THE DEVELOPMENT OF AN AUTOMATIC 3D ANIMATION BUILDER FOR DISPLAYING UKULELE PLAYING

Thawatphong Phithak

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Information Science in Information Technology

Suranaree University of Technology

Academic Year 2015

การพัฒนาระบบสร้างภาพเคลื่อนไหว 3 มิติแบบอัตโนมัติ
สำหรับแสดงการเล่นอูคูเลเล่

นายธวัชพงษ์ พิทักษ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาการสารสนเทศดุษฎีบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

มหาวิทยาลัยเทคโนโลยีสุรนารี

ปีการศึกษา 2558

THE DEVELOPMENT OF AN AUTOMATIC 3D ANIMATION BUILDER FOR DISPLAYING UKULELE PLAYING

Suranaree University of Technology has approved this thesis submitted in partial fulfillment of the requirements for the Degree of Doctor of Information Science in Information Technology.

Thesis Examining Committee

____
(Assoc. Prof. Dr. Weerapong Polnigongit)
Chairperson

____
(Asst. Prof. Dr. Jitimon Angskun)
Member (Thesis Advisor)

____
(Assoc. Prof. Dr. Punpiti Piamsa-nga)
Member

____
(Assoc. Prof. Dr. Kittisak Kerdprasop)
Member

____
(Asst. Prof. Dr. Suphakit Niwattanakul)
Member

____
(Asst. Prof. Dr. Thara Angskun)
Member

____
(Prof. Dr. Sukit Limpijumnong)
Vice Rector for Academic Affairs
and Innovation

____
(Dr. Peerasak Siriyothin)
Dean of Institute of Social Technology

ธวัชพงษ์ พิทักษ์ : การพัฒนาระบบสร้างภาพเคลื่อนไหว 3 มิติแบบอัตโนมัติ สำหรับแสดงการเล่นอูคูเลเล่ (THE DEVELOPMENT OF AN AUTOMATIC 3D ANIMATION BUILDER FOR DISPLAYING UKULELE PLAYING) อาจารย์ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร.จิติมนต์ อั่งสกุล, 192 หน้า.

ปัจจุบัน อูคูเลเล่เป็นที่นิยมอย่างมาก และถูกเลือกให้เป็นเครื่องดนตรีสายชนิดแรกสำหรับการฝึกเล่นดนตรีของผู้เริ่มต้น อย่างไรก็ตาม ในการฝึกเล่นอูคูเลเล่ด้วยตนเองนั้น ผู้เริ่มต้นจำเป็นต้องค้นหาข้อมูลพื้นฐานจำนวนมาก เช่น การจับถือ รูปแบบการดีด และการจับคอร์ด เป็นต้น จากการสำรวจระบบเกี่ยวกับการฝึกเล่นเครื่องดนตรีสายที่มีอยู่ในปัจจุบัน พบว่า ระบบที่มีอยู่ขาดการให้ข้อมูลที่ครอบคลุมสำหรับการฝึกหัด นอกจากนี้ ระบบส่วนใหญ่ไม่อนุญาตให้ผู้ใช้นำเข้าเพลงที่ต้องการฝึกเล่น และหลายระบบแสดงภาพเคลื่อนไหวในรูปแบบ 2 มิติ ซึ่งไม่ยืดหยุ่นต่อการเรียนรู้

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาระบบสร้างภาพเคลื่อนไหว 3 มิติแบบอัตโนมัติ สำหรับแสดงการเล่นอูคูเลเล่ โดยกระบวนการแรกของการพัฒนาคือ การรู้จำสารสนเทศที่เกี่ยวข้องกับการเล่นอูคูเลเล่ ซึ่งประกอบด้วย 2 ส่วน ได้แก่ ส่วนที่ 1) การรู้จำรูปแบบการดีด เป็นวิธีการวิเคราะห์รูปแบบการดีดที่เหมาะสมของเพลงที่ค้นหา โดยขั้นตอนหลักของการรู้จำรูปแบบการดีด คือการทำนายชนิดของการดีด และการสรุปรูปแบบการดีด ผลการทดลองแสดงให้เห็นว่า ความถูกต้องเฉลี่ยของการทำนายชนิดของการดีดในเพลงทดสอบทั้งหมด 10 เพลง ได้ค่าเฉลี่ยอัตราการรู้จำเท่ากับร้อยละ 89.98 แต่อย่างไรก็ตาม เพลงทดสอบทั้งหมดได้รับการสรุปรูปแบบการดีดที่ถูกต้องทุกเพลง ส่วนที่ 2) การรู้จำเวลาในการเปลี่ยนคอร์ด เป็นวิธีในการคำนวณหาค่าเวลาที่เหมาะสม เพื่อความราบรื่นและสมจริงของการสร้างภาพเคลื่อนไหวของการเปลี่ยนคอร์ด ผลการประมาณเวลาในการเปลี่ยนคอร์ด คือ 0.28 วินาทีโดยเฉลี่ย หรือเท่ากับ 18 เฟรมของภาพเคลื่อนไหว 3 มิติในอัตรา 60 เฟรมต่อวินาที

กระบวนการที่ 2 ของการพัฒนาระบบคือ การสร้างภาพเคลื่อนไหว 3 มิติสำหรับการเล่นอูคูเลเล่ ซึ่งประกอบด้วย 3 ส่วนย่อย ได้แก่ 1) การสร้างภาพเคลื่อนไหวการดีด 2) การสร้างภาพเคลื่อนไหวการจับคอร์ด และ 3) การสร้างภาพเคลื่อนไหวการเปลี่ยนคอร์ด

ผลลัพธ์ที่ได้จากกระบวนการทั้งหมดถูกผสานเป็นระบบใหม่ที่มีชื่อว่า อุ๊คมีทรีดี ซึ่งเป็นระบบที่อนุญาตให้ผู้ใช้นำเข้าเพลงอูคูเลเล่ตระกูลเอ็มพีสาม (MP3) เพื่อการรู้จำเสียงดนตรีและสร้างเป็นภาพเคลื่อนไหว 3 มิติ โดยผู้ใช้สามารถเรียนรู้วิธีการเล่นอูคูเลเล่ พร้อมทั้งควบคุมมุมมองได้จากทุกมุม

ระบบอุ๊คมีทรีดีถูกประเมินโดยผู้ใช้งาน 3 กลุ่ม ได้แก่ 1) ผู้เชี่ยวชาญด้านการเล่นอูคูเลเล่ 2) ผู้เชี่ยวชาญด้านการออกแบบและพัฒนาระบบ และ 3) บุคคลทั่วไปผู้สนใจในการเล่นอูคูเลเล่ การทดสอบทำโดยการใช้แบบสอบถามหลังการใช้งานระบบ ซึ่งผลการทดสอบแสดงให้เห็นว่า ความสามารถในการใช้งานได้ของระบบโดยรวมอยู่ในเกณฑ์ดี ($\bar{X}$ = 2.77 จาก 3) อย่างไรก็ตาม พบความแตกต่างระหว่างผู้ใช้กลุ่มที่ 1 และ 2 อย่างมีนัยสำคัญทางสถิติ ที่ระดับ 0.05 ในแง่ของความสามารถในการใช้งานโดยรวม และด้านการให้ความช่วยเหลือ นอกจากนี้ การทดสอบความสามารถในการใช้งานได้ ยังนำวิธีการคิดออกเสียงมาใช้ ซึ่งทำให้ได้รับความคิดเห็นจากผู้ใช้มากกว่า 60 ความคิดเห็น โดยความคิดเห็นส่วนใหญ่เกี่ยวข้องกับความเป็นประโยชน์ของระบบอุ๊คมีทรีดี รวมทั้งข้อจำกัดบางประการเพื่อการปรับปรุงระบบ

สาขาวิชาเทคโนโลยีสารสนเทศ ลายมือชื่อนักศึกษา ธีรัชพงษ์ พิทักษ์

ปีการศึกษา 2558 ลายมือชื่ออาจารย์ที่ปรึกษา

ลายมือชื่ออาจารย์ที่ปรึกษาร่วม

THAWATPHONG PHITHAK : THE DEVELOPMENT OF AN AUTOMATIC 3D ANIMATION BUILDER FOR DISPLAYING UKULELE PLAYING. THESIS ADVISOR : ASST. PROF. JITIMON ANGSKUN, D.ENG., 192 PP.

MUSIC RECOGNITION/3D ANIMATION/UKULELE

Nowadays, the ukulele is very popular and it is selected by general beginners to learn the first string instrument in music education. However, for self-studying the ukulele, the beginners are required to know about a lot of basic information, such as holding, strumming pattern, and touching chords, etc. A survey in the existing string instrument playing systems shows that information for practicing ukulele is lacking, Moreover, most systems do not allow users to import the desired song files and many systems display an animation in 2D format, which is not flexible for learning.

This research aims to develop an automatic 3D animation builder for displaying ukulele playing. The first process of the development is recognizing information related to the ukulele playing information which consists of two sub-stages. The Stage 1 is strumming pattern recognition which is the method for analyzing a proper strumming pattern of each query song. The main step of strumming pattern recognition is strumming type prediction and strumming pattern summarization. The experimental results reveal that the average correctness of strumming type prediction of ten test songs is 89.98 per cent of F-measure; however, all of test songs are summarized with the correct strumming patterns. The Stage 2 is chord changing time recognition which is operated to calculate the appropriate time to create a smooth and realistic chord changing animation. The result of estimating the chord changing time is 0.28 seconds on average, i.e. 18 frames of 3D animation based on 60 frames per second.

The second process of the development is constructing 3D ukulele playing animation, which consists of three sub-stages: 1) Building a strumming animation, 2) Building a chord touching animation, and 3) Building a chord changing animation.

The output of all the processes is integrated into the new system named UkeMe3D, which allows users to import MP3 ukulele songs for recognizing ukulele playing information and building 3D ukulele playing animation. The users can learn how to play the ukulele and control the viewpoint from any angle.

The UkeMe3D system is tested by three user groups consisting of 1) the experts of ukulele playing, 2) the experts of system design and development, and 3) the people who are interested in ukulele playing. The system usability testing is administered by the post-study questionnaire. The testing results reveal that the overall system usability is in a good level ($\bar{X}$ = 2.77 from 3). However, there is a significant difference between the sample groups 1 and 2 at the significance level of 0.05 in terms of overall system usability and the criterion of helpfulness. Moreover, the usability testing is performed by the thinking-aloud protocol, which receives more than 60 difference opinions. Most opinions are about the advantages of UkeMe3D system and some limitations for improving the system.

School of Information Technology | Student's Signature Tharatphong Phithak

Academic Year 2015 | Advisor's Signature ______

Co-advisor's Signature ______

ACKNOWLEDGEMENTS

I would like to thank all of people who have contributed to the completion of this dissertation. This dissertation would not have been possible without their assistance, advice, encouragement, inspiration, and support, both direct and indirect contribution.

First, I would like to acknowledge my heartfelt gratitude to my supervisors, Assist. Prof. Dr. Jitimon Angskun and Assist. Prof. Dr. Thara Angskun, for their supervision, excellent guidance on my research, their continuous encouragement, and immense knowledge. It was a great time to learn by doing research with them.

I would like to express my gratitude to Assoc. Prof. Dr. Weerapong Polnigongit, the chairman of my dissertation committee, for his valuable suggestion and assistance. My grateful thanks are also extended to the committees for my dissertation who are Assoc. Prof. Dr. Punpiti Piamsa-nga for his insightful comments, Assoc. Prof. Dr. Kittisak Kerdprasop for giving me guidance and useful advices, and Assist. Prof. Dr. Suphakit Niwattanakul for giving me encouragement and inspiration in all aspects of this endeavor.

My special thanks are extended to Assist. Prof. Dr. Nuenghathai Khoponklang and Dr. Sorachai Kamollimsakul, for their suggestion and their continuous encouragement.

My grateful thanks go to Suranaree University of Technology that provides me with a scholarship for studying a doctoral degree. I would also like to express my gratitude to all of my lecturers at the Institute of Social Technology, School of Information Technology for their kindness in teaching me throughout my study.

I would also like to thank friends in the School of Information Technology, for their friendships and support, especially Supachanun Wanapu and Narodom Kittidachanupap.

Finally, my special thanks go to my beloved family, my parents, older sister, older brother, and my friends. Without their endless love, encouragement, and support, none of this would have been possible.

Thawatphong Phithak

TABLE OF CONTENTS

Page

TABLE OF CONTENTS (Continued)

Page

TABLE OF CONTENTS (Continued)

Page

TABLE OF CONTENTS (Continued)

Page

LIST OF TABLES

Table **Page**

LIST OF TABLES (Continued)

Table **Page**

LIST OF TABLES (Continued)

Table **Page**

LIST OF TABLES (Continued)

Table **Page**

LIST OF FIGURES

Figure **Page**

LIST OF FIGURES (Continued)

Figure **Page**

LIST OF FIGURES (Continued)

Figure **Page**

LIST OF FIGURES (Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Introduction

Music is a universal phenomenon, which is studied, performed and enjoyed by a wide and diverse audiences (Liem, Müller, Eck, Tzanetakis, and Hanjalic, 2011). Music is an important role in people's diary life and it plays a role in various activities such as, social and cultural activities, hobbies, sports, parties, and celebrations, etc. In this digital age, people have a huge collection of music for accessing because the popularity of the Internet and the use of high quality audio format, such as MP3 have expedited the growth of digital music libraries (Yang and Chen, 2012). Listeners can retrieve music anywhere and anytime from their devices.

The musical instrument is an invention to make musical sounds. When we play on many musical instruments in the form of rhythm, harmonic will make a melodious sound. The musical instruments are divided into five types; String Instruments, Woodwind Instruments, Brass Instruments, Keyboard Instruments, and Percussion Instruments. For string instruments, the most popular instrument that people play today is the guitar because it can be very harmonious, and in some instances, it can also be comfortable for self-practicing. In addition, other string instruments are similar to guitar and it is populated by originating from many cultures, such as Chinese Ruan moon guitar, Irish Bouzouki, and Hawaii ukulele.

The Hawaii ukulele or ukulele looks like a small guitar, but it has only four strings. The general material which is used to produce the ukulele is the Hawaiian koa tree because it makes a density and sonority sound. The ukulele is a plucked stringed instrument of the Hawaiian people and it is an icon of their culture (Yasui, 2008).

In 2012, the ukulele was interested and very popular in musical instrument lovers, especially in Asia people because it is melodious sound, easy to carry, and inexpensive (Humphrey, www, 2012). In the classroom, the ukulele is a full-sized instrument, which is not scaled down for children and it is small enough for children to comfortably play (Perlmutter, 2011). It can be used as a melodic or a harmonic instrument and the chords are simple and accessible for small hands. In addition, a decent ukulele can be purchased for about $50, making it a feasible addition to a music education program (Jones, 2013).

The popularity of the ukulele can be surveyed from youtube.com, which has uploaded tutorials and covered song videos more than 4,090,000 videos (surveyed by keyword “ukulele” on June 1, 2014). Although online resources have a lot of ukulele lesson videos, they are specific songs because each song is presented by individual presenter interested. However, beginners need to find basic skill information from many resources for self-studying such as tuning technique, holding, touching chords, fingering, and strumming pattern (Kummong, 2013). As discussed above, the ukulele self-studying is difficult for beginners because they lack a self-study system and online resources are specific songs.

In the digital age, there are a lot of resources for accessing electronic information, especially in online space such a website. The information is presented in various formats, for example, text, images, video clips, animations, and the specific existing systems that facilitate the user to play musical instrument. The existing systems about string instrument playing are surveyed consisting of eight systems: 1) iPerform 3D (2014), 2) Guitar Alchemist (2014), 3) Guitar Scales (2013), 4) EMedia Intermediate Guitar (2014), 5) Guitar Encyclopedia (2014), 6) Ukulele Chord Finder (2011), 7) Ukulele Beatles Fun! (2003), and 8) Virtual Ukulele Online (2014).

The eight existing systems provide different features which are divided into three main groups as 1) musical feature extraction, 2) functions and techniques of animation displaying, and 3) system flexibility. A feature comparison of existing systems is shown in Table 1.1.

In the first part of Table 1.1, musical feature extraction, many existing systems can extract features from music such as the harmony (chord, key, notes, score), touching chords, and changing chords. Nevertheless, the strumming patterns feature was not found in all reviewed systems.

In the recent years, there are many research about musical feature extraction. For example, Pauws (2004) evaluated a piano musical key extraction algorithm that works directly on raw audio data. The implementation is based on models of human auditory perception and music cognition. Lee and Slaney (2006) developed an automatic guitar chord recognition from audio using a supervised Hidden Markov Model (HMM) trained with audio from symbolic data. Morman and Rabiner (2006) generated an automatic system for segmenting and classifying of chord sequences. The outcomes displayed an enabled high recognition rates for musical chords. Harte, Sandler, and Gasser (2006) developed a novel method for detecting changes in the harmonic content of musical audio signal by using a new model for equal-tempered pitch class space. McVicar and De Bie (2010) projected the method to solve guitar chords by exploiting noisy, but freely and abundantly available online resources. Kaliakatsos-Papakostas, Floros, and Vrahatis (2013) inspected the automatic segmentation of audio data into parts composed in different keys, using clustering on chroma-related spaces.

Table 1.1 A feature comparison of existing systems related to string instrument playing

Related Features	Related Existing Systems								
	1	2	3	4	5	6	7	8	*
Musical Feature Extraction									
Harmony (chord, key, notes, score)	√	√	√	√	√	√	√	√	√
Strumming pattern	-	-	-	-	-	-	-	-	√
Touching chord	√	√	√	√	√	√	√	√	√
Changing chord	√	-	-	√	-	-	√	-	√
Functions and Techniques of Animation Displaying									
Virtual musician or character animation	√	-	-	-	-	-	-	-	√
Virtual musical instrument animation	√	-	-	-	-	-	-	-	√
Controlling the viewpoint of animation	√	-	-	-	√	-	-	-	√
Creating animation by using motion capture machine	√	-	-	-	-	-	-	-	√
System Flexibility									
Importing MIDI files	-	-	-	-	-	-	-	√	-
Importing MP3 files	-	-	-	-	-	-	-	-	√

Related existing systems: *1 = iPerform 3D; 2 = Guitar Alchemist; 3 = Guitar Scales 2.0; 4 = eMedia Intermediate Guitar; 5 = Guitar Encyclopedia; 6 = Ukulele Chordfinder; 7 = Ukulele Beatles Fun!; 8 = Virtual Ukulele Online; * = The Proposed System in this Research*

As discussed above, a diversity of musical feature extraction techniques is presented. Because music contains many elements, the methods to reach the music information have to specify elements with characteristics, such as a musical key extraction, chord recognition, and changing in the harmonic. However, a study of strumming pattern recognition has not been found.

Strumming is a using the dominant hand for playing a string instrument while the other hand holds down notes on the fretboard. Strumming is in the form of rhythm and the most essential aspect of rhythm is consistency and good timing (Kummong 2013). Normally, one song uses only one strumming pattern that depends on a song rhythm. The best way to understand a strumming pattern is counting "1&2&3&4&" while playing the ukulele. For example, the basic strumming pattern is d-/du/d-/du (d = strumming down, u = strumming up, and - = no strumming).

In the second part of Table 1.1, functions and techniques of animation displaying, this part shows the necessary functions of animation displaying a builder for musical instrument playing. However, the existing system, iPerform 3D, is the only one system which has all functions because this system uses motion capture technology for capturing animation. Hence, iPerform 3D displays animation by using 3D objects of both musician and musical instrument. Moreover, users can control the viewpoint of 3D animation such as zoom in, rotate, and slow down.

The motion capture technique is frequently used for data analysis in biomechanics education for the diagnosis of the human body and clinical problems. Moreover, this technique is used in the field of sport science to analyze and design appropriate postures for playing sport. Later, it is widely used in the entertainment industry for movie and computer game productions. The most popular motion capture method is placing markers on the human skin. After that, all markers are detected by a lot of video cameras while human is moving (Xiao and Shengfeng, 2009).

As discussed above, the motion capture is a popular method in 3D animation production because the 3D animation production by animators requires a lot of time. This machine can display realtime motion or record the motion for future analysis, thus motion capture is different from other movements which are produced by the animators.

Furthermore, the 3D animation is beneficial to display the multidimensional views of a 3D object. The power of the creativity coming from the animator's imagination can be visualized by using 3D Computer Generated Imagery (CGI). The characters, complex objects or impossible situations can be created from a 3D CGI and 3D animation controllers allows users to control the 3D view from any angle, which is different from 2D view (Amanda, 2014).

In the end part of Table 1.1, system flexibility, the existing system, Virtual Ukulele Online, allows the user to import Musical Instrument Digital Interface (MIDI) files for chord extraction while the other systems do not have a music extraction function. However, MIDI is a specification of a communication scheme for digital music devices (Loy, 1985) and it is a set of commands for sound synthesis, which is not real sound from musical instruments. Then, this proposed system does not support MIDI files but it supports MPEG-1 audio layer 3 (MP3) files because the MP3 is a now well-known recording device. By using heavy data compression, the machines can record music for considerable durations on to a solid state storage card (Smith, 2012: 159).

Moreover, there are reviews of previous research related to sound recognition and 3D animation builders for displaying musical instrument playing (Jensen and Arnspang ,1999; Chai and Vercoe, 2001; Feng, Zhuang, and Pan, 2003; Sheh and Ellis, 2003; Shao and Kankanhalli, 2004; Yin, Wang and Hsu, 2005; Lin and Lui, 2006; Ng, Weyde, Larkin, Neubarth, TKoerselman and Ong, 2007; Wang, Wu, Deng, and Yan, 2008; Zhu, Manders, Farbiz and Rahardja, 2009; Sauer and Yang, 2009; Lee, Mower, Busso, Lee, and Narayanan, 2011; Zhu, Ramakrishnan, Hamann and Neff, 2012; Perez Carrillo and Wanderley, 2012 and Xia, Tay, Dannenberg and Veloso, 2012). Those research are surveyed into six parts as follows: 1) research publish year, 2) input data, 3) machine learning or feature extraction approach, 4) output format, 5) animation development technique, and 6) field of research.

The results of surveying showed that there is no research studying about the ukulele musical instrument. Before the year 2005, it is not found any explicit research studying about music feature extraction for animation building. Since the year 2005, there are research findings to integrate between music feature extraction and animation building. In addition, some research use the motion capture technology which was a popular output technique to generate proper human motion because it can be applied in many ways for creating an animation smoothly and naturally.

From reviewing about a capability of existing systems and previous research in various levels, some limitations are found. The summary of problem statement of this study is pointed out as follows:

1) Most of the existing systems concentrates on basic skills for ukulele playing and cannot import audio files. Hence, most systems do not have any process of sound extraction for displaying an animation.

2) Many systems display an animation in 2D format and users can view from front view only.

This research aims to design and develop an automatic 3D animation builder for displaying ukulele playing. The system development of this research consists of two main steps. The first step is extracting features from the audio files by using a machine learning approach. The second step is developing 3D animation for displaying ukulele playing. This step uses a motion capture machine to record a position data of the right hand. All animations are linked into extracted features, which are developed in the first step. The output in the second step is a model of musician with the ukulele in 3D animation format and users can customize the 3D animation from any angle.

1.2 Research Objectives

To design and develop an automatic 3D animation builder for displaying ukulele playing that performs the following tasks.

1.2.1 Develop a method of extracting features from ukulele audio files.

1.2.2 Develop a method of building 3D animation for displaying ukulele playing.

1.3 Research Questions

1.3.1 What is the method of feature extraction from ukulele audio file?

1.3.2 What is the method of building 3D animation for displaying ukulele playing?

1.4 Basic Assumptions

1.4.1 The type of ukulele audio files for data extracting uses MP3 files only.

1.4.2 An automatic 3D animation builder for displaying ukulele playing supports on the Windows and Mac operating system.

1.4.3 The usability evaluation of an automatic 3D animation builder for displaying ukulele playing performs after users' system testing under the research environment, which is prepared by the researcher.

1.5 Scope and Limitations of the Study

This study is the development of an automatic 3D animation builder for displaying ukulele playing, which consists of two main steps. The first step is the feature extraction from the audio files using a machine learning approach. The features extraction is performed by using waveform analysis for ukulele strumming pattern recognition only, while chord recognition is accomplished by a Non-Negative Least Squares (NNLS) Chroma and Chordino plugin of Sonic Visualiser software (Mauch and Dixon, 2010). These recognized chords will be passed through a process of chord changing time recognition. The capability evaluation of strumming pattern recognition is operated by using Precision, Recall, and F-Measure. The waveform analysis in the first step is based on three ukulele types: Soprano, Concert, and Tenor, except Baritone. Because the Baritone is the lower pitched instrument, which differs from the other types (Kummong, 2013).

The second step is using a motion capture machine (optical type) for recording a 3D coordinate data and developing an ukulele playing animation. The 3D coordinate data capture from the right-handed musician, which uses the right hand for strumming.

All information obtained from two main steps will be integrated into an automatic 3D animation builder for displaying ukulele playing. This builder will be tested the system usability by 15 people who are interested in ukulele playing with basic computer skills. The system usability testing is based on the Software Usability Measurement Inventory (SUMI) and a thinking-aloud protocol.

1.6 Expected Results

To achieve the design and development of an automatic 3D animation builder for displaying ukulele playing, the expected results are as follows:

1.6.1 To achieve a method of feature extraction from ukulele audio files.

1.6.2 To achieve a method of building 3D animation for displaying ukulele playing.

1.7 Definitions of Terms

1.7.1 Builder

A builder means a computer system that presents a physical model for supporting a user. The user can receive a simulation situation as same as a real system. Builder helps a user in learning in real behaviour before or during performing.

1.7.2 3D Animation

A 3D animation means the speedy display of a continuity of images to make an illusion of movement, drawings, models, or inanimate objects. The 3D animation can be made with computer software in three dimensional space displaying. The creator can be rotated and moved like real objects. The viewing device displays these images in rapid succession, usually 24 frames per second or higher.

1.7.3 Automatic

An automatic means a capability to operate independently of the system by using a mechanism which is designed by the developer. Users need to enter some basic information and then the system will start the operation for displaying output on screen.

1.7.4 Ukulele

An ukulele means a name of the musical instrument. It looks like a small guitar with four strings and it is a well-known symbol of Hawaiian culture. It is commonly strummed with the fingernails and players can play in various techniques such as, strumming style, finger style, and mixed style. The ukulele has four basic sizes which are soprano, concert, tenor, and baritone.

1.7.5 Motion Capture

A motion capture means a popular method in animation production. It is the 3D realistic movement creation by capturing or recording direct movement of human. This machine can display realtime motion or record the motion for future analysis, hence the motion capture is different from other movements which are produced by the animator. Nowadays, there are three types of motion capture techniques: Optional motion capture systems, Magnetic motion capture systems, and Mechanical motion capture systems. The most popular method of motion capture is the Optional motion capture systems which is placing markers on the human skin. After that, all markers are detected by a lot of video cameras while human is moving.

1.7.6 Music Features Extraction

A music feature extraction means the method to get the music information out automatically. The method of extraction depends on each type of information such as key, chords, and rhythm.

CHAPTER 2

REVIEW OF THE LITERATURE

This dissertation aims to design and develop an automatic 3D animation builder for displaying ukulele playing. This chapter presents the background of the main tools, important techniques and an overview of previous studies on related topics that provide the background of the present research. This literature review is divided into five main topics. Section 2.1 discusses ukulele and its popularity because ukulele is the musical instrument for the experiment. Section 2.2 presents the fundamental of sound and music. Section 2.3 explains the recognition in the music, which is the considerable procedure of the system development. Section 2.4 explains 3D animation which is the main format of the automatic system, and describes basic information about the motion capture machine which is the one of experimental tools. Finally, Section 2.5 presents a review of related work about the existing systems and previous research, which concentrate on sound extraction and the 3D animation builder for displaying musical instrument playing.

2.1 Ukulele Musical Instrument

2.1.1 Origin of the Ukulele and Its Popularity

The ukulele is a string musical instrument of the Hawaiian people and it is a well-known symbol of Hawaiian culture. The ukulele is commonly strummed with the fingernails and it can play in various techniques such as, strumming style that is sliding the thumb across the strings, finger style that is playing by using fingernails close to the ukulele bridge for making a metallic sound (Eargle, 2013).

Harden and Brinkman (1999) states that the word "ukulele" is a Hawaiian word referred as "jumping flea"; flea is wingless insects with long legs for bouncing. The ukulele begins when a Portuguese immigrant jumped away from a boat into the sea while playing a little string instrument. Their posture looks like jumping of fleas.

Beverly (2011) mentions that in the year 2000, many popular singers mix the ukulele sound in their new songs and legions of amateurs are following along. Moreover, the melodious sound of ukulele makes happiness both at home and informal ukulele clubs.

Beverly (www, 2011) report that in the year 2011 the website BeatlesCompleteonUkulele.com released a new recording of a Beatles song every week. This website uploads a Beatles song performed by different guest artists. Eddie Vedder, the singer of Pearl Jam band uses the ukulele in his solo album and the ukulele is very interesting from listeners around the world.

In the music education, Jones (2013) presented "A Cultural Study of Ukulele in the Music Classroom". This research is the initial of a study about how to incorporate Hawaiian culture and history when teaching the instrument in the music classroom. The public performance of ukulele can be a very powerful experience for students as they watch the professionals to create music with the ukulele. The study result showed that the ukulele performs for a greater understanding of music's role in Hawaiian culture and throughout the world.

2.1.2 Types of Ukuleles

The ukulele body looks like a small guitar, but it has only four strings while the guitar has six strings. The general material which is used to produce the ukulele is the Hawaiian koa tree because it makes a density and sonority sound. Kummong (2013) describes that the ukulele has four basic sizes which are soprano, concert, tenor, and baritone. The soprano, concert and tenor are the traditional tuning of GCEA, which is higher pitched instrument. But the baritone is the lower pitched instrument, which differs from the other types. A characteristic of each ukulele type from McQueen (www, n.d.) can be described as follows:

2.1.2.1 Soprano Ukulele

The soprano ukulele is the smallest and thinnest ukulele. It has length about 21 inches and 12 – 15 frets. Because it is very small, it is comfortable for holding, carrying and travelling. However, players who have big fingers or big hands have a problem with playing this ukulele type because the frets are closer together.

2.1.2.2 Concert Ukulele

The concert ukulele is bigger than the soprano, but smaller than tenor and baritone. The player can tune it in a standard scale like the soprano ukulele. The space of the concert ukulele frets is just a little bit more than the soprano. Therefore, it is suitable with larger fingers and it is easier to play. The concert ukulele has length about 23 inches and 15 – 20 frets. It allows players to navigate the diverse notes for playing.

2.1.2.3 Tenor Ukulele

The tenor ukulele is bigger than the soprano and concert size. It has length about 26 inches and 15 frets at least. It is generally tuned in standard tuning (GCEA). However, the player can adjust it lower like a baritone ukulele by using DGBE tuning, which is notes for standard tuning of a baritone ukulele. The tenor ukulele is suitable because the player can get a complete sound and the player is able to get higher notes on the finger board.

2.1.2.4 Baritone Ukulele

The baritone ukulele is the largest size of the ukulele. It has length about 30 inches and 19 frets at least. The baritone can tune down lower to DGBE tuning, which is similar to the tuning of the bottom four strings on a guitar. The baritone makes a deeper and fuller sound. It is proper for playing with the finger style by blues musicians because it generates the lower pitched scale.

Table 2.1 classifies measurements of each ukulele type. Figure 2.1 shows the difference of each ukulele type and Figure 2.2 visualizes the anatomy of an ukulele.

