shows an abbreviation for each instrument name that will be used to reference the classes throughout the paper. The third column assigns a numeric label to each instrument, ranging from 1 for cello to 11 for voice.

These numeric labels correspond to the output classes used when training the neural network models. During the model training, the labels in the IRMAS dataset will be converted to one-hot encoded vectors, representing the presence or absence of each instrument category in the excerpt. Each label vector has 11 elements, corresponding to the 11 instrument categories. For a given excerpt, the element corresponding to the predominant instrument is set to 1, while all other elements are set to 0. Example: If an excerpt features a piano as the predominant instrument, the label vector would be [0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0], where the 7th element represents the piano category. The fourth column indicates the total number of examples available for each instrument. Despite the reduction from the full dataset, there is still a reasonable amount of data for each class to train deep learning models.

The IRMAS dataset was selected for this research because it provides polyphonic musical mixtures with labeled predominant instruments. This allows training and evaluation of multi-instrument recognition models, unlike datasets with only isolated note samples. The large dataset size, diversity of musical styles, and focus on predominant instrument recognition make IRMAS well-suited for this task.

Table 5-II. List of instruments in the project with the number of data

Instruments	Abbreviations	Number represented	Number of data (n)
Cello	cel	1	388
Clarinet	cla	2	505
Flute	flu	3	451
Acoustic guitar	gac	4	637
Electric guitar	gel	5	760
Organ	org	6	682
Piano	pia	7	721
Saxophone	sax	8	626
Trumpet	tru	9	577
Violin	vio	10	580
Voice	voi	11	778

Meanwhile, the musical recordings in IRMAS span a wide range of genres and styles from the past century. Although there is class imbalance in this dataset, it is not necessary to compensate for it due to several reasons:

1 The IRMAS dataset reflects real-world distribution of instruments in music

   - Class imbalance in IRMAS is relatively moderate:
   - Highest: Voice (778 samples)
   - Lowest: Cello (388 samples)

2 The ratio between highest and lowest is approximately 2:1

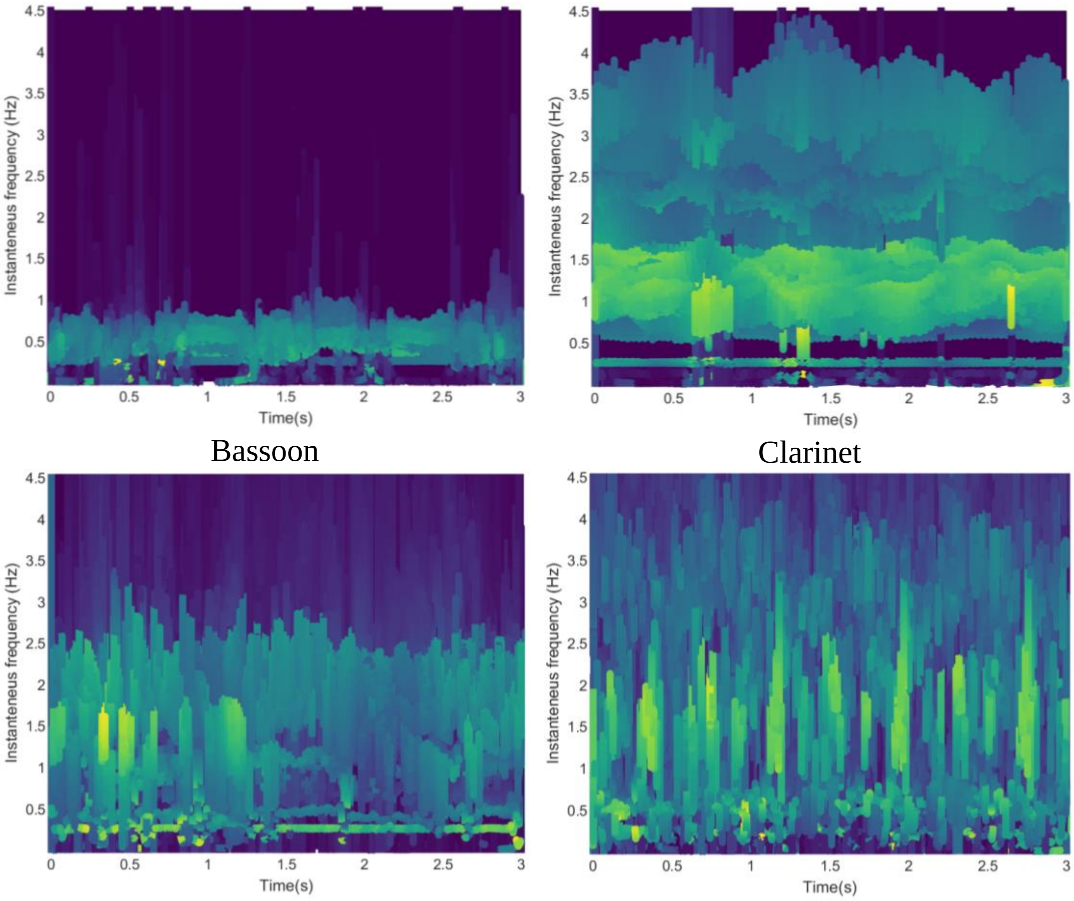
3 Artificially balancing classes would create unrealistic representations

4 In real-world music analysis, certain instruments naturally appear more frequently

Therefore, this dataset aligns with the study's goal of developing practical, robust models for real-world PMIR applications. Furthermore, the dataset's diversity in terms of sound quality, instrumentation, performance styles, and production aesthetics ensures that models trained on IRMAS are robust to different musical contexts. In the following experiments, 70% data is used for training and the rest data is used for evaluation.

5.3.2 Analysis of HHT spectrogram

The visualization of Hilbert-Huang Transform (HHT) spectrograms is usually affected by pitch, volume, and rhythm. To validate the reliability of HHT, a controlled experiment is conducted where musical segments featuring different instruments playing the same note (C4) with identical volume and rhythm are selected, and the corresponding spectrograms are shown in Fig. 5-4. The bassoon is a double-reed woodwind instrument with a deep and expressive tone. Its spectrogram shows strong energy in the lower frequency region, with a smooth energy decay curve as the frequency increases. Due to its unique reed mechanism, different harmonic components appear as specific peaks in the spectrum pattern. The clarinet is also a woodwind instrument, but its tone is brighter than that of the bassoon. Its spectrogram shows a more balanced spectrum distribution, with relatively strong energy in the higher frequencies. Although it shows clear harmonic components, its peak



Bassoon

Clarinet

Saxphone

Violin

Fig. 5-4. Typical samples of HHT spectrogram.

distribution pattern is significantly different from that of the bassoon. The saxophone combines the characteristics of woodwind and brass, and its sound exhibits a unique "breathing quality". Its spectrogram shows rich harmonic content, especially in the mid- and high-frequency ranges, and shows unique spectral characteristics due to its reed and mouth resonance, resulting in characteristic peaks and valleys in its frequency distribution. The violin is a string instrument that produces a clear, penetrating tone with rich harmonic content, especially in the higher frequencies. Its spectrogram shows clear, orderly harmonic peaks, reflecting the multiple vibration modes of the strings. The violin's spectrogram also contains unique overtones and formants due to bow-string interaction and body resonances. This comparative analysis shows that even when the same note is played under the same conditions, the unique sound-producing mechanism of each instrument produces a unique spectral signature.

5.3.3 Overview of Audio Content Analysis

Audio Content Analysis (ACA) is a field that focuses on automatically extracting musically meaningful information and metadata from audio signals. The main stages in ACA are:

- Feature Extraction - Obtain descriptive features and statistics that characterize the audio signal. The raw audio is reduced to a compact feature representation, which captures the essential information while reducing the data size.
- Classification/Inference - In this stage, machine learning models are used to map the extracted feature values to labels, predictions, or other higher-level musical concepts. This allows for the automatic identification and classification of various aspects of the audio content.
- Metadata Generation - The final stage involves producing descriptive outputs and metadata that summarize the semantic content of the audio. This metadata is informed by the results of the classification/inference stage and provides a high-level understanding of the audio content.

The first step in developing a machine learning model for instrument recognition is extracting meaningful features that represent the acoustic qualities of each instrument. In this work, Audio Content Analysis (ACA) is employed to analyze the audio signals and obtain descriptive instantaneous features. As shown in Fig. 5-5, ACA extracts 19 instantaneous features that provide low-level signal statistics, yielding 105 feature values per audio example in total. These features serve as the foundation for distinguishing between different instruments. Although these features may not be directly semantically meaningful, combining them allows machine learning models to learn complex acoustic patterns and profiles specific to each instrument type. The instantaneous features characterize various aspects of short blocks of audio samples and can be categorized into two main domains, i.e., time domain features and frequency domain features.

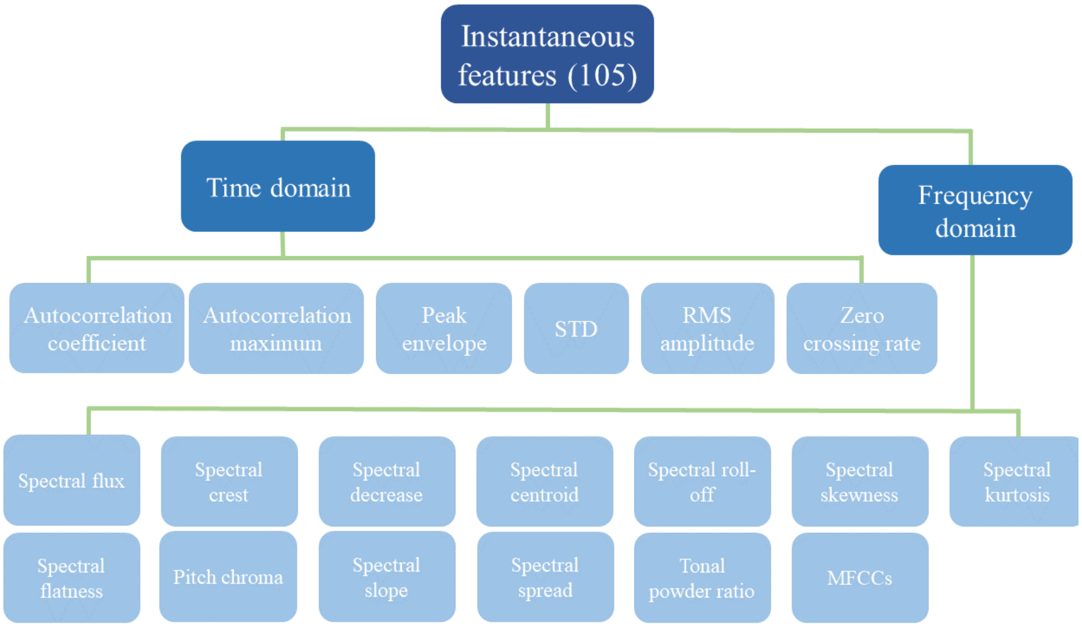


Fig. 5-5. Instantaneous features in Audio Content Analysis

Time domain features include Zero crossing rate, Peak envelope, Standard deviation (STD), Root mean square (RMS) amplitude, Autocorrelation coefficient, Autocorrelation maximum.

Frequency domain features include Spectral features: Spectral flux, Spectral crest, Spectral decrease, Spectral centroid, Spectral rolloff, Spectral skewness, Spectral kurtosis, Spectral flatness, Pitch chroma, Spectral slope, Spectral spread, Tonal powder ratio, MFCCs.

Of particular note are the Mel-Frequency Cepstral Coefficients (MFCCs) which represent the short-time power spectrum based on perceptually meaningful mel frequency bands. MFCCs are commonly used in instrument recognition to mimic human auditory properties.

5.3.4 Selection of Batch Size

Table 5-III evaluates the model using batch sizes ranging from 50 to 300, measuring precision, recall, F1 score, overall accuracy (OA), and Kappa score. The best and the worst results are highlighted in green and red, respectively. This also applies for the rest of the tables in Chapter 5 and Chapter 6. Several trends emerge: Precision remains relatively stable across batch sizes, staying between 86-88%. This indicates the

Table 5-III Results for different batch size.

Batch size	50	100	150	200	250	300
Precision (%)	87.94	87.93	87.65	87.69	86.96	87.05
Recall (%)	85.95	85.71	85.78	85.24	84.53	84.03
F1 (%)	86.93	86.80	86.70	86.45	85.73	85.51
OA (%)	88.54	88.30	88.16	88.06	87.20	87.15
KP*100	87.05	86.78	86.63	86.50	85.54	85.46

fraction of positive predictions that are correct does not vary much. Recall declines steadily from 86% at a batch size of 50 to 84% at a batch size of 300. This suggests smaller batch sizes help the model better detect positive examples. F1 score trends downward from 87% to 86%, driven by the declining recall. As batch size increases, the balance of precision and recall gets slightly worse. Overall accuracy peaks at a batch size of 50, and slowly decreases as batch size goes up. The highest OA is 88.54% with the smallest batch size. Kappa score shows a similar trend, with the highest score of 87.05% occurring at a batch size of 50. Kappa continues decreasing as batch size increases. Therefore, a batch size of 50 is selected for the proposed model.

5.3.5 Selection of Learning Rate

To select the best learning rate for the proposed CNN model, quantitative assessment of learning rate ranging from 0.001 to 0.01 with the interval of 0.001 and 0.01 to 0.1 with the interval of 0.1, has been carried out. As seen in the Fig. 5-6, the proposed model evaluation performance is increasing with the learning rate until it reaches a peak at 0.02. After the peak, the CNN model's evaluation performance begins to decrease. This suggests that the CNN model is learning well at lower learning rates, but it is more likely to overshoot the optimal solution at higher learning rates. In addition, there is a small rebound when the learning rate reaches 0.01. However, the learning rate was not increased further because the CNN model typically overfits the training data at higher learning rates.

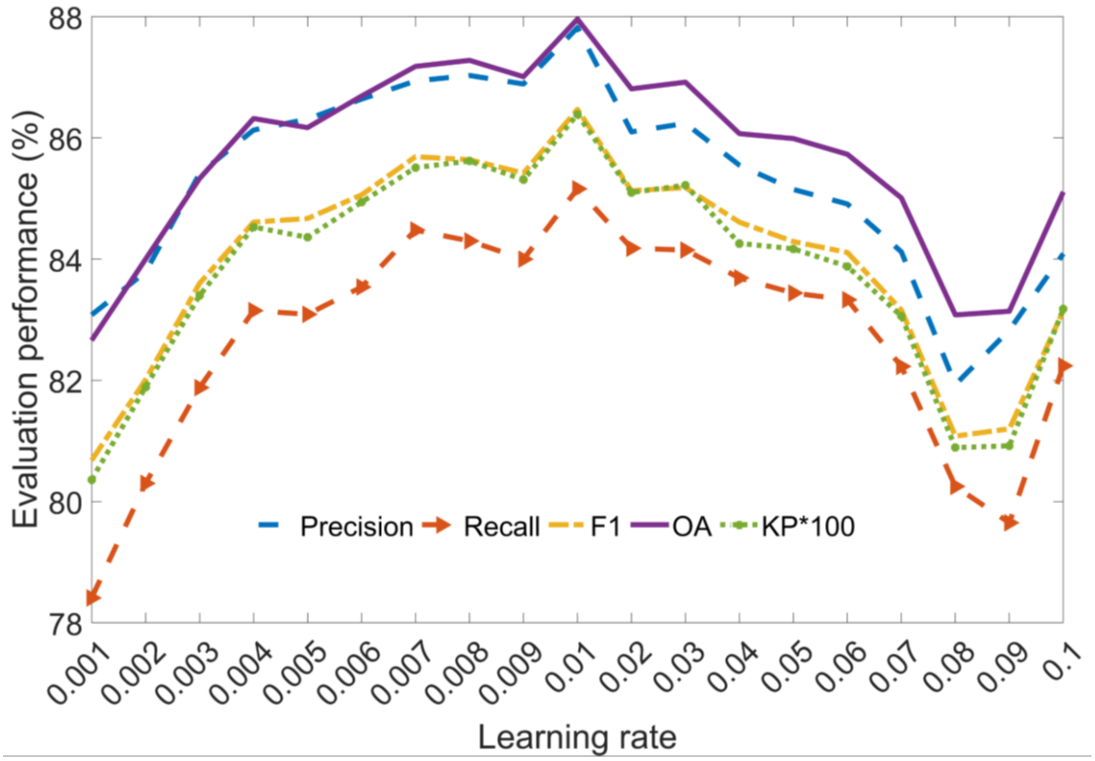


Fig. 5-6. Results for different learning rate

5.3.6 Selection of Optimizer

Table 5-IV evaluates model performance using SGD, Adam, and RMSProp optimizers. As seen, SGD achieves the highest precision, recall, F1 score, overall accuracy, and Kappa. Its scores are 2-3% better across all metrics compared to Adam and RMSProp. Adam and RMSProp perform very similarly, with Adam slightly outperforming on most metrics. But both lag significantly behind SGD. The strong performance of SGD suggests it is well-suited for this instrument recognition model and dataset. SGD's simple approach of adjusting weights based on the gradient appears to be effective for learning the parameters of the CNN architecture.