Table 2.1 A classification measurement of each ukulele type

Measurement (inches)	Types of Ukuleles			
	Soprano	Concert	Tenor	Baritone
Total Length	21”	24”	26”	30”
Body Length	9.5”	11”	12”	14”
Body Width	6.5”	8”	9”	10”
Body Depth	2.5”	2.75”	3”	3.25”
Sound Hole Diameter	1.75”	2”	2.5”	3”

Figure 2.1 Types of ukuleles (Ukuguides, www, 2012)

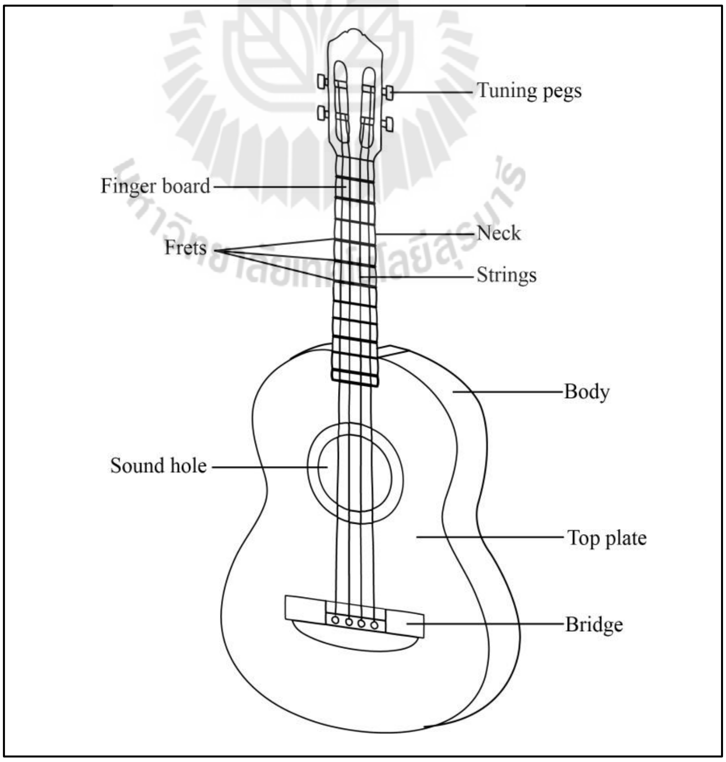


Figure 2.2 The anatomy of an ukulele

2.1.3 Ukulele Strumming

Strumming is using the dominant hand for playing a string instrument while the other hand holds down notes on the finger board. Strumming is in the form of song rhythm. It is a good timing when musicians pluck the ukulele strings (Kummong, 2013). Normally, one song uses only one strumming pattern that depends on a song rhythm. The best way to understand a strumming pattern is counting “1&2&3&4&” while playing the ukulele. For example, the basic pattern is d-/du/d-/du (d = strumming down, u = strumming up), the advanced pattern is du/xu/du/xu (x = chunking, which is muting the strings by covering the strings on the ukulele’s neck with a player’s hand)

However, strumming patterns are various and depend on a player’s style. There are 20 most useful strumming patterns applied from ukulele-tabs.com (Ukukele-Tabs, www, 2013), as shown in Table 2.2.

Table 2.2 The 20 most useful strumming patterns

Pattern No.	Strumming Pattern							
	1	&	2	&	3	&	4	&
1	d	-	d	-	d	-	d	-
2	d	u	d	u	d	u	d	u
3	d	-	d	u	d	-	d	u
4	d	-	d	-	d	u	d	u
5	d	-	d	u	d	u	d	u
6	d	-	d	-	d	u	d	-
7	d	u	d	-	d	u	d	-
8	d	-	d	u	-	u	d	-
9	d	-	d	u	-	u	d	u
10	x	u	x	u	x	u	x	u
11	d	u	x	u	d	u	x	u
12	d	-	d	u	x	u	d	u
13	d	u	-	u	d	-	d	u
14	d	u	x	u	x	u	d	u
15	d	u	d	u	-	u	d	u
16	d	u	x	u	x	u	x	u
17	d	-	d	-	x	u	-	u
18	d	-	d	-	x	u	d	u
19	d	-	d	-	-	u	d	u
20	d	u	-	u	-	u	-	u

* d = strumming down, u = strumming up, x = chunking, and (-) = mute

2.2 Sound and Music

2.2.1 The Sound Wave

Sound is started when an object vibrates. Human can perceive these vibrations by their ears and then the signals are interpreted by the human brain. These vibrations make the moving of molecules in the air and the sound is occurred. The molecules in the air transport via their energy to nearby molecules and the reaction is started.

When the vibration of an object transfers outward, it builds a *compression* (maximum pressure) in the sound wave because molecules in the air are crashed. After that, when the object moves inward by pulling the molecules away from each other, a *rarefaction* (minimum pressure) is originated. (Smith, 2012: 1-3). For example, when the ukulele strings are pulled and released by human hand. Figure 2.3 visualizes the components of a sound wave.

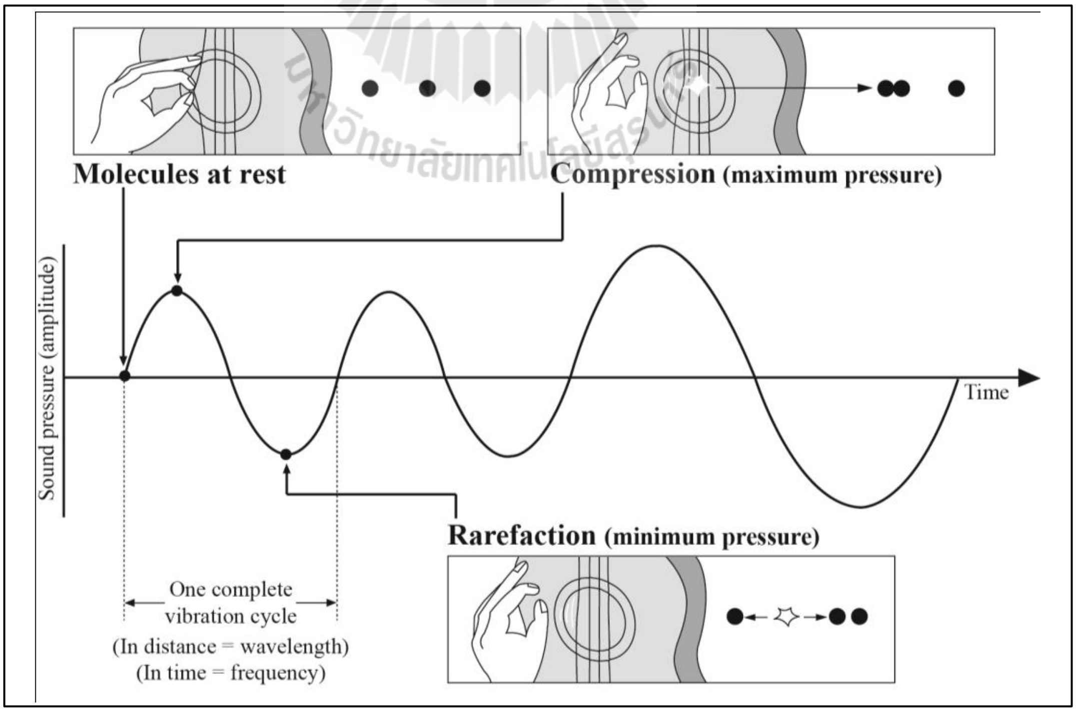


Figure 2.3 The components of a sound wave

2.2.1.1 Frequency

In Figure 2.3, when a vibration completes a motion in one round of up and down from compression to rarefaction, there is finished one cycle. The number of cycles in one second is *the frequency*. For example, if a vibration finishes 10 cycles per second (cps), a frequency will be 10,000 hertz (Hz) or 10 kilohertz (kHz).

In general, all vibrations have a frequency and humans can hear frequencies between 20 Hz to 20,000 Hz. The frequencies less than 20 Hz are called *infrasonic*, and the frequencies more than 20,000 Hz are called *ultrasonic* (Alten, 2013: 14). Figure 2.4 displays the threshold of hearing. It shows a range of frequency which can perceive by human ears. However, it is an attention of the ear that at upper levels the response curve is much flatter (Smith, 2012: 14-15).

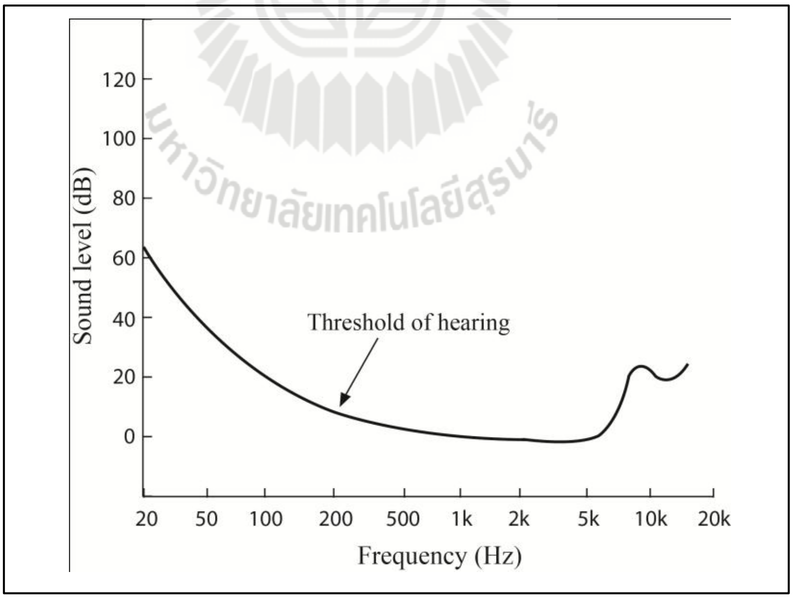


Figure 2.4 The threshold of hearing

2.2.1.2 Amplitude

Amplitude or loudness of sound is the number of molecules displaced by a vibration. Amplitude is measured in decibel (dB), which is a logarithmic unit used to relate to the ratio of two values of a physical quantity, such as sound pressure or intensity. This number is depended on the intensity of an object vibration (Alten, 2013: 14). Figure 2.5 shows the amplitude of sound, which is the loudness of the sound wave. The amplitude of sound in (a) is larger than (b) because the number of molecules in the sound wave in (a) is bigger than (b).

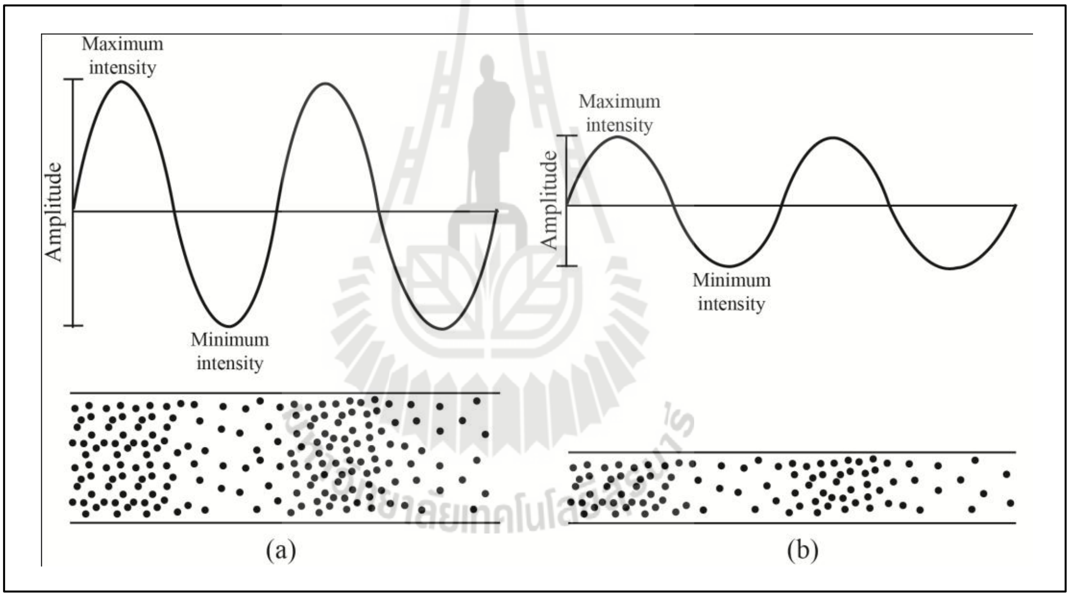


Figure 2.5 The amplitude of sound

2.2.2 The Elements of Music

Human can perceive what kind of sound hearing, such as telephone ringing, a dog barking, music, etc. When humans listen to the music, they will perceive the difference characteristics of music, such as slow or fast, soft or loud, the differentiation of musical instruments, rhythm pattern, etc. All of these are known as the elements of music.

However, music is differentiated from other sounds because it has certain qualities. Music is a multi-dimension and the design of a musical composition can be described on several levels. This part explains three elements of music associated with ukulele sound extraction consisting of harmony, rhythm, and melody as described below.

2.2.2.1 Harmony

Harmony means the blending of notes which plays together and it can refer to the relationship between a set of chords. Harmony is created by playing a group of notes behind the melody thus giving it musical texture and it is built on chord progression or patterns (Leinecker, 1994: 89). There are related in the terms of harmony as follows:

- **Key**: A series of notes based on the standard scale. A key is recognized by a signature of key, which is displayed by the number of sharps or flats in key scale. A key is appointed later its keynote, such as A major's keynote is A and F# minor's keynote is F#. However, the ukulele can play only 12 keys consisting of C, C#, D, D#, E, F, F#, G, G#, A, A#, and B.

- **Tonality:** The musical hierarchy or musical arrangement of all the chords or pitches in a song composition.

- **Chord:** A group of notes which is played at the same time to create a signature of harmony. Three or more notes can combine various chords. For example, B major chord can be played by using three classes of notes consisting of B, D#, and F#. Chords add texture to a melody and build rhythm to a song.

In general, chords are displayed by using alphabets, numbers and special characters, such as A, A9, B#, C#m, Emaj7, etc. The ukulele consists of 168 chords, which are shown in Table 2.3.

Table 2.3 The 168 ukulele chords

Note	**Ukulele Chord**						
C	C	Cm	C7	Cm7	Cmaj7	Csus2	Csus4
	C7sus4	C6	Cm6	C9	Cm9	Caug	Cdim
C#	C#	C#m	C#7	C#m7	C#maj7	C#sus2	C#sus4
	C#7sus4	C#6	C#m6	C#9	C#m9	C#aug	C#dim
D	D	Dm	D7	Dm7	Dmaj7	Dsus2	Dsus4
	D7sus4	D6	Dm6	D9	Dm9	Daug	Ddim
D#	D#	D#m	D#7	D#m7	D#maj7	D#sus2	D#sus4
	D#7sus4	D#6	D#m6	D#9	D#m9	D#aug	D#dim
E	E	Em	E7	Em7	Emaj7	Esus2	Esus4
	E7sus4	E6	Em6	E9	Em9	Eaug	Edim
F	F	Fm	F7	Fm7	Fmaj7	Fsus2	Fsus4
	F7sus4	F6	Fm6	F9	Fm9	Faug	Fdim
F#	F#	F#m	F#7	F#m7	F#maj7	F#sus2	F#sus4
	F#7sus4	F#6	F#m6	F#9	F#m9	F#aug	F#dim
G	C	Cm	C7	Cm7	Cmaj7	Csus2	Csus4
	C7sus4	C6	Cm6	C9	Cm9	Caug	Cdim
G#	G	Gm	G7	Gm7	Gmaj7	Gsus2	Gsus4
	G7sus4	G6	Gm6	G9	Gm9	Gaug	Gdim
A	A	Am	A7	Am7	Amaj7	Asus2	Asus4
	A7sus4	A6	Am6	A9	Am9	Aaug	Adim
A#	A#	A#m	A#7	A#m7	A#maj7	A#sus2	A#sus4
	A#7sus4	A#6	A#m6	A#9	A#m9	A#aug	A#dim
B	B#	B#m	B#7	B#m7	B#maj7	B#sus2	B#sus4
	B#7sus4	B#6	B#m6	B#9	B#m9	B#aug	B#dim

2.2.2.2 Rhythm

Rhythm is the structure of sound generated by the establishment of stressed and unstressed syllables in the sound. Rhythm is directly related to time in music and it is most often thought as a drum part, but the melodic rhythm is important. The term of melodic rhythm refers to the durational patterns of a melody. For example, to understand the rhythm when we tap foot during listening to the music, we are keeping the rhythm of the music (Leinecker, 1994: 71 - 79). There are the related terms of rhythm as follows:

- **Duration:** A range or period of time while hearing a sound or silence in any music. Duration can be also relevant to how long or short of note set in any music.

- **Tempo:** The velocity of the beat which can be shown by the number of beats in a second. In general, the tempo will be often displayed by using a unit of beats per minute (BPM). Nevertheless, in Italian terms, the tempo is classified by a unique name. Figure 2.6 shows a comparative chart of tempos.

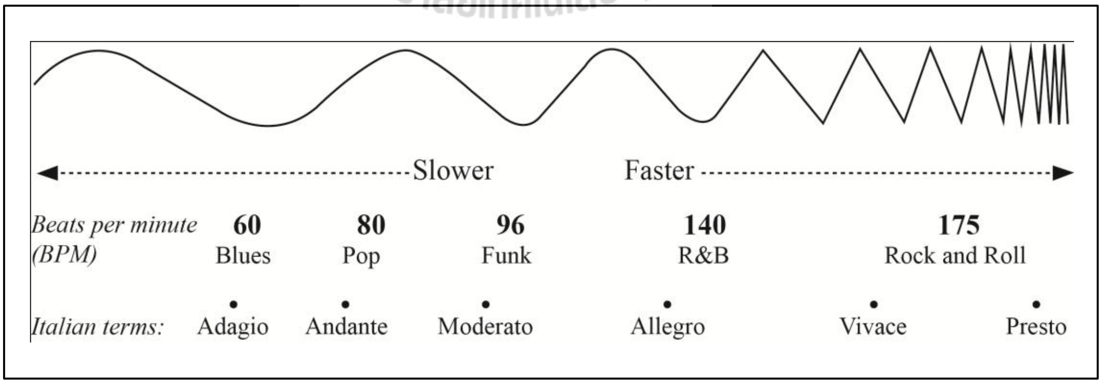


Figure 2.6 A comparative chart of tempos

- **Meter:** The rhythmic structure in a song composition. The comprehensible meter occurs after the beats are arranged into the completed stress patterns. The most common meters are shown in Figure 2.7.

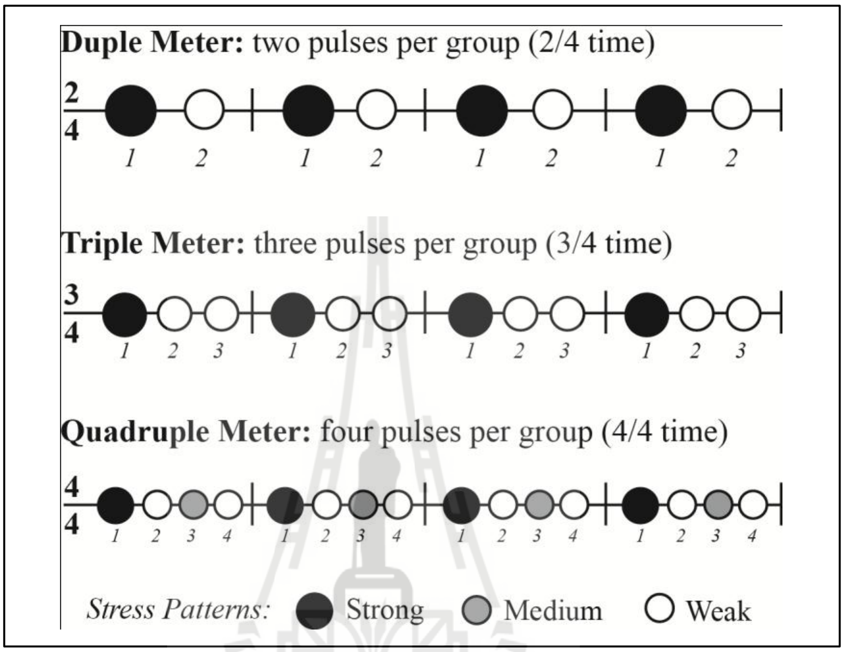


Figure 2.7 The meter diagram

2.2.2.3 Melody

Melody is created by playing a continuity of pitches. It is one of the music elements, which focuses on the plane performance of the pitch. Melody can be received from various scales, such as the traditional major and minor scales of tonal music. A note is a sound with a discriminative pitch and duration. However, the melody of a piece of music is not just any string of notes (Schmidt-Jones, 2008: 32). There is one related term of melody as follows:

- **Pitch:** The maximum or minimum point of a sound. The vibrating frequency affects to the level of pitch. For example, the G string of an ukulele vibrates 196 times per second; therefore, its frequency is 196 Hz. The A string has a frequency of 110 Hz; therefore, the pitch of the G string is higher. Figure 2.8 visualizes the example of waveforms of pitch C from different musical instruments: Ukulele (a), Clarinet (b), and Oboe (c).

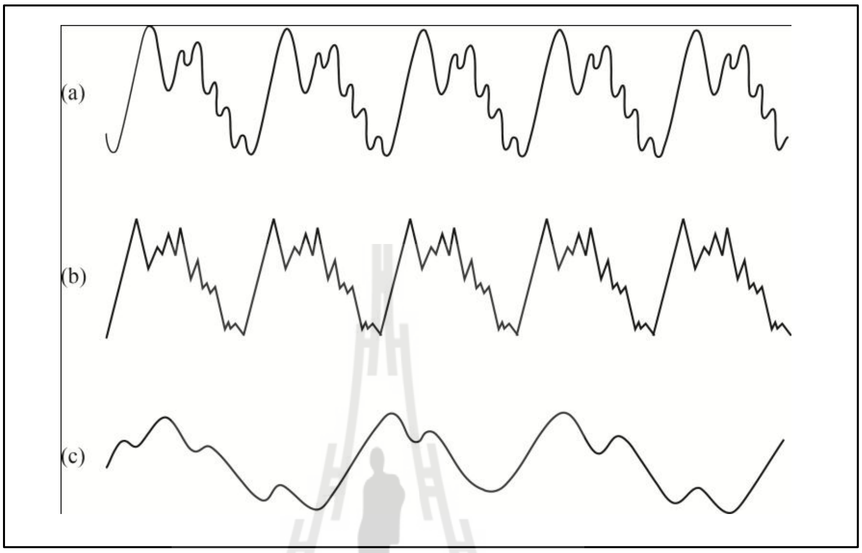


Figure 2.8 The waveforms of pitch C from different musical instruments

2.3 Recognition in Music

2.3.1 Music Recognition Techniques

The music contains and merges information at many elements. The method to get the music information out automatically specifies with each type of information, such as key, chord, and sound intensity. This section presents three recognition techniques for music information extraction, which consists of the decision tree learning, the neural network learning and the hidden Markov model approaches.

2.3.1.1 Decision Tree Learning

The decision tree learning is a popular approach in the field of classification and it is a procedure generally used in data mining. The objective of the decision tree learning is to construct a prediction model based on many input variables. The decision tree is visualized in the form of flow chart structure or decision rule structure. The prior of a rule is a continuity of test as same as the tests at nodes, while the result or output creates the class (or classes) that are adapted into instances covered by rules (or a probability distribution over the classes) (Witten and Frank, 2005: 59).

Pang-Ning, Steinbach, and Kumar (2006: 146) explain that classification by using decision tree is the way of learning a target function *f*, which maps each attribute set *x* to one of the predefined class labels *y* (as shown in Figure 2.9). The target function is also known as a classification model. The advantage of a classification model is separated into two parts:

- **Descriptive Modeling:** The objects of various classes are separated by a classification model as an explanatory tool.

- **Predictive Modeling:** A class label of unseen records is predicted by a classification model and classification techniques are suitable for predicting datasets based on nominal classes.

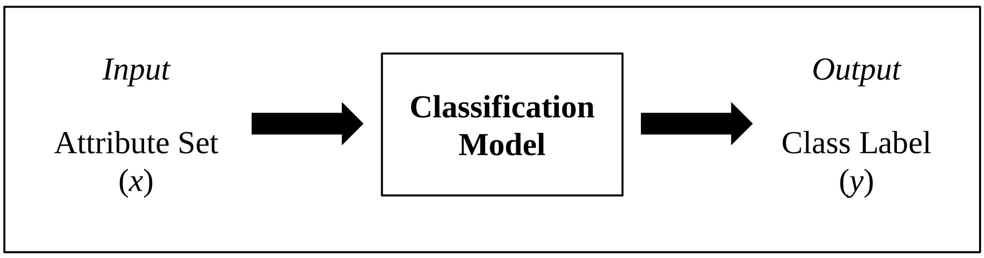


Figure 2.9 Classification as the task of mapping an input attribute set *x* into its class label *y*

Figure 2.9 presents a simple concept for building the classification model. First, a training set with the record of data attributes and recognized class labels must be provided. After that, the training set will be applied to construct a classification model, which is subsequently used to the test set, which consists of data attribute records and unknown class labels.

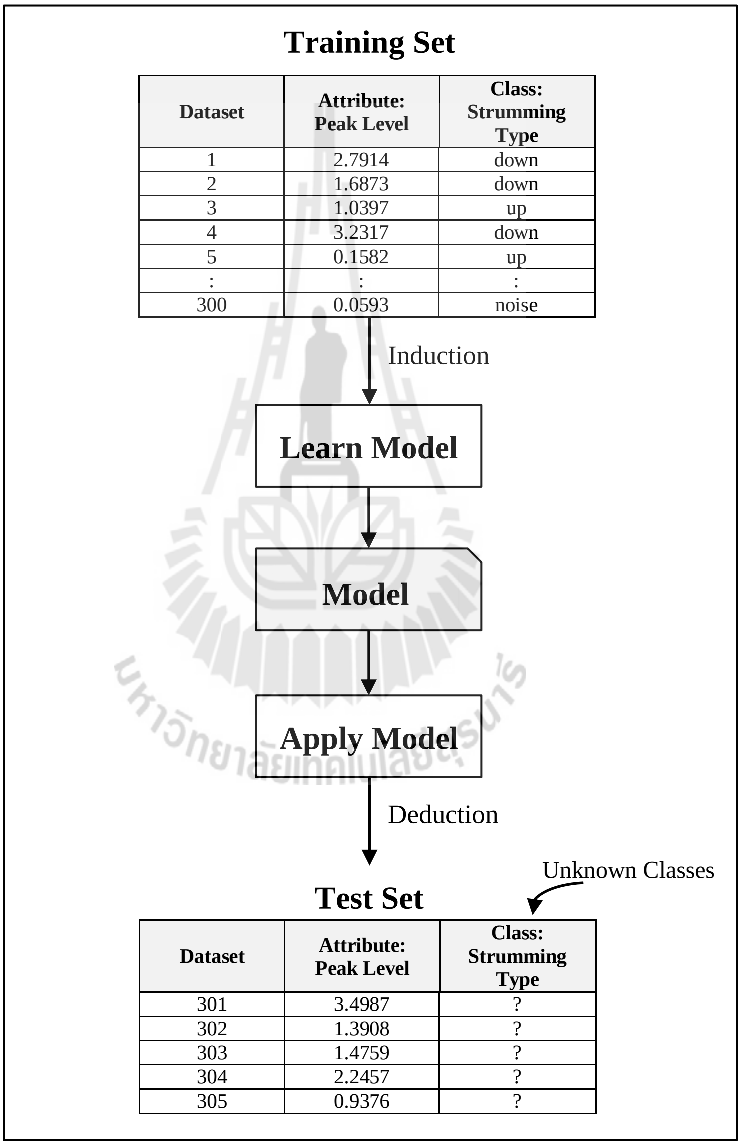


Figure 2.10 A simple concept for building a classification model

Witten, Frank and Hall (2011: 69) describe about the classification rules from a decision tree that this procedure gives rules that are explicit in the operation of irrelevant data. A decision rule is originated for each leaf and the previous rule consists of a condition for every node on the path from the root to that leaf. The outcome of the rule is the class defined by the leaf.

- **The Fundamental of Decision Tree:** The decision tree learning for classification is widely used in both statistics and pattern recognition. The decision tree has three types of nodes as follows:

  - **A root node** is a top-level of the tree. It does not have incoming links and zero or more outgoing links. In general, this node is defined by a significance attribute.

  - **Internal nodes** are consequent nodes under the root node. It has exactly one incoming link and two or more outgoing links. However, the internal nodes are created more than one level in the overall tree.

  - **Leafs or terminal nodes** are the final nodes, which are recognized the unknown classes consisting of one incoming link and no outgoing links. (Pang-Ning, Steinbach, and Kumar, 2006: 150-151). Figure 2.11 shows the example of a decision tree for the basic strumming type classification.

Henery (www, 1994: 7) mentions about the benefits of classification, which can summarize as the lists below.

- **Accuracy:** The received rules of classification are integrity. They are represented by the stability of correct classifications. Although it may be some errors which are more serious than others, it may be important to perform the error rate for any key class.

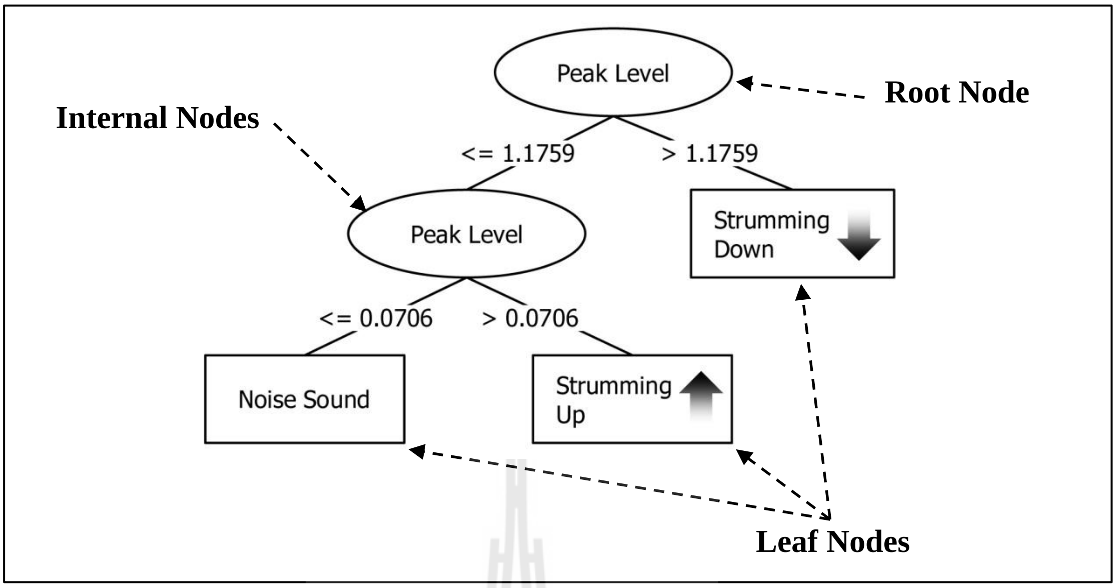


Figure 2.11 The example of a decision tree for the strumming type classification

- **Fastness:** A classifier that is 90 per cent of precision. It may be preferred over one that is 95 per cent of precision, if it is 100 times faster in testing. For example, such considerations would be considerable for the automatic checking or automatic defect detection of products on a production line.

- **Comprehensibility:** The methods of classification should be simple to conceive and absent faults in the decision rules. It is important because it will make a comprehensive system and more efficiency for a user.

- **Learning Time:** It is important for learning a classification rule rapidly or building modifications to an existing rule in real time, especially in an immediately changing situation.

2.3.1.2 Neural Network Learning

Neural network learning or Artificial Neural Network (ANN) is applied in a diversity of tasks, including prediction or function estimation, clustering, forecasting, and pattern classification, as shown in Figure 2.12. Neural networks are very strong when the constructed models are appropriate to data. It can provide informal complex nonlinear models to multidimensional data (as depicted in Figure 2.12b). Moreover, neural networks are also competent of complex data and signal or time-series classification tasks in the term of informal complex nonlinear classification boundaries (as shown in Figure 2.12a) (Samarasinghe, 2006: 12-13).

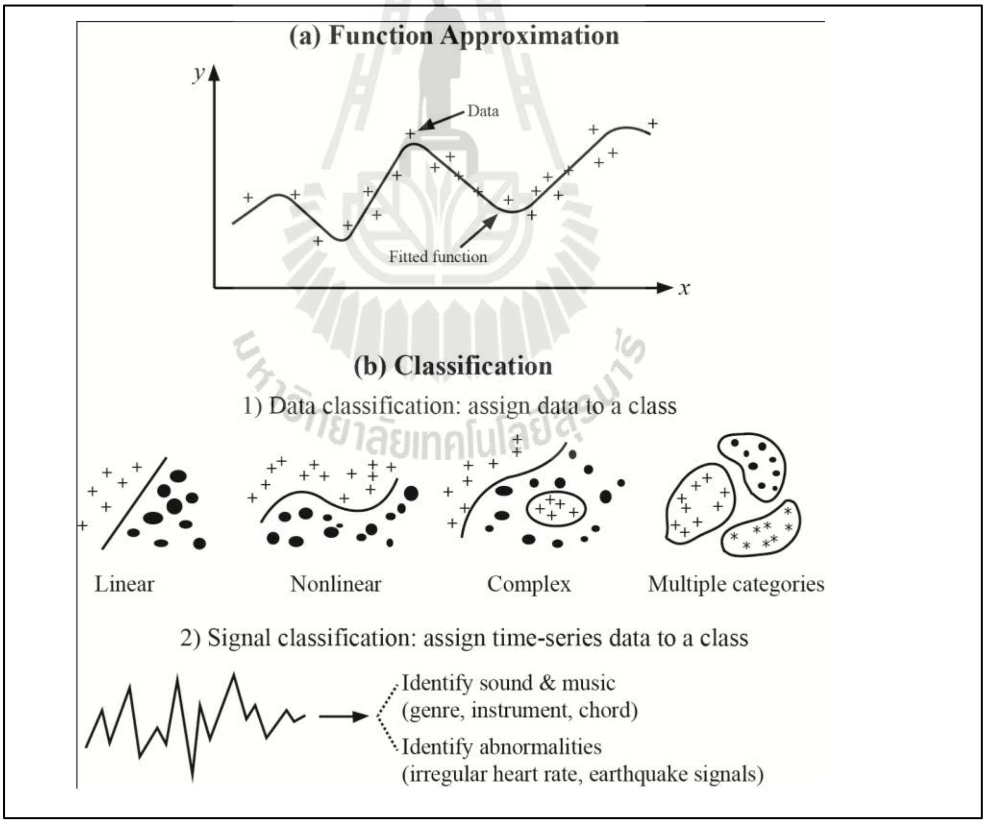


Figure 2.12 Some functions of neural networks suitable for data modeling: (a) fitting models to data, (b) complex classification tasks

Fausett (1994: 3) states about the neural network that

"The neural network has been developed as generalizations of the mathematical models of human cognition or neural biology, based on the assumptions as follows: 1) Information processing occurs at many simple elements called neurons, 2) signal are passed between neurons over connection links, 3) each connection link has an associated weight, which in a typical neural net, multiplies the signal transmitted and 4) each neuron applies an activation function (usually nonlinear) to its net input (sum of weighted input signal) to determine its output signal."

- **The Fundamental of Neural Network:** The operation of the neural network is resolved by many components consisting of the network architecture, the size of the weights, and the processing element's mode of operation. The processing elements work in parallel as the neurons do in the human's brain. The *neuron* is sometime called that *node* or *unit*, which is a processing element that takes a number of inputs for weighting and integrating, and employs the result as the argument for a singular valued function. Figure 2.13 shows the basic model of the neuron.

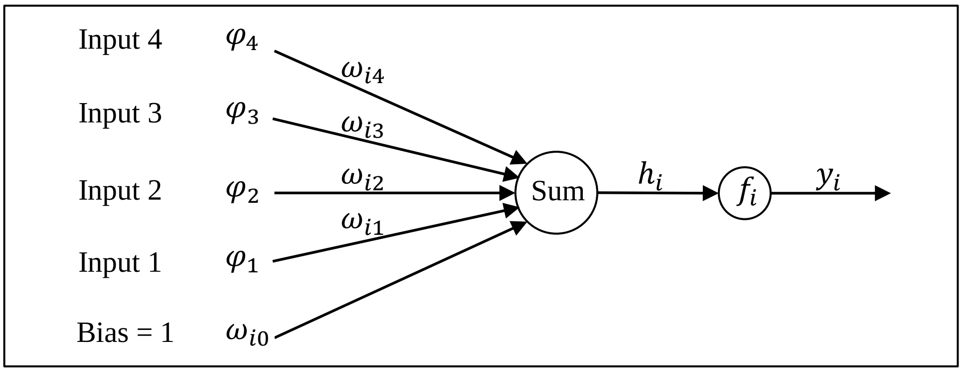


Figure 2.13 A basic model of the neuron

In Figure 2.13, the inputs to a unit can either be outputs of other units. The displacement ω_{i0} is called the *bias* and it can be interpreted as a weight applied to a pseudo input, which is clamped to the constant value 1. Essentially, the activation function f_i can take any form, but most often it is monotonic (Nørgard, Ravn, Poulsen, and Hansen, 2000: 6).

In addition, Doszkocs, Reggia, and Lin (1990) state that a neural network has three components consisting of a network, an activation rule, and a learning rule.

- **The Network** consists of a group of connected nodes or connected units in the connected links. Each node in the network receives a numeric activation level associated with it at time *t*. The completed pattern vector of activation demonstrates the current state of the network at time *t*.
- **Activation Rule or Activation Function** is a local process which is followed by each node for improving its activation level in the context of input from connected nodes.
- **Learning Rule** is a local process that explains how the weights on the adjacent links in the overall network should be modified as a function of time.

- **The Multi-Layer Perceptron**: Multi-layer perceptron (MLP) is another name for a feed-forward neural network. Despite the name, *the neurons* are not usually perceptrons (Ripley, 1996: 351). MLP is one of the most popular ANN being used today and it is pretended that a number of node element layers are existed and added (Singhal, 1988).

MLP consists of a system of simply interconnected neurons or nodes, as displayed in Figure 2.14, which is a nonlinear mapping model between an input layer and output layer. The nodes are linked by weights and output signals which are a summation function of the inputs to the node altered by a simple nonlinear transfer, or activation function (Gardner and Dorling, 1998). Moreover, it may have one or more hidden layers before reaching an output layer.

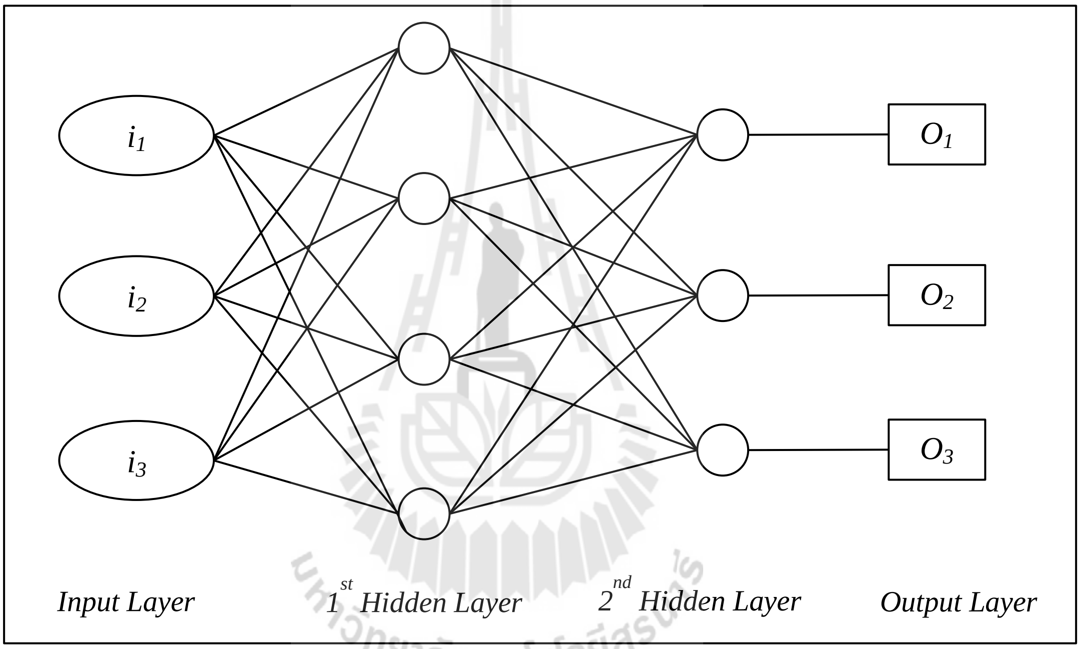


Figure 2.14 A multi-layer perceptron with two hidden layers

Figure 2.14 shows a MLP with two hidden layers. The MLP starts with a set of data into the input layer. The results from the input layer are fed as weighted inputs to the first hidden layer, and the outputs from the first hidden layer are fed as weighted inputs to the second hidden layer, respectively. These processes perform until the output layer is completed (Irwin, Warwick, and Hunt, 1995: 5-6). However, in other cases, the number of input layer may not be equal to the desired output during training the MLP because an error signal is identified as the difference between the desired and actual output.

Haykin (1999: 24 - 26) presented the benefits of properties and capabilities of the neural network as follows:

- **Nonlinearity:** The neural network builds a connection of nonlinear neurons or hidden layer, which is a highly significance property for classifying.

- **Input-Output Mapping:** The network learns from the input layers by constructing an input-output mapping for the prediction.

- **Evidential Response:** In the field of pattern classification, a neural network can use to prepare information for the finest selecting not only about the specific pattern recognition, but also about the confidence in the decision made.

- **Contextual Information:** Every neuron in the network is possible affected by the global activity of all other neurons in the network.

- **Uniformity of Analysis and Design:** The neural network enjoys universality as information processors and it manifests itself in different ways.

2.3.1.3 Hidden Markov Model

A Hidden Markov Model (HMM) is a statistical model. It is essentially adapted in speech recognition, natural language modeling, gesture recognition, on-line handwriting recognition, and the analysis of biological sequences (Bishop, 2006: 610). The HMM has a number of parameters whose values are set to recognize training patterns in the format of data sequences for the known classes. After that, a test pattern is classified by the HMM model with the finest possibility.

- **The Fundamental of HMM:** A HMM is learning for automatic prediction. Dymarski (2011: 5-7) summarizes a type of prediction operation with two following aspects. The first step is defining a condition set, where each condition is mostly connected with a multidimensional probability distribution. The transitions between 6 HMMs are organized by a set of probabilities which is called transition probabilities. The second step of a prediction operation is showing the learning rules, where in some conditions a result can be observed and the conditions are hidden.

Steve (www, 2002) discusses three algorithms concerned in making HMMs work as follows: 1) *Estimating conditional probability:* Calculating the probability of the input sequence for building a model. 2) *Finding the best path through the model:* Finding the path that most closely matches the input sequence. 3) *Train a model:* Estimating the Gaussian parameters and transition probabilities to best describe for a data set.

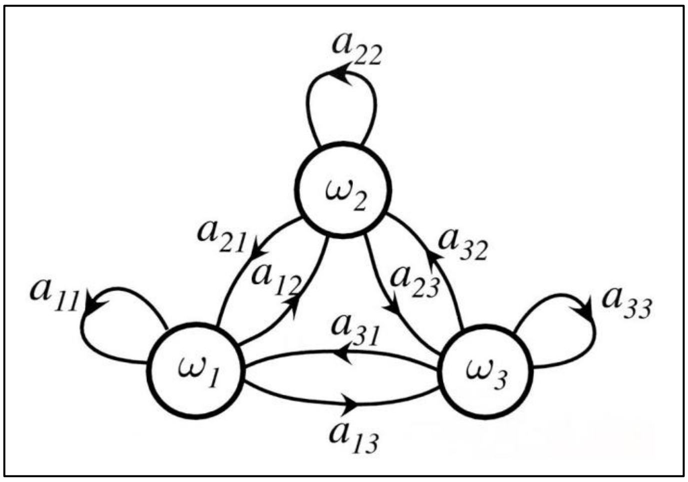


Figure 2.15 A basic Markov model

Figure 2.15 shows a basic Markov model, which is visualized by nodes. The transition probabilities a_{ij} are visualized by links with direction. In a first order discrete time Markov model, at any step t the full system is on a particular state w_t. The state at step $t + 1$ is a random function that solely depends on the state at step t and the transition probabilities (Duda, Hart, and Stork, 2012: 129).

A basic Markov model as shown in Figure 2.15 occurred in three states w_1, w_2, *and* w_3. However, it has a probability of emitting a particular visible state V_t. Because it can access only to the visible states, while the w_i are invisible (Duda, Hart, and Stork, 2012: 129), such a full model is called a Hidden Markov Model as shown in Figure 2.16.

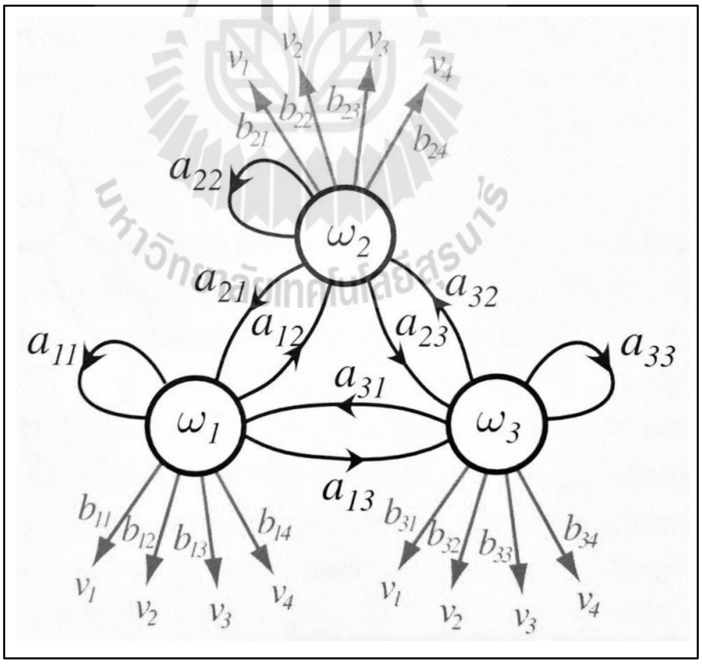


Figure 2.16 Hidden units in a HMM

Figure 2.16 shows Hidden units in a HMM, where this model shows all possible transitions. The black color units are three hidden units in a HMM and the transitions between them while the grey color units are the visible states with the emission probabilities (Duda, Hart, and Stork, 2012: 130-131). However, some candidate transitions are not allowed in other HMMs.

- **The Three Basic Problems of HMM:** The best set of state transitions and observation probabilities is the objective of the HMM learning operation. Consequently, a result of a sequence or a set of sequences is returned. Rabiner (1989) discusses the three basic problems of HMM as follows:

  - **The Evaluation Problem:** This problem is related to the calculation system of the HMM because the model is multidimensional, which is complicated for analyzing. A HMM operation cannot evaluate the completed probability in sometime.

  - **The Decoding Problem:** The users should define suitable probability for HMM operation. This problem is complicated for observing by users.

  - **The Learning Problem:** This problem is related to the number of states and the number of visible states. Given a set of training observations of visible symbols, then these parameters are determined.

2.3.2 Feature Recognition in Music

This part reviews previous research in the field of musical feature extraction. A lot of techniques are used for analyzing sound wave and extracting important features, such as chords, keys, and rhythms as explained below.

2.3.2.1 Chords Recognition

Pardo and Birmingham (2002) presented an algorithm to segment and label chords based on template matching and graph search techniques. Paiement, Eck, and Bengio (2005) designed a distributed representation of chords by constructing a graphical model for chord progressions. Lee and Slaney (2006) employed Hidden Markov Model trained with audio from symbolic data for constructing an automatic guitar chord recognition. Morman and Rabiner (2006) generated an automatic system for segmenting and classifying of chord sequences. The outcomes displayed an enabled high recognition rate for musical chords. Scholz and Ramalho (2008) proposed a new approach to recognize chords from symbolic MIDI guitar data. The system uses contextual harmonic information to solve ambiguous cases, optimization, pattern matching, and rule-based recognition. McVicar and De Bie (2010) projected the method to solve guitar chords by exploiting noisy, but freely and abundantly available online resources.

2.3.2.2 Key Recognition

Chew (2002) used a spiral array model which is a Boundary Search Algorithm (BSA) for assigning modulation’s points in music by performing with a geometric model. Raphael and Stoddard (2008) performed with a Hidden Markov Model to be automatically trainable by using a generic MIDI files. They completed in finding the globally optimal keys and chord labeling. Hu and Saul (2009) operated an unsupervised learning technique on a Latent Dirichlet Allocation (LDA) probabilistic model to determine keys. Wang and Wechsler (2012) presented a Bayesian-based methodology that determines keys of symbolic music using unsupervised learning. Kaliakatsos-Papakostas, Floros, and Vrahatis (2013) used a clustering on chroma-related spaces to build the automatic segmentation of audio data into parts composed in different keys.

2.3.2.3 Rhythms Recognition

Dixon (2001) showed an approach for automatic extraction of tempo and beat from expressive performances. The results are displayed for classifying different musical styles. McKinney and Moelants (2004) presented a procedure for adapting such measures of perceptual tempo. They designed an automatic tempo tracker for more precisely representing the perceived beat in any song. Gulati and Rao (2010) used machine learning approaches to investigate different signal processing methods for extracting rhythm patterns and evaluating them for the music tempo detection task.

2.3.2.4 Others Recognition

Barthélemy and Bonardi (2001) built a procedure for automatic harmonic recognition of a music score by using the extracted figured bass, which is resolved by means of a template matching algorithm. Shenoy and Wang (2005) mixed features of rhythm and harmonic structure between low-level and high-level music for recognizing chord, key, and rhythm of music. They employed a rule-based approach as a main manner. Harte, Sandler, and Gasser (2006) used a new model for equal-tempered pitch class space for finding changing in the harmonic content of musical audio signal. Pérez-García, Pérez-Sancho, and Iñesta (2010) collected two kinds of information which are very different nature for recognizing a musical genre by using classification approach. Chuan (2011) applied *N*-gram models for harmonic analysis for differentiating a composer's style.

As discussed above, a diversity of musical feature extraction techniques is exposed**.** However, this research uses an individual algorithm for extracting strumming patterns. Moreover, it integrates suitable algorithms in the open source software for extracting other features.

2.4 3D Animation and Motion Capture Machine

2.4.1 Definition of 3D Animation

Sanders (www, 2013) describes about 3D animation that

"3D animation is the creation of moving pictures in a three-dimensional digital environment. This is done by sequencing consecutive images, or frames, that simulate motion by each image showing the next in a gradual progression of steps, filmed by a virtual camera and then output to video by a rendering engine. The eye can be fooled into perceiving motion when these consecutive images are shown at a rate of 24 frames per second or faster."

Xiong, Lu, Fang, and Wang (2010: 2) state the development of 3D animation that

"The development of 3D animation production is based on cultural and creative product development. It is the integrating of computer graphic technology and the arts in the 3D virtual space. The 3D object displaying from various angles needs to follow through project planning which is the designing of original painting, 3D modeling, animation, special effects, rendering, post-edit, etc."

In addition, to create 3D characters, Sparga (www, 2008) mentions that

> *"3D animation enables animators to create a smooth, realistic illusion of movement without the burden of creating reams of hand-drawn illustrations. The characters appearance will stay consistent throughout the animation because their proportions are already set in the model. Not only does 3D animation make realistic settings and complex movements possible, but it also cuts down the amount of time required to take an animated feature film from the rough draft and storyboard stages to full-fledged theatre quality. Because of the quicker schedules, film studios have embraced the approach of merging high technology with the art of animation."*

As discussed above, 3D animation is the rapid display of still images in order to make an illusion of motions, drawings, models, or lifeless objects. The 3D animation can be made with computer software in three dimensional space displaying based on the X, Y, and Z axes. The creator can rotate and move like real objects. 3D animation is the important tool of games and virtual reality development. The monitor displays a continuity of images in rapid sequences, usually at least 24 frames per second or higher. Figure 2.17 displays the example of 3D character in the 3D axis.

Moreover, Amanda (www, 2014) explains the benefits of 3D animation into 12 lists. There are 7 lists related to research purposes.

2.4.1.1 Fulfill Human Imagination: The power of the creativity or novel idea coming from the artist or creator's imagination can be illustrated by using 3D Computer Generated Imagery (CGI).

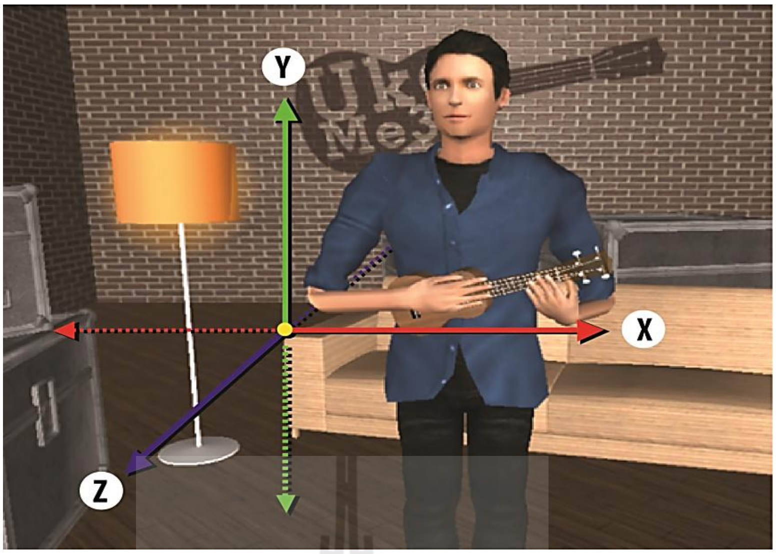


Figure 2.17 The 3D character in the 3D axis

2.4.1.2 Make the Impossible Possible: Numerous digital make-ups on the characters or impossible situations can be created from a 3D CGI program, such as huge waves, rain storm, and world war.

2.4.1.3 Provide the Camera Functionality: The 3D CGI software prepares mathematically controlled motion paths, point of interests, and movement patterns from the widest to the narrow spaces because the 3D CGI software provides a lot of camera functions, without the requirement for dollies or tracks for actual movement. The proposed system in this research applies the 3D camera functions into the system and it allows users to control the 3D view from any angle.

2.4.1.4 Get Low Cost: In some animation productions, the creators may be disregarding the requirement for actual stunts because the realistic 3D stunt can make a realistic action in a movie. The 3D CGI models are now relatively cheaper and more environment-friendly than destroying real objects, such as car, building, etc.

2.4.1.5 Provide the Variety of Special Effects: The elements of 3D objects can be animated according to the natural laws of physics by assigning specific properties to objects, such as mass, gravity, texture, material, and velocity. Moreover, the 3D CGI provides choices of moving around in the most creative and even the most impossible ways via motion paths. The animators can generate various special effects and complicate objects, such as fire, smoke, snow, rain, plants, trees, humans, animals, and monsters.

2.4.1.6 Combinate Animation Styles: The production of animation can integrate traditional animation, stop-motion animation, clay animation, 2D animation, and 3D animation together. The combination of animation styles becomes more interesting and it makes the unique styles of the project.

2.4.1.7 Get Advantages of Motion Capture: Motion capture becomes a desirable machine to project in film and game production. It helps the animators to generate keyframes digitally captured from human's movements and saves time in the 3D animation development. This research uses the motion capture machine for tracking the musician's right hand while strumming the ukulele. Figure 2.18 shows the application of motion capture in this research.

Figure 2.18 The application of motion capture in the ukulele strumming

2.4.2 The Basics of 3D Animation

The development of 3D animation customarily occurs in two main ways consisting of keyframe animation and motion capture as explained below.

2.4.2.1 Keyframe Animation

Keyframe animation is a familiar method in traditional animation where an animation order is acquired by tweening two images which depicts the start and end frames (Vince, 1992: 238). Keyframing animation is altering the figure, shape, form, spacing, position, or timing of an object in successive frames, with major changes to the object being the key frames (Animation Arena, www, 2012).

In traditional, each frame of 2D animation is drawn by the artist’s hand on the plain paper. When all frames are shown in continuity, a little bit different points in each frame will make the illusion of motion, for example, a different position of ball in Figure 2.19. However, in 3D animation production, 3D software such as Adobe Flash makes a keyframe animation by tweening and interpolating. All of frames are calculated the proper position by the software automatically.

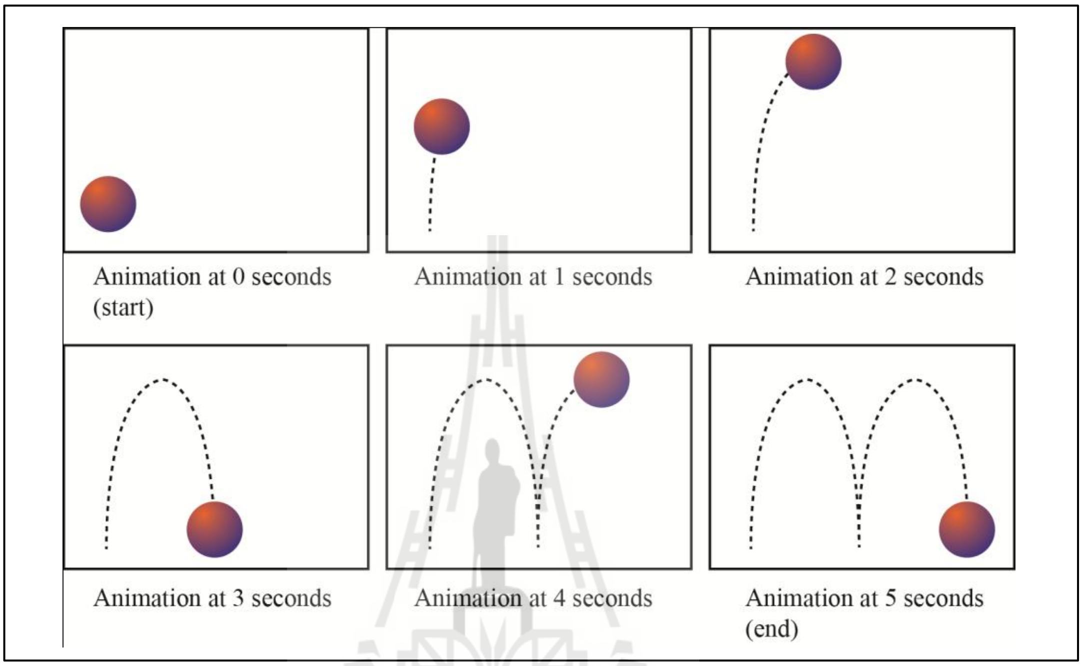


Figure 2.19 Keyframe animation

2.4.2.2 Motion Capture

Motion Capture (MoCap) is a popular technique for creating human animation. It is an advantageous tool for capturing human motion, animal and other movement both in rapid motion and slow motion environment. MoCap for 3D character animation is about the integrating of human motion data into the 3D character for creating motion naturally. Mocap is operated by a motion capture machine as described below.

1) Overview of Motion Capture Machine

The motion capture machine is the 3D realistic movement creation by capturing or recording direct movement of human and it is different from keyframe animation, which uses the animator's hand in creating a frame by frame animation. Motion capture is a popular method in animation production because it helps an animator to create a movement of characters. This machine can display realtime motion or record the motion for future analysis and then motion capture is different from other movements which are produced by the animator because it gives a smooth and naturalistic animation.

Webster (2005) mentions that the invention of motion capture systems and the development of computer programs used with a motion capture system make the motion capture become an efficiency tool for animation production. The motion capture system does not only use programs for displaying the motion, but it also uses programs for editing. The program for editing is created for improving or editing movement. In addition, the program for editing is used for blending many movements together by using the keyframe animation technique.

2) Types of Motion Capture Machine

A motion capture machine reduces the cost of animation production because making animation by the motion capture is easier than hand drawing. Nowadays, there are three types of motion capture techniques: Optional motion capture systems, Magnetic motion capture systems, and Mechanical motion capture systems.

- **Optical Motion Capture Systems**

Optical motion capture systems are the systems which use reflection equipment or sensors attached to an actor or an object. Motions of an actor or an object are recorded via special cameras and then the recorded motions are processed by an application PC.

Figure 2.20 An optical motion capture system (Swangoblues, www, 2014)

At present, there are two main techniques for optional motion capture systems that are reflective markers and Pulsed-LED markers. The high definition cameras are necessary for optical motion capture systems. All cameras will be used for tracking the movement of markers which are connected to the body of the actor or actress. Figure 2.21 presents a component of motion capture system.

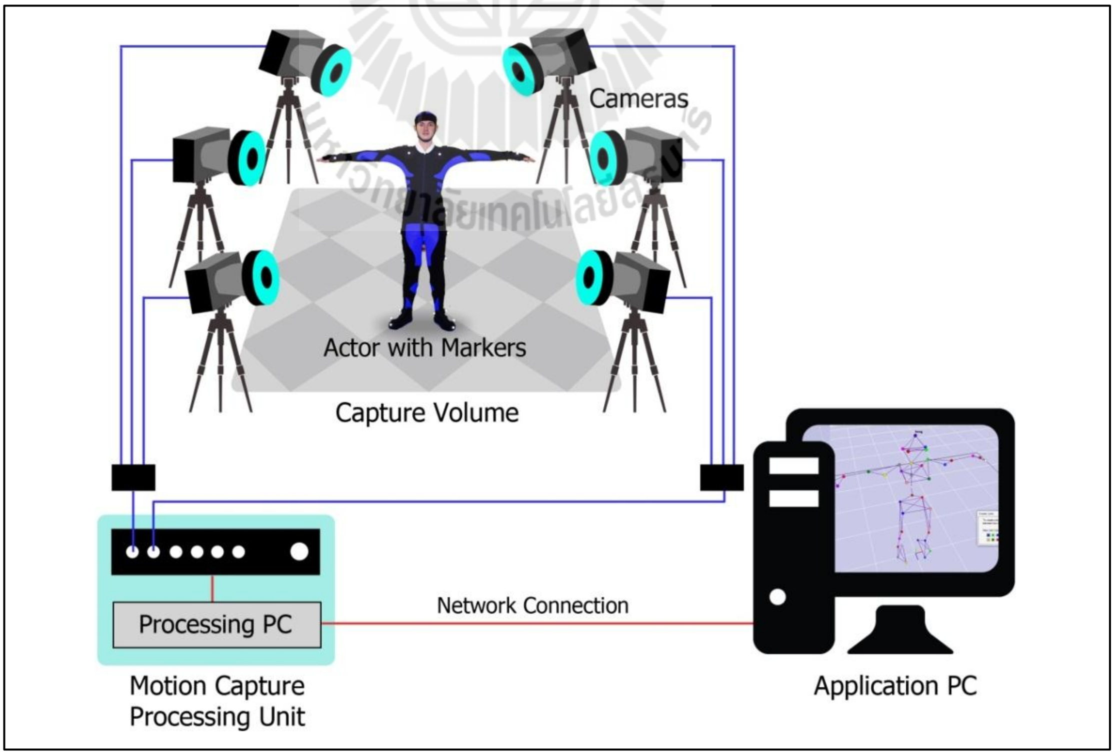


Figure 2.21 A component of motion capture system

Other than the body motion capture, face motion capture requires only one or two cameras while the body motion capture requires 3 - 16 cameras or more (Menache, 2011). Reflective optical motion capture leans on Infrared-LED markers which fix on circumference of special lens. These lenses can detect infrared rays.

Kurihara, Hoshino, Yamane, and Nakamura (2002) presented the real-time operating of optical motion capture by using pan-tilt camera tracking. They found that the pan-tilt camera tracking expanded the scope of capturing area. The algorithm was designed for parallel computed in realtime data analysis. This research showed the effectiveness of the optical motion capture systems.

Zhang, Biswas, and Fan (2010) used the motion capture data and human shape models to create 3D animations. The proposed pipeline integrated two animation software tools, Maya and Motion Builder together. Their objective was to generate both realistic and accurate motion. The method was tested by three data sets of motion capture in various motion types and five existing human shape models. Their method confirmed the better visual realistic and kinematic accuracy when compared with three other animation generation methods.