5.3.7 Selection of Dropout Rate

Dropout is a commonly used regularization technique that can effectively prevent overfitting in deep learning networks by randomly shutting off a portion of neurons during the training process. It also helps alleviate the problem of vanishing gradients. Table 5-V evaluates the impact of using 5 different dropout rates on model

Table 5-IV Results for three optimizers.

Optimizer	SGD	adam	rmsprop
Precision (%)	87.53	84.01	83.96
Recall (%)	84.65	81.90	80.53
F1 (%)	86.06	82.94	82.20
OA (%)	87.50	84.72	84.38
KP*100	85.87	82.75	82.33

Table 5-V Results for five dropout rates.

Dropout rate	0.05	0.15	0.25	0.35	0.45
Precision (%)	84.90	86.39	86.42	84.11	83.81
Recall (%)	81.79	83.18	83.20	81.66	80.43
F1 (%)	83.32	84.75	84.78	82.86	82.09
OA (%)	85.15	86.42	86.48	84.79	84.07
KP*100	83.20	84.63	84.70	82.80	81.97

performance. With no dropout (0.05), performance is decent but lower than optimal. This suggests dropout is helpful for regularization. Increasing dropout from 0.15 to 0.25 consistently improves all metrics. This indicates additional dropout continues to have a positive effect. However, going beyond 0.25 dropout causes the scores to decline again. 0.35 dropout sees a noticeable dip, and 0.45 dropout gives the poorest performance. Therefore, a dropout rate of 0.25 delivers the best results by balancing the benefits of regularization against excessive disruption of learning.

5.3.8 Comparison with Other Methods

To further evaluate the effectiveness of the proposed PMIR framework, three conventional approaches are used to benchmark in terms of precision, recall and F1-measurement[159]. Three conventional frameworks are based on Audio Content Analysis (ACA) system [58, 160] and three machine learning model (i.e. random forest (RF)[6], SVM[2] and shallow neural network (SNN)[4]). The tree number of RF is 300, the LIBSVM toolbox [2] is selected as SVM learner, and the neuron number of hidden layer in SNN is set as 70. For each music piece, ACA system is used to extract

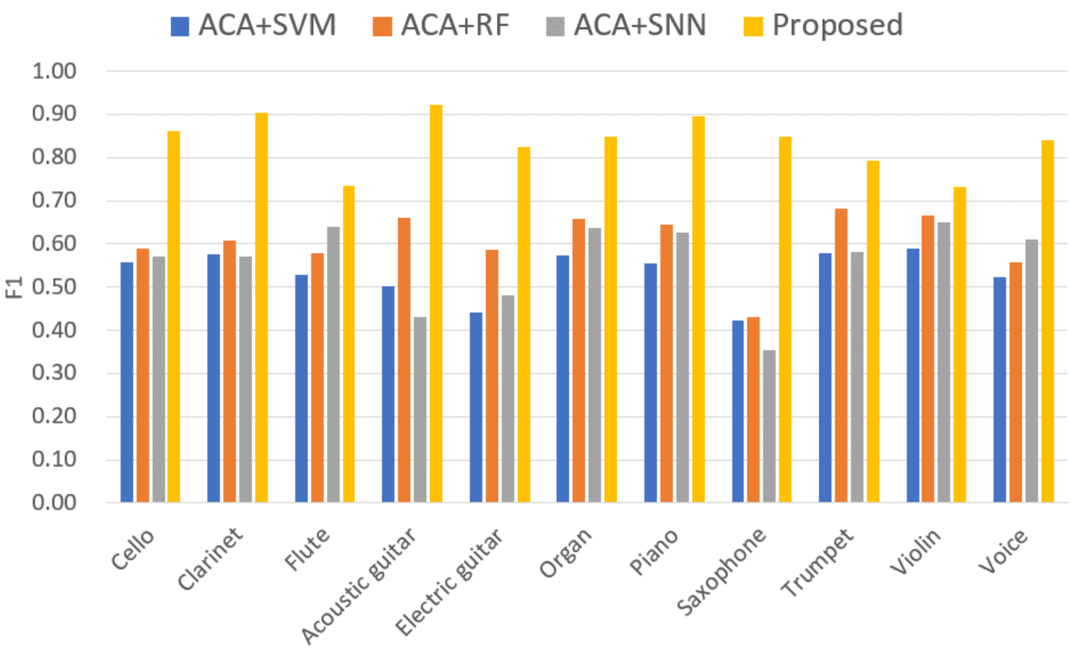


Fig. 5-7. F-measurement (F1) of each instrument of five methods.

19 features from both time and frequency domains such as Peak envelop, autocorrelation coefficients, MFCCs, pitch chroma etc. Then these features will concentrate into a feature vector with the length of 105 and be entered into machine learning model for instruments' classification task. Further, the proposed PMIR framework is compared with a new HAS-IMF algorithm proposed in 2018[8], using the same dataset and computing environment.

Results that includes the overall precision, recall and F1-measurement are presented in Table 5-VI and the F1-measurement of each instrument is shown in Fig. 5-7. As can be seen, the proposed PMIR framework generates the best overall performance and outperforms conventional frameworks in classifying individual instruments. In Table 5-VI, the ACA features are combined in the first three classifiers, i.e. SVM, SNN, and

Table 5-VI. Overall precision, recall and F1 of five methods.

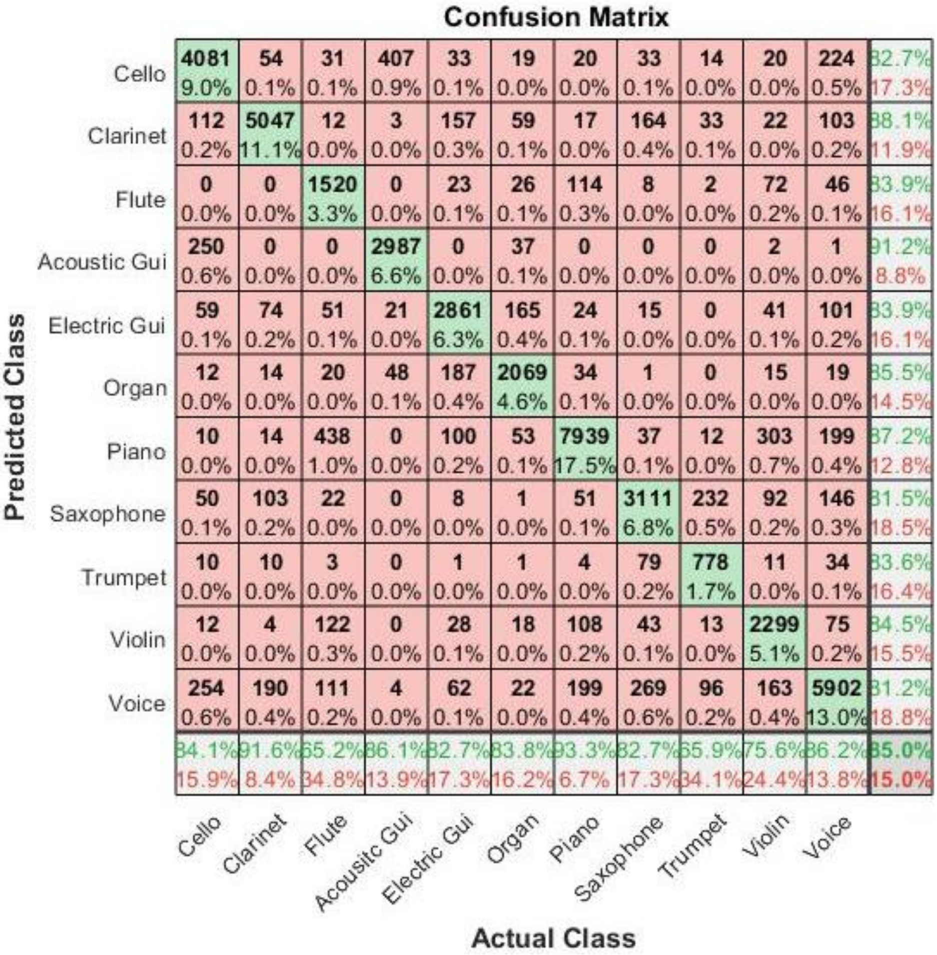
Methods	Precision	Recall	F1
ACA+SVM[2]	0.53	0.54	0.53
ACA+SNN[4]	0.57	0.56	0.56
ACA+RF[6]	0.61	0.62	0.61
HAS-IMF[8]	0.77	0.80	0.78
Proposed	0.82	0.85	0.84

RF. Among them, RF based method seems the best, yet is significantly poorer than the HAS-IMF, due mainly to the integration of the HHT spectrogram and CNN model. However, thanks to the improved CNN structure, the proposed model has significantly outperformed all others including HAS-IMF, where the precision, recall, and F1 measures are improved by 5%, 5% and 6%, respectively.

In the proposed model, the batch normalization and Max pooling are followed by each convolutional layer, and the dropout layers are put before and after the fully connected layer. However, in HAS-IMF, a dropout layer is followed by every two convolutional layers. Although the dropout layer can reduce the training time, too many dropout layers may lead to the network not fully trained. Furthermore, it does not include a batch normalization layer which may lead the data to be unbalanced. Therefore, the proposed CNN model is better than that in HAS-IMF and gives better performance.

In Fig. 5-7 and Fig. 5-8, the classification performance of individual instruments is presented. As can be seen that HAS-IMF and the proposed method are significantly better than the fusion of traditional features (ACA) and machine learning techniques. For the violin, HAS-IMF gives the best result. But with better structure of deep learning network, the proposed method produces better performance on the rest individual instruments.

Another finding is the performance of individual instruments is potentially related to the types of instruments. For example, as can be seen in Fig. 5-8, a saxophone sometimes is misclassified to a clarinet and trumpet. Because both saxophone and clarinet belong to the woodwind family, and both saxophone and trumpet belong to wind family. In addition, piano is mostly misclassified into violin and flute. The main reason is that the pitch of violin and flute is very high, and the piano sometimes compose the main melody by high pitch. Therefore, there is the confusion of piano and violin and flute. Human voice is often misclassified into cello, clarinet, piano and saxophone, etc., since it is very complicated, and its pitch or timbre may be close to some instruments.



	Cello	Clarinet	Flute	Acousitc Gui	Electric Gui	Organ	Piano	Saxophone	Trumpet	Violin	Voice	
Cello	4081 9.0%	54 0.1%	31 0.1%	407 0.9%	33 0.1%	19 0.0%	20 0.0%	33 0.1%	14 0.0%	20 0.0%	224 0.5%	82.7% 17.3%
Clarinet	112 0.2%	5047 11.1%	12 0.0%	3 0.0%	157 0.3%	59 0.1%	17 0.0%	164 0.4%	33 0.1%	22 0.0%	103 0.2%	88.1% 11.9%
Flute	0 0.0%	0 0.0%	1520 3.3%	0 0.0%	23 0.1%	26 0.1%	114 0.3%	8 0.0%	2 0.0%	72 0.2%	46 0.1%	83.9% 16.1%
Acoustic Gui	250 0.6%	0 0.0%	0 0.0%	2987 6.6%	0 0.0%	37 0.1%	0 0.0%	0 0.0%	0 0.0%	2 0.0%	1 0.0%	91.2% 8.8%
Electric Gui	59 0.1%	74 0.2%	51 0.1%	21 0.0%	2861 6.3%	165 0.4%	24 0.1%	15 0.0%	0 0.0%	41 0.1%	101 0.2%	83.9% 16.1%
Organ	12 0.0%	14 0.0%	20 0.0%	48 0.1%	187 0.4%	2069 4.6%	34 0.1%	1 0.0%	0 0.0%	15 0.0%	19 0.0%	85.5% 14.5%
Piano	10 0.0%	14 0.0%	438 1.0%	0 0.0%	100 0.2%	53 0.1%	7939 17.5%	37 0.1%	12 0.0%	303 0.7%	199 0.4%	87.2% 12.8%
Saxophone	50 0.1%	103 0.2%	22 0.0%	0 0.0%	8 0.0%	1 0.0%	51 0.1%	3111 6.8%	232 0.5%	92 0.2%	146 0.3%	81.5% 18.5%
Trumpet	10 0.0%	10 0.0%	3 0.0%	0 0.0%	1 0.0%	1 0.0%	4 0.0%	79 0.2%	778 1.7%	11 0.0%	34 0.1%	83.6% 16.4%
Violin	12 0.0%	4 0.0%	122 0.3%	0 0.0%	28 0.1%	18 0.0%	108 0.2%	43 0.1%	13 0.0%	2299 5.1%	75 0.2%	84.5% 15.5%
Voice	254 0.6%	190 0.4%	111 0.2%	4 0.0%	62 0.1%	22 0.0%	199 0.4%	269 0.6%	96 0.2%	163 0.4%	5902 13.0%	81.2% 18.8%
	84.1% 15.9%	91.6% 8.4%	65.2% 34.8%	86.1% 13.9%	82.7% 17.3%	83.8% 16.2%	93.3% 6.7%	82.7% 17.3%	65.9% 34.1%	75.6% 24.4%	86.2% 13.8%	85.0% 15.0%

Fig. 5-8. Confusion matrix of proposed methods in 11 instruments.

5.4 Chapter Summary

In this chapter, a promising framework for predominant musical instrument recognition (PMIR) in polyphonic music was introduced. The proposed method combines the Hilbert-Huang Transform (HHT) for feature extraction and a deep convolutional neural network (DCNN) for classification. The HHT is employed to generate the Hilbert spectrum of the audio data, which serves as input to the DCNN. The optimal DCNN model is trained on the IRMAS dataset, and objective evaluation demonstrates that the proposed method achieves a classification accuracy of 85%, outperforming three conventional frameworks. This result highlights the potential of image-based deep learning methods for music instrument recognition.

Although a significant progress has been made for predominant musical instrument recognition (PMIR), challenges remain in distinguishing between timbrally similar instruments and achieving high accuracy in complex, multi-instrument settings. These challenges highlight the need for more sophisticated analysis tools that can capture not only the identity of the instruments but also their interaction within a musical piece.

This leads us to the Chapter 6, which extends previous exploration by focusing on identifying and comparing various types of musical shapes, providing deeper insights into their influence on musical expressiveness and performance. Musical shape evaluation aims to address the current limitations in understanding the diverse stylistic nuances and expressive qualities of music. This holistic approach will pave the way for more advanced music analysis and generation techniques, offering a comprehensive framework for intelligent music processing.

Chapter 6

6 S-ResNN: Siamese Residual Neural Network for Musical Shape Evaluation in Piano Performance Assessment

6.1 Introduction

Music analysis encompasses several interconnected aspects that contribute to a comprehensive understanding of musical performances. Chapter 3 focused on AMT, which forms the foundation of music analysis by converting audio signals into music transcriptions, capturing basic elements such as pitch and harmonics. Chapter 4 explored PMIR, which evaluates timbral characteristics and identifies primary instruments within a music piece. Building on these technical aspects, this chapter aims to bridge the gap between sound production and artistic interpretation by focusing on musical shape evaluation (MSE).

Understanding and identifying musical shape plays a crucial role in piano performance assessment and education. Traditionally, MSE has relied on extensive musical training, resulting in a lengthy learning period and high costs in terms of time and resources. However, the potential of artificial intelligence (AI) driven models to address this gap has not been sufficiently explored. To tackle this challenge, MSE is approached as a classification problem. A Siamese Residual Neural Network (S-ResNN) is proposed to automatically identify musical shapes. The S-ResNN combines the strengths of Siamese networks and residual blocks, taking as input the spectrum generated by Constant-Q transform. To assess the performance of the proposed model, a new dataset has been created specifically for this task. The dataset contains 4,116 music pieces derived from 147 piano preparatory exercises, performed in 28 categories of musical shapes. This comprehensive dataset allows for robust testing and validation of the proposed approach. The experimental results demonstrate that the S-ResNN significantly outperforms other baseline models in terms of precision, recall, and F1 score. These promising results suggest that the proposed approach has the potential to greatly benefit piano performance assessment, offering a more efficient and objective method for evaluating musical shape.

The main contributions of this chapter are highlighted below.

1) A new dimension of MPA is discovered, which connects the human perception with music intrinsic properties;

2) A new MSE dataset is collected and released, including 4116 high-quality piano recordings in 28 classes of MSs;

3) S-ResNN method is proposed and released to evaluate the musical shape in the piano pieces.

6.2 Dataset Description

The study of musical shape is a critical aspect of understanding and analyzing expressive performance in music. Despite its importance, there is currently a lack of comprehensive datasets that focus specifically on musical shape in piano performance. Existing datasets often prioritize technical aspects such as note accuracy and timing, while neglecting the expressive and interpretive dimensions of performance. To address this gap, the Musical Shape Evaluation Dataset (MSED) is introduced as a new resource for researchers and musicians to investigate the intricate relationship between musical shape, expression, and piano performance.

6.2.1 Selection of Musical Materials

Inspired by [126] [161], a well-established educational book on piano finger practice needs to be selected to cover the possible correlation of finger strength and construct a comprehensive experiment. Schmitt's work [162] is widely regarded as a cornerstone of piano pedagogy, offering a rich variety of technical and expressive challenges for pianists of all skill levels. By focusing on Schmitt's exercises, MSED provides a musically diverse and pedagogically relevant foundation for the study of musical shape. The proposed MSED comprises 147 music pieces which were chosen to represent a wide range of musical forms, techniques, and expressive possibilities. The breakdown of the selected pieces is as follows: 83 polyphony, 20 scales, 12 arpeggios, and 32 staccato. The characteristics of each piece can be found in Appendix. B. This diverse selection allows for a comprehensive exploration of musical shape across different technical and expressive contexts. Polyphonic pieces, which feature multiple simultaneous melodic lines, offer rich opportunities for studying the shaping of harmonies and textures. Scales and arpeggios present unique challenges for shaping melodic contours and dynamic gradations. Staccato pieces, with their short and detached articulations, require precise control and expressive timing to effectively convey musical shape.

In curating the dataset, careful attention was given to the pedagogical value and technical difficulty of the selected pieces. Schmitt's exercises are known for their progressive arrangement, accommodating pianists from beginner to advanced levels. MSED reflects this range, ensuring that the dataset is accessible and relevant to a broad spectrum of piano students and teachers.

6.2.2 Musical Shape Categorization

To establish a systematic framework for studying musical shape, MSED introduces a comprehensive categorization scheme based on tempo and dynamics, two fundamental aspects of musical expression [34]. The reference point for this categorization is the "normal" musical shape, which is defined as a performance at a moderate tempo of 60 beats per minute (bpm) with standard dynamics. All other musical shape categories are derived from this baseline through systematic variations in tempo and dynamics. A single time pattern in terms of faster and slower speed is performed as 72 bpm and 50 bpm, which are represented as *Adagio* and *Largo* in music

Table 6-I. Description of 8 basic shape.

No.	Shape		Description
1	Forte	Dynamics	Strong dynamics and denoted as *f* on the score
2	Piano		Weak dynamics and denoted as *p* on the score
3	Cresc.		Gradually increase the dynamics from *p* to *f*
4	Decresc.		Gradually reduce the dynamics from *f* to *p*
5	Adagio	Time	Perform the score with 72 bpm
6	Largo		Perform the score with 50 bpm
7	Rit.		Gradually reduce the speed from 60 - 50 bpm
8	Accel.		Gradually increase the speed from 60 - 72 bpm

theory, respectively. It is worth noticing that the actual performing speed is between 50 (50 bpm * 1 note) - 288 (72 bpm * 4 notes) notes per minute. Stronger and weaker dynamics patterns at 60 bpm are represented as *Forte* and *Piano*, respectively. Gradually getting louder and gradually getting softer are represented as Cresc. (crescendo) and Decresc. (decrescendo), respectively. Gradually slowing down and gradually speeding up are represented as Rit. (ritardando) and Accel. (accelerando), respectively. These categories capture the essential contrast between loud and soft playing, which is a fundamental aspect of musical expression and shape. As shown in Table 6-I, MSED defines a set of 8 core musical shapes. These core shapes serve as the building blocks for more nuanced and complex expressions. To further enrich the dataset, an additional 16 musical shape combinations are included, as detailed in Table 6-II. These combinations represent more complexed expressive possibilities afforded by the tempo and dynamics dimensions.

Table 6-II. Description of extended 16 shapes.

	Adagio (72 bpm)	*Largo* (50 bpm)	*Rit.* (60→50 bpm)	*Accel.* (60→72 bpm)
Forte (f)	*f* + 72 bpm	*f* + 50 bpm	*f* + 60→50 bpm	*f* + 60→72 bpm
Piano (p)	*p* + 72 bpm	*p* + 50 bpm	*p* + 60→50 bpm	*p* + 60→72 bpm
Cresc. (p→f)	*p→f* + 72 bpm	*p→f* + 50 bpm	*p→f* + 60→50 bpm	*p→f* + 60→72 bpm
Decresc. (f→p)	*f→p* + 72 bpm	*f→p* + 50 bpm	*f→p* + 60→50 bpm	*f→p* + 60→72 bpm

To enhance the musical and stylistic diversity of the dataset, three supplementary musical shape categories are also included: i.e., *Swing*, *Give* and *Take*. *Swing* is the most important feature in Jazz which is a popular music style 20 century [125]. *Give* and *Take* are advanced and delayed movements of time in the music pieces, respectively [34]. In total, 28 categories (normal and 27 MSs) of shape were performed on 147 music pieces, resulting in 4116 recordings in WAV format, with a sampling rate of 48 KHz and a period of 7 seconds. In this study, music pieces with normal and 27 MSs were played by music trainers and students, respectively. All categories can be clearly separable from listening. Appendix. C shows the relationship between folder name and each MS class. To better understand the difference across different MSs, the spectrograms generated from CQT are shown in Appendix D.

6.2.3 Recording Process and Equipment

In order to ascertain a high standard of audio quality, all recordings in MSED were carried out in a professional audio recording environment setting utilizing advanced recording equipments.

The piano made use of for the recordings was a Yamaha U3H. It was thoroughly tuned and managed by an expert specialist prior to each recording session to ensure optimal tone and playability.

Sound was captured by a Sony Lavalier (Mode No. ECM-LV1). This microphone is renowned for its capacity, openness, and quality to consistently duplicate the nuances of acoustic piano sound. The microphone was set up in a coincident ORTF stereo arrangement, offering a reasonable and spatially well balanced depiction of the piano's audio area.

Throughout the recording process, strict top quality control actions were applied to guarantee uniformity and precision in the efficiencies. Several takes of each piece were tape-recorded, and the finest takes were selected for addition in the last dataset based on criteria such as note accuracy, balanced accuracy, dynamic control, and general music expressiveness.

To decrease exhaustion and preserve optimum performance high quality, taping sessions were limited to a maximum of two hours per day, with normal breaks. Performers were urged to focus on music expressiveness and form over simple technological precision, while still adhering to the offered tempo and dynamic indications.

The resulting recordings were saved in an uncompressed, high-resolution WAV style to protect the complete integrity and dynamic series of the performances. Each audio file was thoroughly annotated with metadata including the item title, music shape classification, performer name, and recording date.

6.2.4 Unique Aspects and Potential Applications

The Musical Forming Evaluation Dataset (MSED) supplies numerous one-of-a-kind function that identify it from existing datasets in the area of music efficiency analysis. Firstly, MSED is the very first dataset to concentrate especially on the principle of music form, offering a extensive and systematic structure for studying the meaningful and interpretive dimensions of piano performance. By prioritizing music shape over purely technical aspects, MSED allows scientists to delve much deeper into the communicative and creative elements of music-making.

Second of all, MSED includes performances by pianists, using an unusual possibility to study the differences in music form implementation and understanding across skill levels. This attribute makes MSED particularly valuable for research study in music education and learning and rearing, as it can educate the growth of mentor techniques and assessment devices that prioritize meaningful abilities alongside technical efficiency.

Third, the inclusion of a varied series of music kinds and forms in MSED allows for a nuanced examination of exactly how various musical aspects engage to produce meaningful meaning. By covering polyphonic structures, scales, arpeggios, and articulation styles, MSED sustains research study into the intricate interplay in between melodic, harmonic, and balanced measurements fit musical expressions.

The prospective applications of MSED are interdisciplinary and substantial. In songs education, MSED can serve as an effective source for making curricula and assessments that emphasize meaningful abilities and music understanding. By offering concrete examples of musical shapes and their understandings, MSED can assist educators and trainees create a common vocabulary and conceptual framework for going over and reviewing music expression.

In the field of music psychology, MSED uses a wealth of data for studying the cognitive and perceptual processes included in the interaction and interpretation of music meaning. Researchers can use MSED to check out exactly how listeners react and regard to different music shapes, and exactly how performers use meaningful methods to convey visual and psychological objectives. Such study can drop light on the culturally-specific and universal facets of musical experience, and inform theories of musical cognition and influence.

For songs information retrieval (MIR) and expert system (AI) research, MSED gives a useful training resource for creating algorithms that can automatically identify, identify, and create expressive efficiencies. By gaining from the professional comments and performances in MSED, artificial intelligence versions can be trained to recognize and forecast music shapes, allowing brand-new applications in meaningful performance analysis, automated songs transcription, and computer-assisted songs make-up and performance.

In the imaginative markets, MSED can offer as a resource of inspiration and referral for songs manufacturers, authors, and performers seeking to develop more mentally engaging and expressive music. By researching the meaningful strategies and approaches used by the specialist entertainers in MSED, musicians can expand their creative scheme and develop new methods to music interpretation and interaction

To demonstrate the potential of MSED in practice, consider a hypothetical study in music education that aims to evaluate the effectiveness of a new teaching method for developing expressive performance skills in novice pianists. Using MSED, researchers could compare the performances of students trained with the new method against those trained with traditional approaches, analyzing differences in musical shape execution and perception. The rich annotations and expert performances in MSED would provide a baseline for evaluating student progress and identifying areas for improvement. Such a study could have significant implications for piano pedagogy, leading to the development of more effective and evidence-based teaching strategies.

6.2.5 Overview of proposed method

Fig. 6-1 illustrates the architecture and essential parameters of the proposed method. At its core, the system processes paired audio inputs including the music piece with specific musical shape and its corresponding one with normal MS. Both inputs undergo Constant-Q transform (CQT) [39] to generate the colorful (RGB) spectrogram sized $3 \times 224 \times 224$. CQT is specifically chosen due to its widespread application in MIR tasks [163] [164] and its ability to provide detailed time-frequency analysis. The network features a sophisticated four-stage structure with the kernel number of 64, 128, 128, and 64. From the first stage to the second stage, the kernel number is increased from 64 to 128. This is because the model tries to learn more complex low-level features such as edges, textures, and local patterns. From the third stage to the fourth stage, the kernel number decreased from 128 to 64. This reduction is because, at higher levels, the model needs to integrate and abstract the high-level features learned from previous stages, rather than keep increasing the features' diversity. Meanwhile, the symmetrical design of the four-stage structure serves multiple purposes. Firstly, it can reduce the computational cost by balancing the number of parameters and the complexity of computations across different stages. This makes the model more efficient and easier to train. Secondly, the symmetrical design helps to maintain the effectiveness of the model by ensuring a smooth transition between different stages, which facilitates the flow of information and gradients during training. Residual blocks were incorporated in each stage to facilitate training of this deep network, allowing for better gradient flow and feature reuse. This choice was influenced by the success of ResNet architectures in various domains. Furthermore, the trend of gradually reducing the convolution kernel size from 7x7 to 5x5, and then to 3x3, is an empirical rule

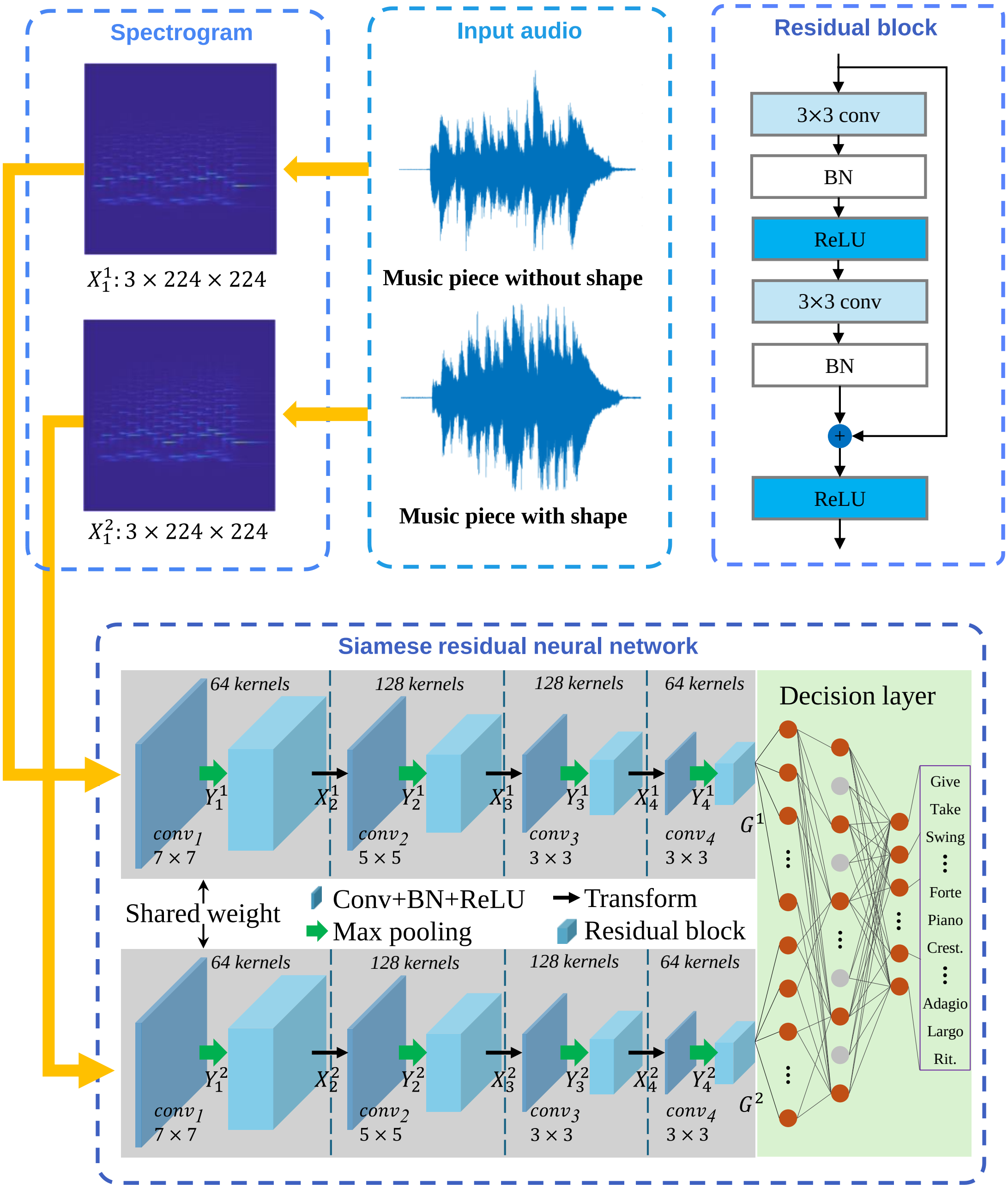


Fig. 6-1. System overview. Conv and BN represent the convolutional layer and the batch normalization layer, respectively.

gradually formed through practice in the field of deep learning. Behind this trend lies the advantage of using multiple layers of small convolution kernels, which can reduce the number of parameters, improve computational efficiency, and introduce more nonlinear transformations while maintaining the receptive field. The Siamese architecture with shared weights between branches enables the network to learn a common feature space for comparing normal and specific musical shapes. This is particularly suitable for MSE task, which aims to identify deviations from normal musical shapes.