- **Magnetic Motion Capture Systems**

Magnetic motion capture systems are performed by using sensors to correctly measure the low frequency magnetic field. One or more electronic control units are merged into the cable source and sensors in the system of magnetic motion capture. The animation in 3D space will be perceived via 3D animation software. These systems employ 6 to 11 sensors per actor and place them on the actor's body to capture the movement of body joint. After that, the sensors show a position and rotational information on a computer display (Meta Motion, www, n. d.).

Mitobe et al. (2006) manipulated the magnetic three dimensional position sensors for hand tracking in the motion capture experiment. They developed the high accuracy systems for 3D hand motion capture by using the electromagnetic tracker for recording the finger movements of a pianist that are high in speed and unpredictable. The result indicated that the accuracy was not affected even if each pianist's finger was adjoining.

Figure 2.22 Magnetic motion capture systems (Kaiga et al, 2007)

- **Mechanical Motion Capture Systems**

Mechanical motion capture systems use three elements to track motion consisting of orientation sensors, bend sensors and potentiometers. These systems record movements directly via a skeletal type suit attached to the actor. While the actor moves with the suit, the motion data are captured by many sensors at each joint and all data are recorded into a system.

Mechanical motion capture systems are cheaper than optical motion capture systems and magnetic motion capture systems. However, the suits have some limitations because only major movements can be recorded.

Tiesel and Loviscach (2006) introduced procedures to extract and tweak kinematical as well as timing data from mechanical motion capture systems. They found that only the measured acceleration data can equip neither complete nor accurate information to remodel the captured motion.

Figure 2.23 Mechanical motion capture systems (Rowan, www, 2011)

The beginning of motion capture was not in the entertainment industry, but it started in medicine. Such devices had been used in medicine for many years before becoming workable tools in the computer graphic field. Nowadays, medicine, entertainment, sports, and law are the main fields which get an advantage from the motion capture.

In the medicine field, the motion capture is called 3D biological measuring or 3D motion analysis. It is beneficial to analyze the biomechanical data for human walk posture diagnostic or gait analysis, and orthopedic applications, such as joint mechanics and sports medicine. In gait analysis, motion capture is very advantageous because it can present the multiple phase of the patient walk cycle and it is simple to detect certain abnormalities and changes (Laybourne, 1998).

In the entertainment industry, motion capture has been widely applied to create specific movement of the character in several recent theatrical releases both in television and film. For example, Avatar, the movie created by James Cameron, used motion capture to animate characters riding on direhorses and flying on the back of mountain banshees (Parent, 2008). In addition, motion capture is vastly used in 3D game production because it is comfortable to capture the human motions, which are applied to simulate 3D characters in games.

As discussed above, the motion capture is a very robust and beneficial tool. It is a proper technology for the computer 3D animation development. Because of its complexity and the speed at which profitable production proceeds, it is a challenging process to adapt in various fields and this research uses the optical motion capture system in the 3D animation development.

2.5 Related Work

This section reviews related literature for research. The content in this section is separated into 2 main parts: 1) The existing systems for string instrument playing, and 2) Previous research related to sound extraction and a 3D animation builder for displaying musical instrument playing.

2.5.1 The Existing Systems for String Instrument Playing

This research aims to design and develop an automatic 3D animation builder for displaying ukulele playing. Hence, the related existing systems in this part are selected especially in the field of string instrument playing (ukulele or guitar). There are eight systems as described below.

iPerform 3D (2014) is a 3D guitar learning system developed by iPerform3D incorporation. This system allows users to practice guitar playing from any angle as shown in Figure 2.24. Users can zoom in, rotate, slow down, and customize learning experiences. The iPerform3D displays basic guitar playing skills, such as strumming, fretting, notes, and much more. 3D animation in iPerform3D is captured by using MoCap360 technology. However, this system does not allow users to import any song from the computer, but the users can choose from hundreds of riffs, rhythms, and solos which are provided in the system.

Figure 2.24 An example of iPerform 3D screenshot

Guitar Alchemist (2014) is a system for finding chords for playing the guitar. This system visualizes scale patterns both the 2D fingerboard (only left hand) as shown in Figure 2.25 and a 2D virtual musician at the same time. In addition, Guitar Alchemist can show a simple chord dictionary for beginners to advance their guitar playing.

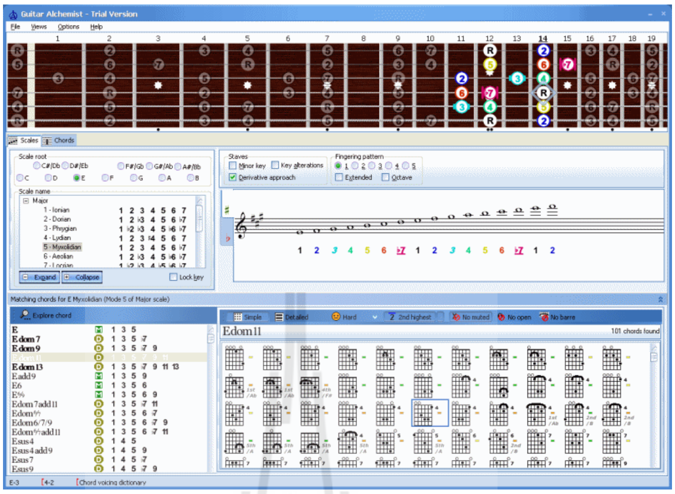


Figure 2.25 An example of Guitar Alchemist screenshot

Guitar Scales (2013) can display chords and fingering for touching chords, but users cannot upload any song to extract chords in this system. This system is used for practicing the guitar. User can watch, listen and play along with accurate time and the appropriate user's skill level. The users can select guitar neck styles, such as number of frets, note color, and background color during the practice as shown in Figure 2.26.

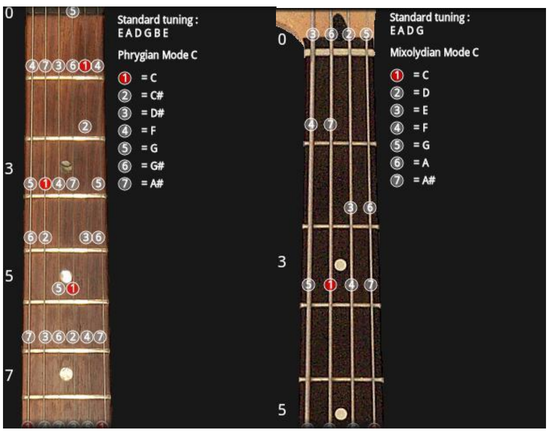


Figure 2.26 An example of Guitar Scales screenshot

EMedia Intermediate Guitar (2014) has more than 170 guitar lessons projected for advance guitar players. This system helps users to learn the way to lead the guitar, fingerpicking patterns, and music theory. Moreover, EMedia Intermediate Guitar's lessons present guitar songs with classic riffs played by Eric Clapton and Jimi Hendrix, who are the famous singers. In the term of displaying a guitar fretting board, this system shows 2D animation and recognizes the finger position by using an animation of color point as shown in Figure 2.27.

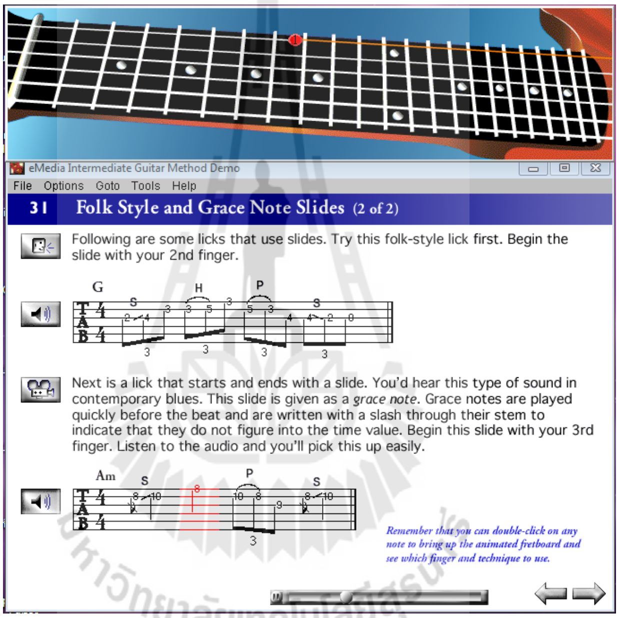


Figure 2.27 An example of EMedia Intermediate Guitar screenshot

Guitar Encyclopedia (2014) supports learners to practice diverse performance techniques based on modern styles and interpretations of guitar playing. The learner can find the techniques to play the electric and acoustic guitars. This system allows the user to watch guitar experts in video format, record and print the lessons, and submit the recordings of learner performances to the instructors for an evaluation as shown in Figure 2.28.

Figure 2.28 An example of Guitar Encyclopedia screenshot

Ukulele Chord Finder (2011) is a 2D graphic website for finding chords in various songs as shown in Figure 2.29. The user can choose songs which are recorded on the web database but they cannot upload songs from their computer. After a user selects a song from the database, this system will display the alphabet of each chord, 2D virtual guitar, fingering positions, song lyrics, and sound of the selected song. However, this system does not allow users to control the viewpoint of animation except playback controlling.

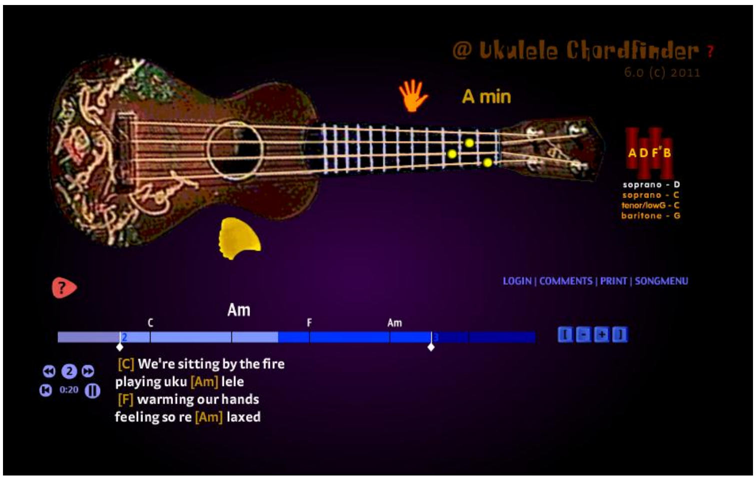


Figure 2.29 An example of Ukulele Chord Finder screenshot

Ukulele Beatles Fun! (2003) presented 64 well known Beatles' songs, such as Yesterday, Lady Madonna, and Yellow Submarine in the format of 2D graphic ukulele as shown in Figure 2.30. Users can select a song for displaying information of chords, song lyrics, fingering positions, and MIDI of Beatles' songs. Unfortunately, Ukulele Beatles Fun! is developed for Beatles songs only.

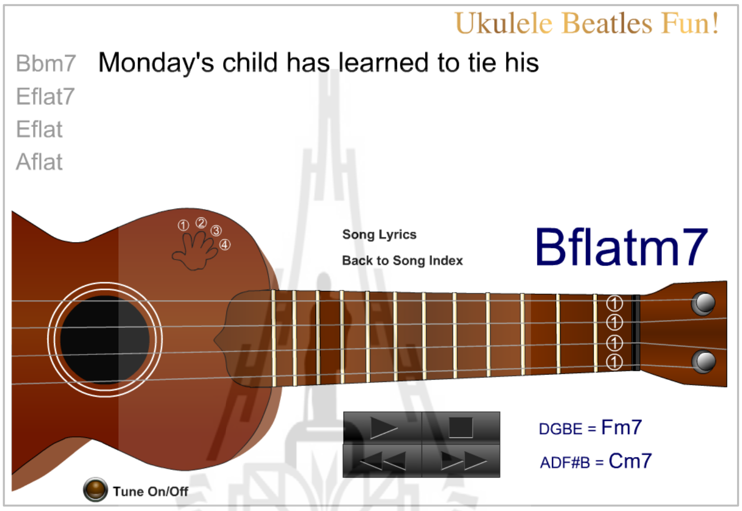


Figure 2.30 An example of Ukulele Beatles Fun! screenshot

Virtual Ukulele Online (2014) is developed for ukulele chord compiling. This system supports all of ukulele types: Soprano, Standard, Concert, and Baritone. A distinctive point of this system allows the user to select any MIDI song from the computer. After selection, the system will analyze the selected song to show all chords which are used for ukulele playing with the 2D ukulele fretting board as shown in Figure 2.31. However, this system does not allow users to improve chords before learning.

Figure 2.31 An example of Virtual Ukulele Online screenshot

As discussed above, a comparison of the existing support systems for playing string instruments is illustrated in Table 1.1 of Chapter 1.

2.5.2 Previous Research

These part reviews previous research related to sound recognition and the 3D animation builder for displaying musical instrument playing. There are 15 related research as described below.

Jensen and Arnspang (1999) used the decision tree for classifying musical sounds in the instrument families. They concentrated on the interpretation of the timbre for attribute extraction. The experiments were performed by recording sounds from various instruments and the results showed that the decision tree was created in a few levels. The binary tree which was originated with the timbre data could be used to classify the musical instrument family.

Chai and Vercoe (2001) employed Hidden Markov Model (HMM) to classify a folk music from different countries based on their monophonic melodies. The songs were collected from Irish (I), German (G), and Austrian (A) folk music as the data set. All of folk music were extracted the attributes into four types of the song melody representations consisting of absolute pitch, absolute pitch with duration, interval, and contour. After that, each attribute would be measured the performance for folk music classification. The results were achieved in 75 per cent of classification accuracy between I-G, 77 per cent between I-A and 66 per cent between G-A for two-way classifications and 63 per cent between I-G-A for three-way classifications using 6-states left-right HMM. Moreover, the results showed that the song interval representation generally performed better than the song pitch or the contour representation.

Feng, Zhuang, and Pan (2003) operated the neural network classifier for music mood recognition. They collected 223 pieces of modern popular music containing multiple instruments and vocal singing from the Internet and personal CD repository, where 200 pieces were used as training data, 23 pieces as testing data. All music pieces were divided into four types of mood consisting of happiness, sadness, anger, and fear. The feature extraction was calculated by only the time domain energy of music signals. The results of model construction consisted of three input nodes, ten hidden nodes and four output nodes. The experimental results showed that the precision was 67 per cent and the recall was 66 per cent of average.

Sheh and Ellis (2003) built a system for automatic chord transcription using speech recognition tools. The pitch class profile vectors of the Beatles songs were used as features to emphasize the tonal content of the signal. The chord sequence recognition was accomplished by using HMM and trained by the same Expectation-Maximization (EM) algorithm. The results on a small set of 20 early Beatles' songs showed that the frame-level accuracy of around 75 per cent on a forced-alignment task and HMM trained by EM could successfully recognize chords in unstructured, polyphonic, and multi-timbre audio.

Shao and Kankanhalli (2004) projected an unsupervised clustering method for an automatic music genre classification based on a given measure of similarity which could be provided by a Hidden Markov Model. They performed two steps of the experiment. Firstly, every music piece was segmented into clips and features were extracted based on these segments. The three types of the features were extracted for each segment consisting of 1) Mel-frequency Cepstral coefficient, 2) Linear prediction coefficients (LPC) derived cepstrum coefficient, and 3) Delta and acceleration. After that, they trained a HMM by using these features. Secondly, they embedded the distance between every pair of music pieces into a distance matrix and performed clustering to generate desired clusters. The results by using the proposed method with five-states presented that the accuracy obtained 88 per cent, 92 per cent, 76 per cent, and 100 per cent in a Pop, Country, Jazz, and Classic music, respectively.

Yin, Wang, and Hsu (2005) designed an experience with Digital Violin Tutor (DVT). DVT was merged violin audio transcription with visualization and combined a system for violin beginners. The operation of violin audio transcription was fast, correct, and low noise while the visualization was designed to be easily understandable by a learner with little music skill. DVT displayed 2D fingerboard animation and 3D virtual musician animation which were created by an animator for helping beginners to learn and practice more capably. In an evaluation process by users, DVT had received very positive results.

Lin and Lui (2006) designed a virtual piano tutoring system. The system used a MIDI song as input and displayed the suggested output by

3D virtual pianist which was created by an inverse kinematic technique. This system created suitable fingering for beginners in order to imitate. The result showed that the fingering solving could be displayed in real time, reduced the searching space of fingering, and provided an ergonomic based on pose evaluation. Moreover, they presented Slicing Fingering Generation (SFG), which was a new framework for solving a polyphonic sequence using a novel fingering generation mechanism.

Ng, Weyde, Larkin, Neubarth, Koerselman, and Ong (2007) projected multimodal interfaces for music learning and teaching. 3D motion capture, motion analysis, visualization, and sonification were used for displaying a method that purposed to support string practice training. They developed a system based on the traditional function of a physical mirror as a violin teaching assistance. Moreover, the approach of developing an “augmented mirror” using 3D motion capture technology was applied. However, this research was interested in only motion of the violinist and violin.

Wang, Wu, Deng, and Yan (2008) explained a procedure of realtime speech and music classification by using a hierarchical oblique decision tree. A realtime features were deployed from harmonic structure stability (HSS) based on a rough harmonic structure estimation. A feature subset selection tool was used to select a subset of short and long term features to feed into a hierarchical oblique decision tree classifier. After that, the oblique decision trees were trained and tested for classification. The experimental result was achieved in 98.3 per cent of accuracy.

Zhu, Manders, Farbiz, and Rahardja (2009) reported a concept for music feature analysis with an application in dance animation. They studied feature extraction of various music styles, such as drum beats, rhythms, and meters, which were used for synchronization with motion sequences from a database of dance animation files. The experiment was based on music in dancing club consisting of hip-hop and R&B. Their concept validated the performance of the proposed techniques.

Sauer and Yang (2009) proposed a music-driven character animation used for extracting musical features from a song and creating an animation. They developed a system for building a new animation directly from musical attributes. Autodesk Maya software was used for animation production. Another feature of the system was its ability to unite multiple characters in the same animation, both with synchronized and asynchronized movements. The system helped users to create charming animations by using different input types of music. An evaluation showed that the majority of animations was found and their method improved an attractiveness of the music.

Lee, Mower, Busso, Lee, and Narayanan (2011) presented a hierarchical computational structure to recognize speech emotions by using a binary decision tree approach. The framework was designed by empirical guidance and experimentation. A multilevel binary decision tree structure was proposed to perform multi-class emotion classification, such as angry, emphatic, neutral, positive, and rest. The results showed that the presented hierarchical approach was effective for classifying emotional pronunciation in multiple database contexts.

Zhu, Ramakrishnan, Hamann, and Neff (2012) presented the system for automatically creating 3D animation of piano playing by importing a midi audio file. This work used a graph theory-based motion planning method for making a set of fingers when a pianist strikes the piano keys for each chord. A motion capture machine was used to collect motion transitions between poses. The outcome of motion capture was realistic because motion capture data were naturally in the transition between poses. The research approach was proved by a comparison between the actual piano playing and simulation of a complete music piece, which required various playing skills.

Perez Carrillo, and Wanderley (2012) introduced a procedure to extract violin instrumental controls from an audio signal. The extraction was based on learning by means of direct measurement with sensors. Sound analysis was based on a signal captured with a violin pickup, which was a signal close to string vibration. The learning consisted of training statistical models with a database of violin performances, which included audio spectral features and instrumental controls. The outcomes showed that an automatic acquisition from any violin recording would be more difficult than the specific violin recording with spectral properties.

Xia, Dannenberg, Tay, and Veloso (2012) studied the method to automate robot dancing by forming schedules of motion primitives that were driven by the emotions and the beats of any music on a humanoid robot, but the algorithms can be used with any robot. The process was started by giving emotion labels for static postures. They estimated the activation valence space locations of the motion primitives and selected the proper motion primitives for emotion detection in the music. This research supported synchronization between robot dance motions with emotion and the music. In addition, they presented the procedure of the schedule of motion primitives and recompensed for any timing errors.

As discussed above, Table 2.4 shows a comparison of previous research into six aspects which are research publish year, input data of feature extraction, machine learning or feature extraction approaches, output format, animation development techniques, and fields of research.

Table 2.4 presents a progression of research in the fields of sound recognition and 3D animation builder for displaying musical instrument playing. Before the year 2005, there are no research findings to study about music feature extraction for animation building, i.e., most research aims to study only the feature extraction method for music recognition by using machine learning in various approaches, such as Decision tree learning, Neural network learning, Hidden Markov model, etc. Moreover, since the year 2005 there are research findings to integrate between music feature extraction and animation building. The motion capture machine was a popular animation development technique to generate proper human motion because it can be applied in many ways for creating an animation smoothly and naturally.

Table 2.4 A comparison of previous research related to sound recognition and the 3D animation builder for displaying musical instrument playing

No.	Year	Input Data of Feature Extraction	Machine Learning / Feature Extraction Approach	Output Format	Animation Development Technique	Field of Research
1	1999	Any Music Files	Decision Tree	Type of Musical Instrument	Animation Not Included	General Music
2	2001	Not Defined	Hidden Markov Model	Classification of Folk Music	Animation Not Included	Folk Music
3	2003	Any Music Files	Neuron Network Learning	Mood of Music	Animation Not Included	General Music
4	2003	Beatles Songs	Hidden Markov Model	Chord Sequence Information	Animation Not Included	General Music
5	2004	Any Music Files	Hidden Markov Model	Genre of Music	Animation Not Included	General Music
6	2005	MIDI Files	Feature Extraction Not Included	2D & 3D Animation	Created by Animator	Violin
7	2006	MIDI Files	Slicing Fingering Generation (SFG)	3D Animation	Inverse Kinematics (IK)	Piano
8	2007	Not Used	Feature Extraction Not Included	3D Animation	Motion Capture Machine	Violin
9	2008	Any Speech / Music Files	Decision Tree	Genre of Speech and Music	Animation Not Included	Speech / General Music

Table 2.4 A comparison of previous research related to sound recognition and the 3D animation builder for displaying musical instrument playing (cont.)

No.	Year	Input Data of Feature Extraction	Machine Learning / Feature Extraction Approach	Output Format	Animation Development Technique	Field of Research
10	2009	Any Music Files	Music Segmentation and Intensity Labeling	3D Animation	Used Motion Sequences from Database	Dancing
11	2009	Any Music Files	Tempo Detection and Onset Tracking	3D Animation	Automatically Created by Maya Software	Body Movement
12	2011	Speech Files	Decision Tree	Emotion of Speech	Animation Not Included	Speech
13	2012	MIDI Files	Graph Theory-Based Motion Planning	3D Animation	Motion Capture Machine	Piano
14	2012	MIDI Files	Neuron Network Learning	Violin Controlling Information	Animation Not Included	Violin
15	2012	Any Music Files	Beats and Emotions Detection	Movement of Real Robot	Markov Dance Model	Robot Dancing
*	2015 (This Research)	MP3 Files	Decision Tree Learning, Neuron Network Learning, and Hidden Markov Model	3D Animation	Created by Motion Capture Machine and Animator	Ukulele

Related works: *1 = Jensen and Arnspang (1999); 2 = Chai and Vercoe (2001); 3 = Feng, Zhuang, and Pan (2003); 4 = Sheh and Ellis (2003); 5 = Shao and Kankanhalli (2004); 6 = Yin, Wang and Hsu (2005); 7 = Lin and Lui (2006); 8 = Ng, Weyde, Larkin, Neubarth, TKoerselman and Ong (2007); 9 = Wang, Wu, Deng, and Yan (2008); 10 = Zhu, Manders, Farbiz and Rahardja (2009); 11 = Sauer and Yang (2009); 12 = Lee, Mower, Busso, Lee, and Narayanan (2011); 13 = Zhu, Ramakrishnan, Hamann and Neff (2012); 14 = Carrillo and Wanderley (2012); 15 = Xia, Tay, Dannenberg and Veloso (2012); * = This research*

As explained above, the system development steps which are operated in this research can be summarized as follows:

1) Extracting features from the ukulele audio files by using machine learning approaches. The contributions of this step are the strumming pattern recognition and the chord changing time recognition.

2) The development of 3D animation for displaying ukulele playing. The contribution of this step is the method to create dynamic animation from unseen music.

As reviewed of the literature and related works in this chapter, this dissertation aims to design and develop an automatic 3D animation builder for displaying ukulele playing. The research framework can be shown as Figure 2.32.

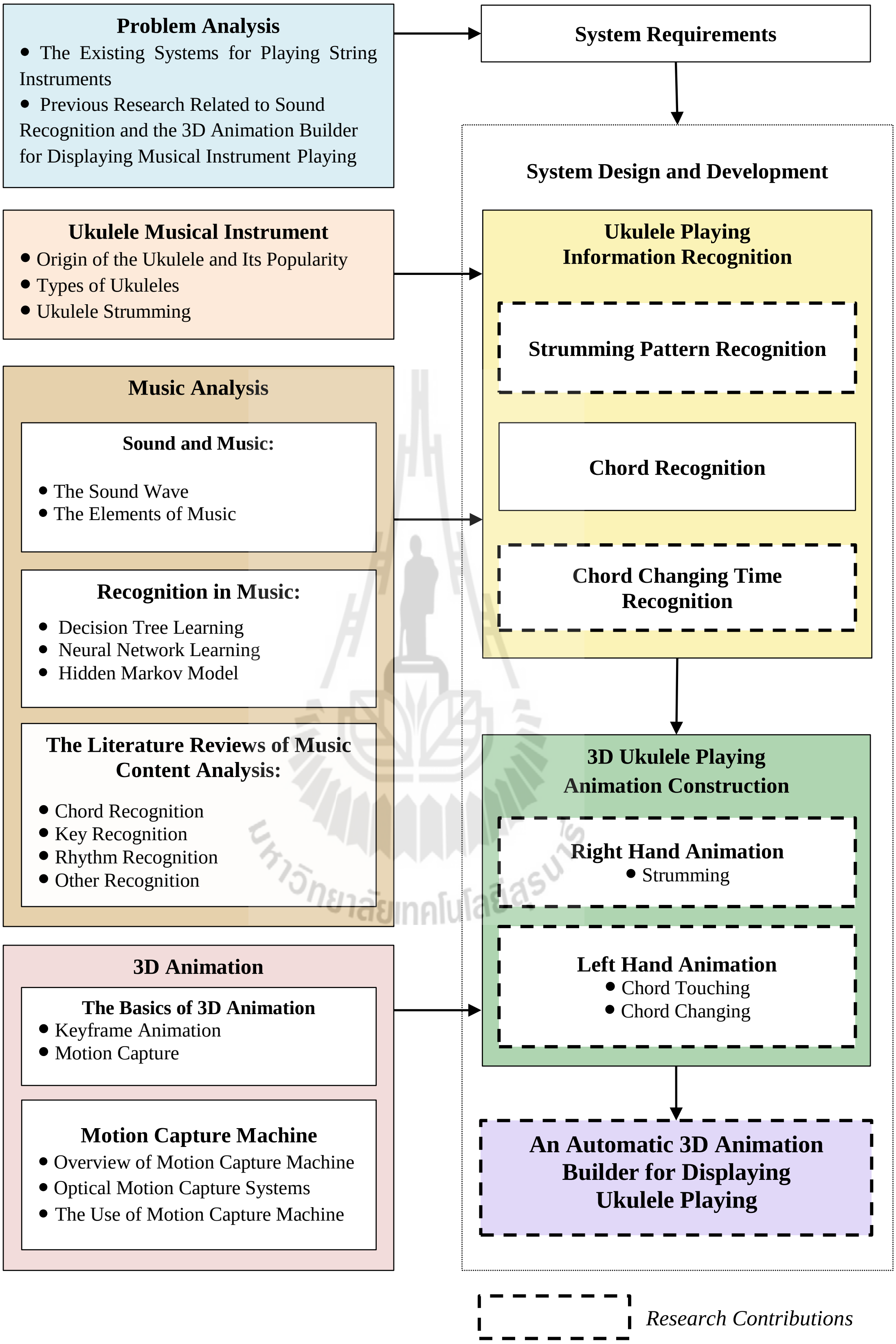


Figure 2.32 Research framework

CHAPTER 3

RESEARCH PROCEDURE

The objective of this chapter is to explain how the research will be carried out. This chapter describes the research methodology including problem analysis, system design and development, and system testing and evaluation. Then, population and sampling, research instruments, data collection and data analysis are presented.

3.1 Research Methodology

This work is an applied research related to the development of an automatic 3D animation builder for displaying ukulele playing. The methodology applies the System Development Life Cycle (SDLC) for designing and developing the 3D animation builder. There are three processes as shown in Figure 3.1.

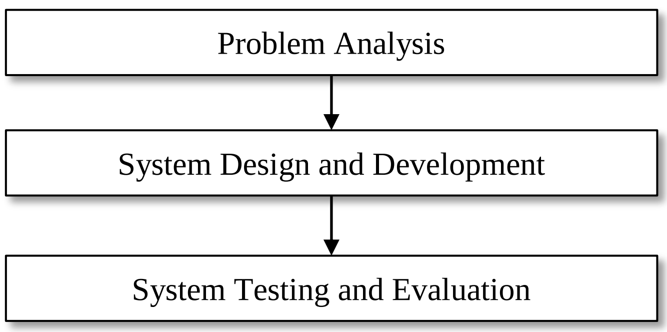


Figure 3.1 Research methodology

3.1.1 Problem Analysis

Problem analysis is performed by surveying an existing automatic 3D animation builder in the field of musical instrument. Then, many factors are analyzed for improving and integrating the purposed animation builder. There are three main surveyed factors as follows:

3.1.1.1 Musical Feature Extraction

This first factor consists of four parts: 1) Displaying data of harmony (i.e., chord, key, notes, and score), 2) Displaying strumming patterns and fingerstyles for strumming, 3) Displaying fingerstyles for touching chord, and 4) Displaying fingerstyles for changing chords.

3.1.1.2 Functions and Techniques of Animation Displaying

This second factor consists of four parts: 1) Displaying a virtual musician or a character animation, 2) Displaying a virtual musical instrument animation 3) Controlling the viewpoint of animation, and 4) Creating an animation by using a motion capture machine.

3.1.1.3 System Flexibility

This third factor consists of two parts: 1) Allowing users to import MIDI files into the system and 2) Allowing users to import MP3 files into the system.

As previously discussed in Chapter 1, a comparison of the existing builders of string instrument playing is illustrated in Table 1.1.

3.1.2 System Design and Development

The system design and development is based on the problem analysis as previously mentioned. The result of system design is shown in a framework of the automatic 3D animation builder consisting of three components which are user interface, Ukulele playing information recognition, and 3D animation construction as depicted in Figure 3.2.