6.2.6 Convolutional Layer

Given an input X_i^s in each stage, where $s \in [1,2]$ is the number of input spectral feature map and $i \in [1,4]$ is the number of stages. The output $Y_i^s$of each convolutional layer can be expressed as

$$Y_i^s = \sigma(BN(conv_i(X_i^s))) \tag{6.1}$$

where $\sigma(\cdot)$ and $BN(\cdot)$ represent the rectified linear unit (ReLU) activation function and batch normalization, respectively. The convolutional layer applies a set of learnable filters to the input, capturing local patterns and features. The ReLU activation function introduces non-linearity, while batch normalization helps in reducing internal covariate shift and accelerating the training process.

6.2.7 Residual Block

After the convolutional layer, the output Y_i^s is passed through a residual block, which is implemented following the standard structure in ResNet50. The residual block allows the model to learn residual functions, enabling it to capture more complex patterns and mitigate the vanishing gradient problem [70]. The input to the next stage, X_{i+1}^s is updated by Eq. (6.2).

$$X_{i+1}^s = ResBl(MP(Y_i^s)) \tag{6.2}$$

where $MP(\cdot)$ denotes a 3×3 max pooling layer with a stride 2, and $ResBl$ denotes the residual block. Max pooling reduces the spatial dimensions of the feature maps, helping to capture translation-invariant features and reduce computational complexity. The residual block is defined as:

$$y = \mathcal{F}(x) + x \tag{6.3}$$

where x and y are the input and output vectors of the layer considered. Here, x is $MP(Y_i^s)$ and y is $ResBl(MP(Y_i^s))$. The function $\mathcal{F}(x)$represents the residual mapping learned by the network. In the residual block, $\mathcal{F} = BN(conv_i\sigma BN\big(conv_i(x)\big))$ in which σ denote ReLU and the biases are omitted for simplifying notations. The shortcut connection, which adds the input *x* to the output of the residual mapping $\mathcal{F}(x)$, allows the model to learn the identity function if necessary.

6.2.8 Concatenation and Classification

The outputs of two branches, G^1 and G^2, are concatenated together, followed by a dropout layer $D(\cdot)$ and a flatten layer $F(\cdot)$. Dropout is used as a regularization technique to prevent overfitting by randomly dropping out a fraction of the units during training. The flatten layer reshapes the feature maps into a one-dimensional vector, preparing them for the final classification stage. To accurately evaluate the MSs in the piano pieces, MSE is considered as a classification task. The cross-entropy loss function $\mathcal{L}$ is adopted, which a commonly used loss function for classification tasks. The loss function is defined by Eq. (6.4).

$$\begin{gathered}\mathcal{L}_{p,l} = -\frac{1}{t}\sum_{i=1}^{t}(l * log(p) + (1-l) * log(1-p)) \\ p = F(D(concatenate[G^1, G^2]))\end{gathered} \tag{6.4}$$

where t, p and l denote the number of inputs, predicted probability, and the classification label, respectively. The loss function measures the dissimilarity between the predicted probabilities and the true labels, guiding the model to learn the correct mapping from the input spectrograms to the corresponding MS labels.

6.3 Experimental Setting

The proposed S-ResNN is trained on NVIDIA Quadro RTX 6000 with 200 epochs and a batch size of 32. For fast convergence, stochastic gradient descent is selected as the optimizer where the learning rate, momentum and weight decay are set as 1e-3, 0.9 and 0.0005, respectively. The spatial size of the input spectrogram image is set to $3 \times 224 \times 224$.

For quantitative evaluation, three widely used metrics including the precision, recall and F1 score are adopted. Each experiment was repeated 10 times, and averaged results are reported in the Section 6.4. Within each repetition, training and testing data are randomly selected without overlap. Different training rates ranging from 10% to 70% in each class have been used for training.

To validate the efficacy of the proposed model, comprehensive experiments are carried out where conventional audio content analysis (ACA) methods and deep learning models are used for benchmarking. For the audio methods, some classic MIR techniques including Zero Cross Rate (ZCR), spectral centroid (SpCen), spectral rolloff (SpRf), spectral flux (SpFlux), spectral skewness (SpSkew), spectral flatness (SpFlat) and Mel-frequency cepstral coefficients (MFCC) are used to extract the audio feature followed by a popular classifier i.e., support vector machine (SVM) for the decision making. The models of audio feature extraction and SVM is employed from ACA system [58] and LIBSVM tool [2], respectively. Deep learning models (e.g., VGG16 [66], ResNet50 [70] and DenseNet161 [71].) are employed from Openmmlab’s image classification toolbox [165].

6.4 Results and Discussions

6.4.1 Comparison with Benchmarking Methods

An objective comparison between the proposed method and other benchmarking methods is shown in Fig. 6-2. As seen, S-ResNN is comparable with Resnet50 under 30% and 40% training rates but always leads to a higher recall for the remaining training rates. This is due to the fusion of Siamese structure and residual blocks, making full use of spectral features to learn the discriminative information from various MSs.

Meanwhile, when the training rate is greater than 20%, the baseline deep learning models consistently produce better results than the ACA methods. This is due to the fact that deep learning models can extract more representative global features than ACA methods with sufficient training data. Thus, the complex time and dynamics patterns in music pieces can be better identified. In addition, the features extracted by ACA methods consist of numerous local temporal and/or spectral features that are

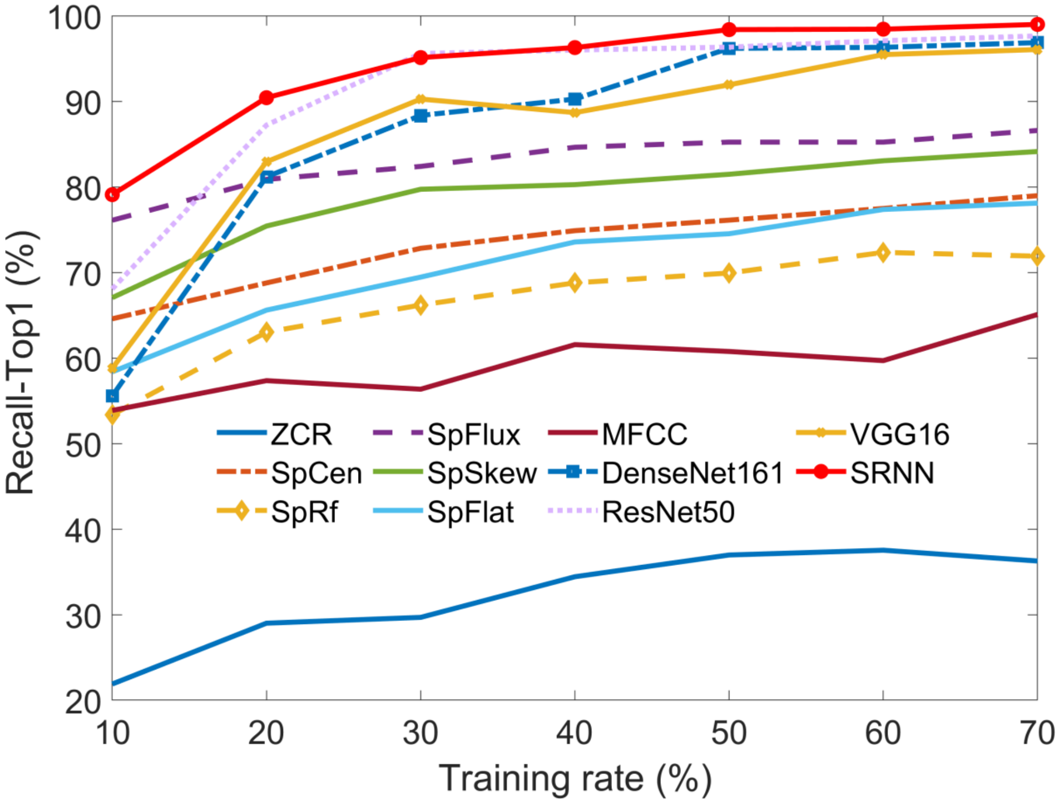


Fig. 6-2. Recall of different approaches with 7 training rates.

insufficient to characterize the MSs adequately. The parameter selection (i.e., hop and block size in ZCR and spectral properties, and number of coefficients in MFCC, etc) also affects the classification performance of ACA methods, making them less practicability.

Table 6-III shows the average classification results and standard deviation (STD) of 7 training rates, where it is seen that S-ResNN is superior to other baseline deep learning models in terms of higher classification accuracy and lower STD. ACA methods have generally lower STD, indicating their stability, but their classification accuracy is much lower than deep learning models under different training rates. On a different point, when the training sample is not sufficient (e.g., 10% training rate as seen in Fig. 6-2), the baseline deep learning models produce inferior results than some ACA methods such as SpFlux and SpSkew, etc. However, the proposed S-ResNN still yields the best accuracy, which further validates its effectiveness when dealing with limited training data.

Table 6-III. Comparison of classification performance with mean value and corresponding STD of 7 training rates

Methods	**Precision (%)**	**Recall (%)**	**F1 score (%)**
ZCR	32.25± 5.29	32.75± 5.25	32.75± 5.25
SpCen	73.37± 4.73	74.03± 4.48	73.38± 4.72
SpRf	66.50± 6.16	67.23± 5.72	66.55± 6.03
SpFulx	83.01± 3.34	83.30± 3.19	83.03± 3.31
SpSkew	78.73± 5.42	79.82± 4.67	78.87± 5.26
SpFlat	70.99± 6.55	71.90± 6.10	71.07± 6.47
MFCC	59.24± 3.44	70.23± 2.56	62.20± 3.26
VGG16	86.39± 13.63	87.12± 12.96	86.29± 13.86
ResNet50	91.13± 9.97	91.26± 9.81	91.06± 10.11
DenseNet161	86.30± 11.95	87.04± 10.87	86.24± 12.07
S-ResNN	93.81± 6.60	93.98± 6.43	93.78± 6.67

Table 6-IV. Comparison of proposed method and baseline deep learning models on efficiency

Method	VGG16	ResNet50	DenseNet161	S-ResNN
Params (M)	138.36	25.56	28.68	14.88
Flops (G)	15.5	4.12	7.82	9.78

6.4.2 Parameter Efficiency

Table 6-IV reveals the efficiency of the proposed method and baseline deep learning models. It is observed that the proposed S-ResNN has fewest parameters but needs adequate computation cost. The main reason is that S-ResNN has fewer weighted layer than baseline deep learning models, but the size of fully connected (FC) layer at the end of each branch is 6400×2048, leading to higher Flops than ResNet50 and DenseNet161. This issue can be potentially solved by reducing the kernel size but increasing the number of convolutional layers with larger stride. With a deeper structure but fewer spatial size of convolutional feature maps, the size of FC layers can be much reduced, and the discriminative information of MSs can be well extracted. Thus, a much better balance between effectiveness and efficiency can be achieved.

Table 6-V Results of different batch size

Batchsize	8	16	32	48	64
Precision (%)	77.76	81.95	81.28	78.56	72.71
Recall (%)	76.6	80.84	80.46	77.40	70.47
F1 (%)	76.34	80.82	80.34	77.14	70.40
OA (%)	76.6	80.84	80.46	77.40	70.47
KP*100	75.76	80.13	79.74	76.56	69.38

6.4.3 Selection of Batch Size

Table 6-V provides a comparison of performance metrics across several batch sizes — 8, 16, 32, 48 and 64 — for a deep learning model. When evaluating OA (Eq. (2.24)), KP (Eq. (2.26)), precision (Eq. (2.27)), recall (Eq. (2.28)) and F1 score (Eq. (2.29)), a batch size of 16 consistently achieves the best results, with over 80% on all metrics. Smaller batch sizes like 8 underperform on all evaluation criteria, likely because they see limited data for each update. Larger sizes beyond 16 also decline in performance somewhat as well, indicating batch statistics and noise reduction diminish after 16. Given the goal of maximizing predictive power across many metrics, the analysis clearly determines that a batch size of 16 is optimal for this particular deep learning architecture and dataset. By properly tuning this hyperparameter and selecting 16, model performance reaches peak efficiency based on precision, recall, accuracy and other vital measures for employing the model successfully in practice.

6.4.4 Selection of Learning Rate

Table 6-VI compares model performance across several learning rate values to determine the optimal learning rate hyperparameter for the deep learning model. The key observations are:

1. A learning rate between 0.002-0.005 results in very comparable performance across all evaluation metrics, where all these metrics reach their peak values at 0.003 and highlighted in bold in the table.
2. When learning is further increased to 0.009 and above, all the evaluation metrics

Table 6-VI Results of different learning rate (lr)

lr	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009	0.01
OA (%)	79.30	85.82	86.90	85.25	85.20	84.28	80.76	76.02	75.07	69.47
Precision(%)	79.84	86.21	87.40	85.85	85.75	85.62	80.90	78.45	78.33	72.15
Recall(%)	79.30	85.82	86.90	85.25	85.19	84.28	80.76	76.01	75.08	69.47
F1(%)	79.22	85.73	86.80	85.13	85.12	84.30	80.53	74.23	73.35	68.63
KP*100	78.53	85.29	86.42	84.70	84.65	83.69	80.05	75.13	74.15	68.34

Table 6-VII Results of different dropout rates

Dropout rate	0.2	0.3	0.4	0.5	0.6	0.7
Precision (%)	73.48	86.21	83.18	81.95	82.10	74.86
Recall (%)	73.48	85.82	82.60	80.84	80.65	72.18
F1 (%)	73.48	85.73	82.51	80.82	80.61	71.77
OA (%)	73.48	85.82	82.60	80.84	80.65	72.18
KP*100	73.48	85.29	81.95	80.13	79.93	71.15

drop down dramatically. This suggests the learning rate has become far too large, causing the training optimization to become unstable and ineffective. Essentially, the model parameters are changing too drastically with each update for learning to properly occur.

3. For the rest of learning rate, the model shows degraded performance. This likely indicates that the model is slightly underfitting and lose generalization capabilities.

In summary, 0.003 is the optimal learning rate for performance based on this comprehensive evaluation using accuracy, precision, recall, F1, and other metrics.

6.4.5 Selection of Dropout Rate

Table 6-VII shows the model performance varying by dropout rate. A dropout rate of 0.3 yields the best performance across most evaluation metrics. It achieves the highest precision of 86.21%, recall of 85.82%, F1 score of 85.73%, overall accuracy of 85.82% and Kappa score of 85.29%.

Higher dropout rates such as 0.5, 0.6 and especially 0.7 have comparatively poorer performance on precision, recall, etc. This is likely caused by too much regularization at very high dropout levels.