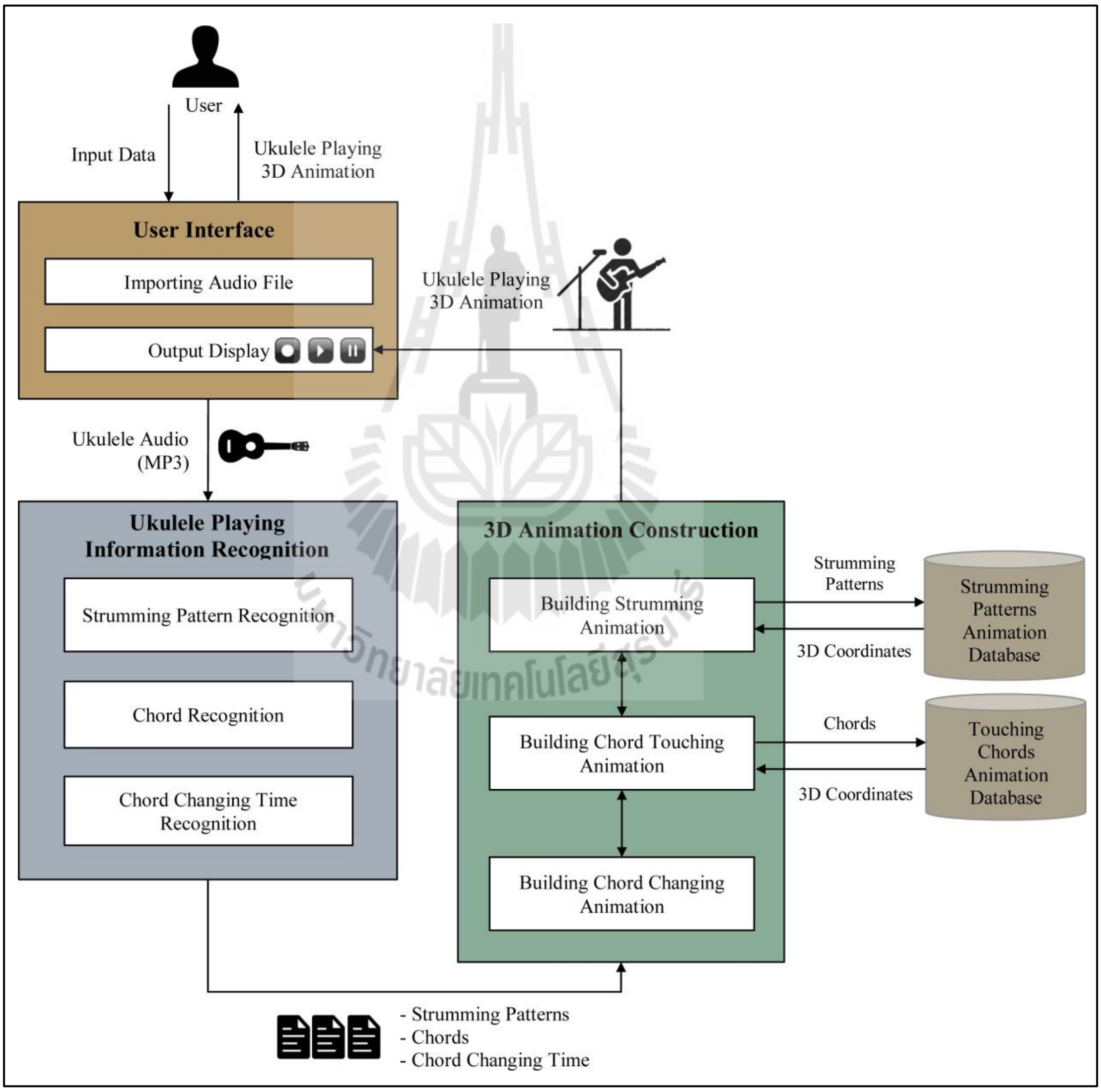


Figure 3.2 The framework of an automatic 3D animation builder for displaying ukulele playing

Figure 3.2 shows the framework of an automatic 3D animation builder for displaying ukulele playing consisting of three main stages as follows:

3.1.2.1 User Interface

The user interface consists of two sub-stages as follows:

1) Importing an Audio File: This is the sub-stage, which allows users to import audio files into the system. The importing in this stage supports only a MPEG-1 audio layer 3 (MP3) file format. Then, the MP3 file will be extracted for displaying an ukulele playing animation in the next step.

2) Displaying a 3D Animation: This is the sub-stage for displaying ukulele playing of a 3D musician via computer display. Media format that appears to users consists of sound, a 3D animation, chord information, and strumming pattern information. Moreover, users can manually control the display of animation by using a computer mouse or the control buttons that appear on a screen.

3.1.2.2 Ukulele Playing Information Recognition: After the numerical data extraction, those data are processed to recognize three types of ukulele playing information as follows:

1) Strumming Pattern Recognition: Strumming pattern is playing styles by using the dominant hand while the another hand holds down chords on the ukulele fretboard. Strumming is in the form of rhythm and the most essential aspect of rhythm is consistency and good timing (Kummong, 2013). Commonly, one song uses only one strumming pattern that depends on a song rhythm. The easiest way to comprehend a strumming pattern is counting “1&/2&/3&/4&” per round of strumming while playing the ukulele in the form of rhythm. The basic pattern consists of two strumming types that are down (d) and up (u). Strumming down is a use of hand for playing a string from top string to bottom string and strumming up is an opposite direction. The example of a basic strumming pattern is “d-/du/d-/du”.

In addition, the advanced strumming technique is interested in the chunking style. Chunking (x) is muting the strings by covering the strings on the ukulele’s neck with a player’s hand. This technique is hard to play because players must have a good skill in strumming. Players can join the chunking style to play the ukulele with strumming down and up such as “d-/du/xu/du”, “d-/xu/-u/xu” .

The overall process of strumming pattern recognition is divided into five stages consisting of 1) data preparation, 2) attribute extraction, 3) model construction for strumming type prediction, 4) strumming type prediction, and 5) strumming pattern summarization as shown in Figure 3.3.

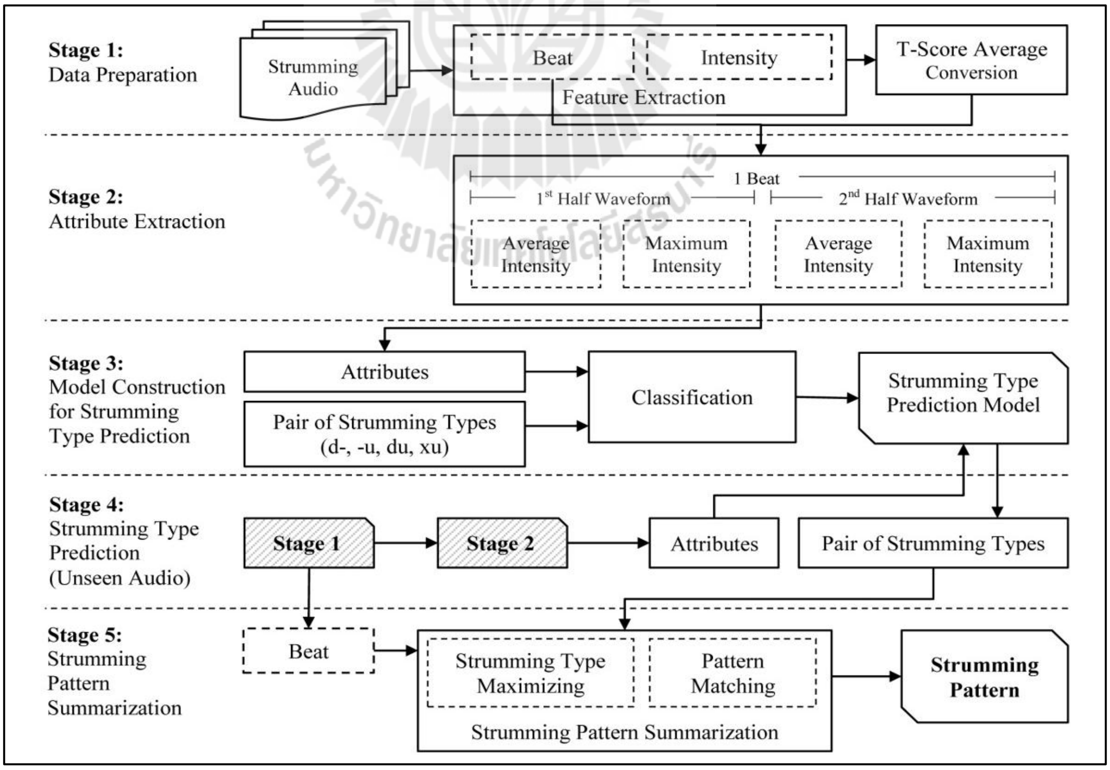


Figure 3.3 The process of strumming pattern recognition

1.1) Stage 1: Data preparation, 15 ukulele songs in various styles of strumming patterns are selected to use for the attribute extraction as shown in Table 3.1.

Table 3.1 The data preparation for the attribute extraction using 15 ukulele songs

Song No.	Strumming Pattern								Number of Pairs of Strumming Types			
	1	**&**	**2**	**&**	**3**	**&**	**4**	**&**	**d-**	**du**	**-u**	**xu**
1	d	-	d	-	d	-	d	-	16	-	-	-
2	d	u	d	u	d	u	d	u	-	16	-	-
3	d	-	d	u	d	-	d	u	8	8	-	-
4	d	-	d	-	d	u	d	u	8	8	-	-
5	d	-	d	u	d	u	d	u	4	12	-	-
6	d	-	d	-	d	u	d	-	12	4	-	-
7	d	u	d	-	d	u	d	-	8	8	-	-
8	d	-	d	u	-	u	d	-	8	4	4	-
9	d	-	d	u	-	u	d	u	4	8	4	-
10	d	-	x	u	d	-	x	u	34	-	-	34
11	d	-	x	u	-	u	x	u	9	-	9	18
12	d	-	x	u	d	-	x	u	8	-	-	8
13	d	u	x	u	d	u	x	u	-	18	-	18
14	d	-	x	u	-	u	x	u	7	-	7	14
15	d	u	x	u	d	u	x	u	-	8	-	8
Total (344)									**126**	**94**	**24**	**100**

Each ukulele song has only one strumming pattern containing four strumming counting “1&/2&/3&/4&” and each song can be strummed indefinitely that depends on the length of the song. One strumming counting comprises one pair of strumming types are d-, du, -u or xu. Table 3.1 shows the strumming pattern style of each song and the number of pairs of strumming types in each the song.

All the pairs of strumming types are imported into a sound analysis software for visualizing waveform and measuring sound intensity. The average waveforms of four pairs of strumming types are very different as shown in Figure 3.4

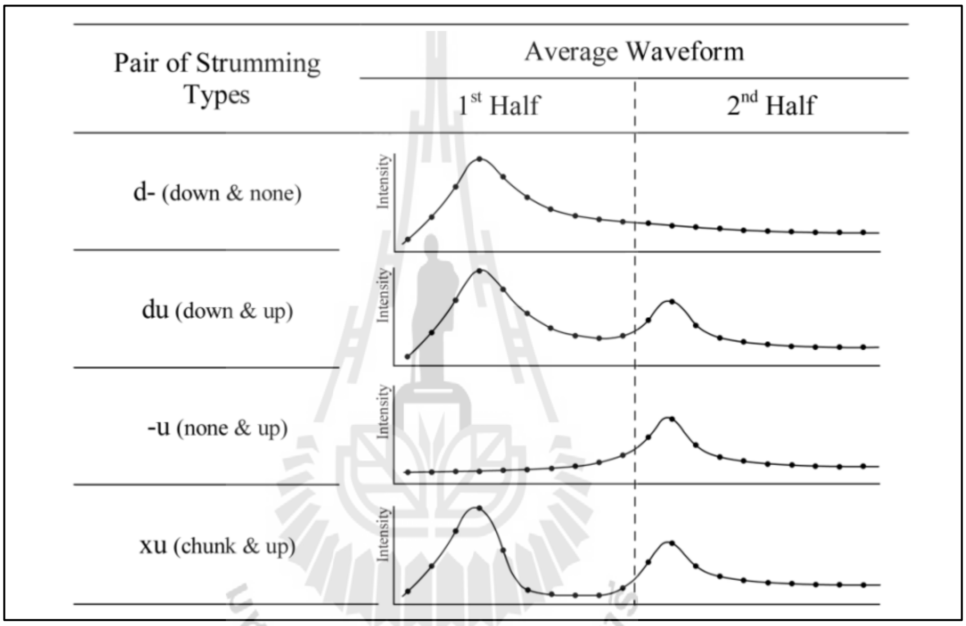


Figure 3.4 The average waveform of four pairs of strumming types

Figure 3.4 shows the sound intensity levels in vertical and horizontal directions which are separated into two parts (1st half and 2nd half) for displaying the curve's appearance between a pair of strumming types. These sound intensity levels are extracted as features for the process of attribute extraction in the next stage. There are 10 sound intensity levels extracted in each half of average waveform and they are transformed from a linear scale to a percentile scale (0 - 100) using T-score as defined in Equation 3.1.

$$T\ Score = 50 + 10Z \tag{3.1}$$

Where $Z = \frac{X - \bar{X}}{S.D.}$

X = Sound intensity level

$\bar{X}$ = Average of a set of sound intensity level

$S.D.$ = Standard deviation of a set of sound intensity level

This transformation is to adjust the different sound intensity of various ukulele songs into the same standard scale as illustrated in Table 3.2. Sound intensity levels in this table are real data of an ukulele song where the number of the sound intensity levels depends on each ukulele song's length.

Table 3.2 An example of the transformation of sound intensity level from a linear scale to a T-score

Data No.	**Sound Intensity Level**		**Data No.**	**Sound Intensity Level**	
	Linear	**T-Score (%)**		**Linear**	**T-Score (%)**
1	483.78	62.43	9	358.40	56.53
2	605.24	68.15	10	295.70	53.57
3	511.54	63.74	11	315.61	54.51
4	390.92	58.06	12	343.94	55.84
5	426.49	59.73	13	278.14	52.75
6	426.68	59.74	14	257.17	51.76
7	362.05	56.70	⋮	⋮	⋮
8	372.96	57.21	2,693	119.08	45.26

1.2) Stage 2: Attribute extraction, this stage aims to extract data attributes for strumming type prediction. The average waveforms as shown in Figure 3.4 are dissimilar curve of each strumming type's pair. Consequently, the most important attribute in this stage is the characteristics of sound intensity levels in the 1st half and 2nd half of the average waveform of each pair. The sound intensity levels of each strumming type's pair are employed to extract four attributes consisting of 1) Average of 1st half, 2) Maximum of 1st half, 3) Average of 2nd half, and 4) Maximum of 2nd half. An example of attribute extraction is shown in Figure 3.5.

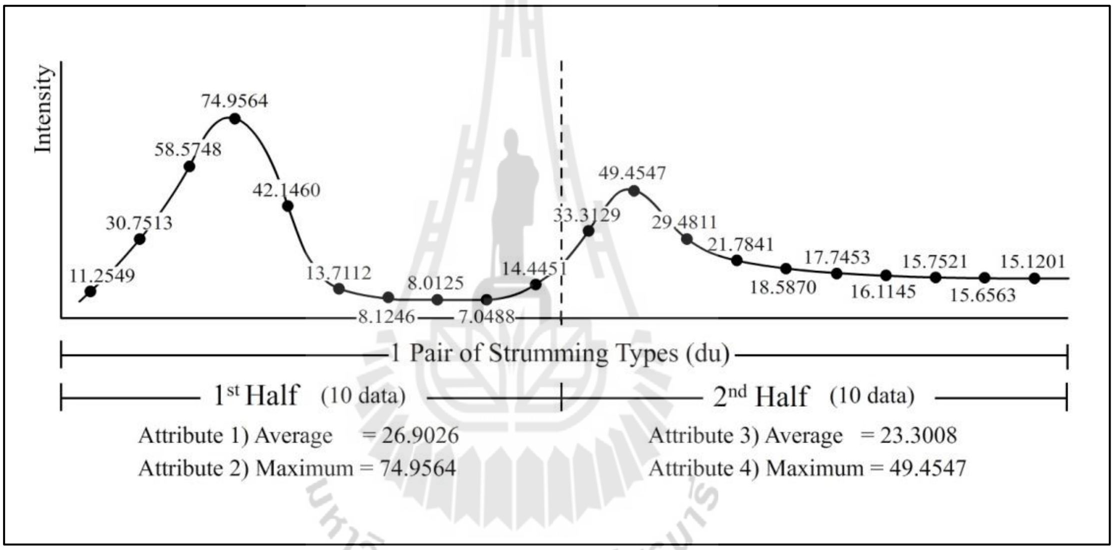


Figure 3.5 An example of attribute extraction

1.3) Stage 3: Model construction for strumming type prediction, 15 ukulele songs for the attribute extraction can be extracted to 344 datasets and all datasets are used as training data as illustrated in Table 3.3. Then the machine learning technique is employed.

Table 3.3 An example of training data for strumming type prediction

Dataset	Attribute* (Input)				Pair of Strumming Types (Output)
	AVG1	MAX1	AVG2	MAX2	
1	72.4805	98.8001	51.0328	60.2593	d-
2	56.4545	75.7346	46.2587	53.8033	du
3	48.1698	67.4124	45.4046	55.4235	du
4	33.5291	34.8841	43.7607	49.6384	-u
5	56.4041	75.1724	50.9940	58.4398	xu
6	66.1661	94.6571	46.6018	51.4527	d-
7	57.1425	81.9701	50.6154	69.3571	du
8	48.2853	82.0156	46.7202	49.6332	xu
9	57.4090	80.9542	47.0900	69.4564	du
10	42.9438	58.3466	47.2499	56.8192	xu
11	63.5848	93.0759	46.3067	74.7470	du
12	46.9386	78.5754	48.7349	52.4342	xu
13	45.8651	51.4009	60.8608	73.3317	-u
14	45.4893	49.8378	59.5611	71.1178	-u
⋮	⋮	⋮	⋮	⋮	⋮
344	63.9996	99.1127	49.5158	56.5196	xu

* *AVG1 = Average intensity of* 1^{st} *half, MAX1= Maximum intensity of* 1^{st} *half*
AVG2 = Average intensity of 2^{nd} *half, MAX2= Maximum intensity of* 2^{nd} *half*

The machine learning approaches are applied for predicting the pair of strumming types consisting of decision tree learning, neural network learning, and hidden Markov model. Prediction model validation uses 10-fold cross validation on the training dataset. The models are appraised using retrieval performance measures, which are Precision, Recall, and F-Measure (Galitsky, 2013). The validation results of each prediction model are shown in Table 3.4 – 3.6, respectively.

Table 3.4 The validation results of the strumming type prediction model using 10-fold cross validation constructed by the decision tree learning

Validation Measurement	**Weighted Average (%)**				
	d-	**du**	**-u**	**xu**	**Avg.**
Precision	75.40	77.30	82.40	80.00	**77.70**
Recall	77.80	79.80	58.30	80.00	**77.60**
F-measure	76.60	78.50	68.30	80.00	**77.50**

Table 3.4 reveals that the prediction model using 10-fold cross validation designed by the decision tree learning is achieved in 77.70 per cent of Precision, 77.60 per cent of Recall, and 77.50 per cent of F-measure by average.

Table 3.5 The validation results of the strumming type prediction model using 10-fold cross validation constructed by the neural network learning

Validation Measurement	**Weighted Average (%)**				
	d-	**du**	**-u**	**xu**	**Avg.**
Precision	76.30	83.50	76.20	89.20	**82.00**
Recall	84.10	80.90	66.70	83.00	**81.70**
F-measure	80.00	82.20	71.10	86.00	**81.70**

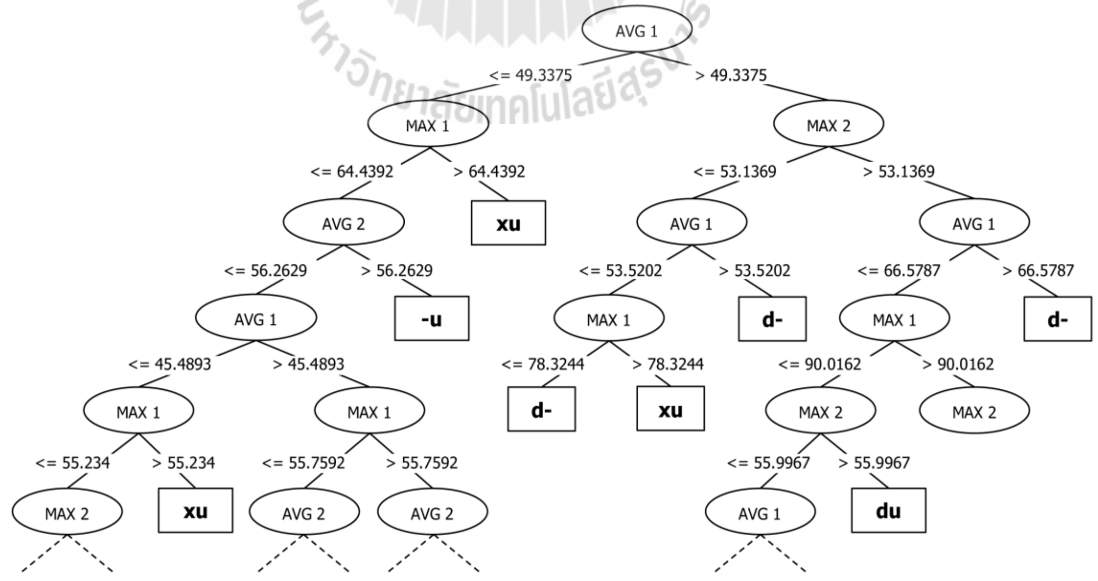
Table 3.5 reveals that the prediction model using 10-fold cross validation designed by the neural network learning is achieved in 82.00 per cent of Precision, 81.70 per cent of Recall, and 81.70 per cent of F-measure by average.

Table 3.6 The validation results of the strumming type prediction model using 10-fold cross validation constructed by the hidden Markov model

Validation Measurement	Weighted Average (%)				
	d-	**du**	**-u**	**xu**	**Avg.**
Precision	75.60	84.30	58.80	93.30	**81.90**
Recall	78.60	79.80	83.30	84.00	**80.80**
F-measure	77.00	82.00	69.00	88.40	**81.10**

Table 3.6 reveals that the prediction model using 10-fold cross validation designed by the hidden Markov model is achieved in 81.90 per cent of Precision, 80.80 per cent of Recall, and 81.10 per cent of F-measure by average.

The results of model construction using three different approaches are shown in Figure 3.6 – 3.8, respectively. The output of this stage is to construct the strumming type prediction model.



**some parts of 26 rules*

Figure 3.6 The prediction model of the decision tree learning

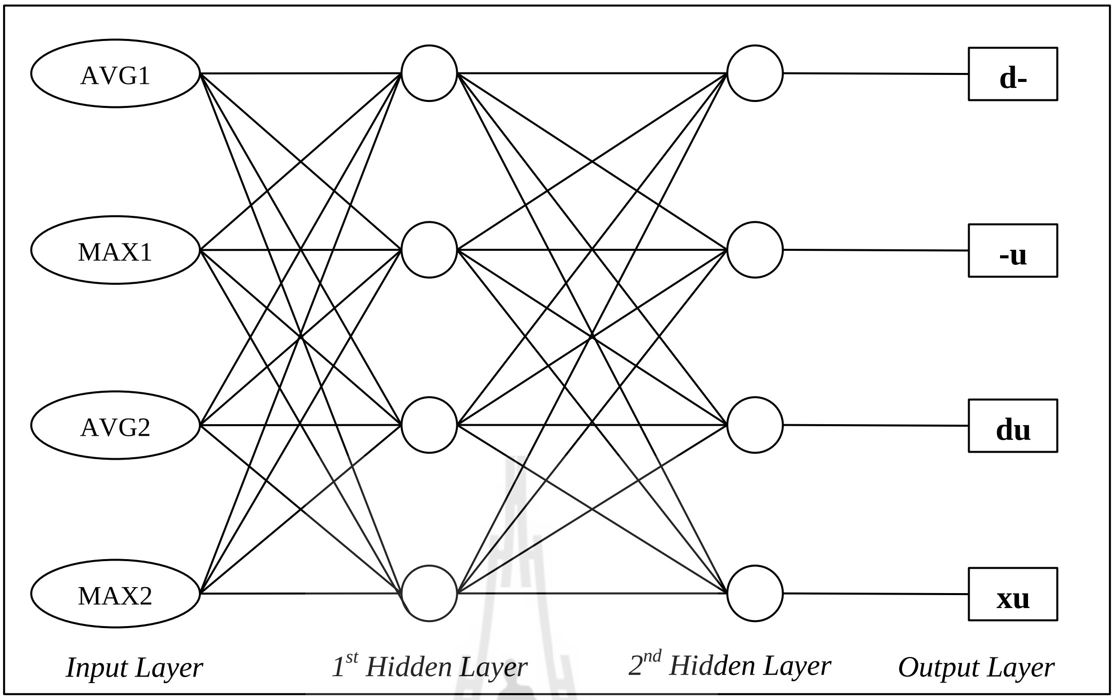


Figure 3.7 The prediction model of the neural network learning

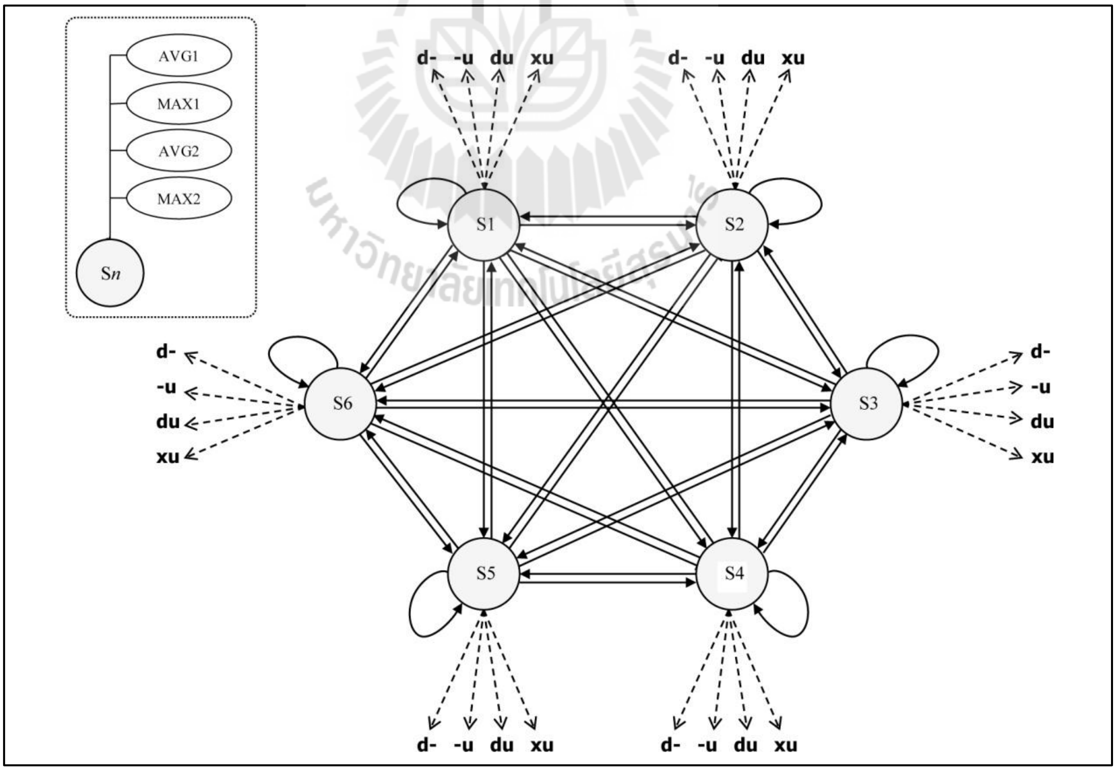


Figure 3.8 The prediction model of the hidden Markov model

1.4) Stage 4: Strumming type prediction, this stage performs for predicting the strumming type of the unseen ukulele audio. Before starting this stage, the unseen ukulele audio will be passed through the processes of stage 1 and stage 2 for extracting attributes. After that, all attributes will be predicted the pairs of strumming types by using the prediction model, which is constructed in the stage 3. All received strumming type's pairs will be used to summarize a proper strumming pattern of the unseen ukulele song in the next stage.

1.5) Stage 5: Strumming pattern summarization, the results from the previous stage are pairs of the strumming types, which are occurred in each strumming counting. However, the predicted strumming type cannot decide that the recognized ukulele song has any strumming pattern because each song has a lot of strumming types. Consequently, this stage is performed for summarizing a strumming pattern of the song. The summarization process can be divided into two steps:

1.5.1) Strumming Type Maximizing: Strumming type maximizing is the calculation of a maximum strumming type of strumming counting in every round of one song. The strumming types which are predicted in Stage 4 can be arranged into a pattern of counting "1&/2&/3&/4&", as shown in Table 3.4.

Table 3.7 The maximum strumming type of strumming counting pattern

Round	Strumming Counting							
	1	**&**	**2**	**&**	**3**	**&**	**4**	**&**
1	-	u	d	u	x	u	d	u
2	-	u	x	u	x	u	d	u
3	x	u	d	u	d	u	d	u
4	-	u	d	u	x	u	d	u
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Maximum Type	**-**	**u**	**d**	**u**	**x**	**u**	**d**	**u**

In Table 3.7, the bottom row is the maximum type (-, u, d, u, x, u, d, u) of strumming counting in every round. It is calculated by counting a type in each counting column. Then, the set of sequenced maximum types will be correlated with the 20 most useful strumming patterns applied from ukulele-tabs.com (Ukukele-Tabs, www, 2013), as shown in Table 3.8.

Table 3.8 The 20 most useful strumming patterns

Pattern No.	Strumming Pattern							
	1	**&**	**2**	**&**	**3**	**&**	**4**	**&**
1	d	-	d	-	d	-	d	-
2	d	u	d	u	d	u	d	u
3	d	-	d	u	d	-	d	u
4	d	-	d	-	d	u	d	u
5	d	-	d	u	d	u	d	u
6	d	-	d	-	d	u	d	-
7	d	u	d	-	d	u	d	-
8	d	-	d	u	-	u	d	-
9	d	-	d	u	-	u	d	u
10	x	u	x	u	x	u	x	u

Pattern No.	Strumming Pattern							
	1	**&**	**2**	**&**	**3**	**&**	**4**	**&**
11	d	u	x	u	d	u	x	u
12	d	-	d	u	x	u	d	u
13	d	u	-	u	d	-	d	u
14	d	u	x	u	x	u	d	u
15	d	u	d	u	-	u	d	u
16	d	u	x	u	x	u	x	u
17	d	-	d	-	x	u	-	u
18	d	-	d	-	x	u	d	u
19	d	-	d	-	-	u	d	u
20	d	u	-	u	-	u	-	u

** d = strumming down, u = strumming up, x = chunking, and (-) = mute*

However, when compared with those 20 strumming patterns, if there are no strumming patterns similar to the set of sequenced maximum types, the strumming pattern for ukulele playing of the sample song is ambiguous. Thus, the pattern matching is proceeded in the next step.

1.5.2) Pattern Matching: The pattern matching is a process of strumming pattern checking. Similarity score computation is used when strumming type maximization cannot decide that the recognized ukulele song has any strumming pattern. This pattern matching technique will estimate only one possible pattern from 20 patterns by measuring the similarity score between the ambiguous set of maximum types and each strumming pattern in the 20 patterns. The similarity score is counting the number of matched types (maximum = 8) and the number of matched type's pairs (maximum = 4). Then both the numbers are combined and converted into a percentage by using Equation 3.2.

$$\text{Summation of similarity score (\%)} = \left(\frac{T+P}{12}\right) \times 100 \qquad (3.2)$$

Where T = Similarity score of strumming type (maximum = 8)

P = Similarity score of strumming type's pairs (maximum = 4)

The strumming pattern with the maximum percentage will be selected as the strumming pattern of the ukulele song. An example of similarity score computation is shown in Table 3.9.

Table 3.9 An example of similarity score computation

The Ambiguous Set of Maximum Type: -, u, d, u, x, u, d, u											
Pattern No.	**Strumming Counting**								**Similarity Score**		
	1	**&**	**2**	**&**	**3**	**&**	**4**	**&**	**Type (8)**	**Pair of Types (4)**	**Summation (12)**
1	d	-	d	-	d	-	d	-	2	0	2 (16.67 %)
2	d	u	d	u	d	u	d	u	6	2	8 (66.67 %)
3	d	-	d	u	d	-	d	u	4	2	6 (33.33 %)
4	d	-	d	-	d	u	d	u	4	1	5 (41.67 %)
5	d	-	d	u	d	u	d	u	5	2	7 (58.33 %)
6	d	-	d	-	d	u	d	-	3	0	3 (25.00 %)
7	d	u	d	-	d	u	d	-	4	0	4 (33.33 %)
8	d	-	d	u	-	u	d	-	4	1	5 (41.67 %)
9	d	-	d	u	-	u	d	u	5	2	7 (58.33 %)
10	x	u	x	u	x	u	x	u	5	1	6 (50.00 %)
11	d	u	x	u	d	u	x	u	4	0	4 (33.33 %)
12	**d**	**-**	**d**	**u**	**x**	**u**	**d**	**u**	**6**	**3**	**9 (75.00 %)** ✓
13	d	u	-	u	d	-	d	u	4	1	5 (41.67 %)
14	d	u	x	u	x	u	d	u	6	2	8 (66.67 %)
15	d	u	d	u	-	u	d	u	6	2	8 (66.67 %)
16	d	u	x	u	x	u	x	u	5	1	6 (50.00 %)
17	d	-	d	-	x	u	-	u	4	1	5 (41.67 %)
18	d	-	d	-	x	u	d	u	5	2	7 (58.33 %)
19	d	-	d	-	-	u	d	u	4	1	5 (41.67 %)
20	d	-	x	u	d	u	x	u	3	0	3 (25.00 %)

In Table 3.9, the ambiguous set of maximum types (-, u, d, u, x, u, d, u) received from the stage of strumming type prediction is used as the example of similarity score computation. The ambiguous set does not match with any strumming pattern in these 20 patterns, therefore a similarity score is computed. As shown in Table 3.9, a grey cell is marked when the maximum types in the ambiguous set match with strumming types in those 20 patterns, and then the percentages of similarity scores are computed. The results show that the maximum similarity scores occurs in pattern number 12 (75.00 per cent). Hence, the strumming pattern of the predicted song is d-/du/xu/du.

2) Chord Recognition: Chord is a set of notes which is simultaneously played to create harmony. Chords add texture to a melody and provide rhythm to a song. Datasets are analyzed to obtain a chord label. There are 184 chords for playing ukulele (Ukukele-Tabs, www, 2013) and each song can have unlimited chords.

Chord recognition is performed by a Non-Negative Least Squares (NNLS) Chroma and Chordino (Mauch and Dixon, 2010), which are an open source Vamp plugin library for harmony and chord extraction. Chordino prepares a simple chord transcription based on NNLS Chroma. NNLS Chroma and Chordino are achieved in very good results of 80 per cent accuracy using the song collection and metric chord detection tasks. The process of chord recognition is started by importing a song into Sonic Visualiser software which installs NNLS Chroma and Chordino plugin. The example of chord recognition information recognized by the NNLS Chroma and Chordino plugin is shown in Table 3.10.

Table 3.10 shows the example of ukulele chord recognition information by using NNLS Chroma and Chordino plugin which is achieved in 100 per cent of accuracy. Moreover, this stage calculates the duration for playing each chord. The purposed results are linked with a 3D virtual musician to show the posture of hands and fingers during ukulele playing. The duration makes a realistic animation.

Table 3.10 An example of Information obtained from ukulele chord recognition

No.	Chord	Start Time	End Time	Duration (Seconds)
1	C	0.05	3.25	3.20
2	G	3.25	5.94	2.69
3	Am	5.94	12.77	6.83
4	F	12.77	15.25	2.48
⋮	⋮	⋮	⋮	⋮
102	D	206.34	214.35	8.01

3) Chord Changing Time Recognition: After the chord recognition is applied, the purposed 3D animation builder is necessary to receive a chord changing time in order to create a smooth and realistic ukulele playing in the 3D animation construction process. This section aims to develop a method for chord changing time recognition, which is an approach to recognize the movement duration of the left hand's finger while changing a chord on the ukulele strings. The final output will be used to generate the animation of the left hand's finger while playing a chord on the ukulele strings.

The experiment is based on two presumptions: 1) Chord changing time seems to reverse with song tempo and 2) the same connected chords should take the same chord changing time in any song.

From the first presumption, song tempo is the considered element about ukulele chord changing. That is chord changing time seems to reverse with song tempo. For example, a slow tempo song should take longer chord changing time than a fast tempo song. Hence, the recognition solution is operated for verifying this prediction. The simple way to understand chord changing time, a location of chord changing in any song is conceptualized as shown in Figure 3.9.

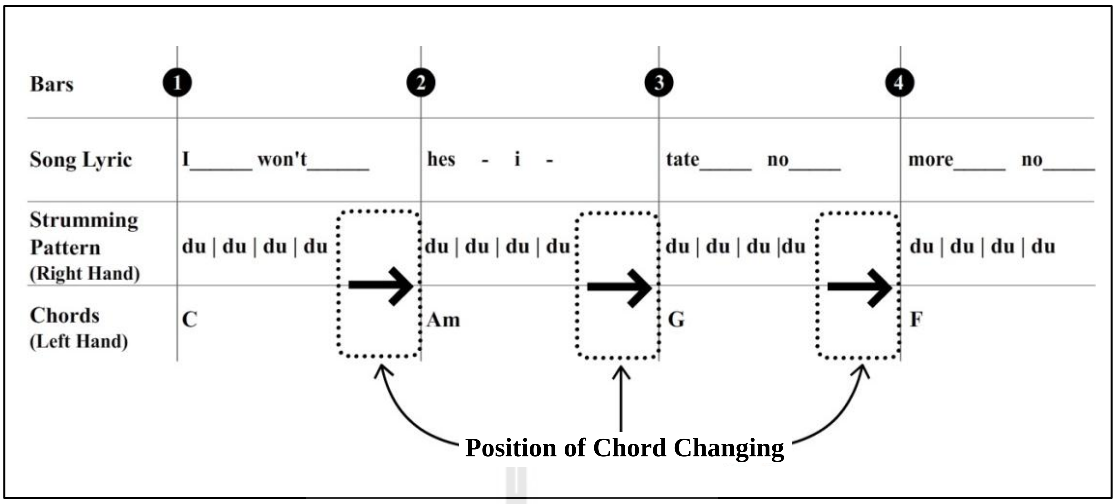


Figure 3.9 A position of chord changing

The overall procedure of chord changing time recognition is divided into three stages consisting of 1) feature extraction, 2) attribute extraction, and 3) chord changing time prediction as shown in Figure 3.10.

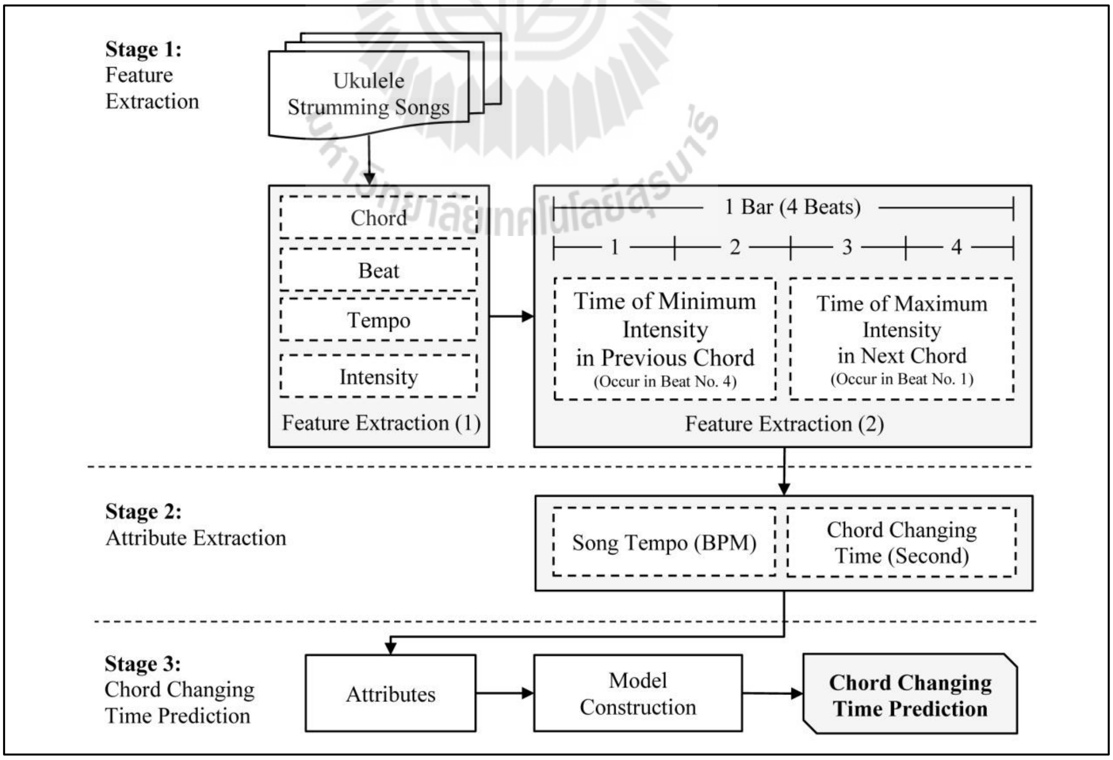


Figure 3.10 The framework of chord changing time recognition

3.1) Stage 1: Feature Extraction, at this beginning stage, 22 ukulele songs with various tempos from 80 to 160 beats per minute (BPM) and diverse strumming patterns are selected to use as sample songs. The four examined features consist of chord, beat, tempo, and intensity as follows:

3.1.1) Chord: Chord recognition is performed by a Non-Negative Least Squares (NNLS) Chroma and Chordino (Mauch and Dixon, 2010) as previously described in section 2.

3.1.2) Beat: Beat is the flow of sound created by the arrangement of stressed and unstressed syllables in accentual verse (Leinecker, 1994). The beat is directly related to time in music. The best way to perceive a beat is loop counting from 1 to 4 in a regular speed. For example, when foot tapping while listening to the music, the beat of the music is kept. In this stage, beat tracker is used to display a segmentation line in the form of loop counting.

3.1.3) Tempo: Tempo is the speed of the beat. It can be represented by the amount of beats in a minute. However, sample ukulele songs are played by general musicians without a metronome, which is a machine that marks time at a selected rate by giving a regular tick. Therefore, tempo data in a whole song is not regular rate. Tempo tracker is performed for measuring the amount of beats in a period of time (Davies and Plumbley, 2007). This tempo rate is used together with chord changing time in the prediction stage (Stage 3).

3.1.4) Intensity: Intensity is the loudness level of a sound. It can be analyzed by using Scale Power Slope plugin on Sonic Visualiser (Sapp, 2015). The Scale Power Slope will create the average waveform with the intensity points on a curve. Intensity points are observed by researchers for building the complete feature extraction.

An example waveform with four features as shown in Figure 3.11.

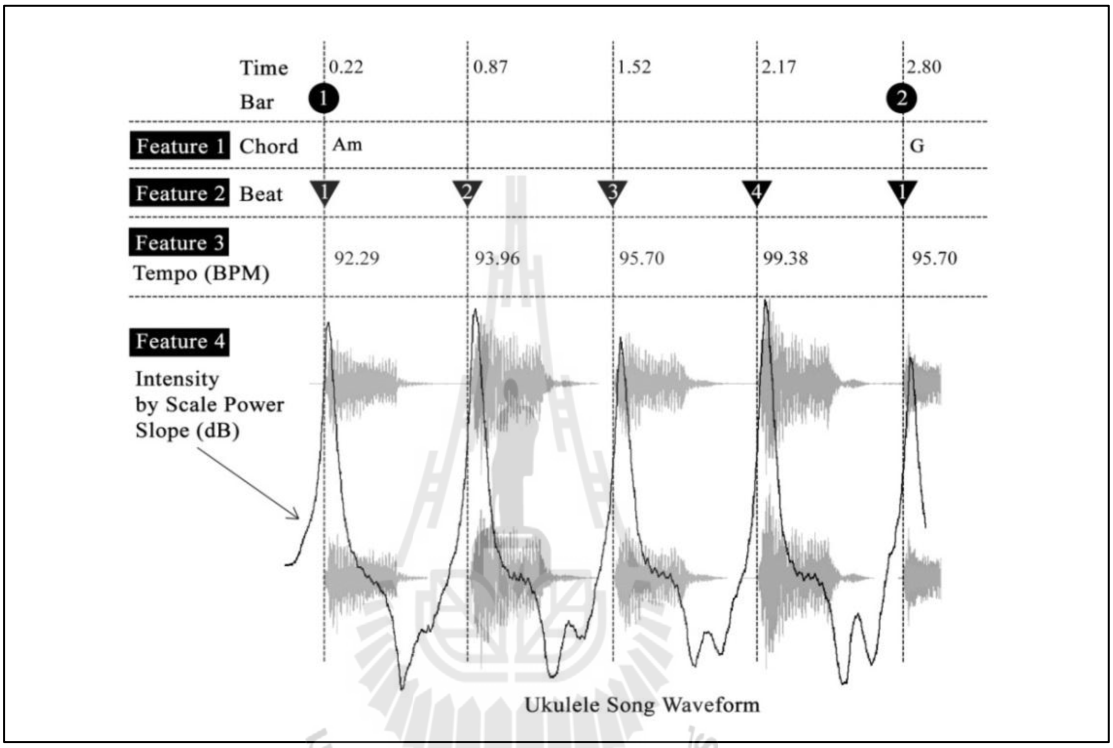


Figure 3.11 An example waveform with four features: Chord, beat, tempo, and intensity

The next step of feature extraction is observing the intensity points on the curve of Scale Power Slope. The related points are divided into two positions, which occur between two connected chords consisting of 1) time point of minimum intensity in the previous chord, and 2) time point of maximum intensity in the next chord as shown in Figure 3.12.

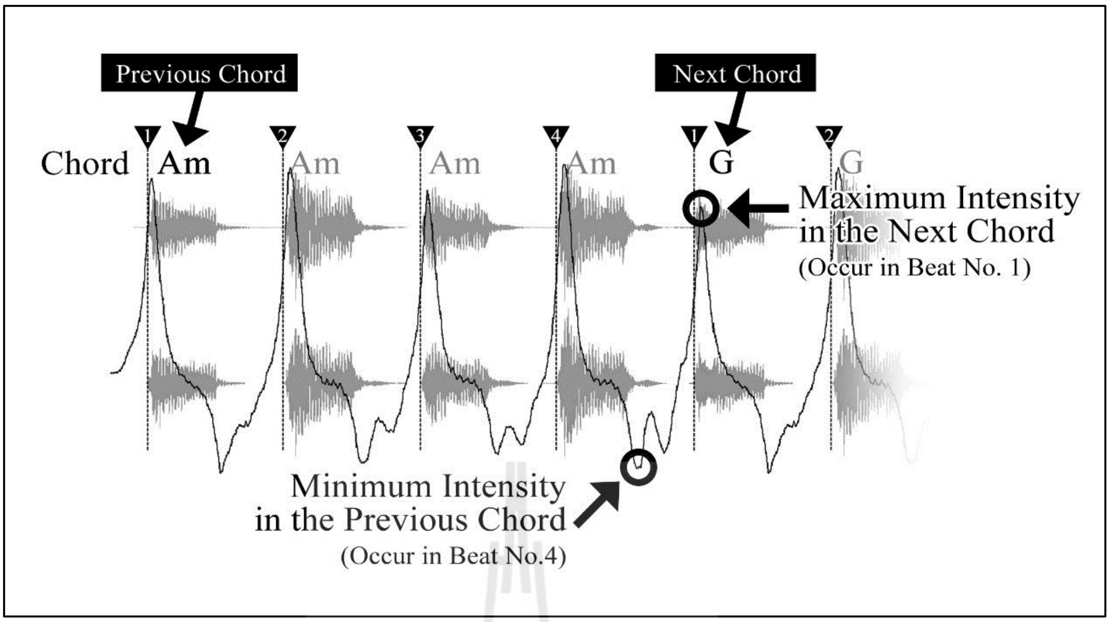


Figure 3.12 An example of minimum and maximum intensity between 2 connected chords

Figure 3.12 shows the time point of minimum intensity in the previous chord that is the end point of touching the string before moving the hand to the next chord and it will occur in the fourth beat counting. Moreover, the time point of maximum intensity in the next chord is shown, it is the first point of strumming the string in new chord and it will occur in the first beat counting. Both of time points will be used to calculate chord changing time later.

3.2) Stage 2: Attribute Extraction, this stage objects to extract data attributes for chord changing time prediction. The important attributes are divided into two parts as follows:

3.2.1) Song Tempo: Beat information from beat tracker in the feature extraction stage are used to song tempo estimation. Song tempo information is estimated by a computing beat average of the whole song. An example of song tempo estimation is shown in Table 3.11.

Table 3.11 An example of song tempo estimation

Data No.	Time Point of Beat Change	Tempo (BPM)
1	0.00	101.33
2	4.46	103.36
3	19.32	105.47
4	20.81	107.67
5	23.78	109.96
6	25.26	107.67
7	28.24	112.35
8	31.21	109.96
9	34.18	103.36
10	35.67	105.47
Average		**106.66**

3.2.2) Chord Changing Time: One song has chord changing in many times. Therefore, the changing time estimation is completed by computing the range between the time point of minimum intensity in the previous chord and the time point of maximum intensity in the next chord. An example of chord changing time calculation is shown in Table 3.12.

Table 3.12 An example of chord changing time calculation

Data No.	Time Point of Minimum Intensity in the Previous Chord	Time Point of Maximum Intensity in the Next Chord	Chord Changing Time Period (Seconds)
1	4.64	5.01	0.37
2	9.38	9.72	0.34
3	14.12	14.49	0.37
4	18.73	19.10	0.37
5	21.30	21.48	0.18
6	23.53	23.99	0.46
7	26.20	26.46	0.26
8	28.43	28.89	0.46
9	30.86	31.30	0.44
10	33.15	33.61	0.46
Average			**0.37**

3.3) Stage 3: Chord Changing Time Prediction, in the previous stage, the average song tempo and the average chord changing time from 22 ukulele songs are used as attributes. An example of datasets is shown in Table 3.13.

Based on the first presumption, 22 datasets from 22 ukulele songs will be used as training data for model construction. Nevertheless, the result shows that chord changing time is not related to song tempo, i.e., a slow tempo song does not take more chord changing time than a fast tempo song in regular sequences. The scatter diagram is visualized to verify appeared result, as shown in Figure 3.13.

Table 3.13 An example of the first presumption experiment datasets

Dataset	Attribute 1: Average Tempo (BPM)	Attribute 2: Average Chord Changing Time Period (Second)
1	82.03	0.33
2	83.37	0.50
3	85.43	0.16
4	88.87	0.12
5	91.48	0.28
6	94.83	0.47
7	95.70	0.32
8	96.91	0.32
9	101.33	0.23
10	102.06	0.37
⋮	⋮	⋮
22	161.31	0.26

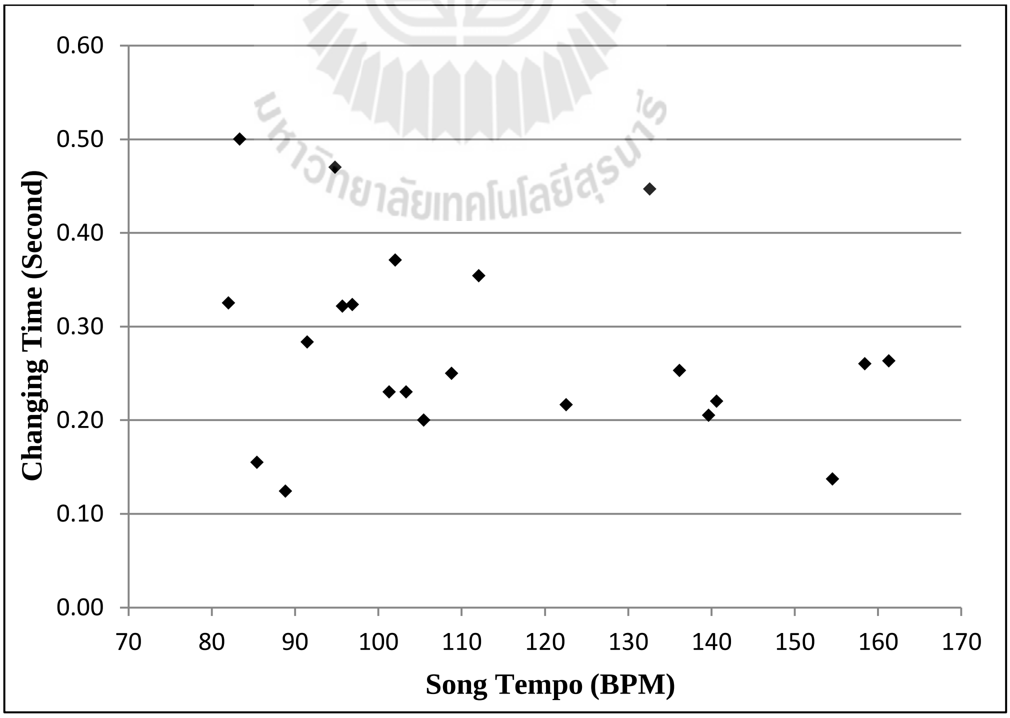


Figure 3.13 The scatter diagram of 22 datasets

Then, the second presumption is proved that is the same connected chords should equally use chord changing time in any song.

The experiment based on the second presumption works by collecting data of chord changing time from five connected chords. The five connected chords which are compiled consist of C to Em, Em to Am, Am to D, F to C, and Am to F. Table 3.14 shows an example of chord changing time data of connected chords between C chord and Em chord, which are extracted from 6 different song tempos.

Table 3.14 shows only one pair of connected chords (C to Em), which appeared in six different song tempos. Each song contains at least five times of chord changing from C to Em consisting of various tempo and chord changing time. An average of tempo and chord changing time is used to be each song's representative. However, this step uses five connected chords (C to Em, Em to Am, Am to D, F to C, and Am to F) but each pair of connected chords can be generated six pairs of the attributes, which are collected from six songs. Therefore, the attributes which are used in the second presumption experiment consisting of 1) Average tempo and 2) Average chord changing time. The datasets of the experiment based on the second presumption as shown is Table 3.15.

Table 3.14 An example of chord changing time data of connected chords between C chord and Em chord

Connected Chords: C to Em			
Data No.	**Song No.**	**Tempo (BPM)**	**Chord Changing Time (Seconds)**
1	1	80.56	0.08
2		80.73	0.08
3		81.23	0.10
4		81.09	0.17
5		80.32	0.08
Avg.		**80.79**	**0.10**
6	2	90.11	0.24
7		90.23	0.33
8		91.65	0.18
9		91.50	0.22
10		90.45	0.38
Avg.		**90.79**	**0.27**
11	3	115.23	0.29
12		113.48	0.05
13		115.33	0.12
14		114.65	0.21
15		114.04	0.23
Avg.		**114.55**	**0.18**
⋮	⋮	⋮	⋮
26	6	155.32	0.12
27		154.67	0.23
28		155.34	0.28
29		154.23	0.34
30		154.87	0.33
Avg.		**154.89**	**0.26**

Table 3.15 An example of the second presumption experiment datasets

Dataset No.	Connected Chords	Attribute 1: Average Tempo (BPM)	Attribute 2: Average Chord Changing Time (Seconds)
1.1	C to Em	80.79	0.10
1.2		90.79	0.27
1.3		114.55	0.18
1.4		121.01	0.25
1.5		142.26	0.24
1.6		154.89	0.26
2.1	Em to Am	90.39	0.56
2.2		92.21	0.27
2.3		113.62	0.35
2.4		121.04	0.26
2.5		146.54	0.32
2.6		153.96	0.22
3.1	Am to D	81.09	0.40
3.2		102.55	0.34
3.3		121.73	0.31
3.4		131.09	0.47
3.5		142.21	0.58
3.6		143.35	0.28
4.1	F to C	82.88	0.40
4.2		92.19	0.28
4.3		110.93	0.50
4.4		134.60	0.36
4.5		146.16	0.38
4.6		154.36	0.42
5.1	Am to F	91.75	0.26
5.1		101.82	0.69
5.3		112.95	0.36
5.4		114.86	0.34
5.5		142.22	0.54
5.6		153.77	0.59

All datasets which are presented in Table 3.15 can be visualized by the scatter diagram as shown in Figure 3.14.

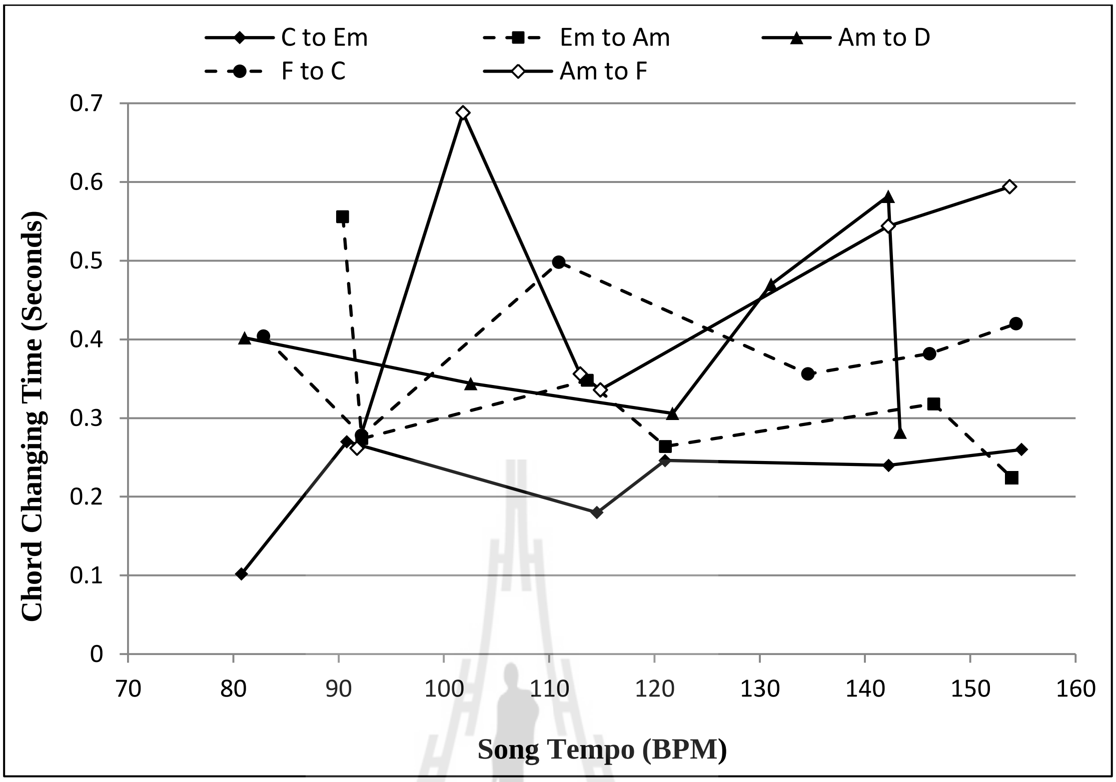


Figure 3.14 The scatter diagram of the second presumption experiment datasets

The scatter diagram in Figure 3.14 visualizes five link lines of each pair of connected chords. Each link line displays the direction of chord changing time, which shows the inconsistency of each pair of connected chords because all lines does not slope in the same direction. This result shows that the same connected chords do not use the same chord changing time in different songs.

As discussed in Figure 3.13 and Figure 3.14, the chord changing time is not depended on song tempo and the type of connected chords. However, the main research objective is the development of an automatic 3D animation builder for displaying ukulele playing. Therefore, the solution of chord changing time recognition must be summarized. The result of chord changing time estimation will be presented in the Chapter 4.

3.1.2.3 3D Animation Construction consists of 3 sub-stage as follows:

1) Building a Strumming Animation: This section matches a strumming animation with the best strumming pattern received from the recognition stage. All strumming animations in the database are created by a motion capture system. The motion capture system captures the musician's right hand while strumming.

The motion capture system uses high definition cameras for following the movement of reflective markers which attached to the joints of the actor's fingers as shown in Figure 3.15.

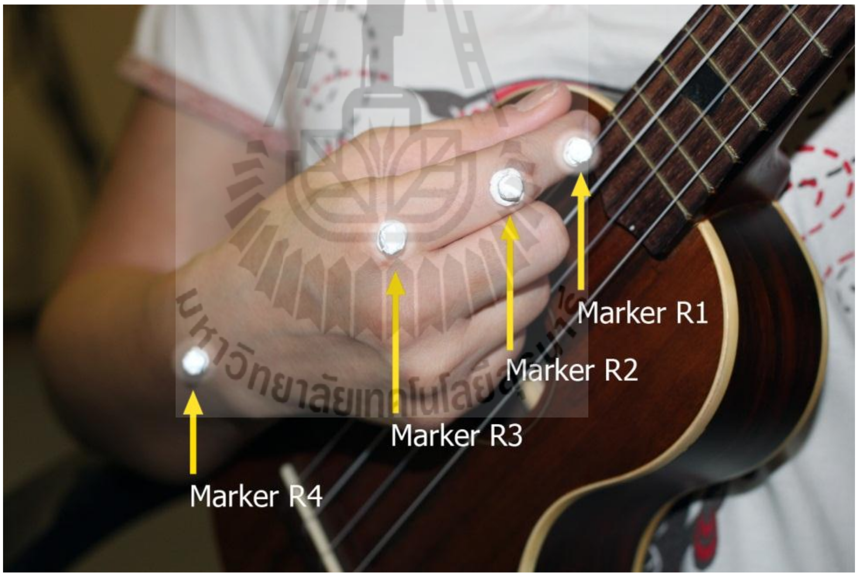


Figure 3.15 Reflective markers on the joints of the musician's finger

Capturing data are stored in three-dimensional coordinates (X, Y, Z) separated by each reflective marker. An example of capturing data is shown in Table 3.16.

Table 3.16 An example of capturing data

Frame	Time	Marker's 3D Coordinates											
		Marker R1			Marker R2			Marker R3			Marker R4		
		X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
1	1.31	88.79	747.21	216.81	65.63	751.04	226.65	32.30	754.46	236.29	-45.48	721.08	215.16
2	1.33	88.34	747.50	218.25	65.24	751.21	227.95	31.86	754.57	237.37	-45.77	721.20	215.73
3	1.35	88.10	747.36	220.05	64.75	751.49	229.47	31.62	754.92	238.90	-46.21	721.35	216.46
4	1.37	87.58	747.58	221.73	64.18	751.58	230.91	30.74	754.86	239.88	-46.62	721.54	217.13
5	1.38	87.03	747.64	223.61	63.73	751.80	232.62	30.44	755.24	241.49	-47.13	721.78	217.81
6	1.40	86.52	747.49	224.95	63.37	751.87	234.72	30.08	755.34	242.95	-47.58	722.00	218.38
7	1.42	86.09	747.22	226.83	62.91	751.67	236.62	29.35	755.16	244.05	-47.96	722.09	218.91
8	1.43	85.72	746.72	228.93	62.42	751.15	238.16	28.65	754.72	245.21	-48.29	722.07	219.41
9	1.45	85.48	745.90	230.90	62.09	751.01	241.01	28.48	754.17	246.79	-48.52	721.98	219.87
10	1.47	85.28	743.99	231.95	61.93	749.24	241.33	28.19	753.30	247.89	-48.71	721.78	220.41
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
175	4.23	80.88	749.89	236.40	57.53	758.08	247.46	23.35	765.11	249.71	-50.87	732.50	218.29

The marker's 3D coordinates will be used to animate a strumming pattern of 3D virtual musician's right hand. This method makes a smooth and realistic animation because all coordinates receive from human's hand movement.

2) Building a Chord Touching Animation: The chord touching is using the non-dominant hand to hold down notes on the fretboard while the dominant hand is playing a string instrument. In this research, the non-dominant hand is the left hand which consists of 20 important points on hand's skeleton. Figure 3.16 shows the bones and joints of the left hand applied from El-Sawah, Georganas, and Petriu (2006).