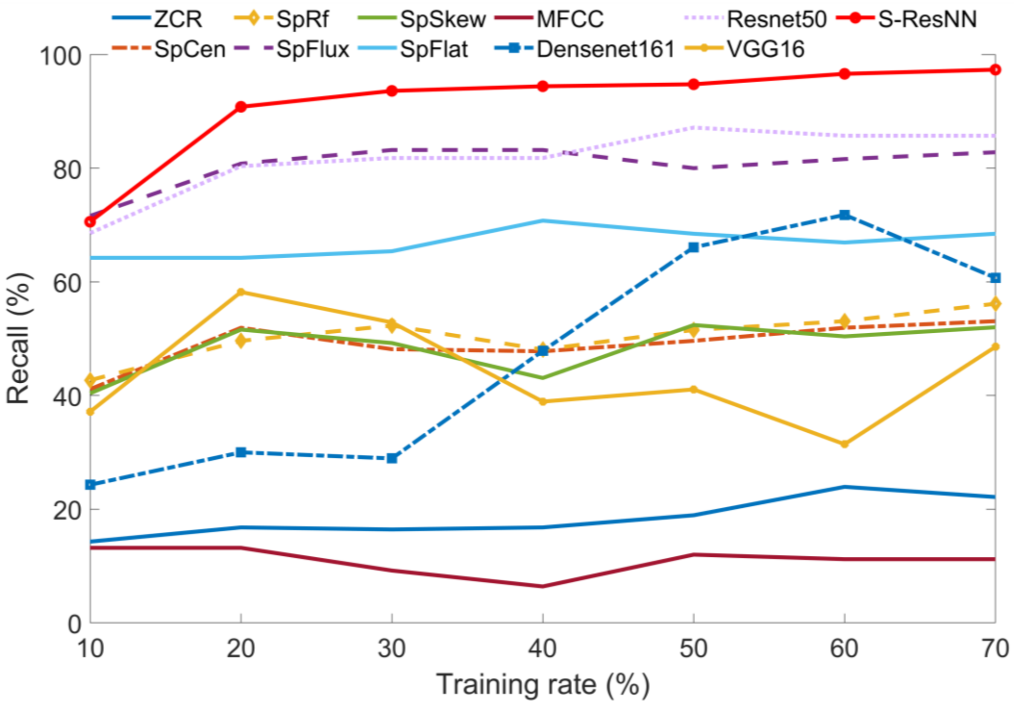


Fig. 6-3. Comparison of various approaches in dependence of different training rates on Hanon music recordings.

In contrast, lower dropout rates like 0.2 deteriorates performance substantially as seen by dramatic reductions in accuracy, F1 and other scores. This indicates model generalization suffers greatly due to too little regularization.

6.4.6 Generalization Experiment

To further validate the robustness and reliability of the proposed model, a generalization experiment is conducted where 10 music pieces selected from Hanon [166] were performed, resulting 250 music recordings with 25 classes of musical shapes (excluding *Swing, Give and Take*). Then all models trained on Schmitt music pieces will be directly tested on Hanon music pieces. Comparison of various models using different training rates is shown in Fig. 6-3. As seen, the proposed S-ResNN has much better generalization ability than other benchmark methods though it produces comparable accuracy to SpFlux in 10% training rate. This actually motivates us to improve the few-shot learning ability of the model by combining data augmentation, attention mechanisms and meta-learning strategies in the future.

6.5 Chapter Summary

This chapter introduces a new insight of MSE into MPA. MSE is a bridge to link the human perception with music's intrinsic properties, which addresses the shortage of existing MPA framework. A new architecture S-ResNN[2] is proposed for MSE, where a new MSE dataset[3] is also released as an extra outcome. Experimental results have shown the proposed method outperforms not only conventional benchmarking approaches but also several deep learning models. Future work will focus on further improving and validating the model on wider application scenarios, where the current dataset will be extended to include increased categories of music scores and MSs. Furthermore, some open-source tools such as MusicXML and MusPy [167] can be used to adjust the time and dynamics to various levels that are sometimes hard to adjust manually.

[2] https://github.com/lixiaoquan731/ICASSP2023

[3] https://zenodo.org/record/7225090#.Y0_mCXbMKUk

Chapter 7

7 Conclusion

The present thesis mainly focused on the methodologies of pattern recognition for computational musicology. The main contributions cover three different area such as Multi-pitch Estimation, Predominant Instrument Recognition, and Music Shape Analysis.

7.1 Thesis Summary

Chapter 2 covers fundamental concepts of music cognition, including pitch, timbre, shape, and music notation tools. It provides the understanding of music knowledge and their role in supporting the study aims in this thesis. This theoretical framework establishes the essential cognitive and perceptual foundations necessary for understanding how humans process and interpret musical information, forming the conceptual basis for the advanced music analysis methodologies developed in subsequent chapters.

Chapter 3 then delves into music signal pre-processing techniques, exploring time-frequency representations like Short-time Fourier Transform (STFT) and Constant-Q transform (CQT), as well as matrix factorization methods such as Non-negative Matrix Factorization (NMF) and Probabilistic Latent Component Analysis (PLCA). It also discusses machine learning approaches for music information retrieval, including Artificial Neural Networks (ANNs), Convolutional Neural Networks (CNNs), and specific architectures like VGG, ResNet, and DenseNet. Moreover, state-or-the-art works for AMT, PMIR and MSE are presented and the challenges and opportunities in each field are discussed. Finally, the chapter outlines various evaluation metrics used to assess the performance of predictive models in music analysis tasks. This comprehensive overview lays a solid foundation for understanding the current state of research in music information retrieval and sets the stage for the novel contributions presented in subsequent chapters.

In Chapter 4, a new multi-pitch estimation model is proposed. For most music pieces, pitchs (fundamental frequencies) are their main characteristics. Therefore, by taking the advantage of constant-Q transform and SI-PLCA, the pianoroll description map can be extracted. Then a harmonic structure detection model is proposed to detect and remove the harmonics from the pianoroll description map and only keep the pitch (fundamental frequencies). Finally, a simply yet effective note tracking strategy is proposed to link the breaking pitches. The proposed MPE model is evaluated on three datasets (MAPs, BACH10, and TRIOS) and compared with 11 state-of-the-art methods, demonstrating superior performance. This accurate pitch detection forms a crucial first step in understanding the complexities of musical compositions.

In Chapter 5, a promising framework for predominant musical instrument recognition (PMIR) in polyphonic music is introduced. This step moves the analysis from basic pitch detection to understanding the timbral characteristics of the music. Two stages are included in the proposed framework, i.e., Hilbert Huang transform (HHT) for spectrogram generation and end-to-end convolutional neural network (DCNN) for deep feature extraction and decision making. The Hilbert spectrograms generated from audio data serve as input to the DCNN, providing a rich representation of the musical signal that captures both pitch and timbral information. The proposed system is tested on IRMAS dataset containing 6705 audio clips of 11 instrument categories. The classification accuracy reaches 85% which outperforms other three conventional frameworks. This high accuracy in instrument recognition, combined with the pitch estimation from chapter 3, provides a comprehensive understanding of the musical content, setting the stage for more nuanced analysis of musical performance.

Chapter 6 builds on the foundational work of Chapters 4 and 5, moving beyond pitch and instrument recognition to assess the expressive qualities of musical performances. This chapter introduces a new approach to musical performance assessment by focusing on musical shape evaluation is introduced. Leveraging the spectral analysis techniques introduced in previous chapters, a Siamese residual neural network (S-ResNN) is proposed to automatically identify musical shapes, treating MSE as a classification problem. The input to the S-ResNN is a spectrogram generated using the Constant-Q transform (CQT). On the other hand, a new dataset, the Musical Shape Evaluation Dataset (MSED), is created to train and evaluate the proposed model. The dataset contains 4116 piano recordings derived from 147 piano preparatory exercises performed in 28 categories of musical shapes. The dataset is carefully curated, considering factors such as musical materials, shape categorization, recording process, and equipment. Experimental results demonstrate that the S-ResNN outperforms conventional audio content analysis (ACA) methods and other deep learning models, such as VGG16, ResNet50, and DenseNet161. This advancement in musical shape evaluation represents a significant step towards automated assessment of the artistic and expressive aspects of music performance, complementing the technical analysis of pitch and instrumentation developed in the previous chapters.

7.2 Future Work

Although the contributions in the present thesis have achieved a certain level of success, there are still several challenges which can be translated to potential improvements and further investigation as summarized below:

In the multi-pitch estimation system, integrating blind source separation techniques would address the assumption of prior knowledge about the instruments present in the music piece, thereby enhancing the system's autonomy. To further improve pitch estimation accuracy, analyzing beat and chord information and incorporating deep learning models such as transformer networks can be explored. Moreover, introducing additional music perceptions, such as ornaments and rhythm, into the model can lead to a more comprehensive and accurate interpretation of the music pieces.

For predominant musical instrument recognition, the feature extraction capability of the system can be enhanced by fusing multiple spectrogram representations, such as the constant-Q transform and MFCC, with the Hilbert-Huang Transform (HHT). Incorporating popular deep feature extraction modules, such as multiscale convolutional layers, dynamic convolution, and self-attention mechanisms, can further improve the system's feature extraction and decision-making abilities. The effectiveness and robustness of the improved model will be validated using additional datasets, including MedleyDB and OpenMIC-2018, to ensure its applicability in diverse musical contexts.

Future work in musical shape evaluation will focus on expanding the dataset and refining the model. The current dataset will be extended to encompass a wider range of music scores and musical shapes, enhancing the model's ability to generalize to various musical styles and expressions. Open-source tools, such as MusicXML and MusPy, will be employed to adjust time and dynamics to various levels, overcoming the limitations of manual performance and enabling the creation of a more comprehensive and diverse dataset. To improve the decision-making capability of the current model, data augmentation techniques [170], and attention mechanisms [171] will be investigated and incorporated. These enhancements aim to enable the model to learn effectively from limited examples and focus on the most relevant features for accurate musical shape evaluation.

References

[1] M. Müller, "Fourier analysis of signals," in Fundamentals of music processing: Springer, pp. 39-117, 2021.

[2] C. C. Chang, 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.

[3] E. Benetos, S. Cherla, and T. Weyde, "An efficient shift-invariant model for polyphonic music transcription," in 6th International Workshop on Machine Learning and Music, 2013.

[4] R. Battiti, "First-and second-order methods for learning: between steepest descent and Newton's method," Neural computation, vol. 4, no. 2, pp. 141-166, 1992.

[5] E. Benetos and S. Dixon, "A shift-invariant latent variable model for automatic music transcription," Computer Music Journal, vol. 36, no. 4, pp. 81-94, 2012.

[6] A. Liaw and M. Wiener, "Classification and regression by randomForest," R news, vol. 2, no. 3, pp. 18-22, 2002.

[7] E. Vincent, N. Bertin, and R. Badeau, "Adaptive harmonic spectral decomposition for multiple pitch estimation," IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 3, pp. 528-537, 2010.

[8] D. Kim, T. T. Sung, S. Cho, G. Lee, and C. B. Sohn, "A Single Predominant Instrument Recognition of Polyphonic Music Using CNN-based Timbre Analysis," International Journal of Engineering & Technology, vol. 7, no. 3.34, pp. 590-593, 2018.

[9] Z. Duan, B. Pardo, and C. Zhang, "Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions," IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 8, pp. 2121-2133, 2010.

[10] A. Klapuri, "Multipitch analysis of polyphonic music and speech signals using an auditory model," IEEE Transactions on Audio, Speech, and Language Processing, vol. 16, no. 2, pp. 255-266, 2008.

[11] L. Su and Y.-H. Yang, "Combining spectral and temporal representations for multipitch estimation of polyphonic music," IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), vol. 23, no. 10, pp. 1600-1612, 2015.

[12] M. Marolt, "A connectionist approach to automatic transcription of polyphonic piano music," IEEE Transactions on Multimedia, vol. 6, no. 3, pp. 439-449, 2004.

[13] S. Sigtia, E. Benetos, and S. Dixon, "An end-to-end neural network for polyphonic piano music transcription," IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), vol. 24, no. 5, pp. 927-939, 2016.

[14] L. Su, "Between homomorphic signal processing and deep neural networks: Constructing deep algorithms for polyphonic music transcription," in Proceedings of 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 884-891, Aloft Kuala Lumpur Sentral, Malaysia, December, 2017.

[15] R. Kelz and G. Widmer, "Towards interpretable polyphonic transcription with invertible neural networks," in Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR, Delft, Netherlands, 2019.

[16] AstuteAnalytica, "Online Music Education Market - Industry Dynamics, Market Size, and Opportunity Forecast To 2031," Available: https://www.astuteanalytica.com/industry-report/online-music-education-market, accessed 27 Jan 2023.

[17] ABRSM. "ABRSM marking criteria." Available: https://gb.abrsm.org/en/our-exams/information-and-regulations/graded-music-exam-marking-criteria/, accessed 31 Oct 2023.

[18] G. Kaye, "International Standard of Concert Pitch," Nature, vol. 143, no. 27, pp. 905-6, 1939.

[19] E. Benetos, S. Dixon, Z. Duan, and S. Ewert, "Automatic music transcription: An overview," IEEE Signal Processing Magazine, vol. 36, no. 1, pp. 20-30, 2018.

[20] G. Doras and G. Peeters, "A prototypical triplet loss for cover detection," in Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, pp. 3797-3801, 2020.

[21] S. M. Demorest and P. Q. Pfordresher, "Singing accuracy development from K-adult: A comparative study," Music Perception: An Interdisciplinary Journal, vol. 32, no. 3, pp. 293-302, 2015.

[22] D. Molero, S. Schez-Sobrino, D. Vallejo, C. Glez-Morcillo, and J. Albusac, "A novel approach to learning music and piano based on mixed reality and gamification," Multimedia Tools and Applications, vol. 80, no. 1, pp. 165-186, 2021.

[23] N. H. Fletcher and T. D. Rossing, "The physics of musical instruments," Springer Science & Business Media, 2012.

[24] M. Vail, "The synthesizer: a comprehensive guide to understanding, programming, playing, and recording the ultimate electronic music instrument," Oxford University Press, 2014.

[25] S. Handel, "Timbre perception and auditory object identification," Hearing, vol. 2, pp. 425-461, 1995.

[26] R. Füg, A. Niedermeier, J. Driedger, S. Disch, and M. Müller, "Harmonic-percussive-residual sound separation using the structure tensor on spectrograms," in Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, pp. 445-449, 2016.

[27] G. Peeters, B. L. Giordano, P. Susini, N. Misdariis, and S. McAdams, "The timbre toolbox: Extracting audio descriptors from musical signals," The Journal of the Acoustical Society of America, vol. 130, no. 5, pp. 2902-2916, 2011.

[28] S. Li and R. Timmers, "Teaching and learning of piano timbre through teacher–student interactions in lessons," Frontiers in psychology, vol. 12, p. 576056, 2021.

[29] S. K. Langer, "Philosophy in a new key: A study in the symbolism of reason, rite, and art," Harvard University Press, 2009.

[30] D. N. Stern, "The present moment in psychotherapy and everyday life (norton series on interpersonal neurobiology)," WW Norton & Company, 2004.

[31] D. N. Stern, "Forms of vitality: Exploring dynamic experience in psychology, the arts, psychotherapy, and development," Oxford University Press, 2010.

[32] Z. Eitan and R. Granot, "How music moves: Musical parameters and listeners images of motion," Music perception: An Interdisciplinary Journal, vol. 23, no. 3, pp. 221-248, 2006.

[33] M. B. Küssner and D. Leech-Wilkinson, "Investigating the influence of musical training on cross-modal correspondences and sensorimotor skills in a real-time drawing paradigm," Psychology of Music, vol. 42, no. 3, pp. 448-469, 2014.

[34] D. Leech-Wilkinson, "Musical shape and feeling," Music and Shape, Oxford University Press, vol. 3, p. 359, 2017.

[35] A. Ockelford, "Zygonic theory: introduction, scope, and prospects," vol. 6, no. 1, pp. 91-172, 2009.

[36] A. Ockelford, "Shape in music notation," Music and Shape, Oxford University Press, vol. 3, p. 129, 2017.

[37] A. P. Klapuri, A. J. Eronen, and J. T. Astola, "Analysis of the meter of acoustic musical signals," IEEE Transactions on Audio, Speech, and Language Processing, vol. 14, no. 1, pp. 342-355, 2006

[38] J. C. Brown, "Calculation of a constant Q spectral transform," The Journal of the Acoustical Society of America, vol. 89, no. 1, pp. 425-434, 1991.

[39] C. Schörkhuber and A. Klapuri, "Constant-Q transform toolbox for music processing," in Proceedings of the 7th Sound and Music Computing (SMC) Conference, Barcelona, Spain, pp. 3-64, 2010.

[40] G. Ballou, "Handbook for sound engineers," Taylor & Francis, 2013.

[41] B. C. Moore and B. R. Glasberg, "Suggested formulae for calculating auditory-filter bandwidths and excitation patterns," The journal of the acoustical society of America, vol. 74, no. 3, pp. 750-753, 1983.

[42] B. R. Glasberg and B. C. Moore, "Derivation of auditory filter shapes from notched-noise data," Hearing research, vol. 47, no. 1-2, pp. 103-138, 1990.

[43] N. E. Huang et al., "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," in Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences, vol. 454, no. 1971, pp. 903-995, 1998.

[44] G. Rilling, P. Flandrin, and P. Goncalves, "On empirical mode decomposition and its algorithms," in IEEE-EURASIP workshop on nonlinear signal and image processing, vol. 3, no. 3, pp. 8-11, IEEE, Grado, 2003.

[45] D. Donnelly, "The fast Fourier and Hilbert-Huang transforms: a comparison," in The Proceedings of the Multiconference on" Computational Engineering in Systems Applications", IEEE, vol. 1, pp. 84-88, 2006.

[46] Z. Peng, W. T. Peter, F. L. Chu, "A comparison study of improved Hilbert–Huang transform and wavelet transform: Application to fault diagnosis for rolling bearing," Mechanical systems and signal processing, vol. 19, no. 5, pp. 974-988, 2005.

[47] D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," Advances in neural information processing systems, vol. 13, 2000.

[48] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, IEEE, pp. 177-180, 2003.

[49] N. Bertin, R. Badeau, and G. Richard, "Blind signal decompositions for automatic transcription of polyphonic music: NMF and K-SVD on the benchmark," in Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE,vol. 1, pp. I-65, 2007.

[50] P. Comon, "Independent component analysis, a new concept?," Signal processing, vol. 36, no. 3, pp. 287-314, 1994.

[51] S. A. Abdallah and M. D. Plumbley, "Polyphonic music transcription by non-negative sparse coding of power spectra," in 5th International Conference on Music Information Retrieval (ISMIR), pp. 318-325, 2004.

[52] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 3, pp. 1066-1074, 2007.

[53] P. Smaragdis, B. Raj, and M. Shashanka, "A probabilistic latent variable model for acoustic modeling," Advances in models for acoustic processing, NIPS, vol. 148, pp. 8-1, 2006.

[54] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," Journal of the royal statistical society. Series B (methodological), pp. 1-38, 1977.

[55] P. Smaragdis, B. Raj, and M. Shashanka, "Supervised and semi-supervised separation of sounds from single-channel mixtures," in Proceedings of the International Conference on Independent Component Analysis and Signal Separation, Springer, pp. 414-421, 2007.

[56] G. Grindlay and D. P. Ellis, "Transcribing multi-instrument polyphonic music with hierarchical eigeninstruments," IEEE Journal of Selected Topics in Signal Processing, vol. 5, no. 6, pp. 1159-1169, 2011.

[57] E. Benetos, R. Badeau, T. Weyde, and G. Richard, "Template adaptation for improving automatic music transcription," in Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR, 2014.

[58] A. Lerch, "An introduction to audio content analysis: Applications in signal processing and music informatics," Wiley-IEEE Press, 2012.

[59] I. Goodfellow, Y. Bengio, and A. Courville, "Deep learning," MIT press, 2016.

[60] M. Müller, "Fundamentals of music processing: Audio, analysis, algorithms, applications," Springer, 2015.

[61] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," IEEE Transactions on speech and audio processing, vol. 10, no. 5, pp. 293-302, 2002.

[62] J.P. Briot, G. Hadjeres, and F. D. Pachet, "Deep learning techniques for music generation," Springer, 2020.

[63] A. Roberts, J. Engel, C. Raffel, C. Hawthorne, and D. Eck, "A hierarchical latent vector model for learning long-term structure in music," in Proceedings of the International conference on machine learning, PMLR, pp. 4364-4373, 2018

[64] M. Schedl, "Deep learning in music recommendation systems," Frontiers in Applied Mathematics and Statistics, vol. 5, p. 457883, 2019.

[65] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.

[66] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.

[67] K. Choi, G. Fazekas, M. Sandler, and K. Cho, "Convolutional recurrent neural networks for music classification," in Proceedings of the 2017 IEEE International conference on acoustics, speech and signal processing (ICASSP), IEEE, pp. 2392-2396, 2017.

[68] C. Hawthorne et al., "Onsets and frames: Dual-objective piano transcription," in Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR, 2017.

[69] C. Cancino-Chacón, S. Peter, S. Chowdhury, A. Aljanaki, and G. Widmer, "On the characterization of expressive performance in classical music: First results of the con espressione game," in Proceedings of the 21st International Society for Music Information Retrieval Conference, ISMIR, 2020..

[70] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.

[71] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700-4708, 2017.

[72] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132-7141, 2018.

[73] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1492-1500, 2017.

[74] P. C. Chang, Y. S. Chen, and C. H. Lee, "MS-SincResnet: Joint learning of 1D and 2D kernels using multi-scale SincNet and ResNet for music genre classification," in Proceedings of the 2021 international conference on multimedia retrieval, pp. 29-36, 2021

[75] A. Geroulanos and T. Giannakopoulos, "Emotion Recognition in Music Using Deep Neural Networks," in Advances in Speech and Music Technology: Computational Aspects and Applications: Springer, pp. 193-213, 2022..

[76] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 6, pp. 1643-1654, 2010.

[77] M. Bay, A. F. Ehmann, and J. S. Downie, "Evaluation of Multiple-F0 Estimation and Tracking Systems," in Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR, pp. 315-320, 2009.

[78] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic music transcription: challenges and future directions," Journal of Intelligent Information Systems, vol. 41, no. 3, pp. 407-434, 2013.

[79] Y. Chunghsin, "Multiple fundamental frequency estimation of polyphonic recordings," Ph. D. dissertation, University Paris 6, 2008.

[80] E. Benetos and S. Dixon, "Joint multi-pitch detection using harmonic envelope estimation for polyphonic music transcription," IEEE Journal of Selected Topics in Signal Processing, vol. 5, no. 6, pp. 1111-1123, 2011.

[81] B. Fuentes, R. Badeau, and G. Richard, "Adaptive harmonic time-frequency decomposition of audio using shift-invariant PLCA," in Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 401-404, 2011.

[82] H. Wu, A. Marmoret, and J. E. Cohen, "Semi-Supervised Convolutive NMF for Automatic Piano Transcription," in Proceedings of Sound and Music Computing (SMC) Conference, St Etienne, France, 2022.

[83] E. Vincent and M. D. Plumbley, "Efficient Bayesian inference for harmonic models via adaptive posterior factorization," Neurocomputing, vol. 72, no. 1-3, pp. 79-87, 2008.

[84] K. W. Cheuk, Y.-J. Luo, E. Benetos, and D. Herremans, "The Effect of Spectrogram Reconstruction on Automatic Music Transcription: An Alternative Approach to Improve Transcription Accuracy," in Proceedings of the 25th International Conference on Pattern Recognition (ICPR), pp. 9091-9098, 2021.

[85] K. W. Cheuk, D. Herremans, and L. Su, "Reconvat: A semi-supervised automatic music transcription framework for low-resource real-world data," in Proceedings of the 29th ACM International Conference on Multimedia, pp. 3918-3926, 2021.

[86] H. Mukherjee, S. M. Obaidullah, S. Phadikar, and K. Roy, "MISNA-A musical instrument segregation system from noisy audio with LPCC-S features and extreme learning," Multimedia Tools and Applications, vol. 77, no. 21, pp. 27997-28022, 2018.

[87] H. Mukherjee, A. Dhar, S. M. Obaidullah, K. Santosh, S. Phadikar, and K. Roy, "Segregating Musical Chords for Automatic Music Transcription: A LSTM-RNN Approach," in Proceedings of the International Conference on Pattern Recognition and Machine Intelligence, Springer, pp. 427-435, 2019.

[88] Z. Fan, J. Jang, and C. Lu, "Singing voice separation and pitch extraction from monaural polyphonic audio music via DNN and adaptive pitch tracking," in Proceedings of the 2016 IEEE Second International Conference on Multimedia Big Data (BigMM), pp. 178-185, 2016.

[89] Y. Yan et al., "Unsupervised image saliency detection with Gestalt-laws guided optimization and visual attention based refinement," Pattern Recognition, vol. 79, pp. 65-78, 2018.

[90] R. M. Bittner, J. J. Bosch, D. Rubinstein, G. Meseguer-Brocal, and S. Ewert, "A lightweight instrument-agnostic model for polyphonic note transcription and multipitch estimation," in Proceedings of the 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 781-785, 2022.

[91] Q. Kong, B. Li, X. Song, Y. Wan, Y. Wang, "High-resolution piano transcription with pedals by regressing onset and offset times," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 3707-3717, 2021.

[92] Y. T. Wu et al., "Omnizart: A general toolbox for automatic music transcription," Journal of Open Source Software, vol. 6, no.68, p.3391, 2021.

[93] K. Toyama, T. Akama, Y. Ikemiya, Y. Takida, W. Liao, and Y. Mitsufuji, "Automatic Piano Transcription With Hierarchical Frequency-Time Transformer," in Proceedings of the 24th International Society for Music Information Retrieval Conference, ISMIR, 2023.

[94] J. Gardner, I. Simon, E. Manilow, C. Hawthorne, and J. Engel, "MT3: Multi-task multitrack music transcription," in Proceedings of the 10th International Conference on Learning Representations (ICLR), 2022.

[95] C. Hawthorne, I. Simon, R. Swavely, E. Manilow, and J. Engel, "Sequence-to-sequence piano transcription with transformers," in Proceedings of the 22nd Int. Society for Music Information Retrieval Conference, ISMIR, 2021.

[96] L. Ou, Z. Guo, E. Benetos, J. Han, and Y. Wang, "Exploring transformer’s potential on automatic piano transcription," in Proceedings of the 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 776-780, 2022.

[97] R. Pichevar and J. Rouat, "Monophonic sound source separation with an unsupervised network of spiking neurones," Neurocomputing, vol. 71, no. 1-3, pp. 109-120, 2007.

[98] T. C. Justus and J. J. Bharucha, "Music Perception and Cognition," Stevens' Handbook of Experimental Psychology, Sensation and Perception, p. 453, 2002.

[99] J. M. Bernardo and A. F. Smith, "Bayesian theory," John Wiley & Sons, vol. 405, 2009.

[100] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of quasi-harmonic sounds in colored noise," in Proceedings of the 10th International Conference on Digital Audio Effects (DAFx-07), 2007.

[101] Z. Duan and D. Temperley, "Note-level Music Transcription by Maximum Likelihood Sampling," in Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR, pp. 181-186, 2014.

[102] B. Maman and A. H. Bermano, "Unaligned supervision for automatic music transcription in the wild," in Proceedings of International Conference on Machine Learning Research, PMLR, pp. 14918-14934, 2022.

[103] P. A. Alvarado Duran, "Acoustically Inspired Probabilistic Time-domain Music Transcription and Source Separation," Ph. D. dissertation, Queen Mary University of London, 2020.

[104] R. Nishikimi, E. Nakamura, K. Itoyama, and K. Yoshii, "Musical Note Estimation for F0 Trajectories of Singing Voices Based on a Bayesian Semi-Beat-Synchronous HMM," in Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR, pp. 461-467, 2016.

[105] B. S. Gowrishankar and N. U. Bhajantri, "An exhaustive review of automatic music transcription techniques: Survey of music transcription techniques," in Proceedings of the 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), pp. 140-152, 2016.

[106] J. S. Downie, A. F. Ehmann, M. Bay, and M. C. Jones, "The music information retrieval evaluation exchange: Some observations and insights," in Advances in music information retrieval, pp. 93-115, 2010.

[107] Y. Han, J. Kim, K. Lee, Y. Han, J. Kim, and K. Lee, "Deep convolutional neural networks for predominant instrument recognition in polyphonic music," IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), vol. 25, no. 1, pp. 208-221, 2017.

[108] J. Pons, O. Slizovskaia, R. Gong, E. Gómez, and X. Serra, "Timbre analysis of music audio signals with convolutional neural networks," in Proceedings of the 2017 25th European Signal Processing Conference (EUSIPCO), pp. 2744-2748, 2017.

[109] C. Hernandez-Olivan and J. R. Beltran, "Timbre Classification of Musical Instruments with a Deep Learning Multi-Head Attention-Based Model," arXiv preprint arXiv:2107.06231, 2021.

[110] D. Bhalke, C. R. Rao, and D. S. Bormane, "Automatic musical instrument classification using fractional fourier transform based-MFCC features and counter propagation neural network," Journal of Intelligent Information Systems, vol. 46, no. 3, pp. 425-446, 2016.

[111] B. Toghiani-Rizi and M. Windmark, "Musical Instrument Recognition Using Their Distinctive Characteristics in Artificial Neural Networks," arXiv preprint arXiv:1705.04971, 2017.

[112] A. Banerjee, A. Ghosh, S. Palit, and M. Ballester, "A Novel Approach to String Instrument Recognition," in Proceedings of the 8th International Conference on Image and Signal Processing, pp. 165-175, 2018.

[113] O. Slizovskaia, E. Gómez Gutiérrez, and G. Haro Ortega, "Automatic musical instrument recognition in audiovisual recordings by combining image and audio classification strategies," in Proceedings of the 13th Sound and Music Computing (SMC) Conference, Hamburg, Germany, 2016.

[114] J. J. Bosch, J. Janer, F. Fuhrmann, and P. Herrera, "A Comparison of Sound Segregation Techniques for Predominant Instrument Recognition in Musical Audio Signals," in Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR, pp. 559-564, 2012.

[115] P. Li, J. Qian, and T. Wang, "Automatic instrument recognition in polyphonic music using convolutional neural networks," arXiv preprint arXiv:1511.05520, 2015.

[116] R. M. Bittner, J. Salamon, M. Tierney, M. Mauch, C. Cannam, and J. P. Bello, "Medleydb: A multitrack dataset for annotation-intensive mir research," in Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR, vol. 14, pp. 155-160, 2014.

[117] Y.-N. Hung and Y.-H. Yang, "Frame-level instrument recognition by timbre and pitch," arXiv preprint arXiv:1806.09587, 2018.

[118] S. Gururani, M. Sharma, and A. Lerch, "An Attention Mechanism for Musical Instrument Recognition," in Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR, Delft, Netherlands, 2019.

[119] E. Humphrey, S. Durand, and B. McFee, "OpenMIC-2018: An Open Data-set for Multiple Instrument Recognition," in Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR, pp. 438-444, 2018

[120] Y. N. Hung, Y. A. Chen, and Y. H. Yang, "Multitask learning for frame-level instrument recognition," in Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 381-385, 2019

[121] K. Racharla, V. Kumar, C. B. Jayant, A. Khairkar, and P. Harish, "Predominant musical instrument classification based on spectral features," in Proceedings of the 2020 7th International Conference on Signal Processing and Integrated Networks (SPIN), IEEE, pp. 617-622, 2020.

[122] A. Kratimenos, K. Avramidis, C. Garoufis, A. Zlatintsi, and P. Maragos, "Augmentation methods on monophonic audio for instrument classification in polyphonic music," in Proceedings of the 2020 28th European Signal Processing Conference (EUSIPCO), IEEE, pp. 156-160, 2021.

[123] H. Flores Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition," in Proceedings of the 22nd Int. Society for Music Information Retrieval Conference, 2021.

[124] H. H. Chen and A. Lerch, "Music Instrument Classification Reprogrammed," in International Conference on Multimedia Modeling, Springer, pp. 345-357, 2023.

[125] A. Lerch, C. Arthur, A. Pati, and S. Gururani, "An interdisciplinary review of music performance analysis," Transactions of the International Society for Music Information Retrieval, vol 3, no 1, pp. 221-245, 2020.

[126] H. Kim, P. Ramoneda, M. Miron, and X. Serra, "An overview of automatic piano performance assessment within the music education context," 2022.

[127] S. D. Peter et al., "Automatic Note-Level Score-to-Performance Alignments in the ASAP Dataset," Transactions of the International Society for Music Information Retrieval, vol 6, no 1, pp. 27-42, 2023.

[128] M. Ištvánek and Š. Miklánek, "Towards Automatic Measure-Wise Feature Extraction Pipeline for Music Performance Analysis," in Proceedings of the 2022 45th International Conference on Telecommunications and Signal Processing (TSP), IEEE, pp. 192-195, 2022.