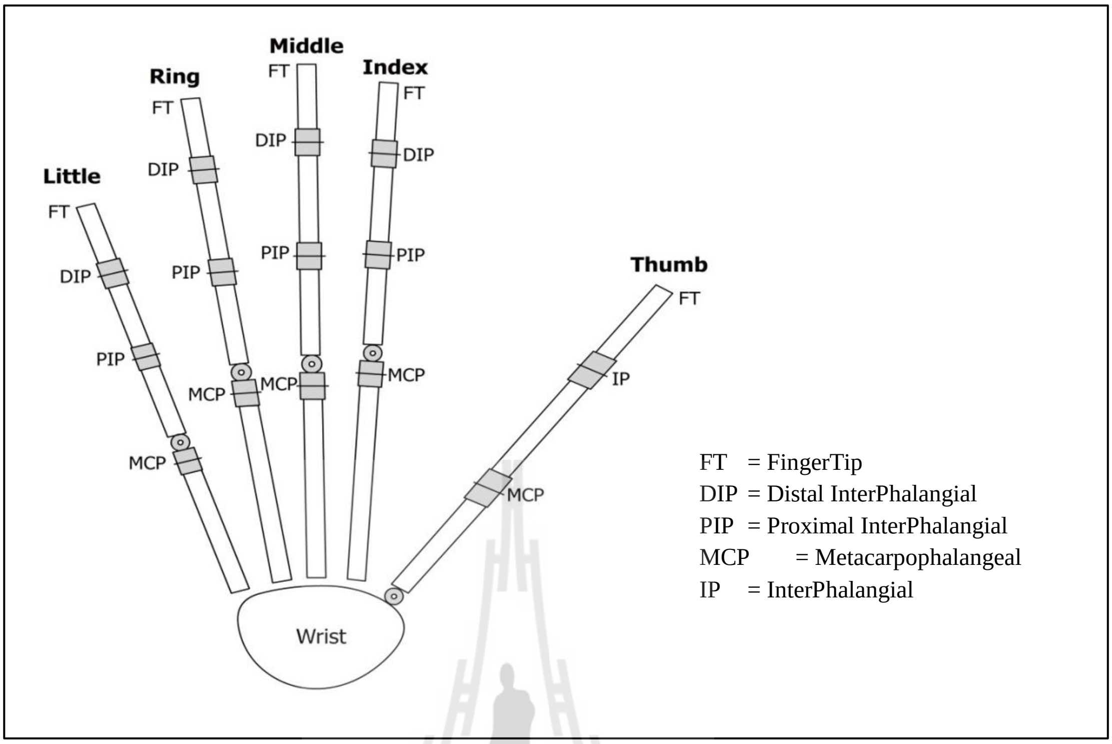


Figure 3.16 Bones and joints of the left hand

Figure 3.16 shows 20 points on the left hand, which is important to touch ukulele chords consisting of 1) Wrist, 2) Thumb-FT, 3) Thumb-IP, 4) Thumb-MCP, 5) Index-FT, 6) Index-DIP, 7) Index-PIP, 8) Index-MCP, 9) Middle-FT, 10) Middle-DIP, 11) Middle-PIP, 12) Middle-MCP, 13) Ring-FT, 14) Ring-DIP, 15) Ring-PIP, 16) Ring-MCP, 17) Little-FT, 18) Little-DIP, 19) Little-PIP, and 20) Little-MCP.

All points on the left hand in each chord from 168 chords will be aligned to the most realistic of the left hand touching. This procedure is performed by the researcher in the system development step powered by the Unity3D software. Figure 3.17 shows the Unity3D screenshot while the researcher aligns the proper coordinate points on the virtual musician's left hand.

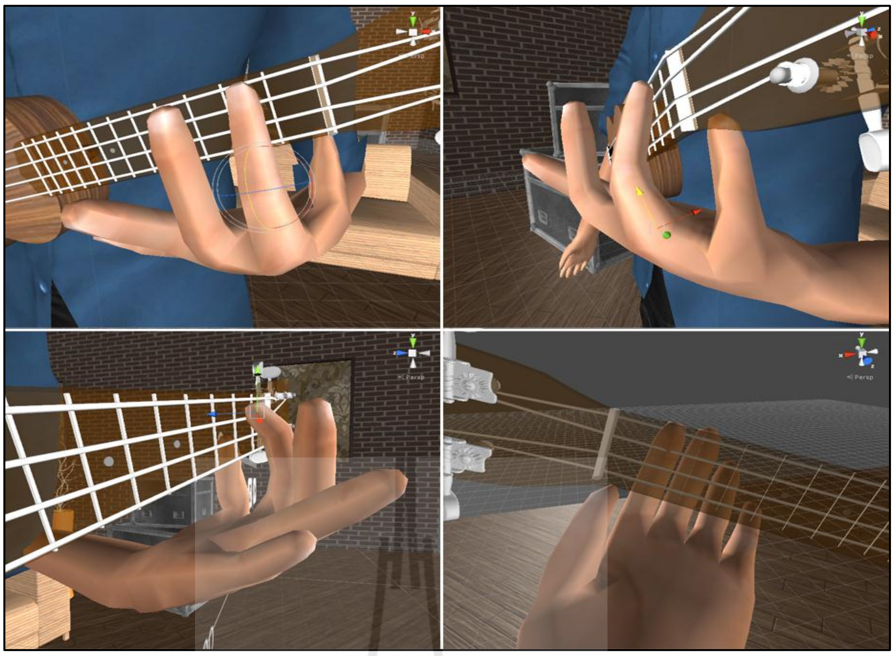


Figure 3.17 Alignment of the proper coordinate points in the Unity3D

An example of 3D coordinate data of chord touching which is received from the Unity3D software is shown in Table 3.17.

Table 3.17 An example of coordinate data of chord touching

Chord: A				
No.	**Point Name**	**3D Coordinates**		
		X	**Y**	**Z**
1	Wrist	111.85	713.63	136.51
2	Thumb-FT	26.36	835.14	139.83
3	Thumb-IP	22.44	822.46	137.76
4	Thumb-MCP	19.36	818.04	133.87
5	Index-FT	101.76	837.26	103.40
6	Index-DIP	98.45	835.65	112.98
7	Index-PIP	84.41	834.92	165.47
8	Index-MCP	93.94	789.86	166.68
9	Middle-FT	52.16	857.29	109.75
10	Middle-DIP	48.76	834.24	137.71

Table 3.17 An example of coordinates data of chord touching (cont.)

Chord: A				
No.	**Point Name**	**3D Coordinates**		
		X	**Y**	**Z**
11	Middle-PIP	45.95	828.36	159.04
12	Middle-MCP	71.37	779.25	164.63
13	Ring-FT	75.68	856.10	143.79
14	Ring-DIP	55.89	843.19	156.03
15	Ring-PIP	28.77	809.10	162.98
16	Ring-MCP	55.83	769.66	151.31
17	Little-FT	47.70	850.64	135.42
18	Little-DIP	22.87	831.55	130.93
19	Little-PIP	12.88	776.37	143.73
20	Little-MCP	43.95	755.75	140.66

3) **Building a Chord Changing Animation:** The chord changing is a movement of the left hand's finger while playing a chord on the ukulele strings. The coordinate data of touching chords received from the previous step will be used for automatically generating a chord changing animation by the Unity3D software. The chord changing time animations are not created by the motion capture system because the number of possible chord changing is very large, i.e., n x (n-1) possible cases where n is the total number of chords. Thus there are 28,056 possible cases of changing chords for 168 ukulele chords.

An example of movement direction and movement distance data is shown in Table 3.18 which are calculated by subtracting 3D coordinates between two contiguous chords. The plus (+) and minus (-) symbols show the movement direction of each point on 3D axis. All information obtained in this step will be indicated by specifying a builder for generating a realistic chord changing animation.

Table 3.18 An example of movement direction and movement distance data between chord A and A#m7

Chord		A			A#m7			Movement Direction and Distance		
No.	Point Name	3D Coordinates			3D Coordinates					
		X	Y	Z	X	Y	Z	X	Y	Z
1	Wrist	111.85	713.63	136.51	43.84	750.78	127.14	**-68.01**	**+37.15**	**-9.37**
2	Thumb-FT	26.36	835.14	139.83	34.14	822.65	98.54	**+7.78**	**-12.49**	**-41.29**
3	Thumb-IP	22.44	822.46	137.76	9.57	842.91	89.52	**-12.87**	**+20.45**	**-48.24**
4	Thumb-MCP	19.36	818.04	133.87	6.98	866.95	84.01	**-12.38**	**+48.91**	**-49.86**
5	Index-FT	101.76	837.26	103.40	-26.57	897.55	104.81	**-128.33**	**+60.29**	**+1.41**
6	Index-DIP	98.45	835.65	112.98	-14.55	883.90	117.27	**-113**	**+48.25**	**+4.29**
7	Index-PIP	84.41	834.92	165.47	-2.38	868.06	130.41	**-86.79**	**+33.14**	**-35.06**
8	Index-MCP	93.94	789.86	166.68	20.92	833.26	147.46	**-73.02**	**+43.4**	**-19.22**
9	Middle-FT	52.16	857.29	109.75	-48.23	899.65	122.28	**-100.39**	**+42.36**	**+12.53**
10	Middle-DIP	48.76	834.24	137.71	-40.25	883.43	132.02	**-89.01**	**+49.19**	**-5.69**
11	Middle-PIP	45.95	828.36	159.04	-26.29	859.42	147.22	**-72.24**	**+31.06**	**-11.82**
12	Middle-MCP	71.37	779.25	164.63	-0.32	815.35	151.26	**-71.69**	**+36.1**	**-13.37**
13	Ring-FT	75.68	856.10	143.79	-78.51	868.45	146.32	**-154.19**	**+12.35**	**+2.53**
14	Ring-DIP	55.89	843.19	156.03	-65.94	855.19	145.20	**-121.83**	**+12**	**-10.83**
15	Ring-PIP	28.77	809.10	162.98	-49.15	832.84	150.59	**-77.92**	**+23.74**	**-12.39**
16	Ring-MCP	55.83	769.66	151.31	-16.63	800.98	143.56	**-72.46**	**+31.32**	**-7.75**
17	Little-FT	47.70	850.64	135.42	-89.94	821.48	130.58	**-137.64**	**-29.16**	**-4.84**
18	Little-DIP	22.87	831.55	130.93	-77.96	811.55	122.24	**-100.83**	**-20**	**-8.69**
19	Little-PIP	12.88	776.37	143.73	-61.01	803.17	131.25	**-73.89**	**+26.8**	**-12.48**
20	Little-MCP	43.95	755.75	140.66	-28.87	786.09	132.19	**-72.82**	**+30.34**	**-8.47**

3.1.3 System Testing and Evaluation

After the system development step, system usability testing is implemented. This research uses a Discount Usability defined by Nielsen (www, 1997) as the system usability testing because this method is more efficient heuristic evaluation of system testing. If the system usability testing is evaluated by 5 users, the testing can find problems more than 85 per cent. If the testing is evaluated by 15 users, it can find overall problems in the system. Therefore, this research tests the system usability with 15 users as details in the next section.

3.2 Population and Sampling

This research assigns the population and sampling as details below.

3.2.1 Population

The population is separated into two parts as follows:

3.2.1.1 Population for Model Construction

1) Model Construction of Strumming Type Prediction: Ukulele songs are played by musicians.

2) Model Construction of Chord Changing Time Recognition: Ukulele songs are played by musicians.

3.2.1.2 Population for Model and System Testing

1) Model Testing of Strumming Type Prediction: A set of ukulele songs are played by musicians.

2) System Usability Testing:

2.1) People for System Usability Testing: People who are interested in ukulele playing with basic computer skills.

2.2) Song for System Usability Testing: Ukulele songs are played by musicians.

3.2.2 Sampling

The sampling is separated into two parts as follows:

3.2.2.1 Sampling for Model Construction

1) Model Construction of Strumming Type Prediction: 15 ukulele songs in various styles of strumming patterns, which are extracted into 344 datasets.

2) Model Construction of Chord Changing Time Recognition: 22 ukulele songs in various chords, which are extracted into 135 datasets.

3.2.2.2 Sampling for Model and System Testing

1) Model Testing of Strumming Type Prediction: 10 ukulele songs for supplying test set.

2) System Usability Testing:

2.1) People for System Usability Testing: 15 people who are interested in ukulele playing with basic computer skills. These people are divided into three groups, five people per group. The first group is experts of ukulele playing with basic computer skills. The second group is experts of system design and development. The third group is persons who are interested in ukulele playing with basic computer skills.

2.2) Song for System Usability Testing: The 20 ukulele songs are based on 20 different strumming patterns, which are various chords and tempos, will be prepared as the test songs.

3.3 Research Instruments

3.3.1 System Development Instruments

3.3.1.1 Hardware: Laptop is used for developing an automatic 3D animation builder for displaying ukulele playing with performances as follows:

- *Processor:* Intel Core i7-2640M 2.80GHz
- *Main memory:* DDR2 4 GB DRAM
- *Hard drive:* 250 GB
- *Internet connection:* Integrated 802.1 1a/b/g
- *Accessories:* keyboard, optical drive, mouse, etc.

3.3.1.2 Software: The operating system and several applications are used for developing an automatic 3D animation builder for displaying ukulele playing which has the competency in developing 3D computer software. The software tools used in this research have specifications as described below:

- *Operating system:* Microsoft Windows 7 Professional
- *Software development:* Unity3D 4.6.8
- *Sound analysis:* Sonic Visualiser
- *Sound editor:* Adobe Audition 2.0

- *3D modeling and animation:* Autodesk Motion Builder 2013 and Autodesk Maya 2013
- *Motion capture:* Cortex 1.0
- *User interface design:* Adobe Photoshop CS5
- *Machine learning tool:* Weka 3.7
- *Web server:* Apache web server 2.2.8
- *Web browser:* Google chrome 37.0.2031.2 dev-m
- *Web development tool:* PHP script language 5.2.6
- *Database management system:* phpMyAdmin 2.10.3
- *Statistic tool:* IBM SPSS Statistics 19

3.3.1.3 Motion Capture System: An optical Motion Capture System is used for capturing ukulele playing. The system consists of eight Raptor-4 digital cameras and Cortex software, which captures complex motion with extreme accuracy.

3.3.2 Evaluation Instruments

3.3.2.1 Model Testing Instruments: The strumming type prediction model is evaluated by using Weka 3.5.8 software (Hall et al, 2009), which is machine learning tools for analyzing the weighted average of Precision, Recall, and F-measure.

3.3.2.2 System Usability Testing Instruments: The system usability testing instruments consist of two types which are a post-study system usability questionnaire and a thinking-aloud protocol.

1) Post-Study System Usability Questionnaire

The post-study system usability questionnaire is used when each user completed the system testing under research environment. All questions in the questionnaire are based on the concept of Software Usability Measurement Inventory or SUMI (Kirakowski and Corbett, 1993). SUMI is a questionnaire for evaluating five question sections. It was primarily developed as a summative instrument which measures a user's perception of the usability of software. The SUMI consists of 50 questions internationally standardized for quantitative measurement in the view of the user. The answer format for the items consists of "agree", "undecided" and "disagree".

Kirakowski and Corbett (1993) describe the topics of the scales which are subdivided into five criteria (10 items per criteria) as follows:

1.1) Efficiency: This refers to the user feeling that the system is enabling the task to be completed in a prompt, effective and economical manner.

1.2) Affect: This is a psychological term for emotional feeling. It refers to the user feeling subjectively stimulated and pleasant or the opposite as a result of interacting with the software.

1.3) Helpfulness: Helpfulness is expected to measure the degree to which the software is self-explanatory, and also the suitability of the help system.

1.4) Control: The control scale is used to measure the degree of the user's mood for controlling the software.

1.5) Learnability: Learnability measures time and effort for learning the handling of the software from the user's point of view.

The highlights of SUMI questionnaire are short question and comfortable to understand. In addition, 50 questions are shifted by the criteria and mixed between positive and negative questions for reducing answerers' bias. Table 3.19 presents an example of questions adapted from the SUMI.

Table 3.19 An example of questions is adapted from the SUMI

No.	Question	Question Type*	Criteria No.**
1	This software responds too slowly to inputs.	−	1
2	I would recommend this software to my colleagues.	+	2
3	The instructions and prompts are helpful.	+	3
4	This software has at some time stopped unexpectedly.	−	4
5	Learning to operate this software initially is full of problems.	−	5
6	I sometimes don't know what to do next with this software.	−	1
7	I enjoy the time I spend using this software.	+	2
8	I find that the help information given by this software is not very useful.	−	3
9	If this software stops it is not easy to restart it.	−	4
10	It takes too long to learn the software functions.	−	5
⋮	⋮	⋮	⋮
50	I have to look for assistance most times when I use this software.	−	5

** Question types: (+) = Positive question, (−) = Negative question*
*** Criterias no.: 1 = Efficiency, 2 = Affect, 3 = Helpfulness, 4 = Control and 5 = Learnability*

Table 3.19 details the example of questions in the post-study system usability questionnaire which are sorted and switched by five criteria: 1 = Efficiency, 2 = Affect, 3 = Helpfulness, 4 = Control and 5 = Learnability. These questions can be both of positive and negative questions. Hence, the interpretation of each question is different as depicted in Table 3.20.

Table 3.20 The interpretation of positive and negative questions

Question Type	**Interpretation of Each Question Type**		
	Agree	**Undecided**	**Disagree**
Positive (+)	3	2	1
Negative (−)	1	2	3

Table 3.20 presents the interpretation of positive and negative questions which their values are during 1 – 3. The measurement of the level of usability is divided into three levels: good, fair, and poor. To construct a frequency of class interval, Equation 3.3 is implemented as follows:

$$\text{Frequency of class interval} = \frac{Highest\ value - Smallest\ value}{Number\ of\ class} \tag{3.3}$$

$$= \frac{3-1}{3}$$

$$= 0.66$$

From the above calculation, the level of system usability can be determined as details below.

2.34 – 3.00 = Good

1.67 – 2.33 = Fair

1.00 – 1.66 = Poor

2) Thinking-Aloud Protocol

The thinking-aloud protocol (Nielsen, 1994) is the basic usability testing technique where users think out loud while the system testing. In HCI practice, the thinking-aloud protocol seems to be one of the most popular techniques. It is often referred to the usability method and used both in laboratory settings, workshops and field testing (Nielsen, 1992).

Nielsen (www, 2012) defines that *"In a thinking-aloud test, you ask test participants to use the system while continuously thinking out loud. That is simply verbalizing their thoughts as they move through the user interface. To run a basic thinking aloud usability study, you need to do only 3 things: 1) Recruit representative users, 2) Give them representative tasks to perform, and 3) Shut up and let the users do the talking."*

The benefits of the thinking-aloud testing can be summarized below.

2.1) Low Cost: No special tool is needed because the researcher will sit next to a user and take notes as a user talks.

2.2) Robust: The researcher will get reasonably good findings directly from a user, particularly in the quantitative usability testing. The study may be found many problems and the smallest mistake can judge a research.

2.3) Adjustable: The researcher can use the thinking-aloud protocol at any stage in the development process, from initially document prototypes to fully implemented systems, e.g., websites, software applications, intranets, consumer products, enterprise software, and mobile design.

3.4 Data Collection

This research collects both of primary data and secondary data as follows:

3.4.1 Primary Data: Data are observed or collected directly from surveying the sampling by using questionnaires.

3.4.2 Secondary Data: Data are collected in the past or other parties, for example, information of the existing 3D animation builder for displaying the musical instruments, including the review of literature, such as theses and dissertations, reference books, research reports, and journals.

3.5 Data Analysis

This section is related to two data analysis stages as follows:

3.5.1 The Result Analysis of Model Construction and Model Testing

The result analysis in this part is accomplished in the stage of ukulele playing information recognition. The analysis will be occurred when the recognition algorithms are developed. The related formulas used for the result analysis of recognition comprise two parts in accordance with the recognition processes as follows:

3.5.1.1 Strumming Pattern Recognition: This recognition is analyzed by using Precision, Recall, and F-measure. The preparatory evaluation of model uses 10-cross validation (Noureldin, El-Shafie, and Reda Taha, 2007) on the training dataset. The model is appraised using retrieval performance measures, which are Precision, Recall and F-measure (Galitsky, 2013). Table 3.21 defines the categorical output in four terms.

Table 3.21 The categorical output

	Data Recognized (+)	Data Not Recognized (-)
Relevant Data (+)	*TP*	*FN*
Irrelevant Data (-)	*FP*	*TN*

1) *TP* (True Positive): The number of relevant data (i.e., the predicted strumming types) that can be correctly recognized.

2) *FP* (False Positive): The number of irrelevant data that is recognized.

3) *FN* (False Negative): The number of relevant data that cannot be recognized.

4) *TN* (True Negative): The number of irrelevant data that is not recognized.

The calculation of Precision, Recall, and F-measure is defined in Equation 3.4 - 3.6, respectively (Martino, Hernández, Fiori, and Fernández, 2013).

$$\text{Precision} = \frac{TP}{(TP+FN)} \, x \; 100\ \% \qquad (3.4)$$

$$\text{Recall} = \frac{TP}{(TP+FP)} \, x \; 100\ \% \qquad (3.5)$$

$$\text{F-measure} = \frac{(b^2 \; + \; 1)\, PR}{b^2\, P + R} \qquad (3.6)$$

Where $b = 1$ means recall (R) and precision (P) are equally weighted

3.5.1.2 Chord Changing Time Recognition: This recognition is analyzed by using foundation statistics, such as Mean and Percentage.

3.5.2 The Result Analysis of System Usability Testing

The result analysis in this part relates to the post-study usability testing questionnaires and the thinking-aloud protocol.

3.5.2.1 Post-Study System Usability Questionnaire

The closed-ended questions are analyzed as details below.

1) User Personal Data: These data are analyzed by IBM SPSS Statistics 19 software, which uses Frequencies, and Percentage.

2) System Usability Data: These data are analyzed by using Descriptive Statistics, such as Mean, Standard Deviation, and Percentage, and Inferential Statistics, such as Post-Hoc test in ANOVA statistic for significant testing.

3.5.2.2 Thinking-Aloud Protocol

The think out loud data from participants are analyzed by a summary description based on five criteria: Efficiency, Affect, Helpfulness, Control and Learnability.

CHAPTER 4

THE RESULTS OF THE STUDY AND DISCUSSIONS

This chapter presents the results of the development of an automatic 3D animation builder for displaying ukulele playing. The explanation of this chapter is organized according to the main process of the system development. Section 4.1 reports the results of system capability evaluation. This section considers and discusses the results of the model construction consisting of a strumming pattern recognition model and a chord changing time recognition model. Section 4.2 represents the outcomes of the system development based on an overall system operation. The last section of this chapter, Section 4.3 presents the results of system usability testing, which are an analysis of the data collected from samples in the term of quantitative and qualitative data.

4.1 The Results of System Capability Evaluation

4.1.1 Strumming Pattern Recognition

As previously discussed in the Chapter 3, the strumming pattern recognition is operated by five stages, consisting of 1) data preparation, 2) attribute extraction, 3) model construction for strumming type prediction, 4) strumming type prediction, and 5) strumming pattern summarization. To make it easier to comprehend the process of strumming pattern recognition, Figure 4.1 is illustrated.

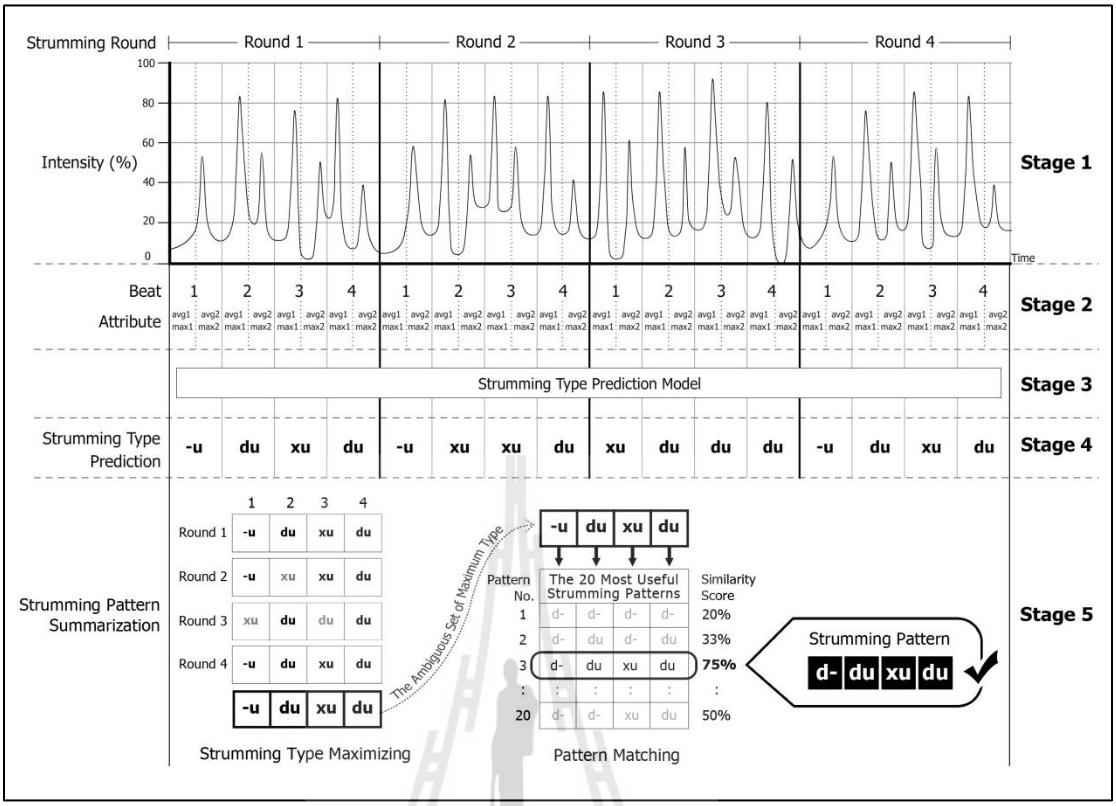


Figure 4.1 The process of strumming pattern recognition

Figure 4.1 shows only 4 strumming rounds of a sample song. A vertical solid line divides a strumming round, which is generated by tempo and beat tracker on Sonic Visualiser software. The output of stage 4 is the pairs of the strumming types, which are predicted by the strumming type prediction model. These strumming type prediction datasets will be summarized by the stage of strumming pattern summarization. This section presents the evaluation results of strumming type prediction and strumming pattern summarization as follows:

4.1.1.1 The Results of Strumming Type Prediction

The strumming type prediction models are constructed by training data, which collects 344 datasets extracted in the attribute extraction stage. According to Table 3.4 - 3.6 of Chapter 3, Figure 4.2 illustrates a comparison of the weighted average results of Precision, Recall, and F-measure of each prediction model using 10-fold cross validation on the training dataset.

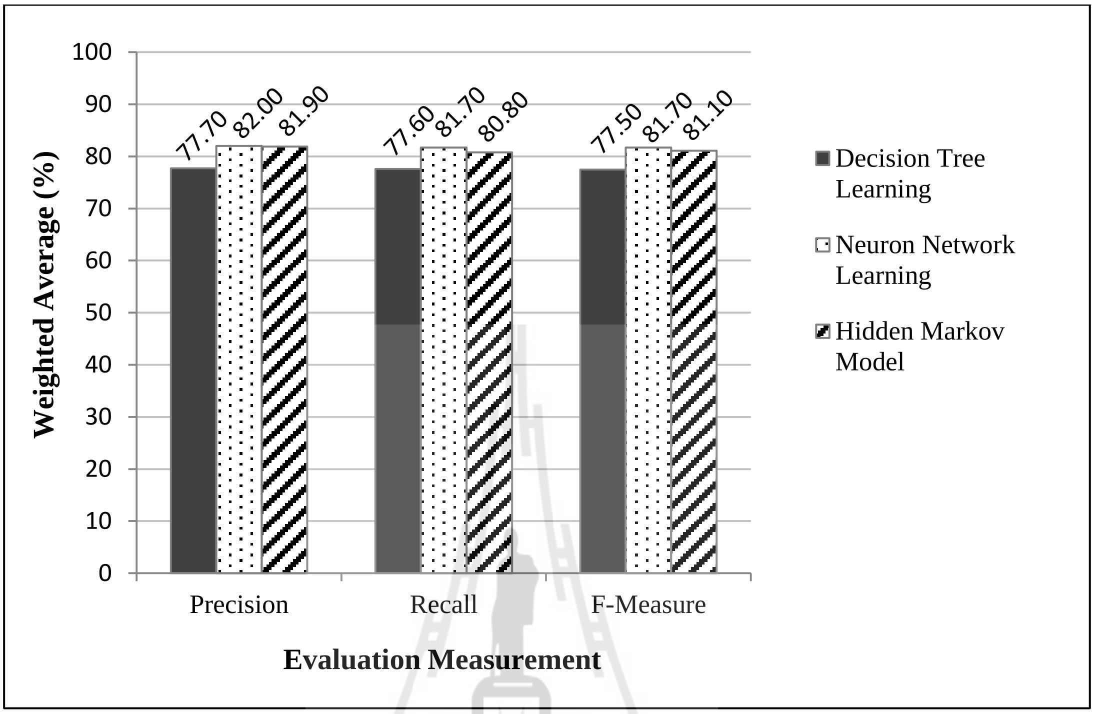


Figure 4.2 The weighted average results of Precision, Recall, and F-measure of the strumming type prediction models using 10-fold cross validation

Figure 4.2 shows that the validation results of each model by using 10-fold cross validation are in the similar range of Precision, Recall, and F-measure. However, these results cannot refer to the finest model for strumming pattern summarization. Consequently, the prediction model evaluation by using test songs is performed.

The complementary evaluation of each model uses the extracted attributes from 10 ukulele songs with difference strumming patterns as the supplied test set. Table 4.1 shows the data environment of 10 ukulele test songs into five aspects which are average tempo, duration, resource, strumming pattern, and number of strumming type's pairs. In the resource aspect, test songs are collected from two types of resource consisting of 1) recording from musicians (two persons) and 2) downloading from the Internet.

The classifiers are used to construct the model consisting of three prediction approaches: 1) decision tree learning, 2) neuron network learning, and 3) hidden Markov model. The model of each approach is appraised using retrieval performance measures, which are Precision, Recall and F-measure (Galitsky, 2013).

The strumming type prediction model of the decision tree learning is constructed by using J48 algorithm, which is the most useful and efficient algorithm for the solution of prediction (Zarkami, 2011). The J48 builds a binary tree to construct the model and allows the classification via rules generated from them (Patil and Sherekar, 2013). The evaluation of the strumming type prediction model is performed by using retrieval performance measures, which are Precision, Recall and F-measure (Galitsky, 2013). The evaluation results of the prediction model constructed by the decision tree learning are shown in Table 4.2.

Table 4.1 The data environment of 10 ukulele test songs for prediction model evaluation

Song No.	Average Tempo (BPM)	Duration (Minutes)	Resource	Strumming Pattern								Number of Strumming Type's Pairs				
				1	&	2	&	3	&	4	&	d-	du	-u	xu	Total
1	81.71	02:13	Recording from musician (1)	d	-	d	-	d	-	d	-	180	-	-	-	**180**
2	111.16	02:31	Recording from musician (2)	d	-	d	u	d	u	d	u	68	204	-	-	**272**
3	149.40	02:40	Downloading from Soundcloud.com	d	u	d	-	d	u	d	-	194	194	-	-	**388**
4	127.62	02:09	Downloading from Youtube.com	x	u	x	u	x	u	x	u	-	-	-	288	**288**
5	131.27	02.55	Recording from musician (1)	d	u	x	u	d	u	x	u	-	150	-	150	**300**
6	90.15	01:44	Downloading from Youtube.com	d	u	x	u	x	u	x	u	-	28	-	84	**112**
7	141.04	03:01	Recording from musician (1)	d	-	d	u	-	u	d	-	212	106	106	-	**424**
8	90.66	01:23	Downloading from Soundcloud.com	d	-	d	u	-	u	d	u	67	134	67	-	**268**
9	95.41	01:12	Recording from musician (2)	d	-	d	-	x	u	-	u	100	-	50	50	**200**
10	156.82	01:17	Recording from musician (2)	d	-	d	-	-	u	d	u	88	44	44	-	**176**
Avg.	**117.52**	**02:06**	**-**	**Total**								**909**	**860**	**267**	**572**	**2,608**

Table 4.2 The evaluation results of the strumming type prediction model using 10 ukulele test songs constructed by the decision tree learning

Song No.	Weighted Average (%)		
	Precision	**Recall**	**F-measure**
1	100.00	92.80	96.30
2	90.60	77.60	82.50
3	86.60	81.40	83.10
4	100.00	94.80	97.30
5	94.50	82.00	87.80
6	94.30	89.30	91.70
7	91.00	87.50	89.00
8	95.60	91.00	92.90
9	96.30	87.50	91.60
10	89.60	86.90	87.60
Avg.	**93.85**	**87.08**	**89.98**

Table 4.2 reveals that the prediction model constructed by the decision tree learning gets good performance for classifying strumming types. The experimental results are achieved in 93.85 per cent of Precision, 87.08 per cent of Recall, and 89.98 per cent of F-measure by average.