[129] M. Ištvánek, Š. Miklánek, K. H. Mühlová, L. Spurný, and Z. Smékal, "Application of Computational Methods for Comparative Music Analysis," in Proceedings of the 2023 4th International Symposium on the Internet of Sounds, IEEE, pp. 1-6, 2023.

[130] J. Huang, Y. N. Hung, A. Pati, S. K. Gururani, and A. Lerch, "Score-informed networks for music performance assessment," in Proceedings of the 21st International Society for Music Information Retrieval Conference, ISMIR, 2020.

[131] H. Schreiber, F. Zalkow, and M. Müller, "Modeling and Estimating Local Tempo: A Case Study on Chopin's Mazurkas," in Proceedings of the 21st of Int. Society for Music Information Retrieval Conference, pp. 773-779, 2020.

[132] A. Vidwans, S. Gururani, C. W. Wu, V. Subramanian, R. V. Swaminathan, and A. Lerch, "Objective descriptors for the assessment of student music performances," in Proceedings of the Audio Engineering Society Conference: 2017 AES International Conference on Semantic Audio, Audio Engineering Society, 2017.

[133] K. A. Pati, S. Gururani, and A. Lerch, "Assessment of student music performances using deep neural networks," Applied Sciences, vol. 8, no. 4, p. 507, 2018.

[134] P. Parmar, J. Reddy, and B. Morris, "Piano skills assessment," in Proceedings of the 2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP), IEEE, pp. 1-5, 2021.

[135] N. Kato, E. Nakamura, K. Mine, O. Doeda, and M. Yamada, "Computational Analysis of Audio Recordings of Piano Performance for Automatic Evaluation," in Proceedings of the European Conference on Technology Enhanced Learning, Springer, pp. 586-592, 2023.

[136] H. Kim, M. Miron, and X. Serra, "Score-Informed MIDI Velocity Estimation for Piano Performance by FiLM Conditioning," in Proceedings of 2022 Sound and Music Computing (SMC) Conference, Stockholm, Sweden, June 2023.

[137] P. Seshadri and A. Lerch, "Improving music performance assessment with contrastive learning," in Proceedings of the 22nd International Society for Music Information Retrieval Conference, ISMIR, 2021.

[138] B. E. Köktürk-Güzel, O. Büyük, B. Bozkurt, and O. Baysal, "Automatic assessment of student rhythmic pattern imitation performances," Digital signal processing, vol. 133, p. 103880, 2023.

[139] C. Wang, "Scattering Transform for Playing Technique Recognition," Ph. D. dissertation, Queen Mary University of London, 2021.

[140] C. Wang, V. Lostanlen, and M. Lagrange, "Explainable Audio Classification of Playing Techniques with Layer-wise Relevance Propagation," in Proceedings of the 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, pp. 1-5, 2023.

[141] S. D. Peter, C. E. Cancino-Chacón, E. Karystinaios, and G. Widmer, "Sounding Out Reconstruction Error-Based Evaluation of Generative Models of Expressive Performance," in Proceedings of the 10th International Conference on Digital Libraries for Musicology, pp. 58-66, 2023.

[142] H. Zhang, J. Tang, S. Rafee, S. Dixon, G. Fazekas, and G. Wiggins, "ATEPP: A dataset of automatically transcribed expressive piano performance," in Proceedings of the 24th International Society for Music Information Retrieval Conference, ISMIR, India, 2023.

[143] S. Rafee, G. Fazekas, and G. Wiggins, "HIPI: A Hierarchical Performer Identification Model Based on Symbolic Representation of Music," in Proceedings of the 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, pp. 1-5, 2023.

[144] S. Rhyu, S. Kim, and K. Lee, "Sketching the Expression: Flexible Rendering of Expressive Piano Performance with Self-Supervised Learning," in Proceedings of the 23rd International Society for Music Information Retrieval Conference, ISMIR, India, 2022.

[145] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in Proceedings of the ICML deep learning workshop, vol. 2, no. 1, 2015.

[146] S. Jadon, "An overview of deep learning architectures in few-shot learning domain," arXiv preprint arXiv:2008.06365, 2020.

[147] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," Neural computation, vol. 29, no. 9, pp. 2352-2449, 2017.

[148] D. Bendor and X. Wang, "The neuronal representation of pitch in primate auditory cortex," Nature, vol. 436, no. 7054, pp. 1161-1165, 2005.

[149] J. Han and C. Moraga, "The influence of the sigmoid function parameters on the speed of backpropagation learning," in International Workshop on Artificial Neural Networks, Springer, pp. 195-201, 1995.

[150] L. M. Smith, "A multiresolution time-frequency analysis and interpretation of musical rhythm," Ph. D. dissertation, University of Western Australia, Perth, Australia, 2000.

[151] C. d'Alessandro and M. Castellengo, "The pitch of short - duration vibrato tones," The Journal of the Acoustical Society of America, vol. 95, no. 3, pp. 1617-1630, 1994.

[152] X. Li, K. Wang, J. Soraghan, and J. Ren, "Fusion of Hilbert-Huang Transform and Deep Convolutional Neural Network for Predominant Musical Instruments Recognition," in Proceedings of the 9th International Conference on Computational Intelligence in Music, Sound, Art and Design (Part of EvoStar), pp. 80-89, Seville, Spain, 2020.

[153] L. E. Kinsler, A. R. Frey, A. B. Coppens, and J. V. Sanders, Fundamentals of acoustics. John wiley & sons, 2000.

[154] V. Emiya, N. Bertin, B. David, and R. Badeau, "MAPS-A piano database for multipitch estimation and automatic transcription of music," 2010.

[155] J. Fritsch and M. D. Plumbley, "Score informed audio source separation using constrained nonnegative matrix factorization and score synthesis," in Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, pp. 888-891, 2013.

[156] N. E. Huang, "Hilbert-Huang transform and its applications," World Scientific, 2014.

[157] U. B. de Souza, J. P. L. Escola, and L. da Cunha Brito, "A survey on Hilbert-Huang transform: Evolution, challenges and solutions," Digital Signal Processing, vol. 120, p. 103292, 2022.

[158] A. Ayenu-Prah and N. Attoh-Okine, "Comparative study of Hilbert–Huang transform, Fourier transform and wavelet transform in pavement profile analysis," Vehicle System Dynamics, vol. 47, no. 4, pp. 437-456, 2009.

[159] Y. Yan et al., "Cognitive fusion of thermal and visible imagery for effective detection and tracking of pedestrians in videos," Cognitive Computation, vol. 10, no. 1, pp. 94-104, 2018.

[160] G. Peeters, "A large set of audio features for sound description (similarity and classification)," CUIDADO project Ircam technical report, 2004.

[161] P. Ramoneda, D. Jeong, E. Nakamura, X. Serra, and M. Miron, "Automatic Piano Fingering from Partially Annotated Scores using Autoregressive Neural Networks," in Proceedings of the 30th ACM International Conference on Multimedia, pp. 6502-6510, 2022.

[162] A. Schmitt, "Preparatory exercises: for the piano," G. Schirmer, 1922.

[163] J. Pan et al., "An Audio Based Piano Performance Evaluation Method Using Deep Neural Network Based Acoustic Modeling," in Proceedings of the 18th Annual Conference of International Speech Communication Association (INTERSPEECH), pp. 3088-3092, 2017.

[164] X. Li, Y. Yan, J. Soraghan, Z. Wang, and J. Ren, "A music cognition–guided framework for multi-pitch estimation," Cognitive computation, vol. 15, no. 1, pp. 23-35, 2023

[165] MMClassification Contributors, "Openmmlab’s image classification toolbox and benchmark," Available: https://github.com/open-mmlab/mmclassification, 2020.

[166] C. L. Hanon, "The virtuoso pianist in sixty exercises for the piano: for the acquirement of agility, independence, strength and perfect evenness in the fingers as well as suppleness of the wrist," vol. 925 G. Schirmer, 1928.

[167] H.W. Dong, K. Chen, J. McAuley, and T. Berg-Kirkpatrick, "MusPy: A toolkit for symbolic music generation," in Proceedings of the 21st International Society for Music Information Retrieval Conference, ISMIR, 2020.

[168] X. Li, S. Weiss, et al. “S-ResNN: siamese residual neural network for musical shape evaluation in piano performance assessment.” in 31st European Signal Processing Conference, EUSIPCO 2023 (pp. 216-220). Aalto, Finland, September, 2023.

[169] S. Li, R. Nair, M. Naqvi, "Acoustic and Text Features Analysis for Adult ADHD Screening: A Data-Driven Approach Utilizing DIVA Interview," IEEE Journal of Translational Engineering in Health and Medicine, vol. 12, pp. 359-370, 2024.

[170] E. Branikas, P. Murray, G. West, "A Novel Data Augmentation Method for Improved Visual Crack Detection Using Generative Adversarial Networks," IEEE Access, vol. 11, pp. 22051 - 22059, 2023.

[171] L. Xie, Y. Yang, Z. Fu, and M. Naqvi, "One-shot medical action recognition with a cross-attention mechanism and dynamic time warping," in Proceedings of 2023 IEEE International Conference on Acoustics, Speech and Signal Processing, Rhodes Island, Greece, June, 2023.

Appendix A: Frequencies in Hz of musical pitches.

Here is an index table giving the frequencies in Hz of music standard pitches, covering the full range of all normal musical instruments.

Pitch number	Piano-roll	Letter Name	Frequency (Hz)
21	1	A0	27.50
22	2	A#0	29.14
23	3	B0	30.87
24	4	C1	32.70
25	5	C#1	34.65
26	6	D1	36.71
27	7	D#1	38.89
28	8	E1	41.20
29	9	F1	43.65
30	10	F#1	46.25
31	11	G1	49.00
32	12	G#1	51.91
33	13	A1	55.00
34	14	A#1	58.27
35	15	B1	61.74
36	16	C2	65.41
37	17	C#2	69.30
38	18	D2	73.42
39	19	D#2	77.78
40	20	E2	82.41
41	21	F2	87.31
42	22	F#2	92.50
43	23	G2	98.00
44	24	G#2	103.83
45	25	A2	110.00
46	26	A#2	116.54
47	27	B2	123.47
48	28	C3	130.81
49	29	C#3	138.59
50	30	D3	146.83
51	31	D#3	155.56
52	32	E3	164.81
53	33	F3	174.61
54	34	F#3	185.00
55	35	G3	196.00
56	36	G#3	207.65
57	37	A3	220.00
58	38	A#3	233.08
59	39	B3	246.94
60	40	C4	261.63
61	41	C#4	277.18
62	42	D4	293.66
63	43	D#4	311.13
64	44	E4	329.63
65	45	F4	349.23
66	46	F#4	369.99
67	47	G4	392.00
68	48	G#4	415.30
69	49	A4	440.00
70	50	A#4	466.16
71	51	B4	493.88
72	52	C5	523.25
73	53	C#5	554.37
74	54	D5	587.33
75	55	D#5	622.25
76	56	E5	659.26
77	57	F5	698.46
78	58	F#5	739.99
79	59	G5	783.99
80	60	G#5	830.61
81	61	A5	880.00
82	62	A#5	932.33
83	63	B5	987.77
84	64	C6	1046.50
85	65	C#6	1108.73
86	66	D6	1174.66
87	67	D#6	1244.51
88	68	E6	1308.51
89	69	F6	1396.91
90	70	F#6	1479.98
91	71	G6	1567.98
92	72	G#6	1661.22
93	73	A6	1760.00
94	74	A#6	1864.66
95	75	B6	1975.53
96	76	C7	2093.00
97	77	C#7	2217.46
98	78	D7	2349.32
99	79	D#7	2489.02
100	80	E7	2637.02
101	81	F7	2793.83
102	82	F#7	2959.96
103	83	G7	3135.96
104	84	G#7	3322.44
105	85	A7	3520.00
106	86	A#7	3729.31
107	87	B7	3951.07
108	88	C8	4186.01

Appendix B: Schmitt OP. 16 data list

The Musical Shape Evaluation Dataset (MSED) comprises 147 music pieces carefully selected from Aloys Schmitt's "Preparatory Exercises for Piano, Op. 16" [162]. This well-established educational book is designed to help pianists develop finger independence and evenness, making it an ideal source for studying musical shape in the context of piano performance.

The selected pieces encompass a diverse range of musical forms and techniques, ensuring a comprehensive exploration of musical shape across various contexts. The breakdown of the 147 pieces is as follows:

- Polyphony: 83 pieces
- Scales: 20 pieces
- Arpeggios: 12 pieces
- Staccato: 32 pieces

The first column, "MSED No.", indicates the numbering of the music pieces within the MSED dataset. The second column, "Schmitt Op. 16 No.", shows the corresponding numbering of the polyphonic pieces in the original book. However, it is important to note that the scales and arpeggios pieces do not have specific numbers in the original book.

The third column, "Characteristics", highlights the key features of each group of music pieces. The polyphonic pieces (1-83) feature multiple simultaneous melodic lines, offering rich opportunities for studying the shaping of harmonies and textures. The scales (84-103) and arpeggios (104-115) present unique challenges for shaping melodic contours and dynamic gradations. Finally, the staccato pieces (116-147) were created by performing the scales and arpeggios in a short, detached manner, requiring precise control and expressive timing to effectively convey musical shape.

By including a diverse selection of music pieces with different technical and expressive demands, the MSED provides a comprehensive resource for investigating musical shape in piano performance.

MSED No.	Schmitt Op. 16 No.	Characteristics	
1	1	2 stacked notes	Polyphony
2	2	2 stacked notes	
3	3	2 stacked notes	
4	4	2 stacked notes	
5	5	2 stacked notes	
6	6	2 stacked notes	
7	7	2 stacked notes	
8	8	2 stacked notes	
9	9	2 stacked notes	
10	10	2 stacked notes	
11	11	2 stacked notes	
12	12	2 stacked notes	
13	13	2 stacked notes	
14	14	2 stacked notes	
15	15	2 stacked notes	
16	16	2 stacked notes	
17	17	2 stacked notes	
18	18	2 stacked notes	
19	19	2 stacked notes	
20	20	2 stacked notes	
21	21	2 stacked notes	
22	22	2 stacked notes	
23	23a	2 stacked notes	
24	23b	2 stacked notes	
25	24a	2 stacked notes	
26	24b	2 stacked notes	
27	25a	2 stacked notes	
28	25b	2 stacked notes	
29	26	2 stacked notes	
30	27	2 stacked notes	
31	28	2 stacked notes	
32	29	2 stacked notes	
33	30	2 stacked notes	
34	31	2 stacked notes	
35	32	2 stacked notes	
36	33	2 stacked notes	
37	34	2 stacked notes, 2 hold notes	

38	35	2 stacked notes, 2 hold notes	
39	36	2 stacked notes, 2 hold notes	
40	37	2 stacked notes, 2 hold notes	
41	38	2 stacked notes, 2 hold notes	
42	39	2 stacked notes, 2 hold notes	
43	40	2 stacked notes, 2 hold notes	
44	41	2 stacked notes, 2 hold notes	
45	42	2 stacked notes, 2 hold notes	
46	43	2 stacked notes, 2 hold notes	
47	44	2 stacked notes, 2 hold notes	
48	45	2 stacked notes, 2 hold notes	
49	46	2 stacked notes, 2 hold notes	
50	47	2 stacked notes, 2 hold notes	
51	48	2 stacked notes, 2 hold notes	
52	49	2 stacked notes, 2 hold notes	
53	50	2 stacked notes, 2 hold notes	
54	51	2 stacked notes, 2 hold notes	
55	52	2 stacked notes, 2 hold notes	
56	53	2 stacked notes, 2 hold notes	
57	54	2 stacked notes, 2 hold notes	
58	55	2 stacked notes, 2 hold notes	
59	56	2 stacked notes, 2 hold notes	
60	57	2 stacked notes, 2 hold notes	
61	58	2 stacked notes, 2 hold notes	
62	59	2 stacked notes, 2 hold notes	
63	60	2 stacked notes, 2 hold notes	
64	61	2 stacked notes, 2 hold notes	
65	62	2 stacked notes, 2 hold notes	
66	63	2 stacked notes, 2 hold notes	
67	64	2 stacked notes, 2 hold notes	
68	65	2 stacked notes, 4 hold notes	
69	66	2 stacked notes, 4 hold notes	
70	67	2 stacked notes, 4 hold notes	
71	68	2 stacked notes, 4 hold notes	
72	119	4 stacked notes	
73	120	4 stacked notes	
74	121	4 stacked notes	
75	122	4 stacked notes	
76	123	4 stacked notes	

77	124	4 stacked notes	
78	125	4 stacked notes	
79	126	4 stacked notes	
80	127	4 stacked notes	
81	128	4 stacked notes, 2 hold notes	
82	129	4 stacked notes, 2 hold notes	
83	127	4 stacked notes, 2 hold notes	
84	N/A	Major ascending	Scales
85	N/A	Major descending	
86	N/A	Harmonic minor ascending	
87	N/A	Harmonic minor descending	
88	N/A	Melodic minor ascending	
89	N/A	Melodic minor descending	
90	N/A	Contrary motion divergence	
91	N/A	Contrary motion close	
92	N/A	Minor contrary motion divergence	
93	N/A	Minor contrary motion close	
94	N/A	Chromatic ascending	
95	N/A	Chromatic descending	
96	N/A	Major a third apart ascending	
97	N/A	Major a third apart descending	
98	N/A	Harmonic minor a third apart ascending	
99	N/A	Harmonic minor a third apart descending	
100	N/A	Major a sixth apart ascending	
101	N/A	Major a sixth apart descending	
102	N/A	Harmonic minor a sixth apart ascending	
103	N/A	Harmonic minor a sixth apart descending	
104	N/A	Major triads	Arpeggios
105	N/A	Minor triads	
106	N/A	Dominant 7th	
107	N/A	Diminished 7th	
108	N/A	Chord passages major triads ascending	
109	N/A	Chord passages major triads descending	
110	N/A	Chord passages minor triads ascending	
111	N/A	Chord passages minor triads descending	

112	N/A	Chord passages chord of the diminished 7th ascending	
113	N/A	Chord passages chord of the diminished 7th descending	
114	N/A	Chord passages chord of the dominant 7th ascending	
115	N/A	Chord passages chord of the dominant 7th descending	
116	N/A	Scales No.84	Staccato
117	N/A	Scales No.85	
118	N/A	Scales No.86	
119	N/A	Scales No.87	
120	N/A	Scales No.88	
121	N/A	Scales No.89	
122	N/A	Scales No.90	
123	N/A	Scales No.91	
124	N/A	Scales No.92	
125	N/A	Scales No.93	
126	N/A	Scales No.94	
127	N/A	Scales No.95	
128	N/A	Scales No.96	
129	N/A	Scales No.97	
130	N/A	Scales No.98	
131	N/A	Scales No.99	
132	N/A	Scales No.100	
133	N/A	Scales No.101	
134	N/A	Scales No.102	
135	N/A	Scales No.103	
136	N/A	Arpeggios No.104	
137	N/A	Arpeggios No.105	
138	N/A	Arpeggios No.106	
139	N/A	Arpeggios No.107	
140	N/A	Arpeggios No.108	
141	N/A	Arpeggios No.109	
142	N/A	Arpeggios No.110	
143	N/A	Arpeggios No.111	
144	N/A	Arpeggios No.112	
145	N/A	Arpeggios No.113	
146	N/A	Arpeggios No.114	
147	N/A	Arpeggios No.115	

Appendix C: Dataset description

Class	Folder name	Shape	Description
1	1	Forte	Strong dynamics and denoted as f on the score
2	2	Piano	Weak dynamics and denoted as p on the score
3	3	Cresc.	Gradually increase the dynamics from p to f
4	4	Decresc.	Gradually reduce the dynamics from f to p
5	5	Adagio	Perform the score with 72 bpm
6	6	Largo	Perform the score with 50 bpm
7	7	Rit.	Gradually reduce the speed from 60 - 50 bpm
8	8	Accel.	Gradually increase the speed from 60 - 72 bpm
9	5_1	Forte+adagio	f + 72 bpm
10	5_2	Piano+adagio	p + 72 bpm
11	5_3	Cresc.+adagio	p→f + 72 bpm
12	5_4	Decresc.+adagio	f→p + 72 bpm
13	6_1	Forte+largo	f + 50 bpm
14	6_2	Piano+largo	p + 50 bpm
15	6_3	Cresc.+largo	p→f + 50 bpm
16	6_4	Decresc.+largo	f→p + 50 bpm
17	7_1	Forte+rit.	f + 60→50 bpm
18	7_2	Piano+rit.	p + 60→50 bpm
19	7_3	Cresc.+rit.	p→f + 60→50 bpm
20	7_4	Decresc.+rit.	f→p + 60→50 bpm
21	8_1	Forte+accel.	f + 60→72 bpm
22	8_2	Piano+ accel.	p + 60→72 bpm
23	8_3	Cresc.+ accel.	p→f + 60→72 bpm
24	8_4	Decresc.+ accel.	f→p + 60→72 bpm
25	0_1	Given	Play ahead
26	0_2	Take	delay play
27	27	Jazz	syncopation
28	28	Normal	60 bpm

Appendix D: Dataset visualization

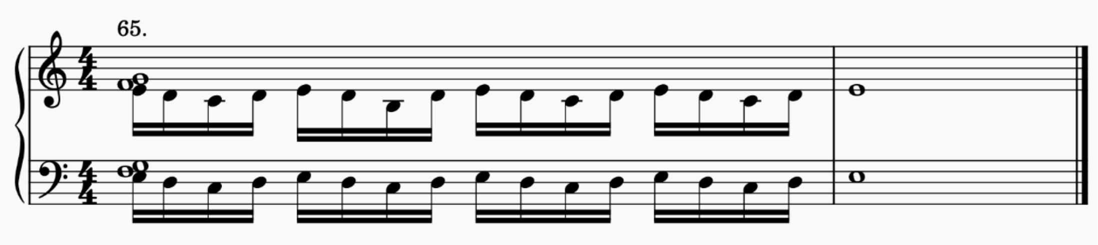


Fig. 0-1. No.65 music piece in Schmitt OP.16.

To better illustrate the differences between normal and the 27 musical shapes (MSs) for the music pieces, let's examine the Constant-Q Transform (CQT) spectrograms of a representative music piece selected from Schmitt OP.16.

Fig. 0-1 shows the musical score of the selected piece, while Fig. 0-2 and Fig. 0-3 display the CQT spectrograms for the normal performance and the 27 MSs. (Please note that the description of the 27 MSs can be found in Appendix C.)

Upon examining the CQT spectrograms, several key observations can be made regarding the impact of tempo and dynamics on the visual representation of musical shape:

Tempo variations:

- When the tempo becomes faster (e.g., Allegro), the length of the bars in the spectrogram appears shorter compared to the normal performance. This is because the music is played at a quicker pace, resulting in less time spent on each note or phrase.
- Conversely, when the tempo becomes slower (e.g., Adagio), the length of the bars in the spectrogram appears longer. The slower tempo allows for more time to be devoted to each musical element, resulting in an extended visual representation.

Dynamic variations:

- When the dynamics are softer (e.g., Piano), the intensity of the bars in the spectrogram appears lighter or less pronounced. This visual change reflects the decreased volume and energy in the musical performance.
- On the other hand, when the dynamics are louder (e.g., Forte), the intensity of the bars in the spectrogram appears darker or more prominent. The increased brightness in the visual representation corresponds to the heightened volume and energy in the performance.

Combined tempo and dynamic variations:

- The CQT spectrograms also showcase the interplay between tempo and dynamics in shaping the visual representation of musical shape. For example, a piece performed in a fast tempo with soft dynamics (e.g., Allegro Piano) will exhibit shorter bar lengths and a lighter intensity compared to the normal performance.
- Similarly, a piece performed in a slow tempo with loud dynamics (e.g., Adagio Forte) will display longer bar lengths and a darker intensity, reflecting the combined effect of the tempo and dynamic changes.

These visual variations in the CQT spectrograms demonstrate how different musical shapes are characterized by distinct combinations of tempo, dynamics, and other expressive elements. By examining the length and intensity of the bars in the spectrograms, insights can be gained into the ways in which performers manipulate these musical parameters to convey different expressive intentions and create diverse musical shapes.

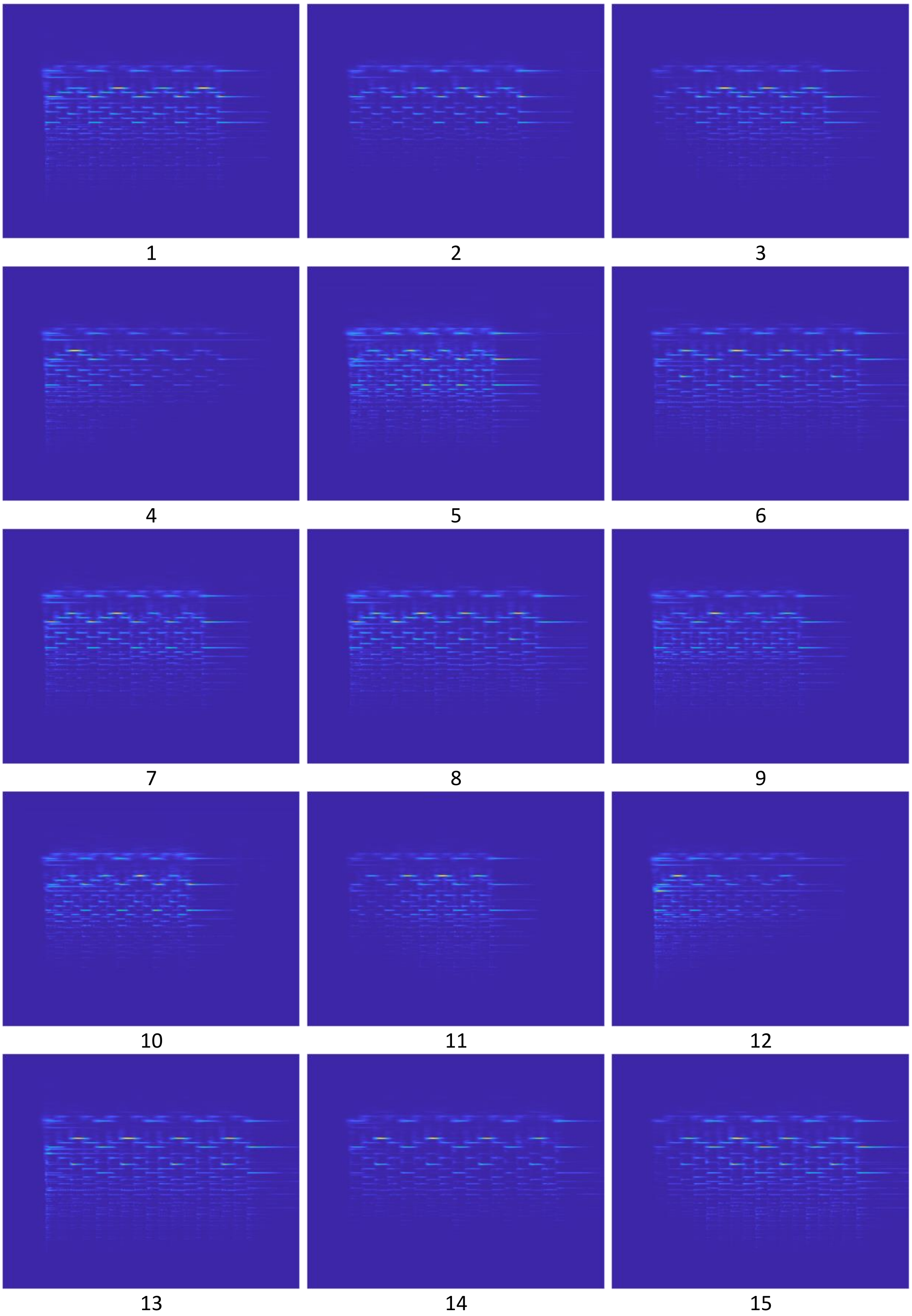


Fig. 0-2. CQT spectrogram of MSs (1-15), where x-axis and y-axis represent time and number of bins, respectively. Detailed description of each MS is given in Appendix C

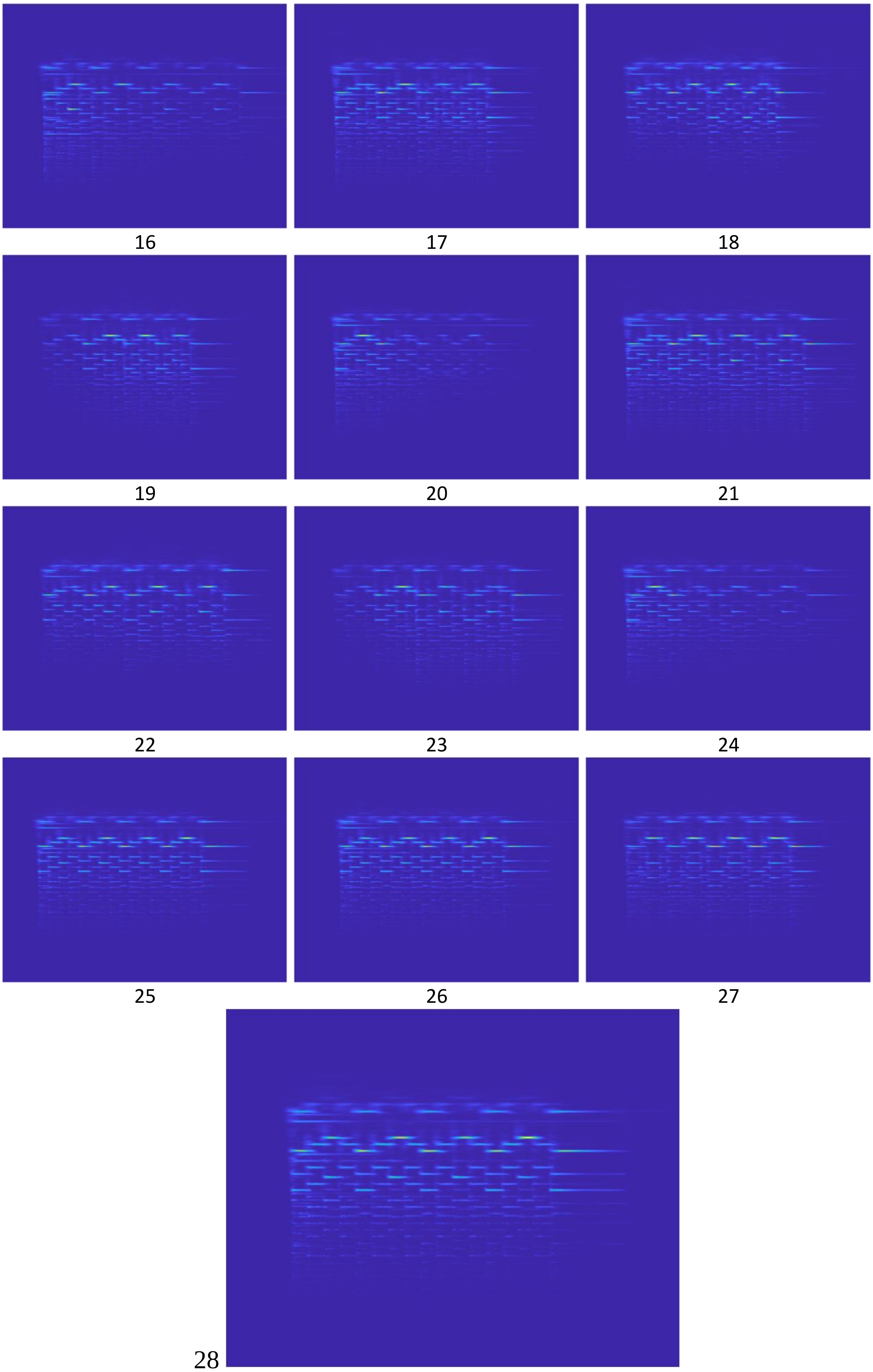


Fig. 0-3. CQT spectrogram of MSs (16-28), where x-axis and y-axis represent time and number of bins, respectively. Detailed description of each MS is given in Appendix C