The strumming type prediction model of the neuron network learning is constructed by the Multilayer Perceptron algorithm. This algorithm has been a great deal of interest in pattern recognition and it is desirable to shorten the number of features that are used while still maintaining enough classification accuracy (Ruck, Rogers, and Kabrisky, 1990). The evaluation results of strumming type prediction model constructed by the neuron network learning are shown in Table 4.3.

Table 4.3 The evaluation results of strumming type prediction model using 10 ukulele test songs constructed by the neuron network learning

Song No.	Weighted Average (%)		
	Precision	**Recall**	**F-measure**
1	100.00	74.40	85.40
2	90.50	77.20	82.30
3	88.30	83.50	84.60
4	100.00	91.30	95.50
5	93.70	77.00	84.50
6	91.30	63.40	74.60
7	76.20	74.30	69.50
8	81.70	68.30	67.40
9	94.10	88.50	91.00
10	68.80	64.80	59.30
Avg.	**88.46**	**76.27**	**79.41**

Table 4.3 reveals that the prediction model designed by the neuron network learning is achieved in 88.46 per cent of Precision, 76.27 per cent of Recall, and 79.41 per cent of F-measure by average, which are less than the results of the evaluation by using the decision tree learning in all measurements.

The strumming type prediction model of the hidden Markov model (HMM) is constructed by using the HMMWeka algorithm (Gillies, 2010) with six states based on a fully connected HMM. The HMMWeka only works with relational attributes. The instances in the relational attributes may consist of single, nominal data instances or multivariate, and numeric attributes. The HMM is a popular classifier because it is simple and flexible. It is typically used in speech recognition and bio-sequence analysis (Murphy, 2002). The evaluation results of strumming type prediction model constructed by the hidden Markov model are shown in Table 4.4.

Table 4.4 The evaluation results of strumming type prediction model using 10 ukulele test songs constructed by the hidden Markov model

Song No.	Weighted Average (%)		
	Precision	**Recall**	**F-measure**
1	100.00	63.90	78.00
2	88.20	70.20	77.30
3	77.70	65.50	68.80
4	100.00	95.50	97.70
5	94.60	76.30	84.00
6	84.30	73.20	78.00
7	86.40	80.20	82.90
8	85.50	77.20	79.70
9	89.20	79.00	81.80
10	85.30	80.70	81.60
Avg.	**89.12**	**76.17**	**80.98**

Table 4.4 reveals that the prediction model designed by the hidden Markov model is achieved in 89.12 per cent of Precision, 76.17 per cent of Recall, and 80.98 per cent of F-measure by average. The prediction results of the HMM are less than that of the decision tree learning in all measurements, but more than that of the neuron network learning.

According to the evaluation of the strumming type prediction models, Figure 4.3 illustrates a comparison of the weighted average results of Precision, Recall, and F-measure of each prediction model.

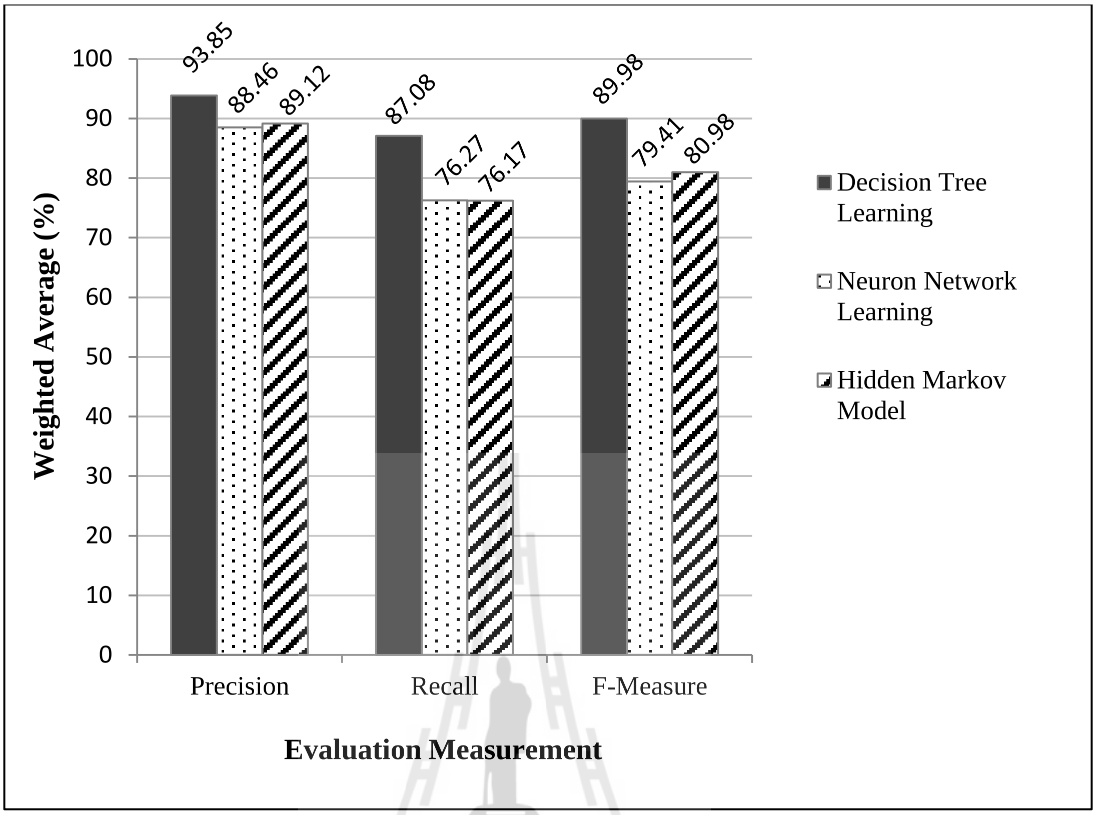


Figure 4.3 The weighted average results of Precision, Recall, and F-measure of the strumming type prediction model using 10 ukulele test songs

Figure 4.3 presents that the decision tree learning-based predictor produces the best accuracy followed by hidden Markov model-based predictor and neuron network learning-based predictor, respectively. Then, the results of strumming type prediction of decision tree learning will be used in the next process which is a process of strumming pattern summarization.

4.1.1.2 The Results of Strumming Pattern Summarization

In the strumming pattern summarization stage, ukulele songs will be passed through the process of strumming type maximizing and pattern matching as previously described in Chapter 3.1.2.2. The 10 ukulele test songs as presented in Table 4.1 are operated again to recognize the strumming pattern. The results of the strumming pattern recognition on 10 ukulele test songs show in Table 4.5.

Table 4.5 The results of the strumming pattern recognition on 10 ukulele test songs

Song No.	Strumming Type Prediction* (%)	Expected Strumming Pattern No.**	Strumming Pattern Matching No.**	Strumming Pattern Summarization
1	96.30	No. 1	No. 1	Correct
2	82.50	No. 5	No. 5	Correct
3	83.10	No. 7	No. 7	Correct
4	97.30	No. 10	No. 10	Correct
5	87.80	No. 11	No. 11	Correct
6	91.70	No. 16	No. 16	Correct
7	89.00	No. 8	No. 8	Correct
8	92.90	No. 9	No. 9	Correct
9	91.60	No. 17	No. 17	Correct
10	87.60	No. 19	No. 19	Correct
Avg.	**89.98**	-	-	-

** Referenced by the percentage of F-measure of decision tree learning-based predictor in Table 4.2*
*** Referenced by the number of 20 most useful strumming patterns in Table 3.1 of Chapter 3*

In Table 4.5, considering on a strumming type in each strumming round, it reveals that some strumming type's pairs are erroneous because of various strumming styles of each ukulele player and each song. However, the results show that all of ukulele test songs are summarized with the correct strumming patterns when are improved by the strumming pattern summarization stage.

4.1.2 Chord Changing Time Recognition

As explained in Chapter 3, the experiment of chord changing time recognition is processed based on two presumptions: 1) Chord changing time seems to reverse with song tempo and 2) the same connected chords should take the same chord changing time in any song. Nevertheless the results of both presumptions showed that the chord changing time is not depended on song tempo and the type of connected chords. However, the solution of chord changing time recognition must be summarized to apply for automatically construing 3D ukulele playing animation. Hence, the attributes of the first presumption experiment as shown in Table 3.10 of Chapter 3 are used to estimate the chord changing time. The results of chord changing time estimation show in Table 4.6.

Table 4.6 shows that the recognized chord changing time is 0.28 seconds that is estimated by the average song tempo and the average chord changing time from 22 ukulele songs. However, the process of 3D animation development is necessary to perceive the animation frame information for generating the chord changing animation. Therefore, the recognized chord changing time must be converted from a time unit to a 3D animation frame unit. In this case, there are 60 frames per second. The calculation of animation frame is performed by Equation 4.1.

$$Number\ of\ frame = \frac{Chord\ changing\ time}{Duration\ of\ 1\ frame} \quad (4.1)$$

$$= \frac{0.28}{1\ /\ 60} = 18\ frames$$

Table 4.6 The results of chord changing time estimation

Song No.	Tempo (BPM)	Average Chord Changing Time (Seconds)
1	82.03	0.33
2	83.37	0.50
3	85.43	0.14
4	91.48	0.27
5	91.48	0.29
6	94.83	0.47
7	95.70	0.39
8	95.70	0.26
9	96.61	0.35
10	96.91	0.30
11	101.33	0.23
12	103.36	0.23
13	105.47	0.20
14	108.81	0.25
15	110.06	0.37
16	122.08	0.35
17	132.57	0.21
18	132.60	0.34
19	139.67	0.21
20	140.64	0.12
21	154.56	0.15
22	161.31	0.26
Average		**0.28**

The 0.28 seconds of chord changing time can be converted to 18 frames of 3D animation based on 60 frames per second. The 18 frames will be used to create the chord changing animation. The purposed system will automatically

recognize the first frame of chord changing before reaching the next chord to create a smooth and realistic ukulele playing animation. Figure 4.4 displays the chord changing from C to Em within the 18 frames.

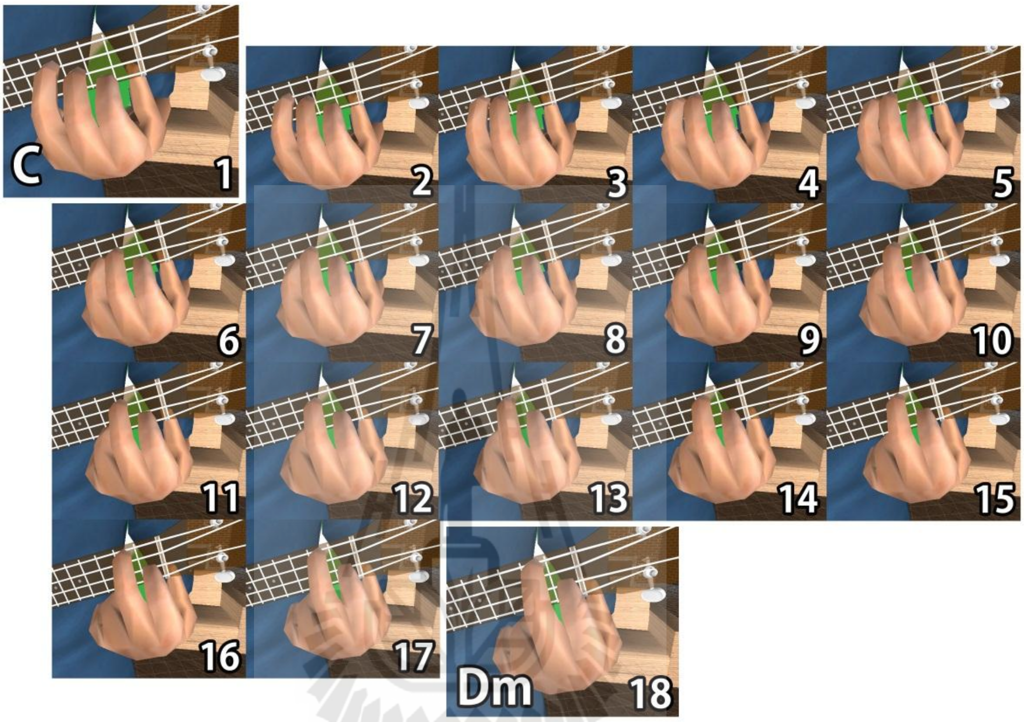


Figure 4.4 The frame by frame animation of changing chord from C to Em

4.2 The Results of System Development

An UkeMe3D is the name of the purposed system which is created to be a research tool. The main tool of system development is the Unity3D software which allows a programmer to create 3D applications with flexible computer languages and high performance. The UkeMe3D works on the Windows and Mac operating system and it represents the output in the 3D animation format. The UkeMe3D is divided by the three main screen of system, e.g., home screen, loading screen, and output screen. There is an overall system operation as shown in Figure 4.5.

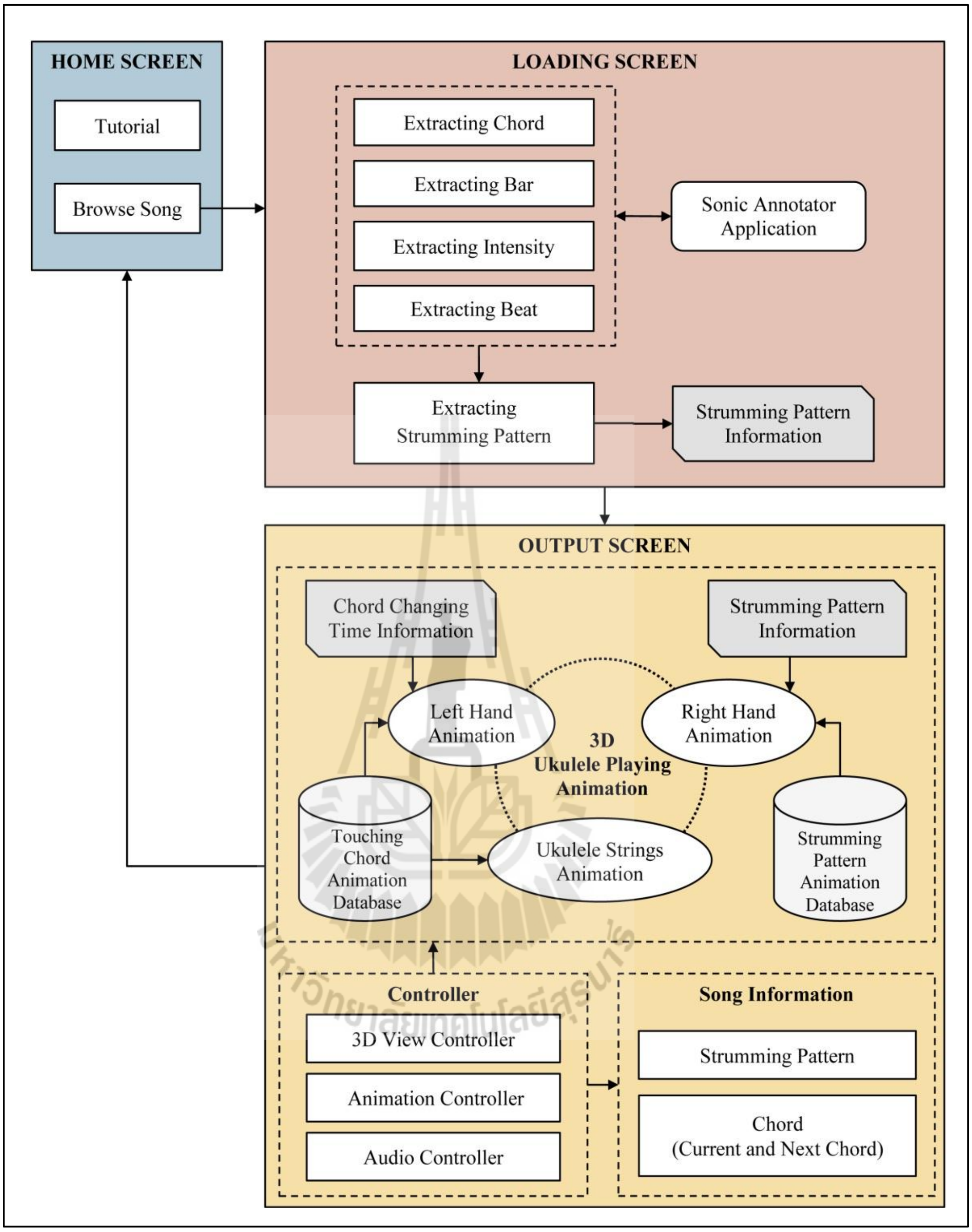


Figure 4.5 An overall system operation

4.2.1 Home Screen: A home screen is the introductory screen that appears when users run the system. On this screen, users can browse the MP3 ukulele song into the system for analyzing the properties of song and building a 3D ukulele playing animation. The system supports MP3 file only because it is small size, good quality and popular today (Yang and Chen, 2012). In addition, users can also learn how to use the system by clicking “How to Uke!” button for showing the tutorial page. The screenshots of the home screen, file selection screen, and tutorial page is shown in Figure 4.6 - 4.8, respectively.

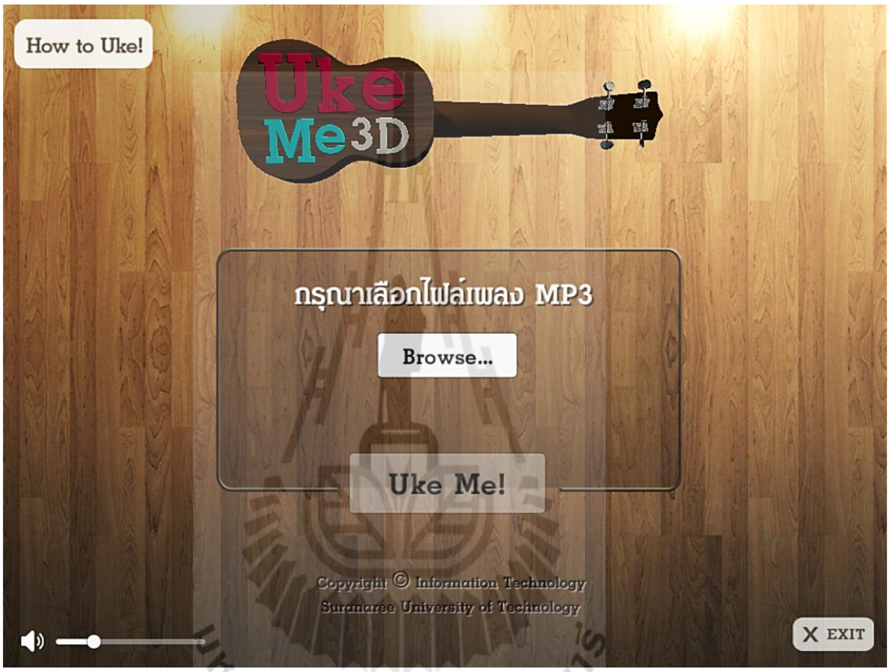


Figure 4.6 Home screen

4.2.2 Loading Screen: A loading screen is a secondary page for notifying the percentage of an overall process. After importing a song, the song will be passed through the attribute extraction process consisting of five components: chord, bar, intensity, beat, and strumming patterns. During the process, all components except strumming pattern are extracted by Sonic Annotator software, which is a batch tool for audio feature extraction. When all processes by Sonic Annotator are completed, UkeMe3D will finish loading all extracted data for analyzing a strumming pattern of the song. The strumming pattern is recognized by a new solution of this research.

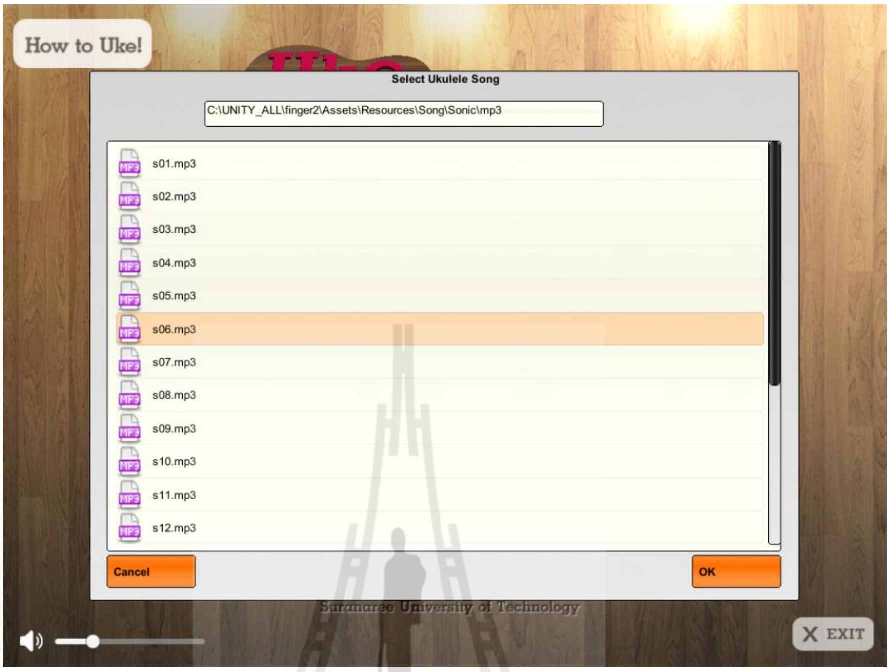


Figure 4.7 File selection screen

Figure 4.8 Tutorial page

The loading screen displays the progress of the extraction on the left side and the instructions of 3D view controller on the right side. When loading is completed, the system will direct users to the output screen automatically. The screenshot of the loading screen is shown in Figure 4.9.

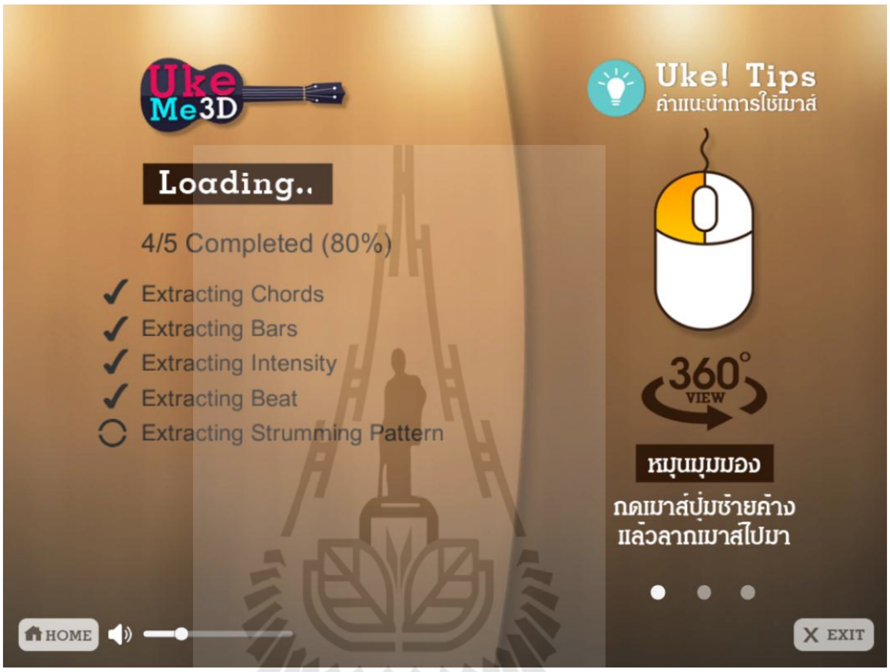


Figure 4.9 Loading screen

4.2.3 Output Screen: An output screen presents the 3D animation of ukulele playing as shown in Figure 4.10 and this screen can be divided into three parts as follows:

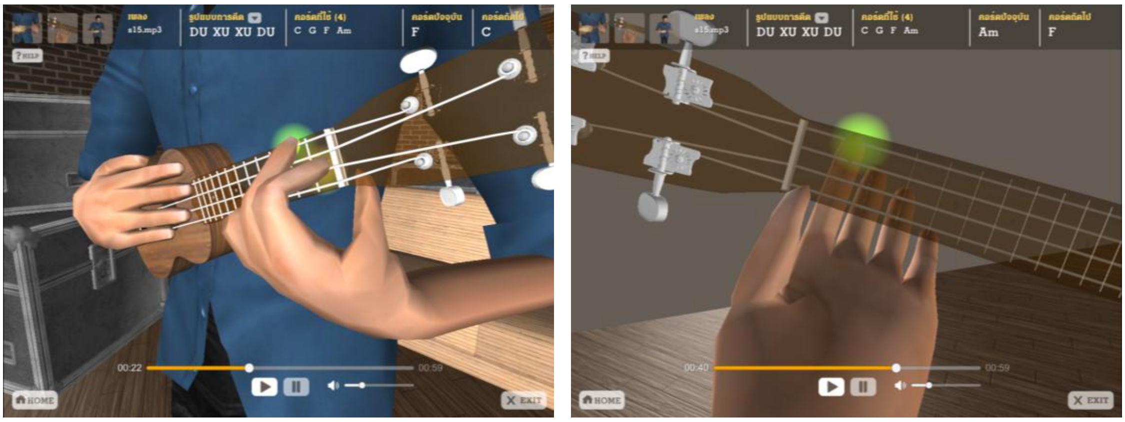
Figure 4.10 Output screen

4.2.3.1 3D Ukulele Playing Animation Part: A 3D musician with ukulele is placed in the center of the screen inside the 3D virtual studio. It is animated by the commands of the system, which are linked to the audio controller. The musician's right hand is animated by retrieving information from the strumming pattern animation database that is consistent with the extracted strumming pattern information. The left hand is animated by chord changing time information, which is received from the stage of chord changing time recognition as previously described in Chapter 3.1.2.2., and retrieving information from chord touching animation database. Moreover, the chord touching animation database is also used for generating the point lights on the ukulele string. The screenshot of 3D musician's hands and 3D ukulele is shown in Figure 4.11.

Figure 4.11 3D musician's hands and 3D ukulele

4.2.3.2 Controller Part: A controller allows users to control the ukulele playing animation and audio playback consisting of three types as follows:

1) 3D View Controller: Users can control the perspective view of 3D musician and 3D ukulele by using the computer mouse with three functions: 1) dragging with the left mouse button for rotating the 3D model in 360 degrees, 2) scrolling with the middle mouse button to zoom in or zoom out, and 3) dragging with the right mouse button for moving the current view to the target view. Moreover, the top-left screen corner also provides three default 3D view buttons consisting of a front view of both hands, a back view of the left hand, and a front view of the musician's body. Figure 4.12 shows the side and back views of the left hand and Figure 4.13 shows an instruction page of mouse button control.



(A) (B)

Figure 4.12 Side view (A) and back view (B) of the left hand

Figure 4.13 An instruction page of mouse button control

2) Animation Controller: The UkeMe3D allows users to control 3D animation from the control bar on the output screen. Users can drag the control bar to skip to a target time and the system will update animation and an ukulele song in real-time. Furthermore, play and pause button help users to more easily control.

3) Audio Controller: Users can increase or decrease the volume as needed by dragging the volume bar. In addition, the ukulele song will be automatically updated by dragging the 3D animation control bar. Figure 4.14 shows the animation and audio controllers.

Figure 4.14 Animation and audio controller

4.2.3.3 Song Information Part: The top of the output screen is the section of song information. Besides, a file name is shown in this section, it also presents the strumming pattern and chord information, which are necessary to play the ukulele. The chord information comprises all chords played in the ukulele song, the current played chord, and the next played chord. Figure 4.15 shows the song information part.

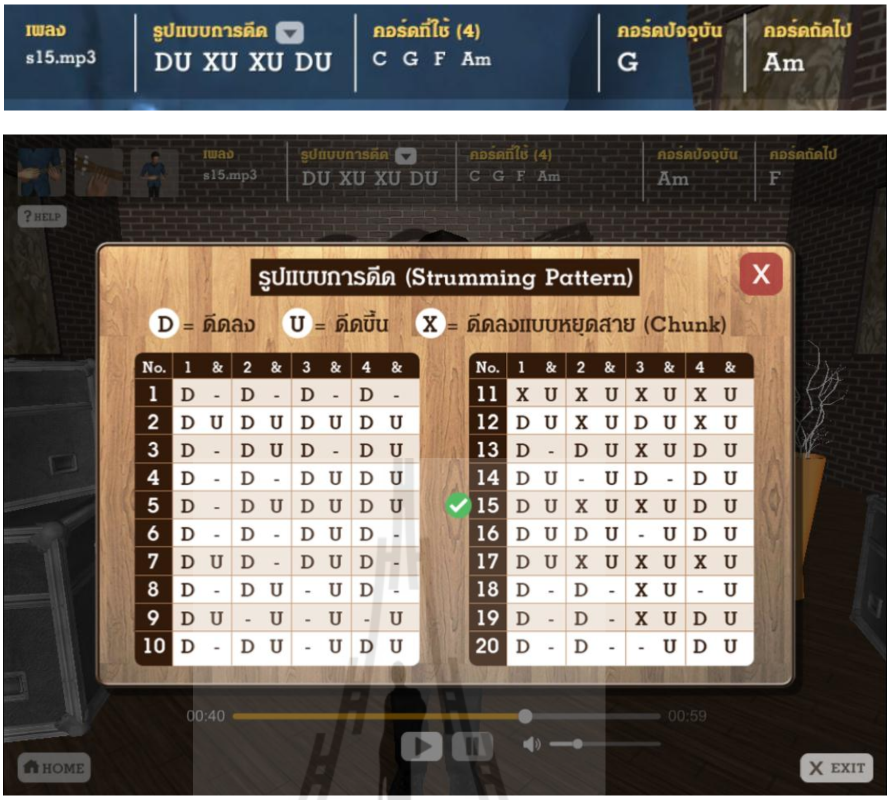


Figure 4.15 Song information part

From the output screen users can return to the home screen by clicking the home button or can log out by clicking the exit button.

4.3 The Result of System Usability Testing

This section presents the results of system usability testing by using two testing instruments: 1) The post-study system usability questionnaire and 2) the thinking-aloud protocol. Both testing is performed with three sample groups. The first group is experts of ukulele playing with basic computer skills. The second group is experts of system design and development. The third group is persons who are interested in ukulele playing with basic computer skills. Table 4.7 presents the number and the percentage of the samples by gender.

Table 4.7 The number and the percentage of the samples by gender

Gender	**Group ID***					
	1		**2**		**3**	
	N	**%**	**N**	**%**	**N**	**%**
Male	2	40	2	40	1	20
Female	3	60	3	60	4	80
Total	**5**	**100**	**5**	**100**	**5**	**100**

** Group ID: 1 = the experts of ukulele playing with basic computer skills*
2 = the experts of system design and development
3 = the persons who are interested in ukulele playing with basic computer skills

Table 4.7 presents the number and the percentage of the samples by gender. There are 15 people in three groups (i.e., 5 people per group). The first group is the experts of ukulele playing consisting of 40 per cent of male and 60 per cent of female. The second group is the experts of system design and development consisting of 40 per cent of male and 60 per cent of female. The third group is the persons who are interested in ukulele playing consisting of 20 per cent of male and 80 per cent of female.

4.3.1 The Usability Testing Results using the Post-Study Questionnaire

This section presents the results which are collected and analyzed from the post-study system usability questionnaire consisting of overall system usability testing and the system usability testing as individual sample group as follows:

4.3.1.1 Overall System Usability Testing: When all samples completed the system testing, they will answer 50 questions which are subdivided into 5 criteria as follows: Efficiency, Affect, Helpfulness, Control, and Learnability. The system usability testing results of all samples are summarized as shown in Table 4.8.

Table 4.8 The system usability testing results of all samples

Criterion	$\bar{X}$	S.D.	Level
Efficiency	2.57	0.42	Good
Affect	2.91	0.19	Good
Helpfulness	2.84	0.24	Good
Control	2.78	0.18	Good
Learnability	2.77	0.33	Good
Average	**2.77**	**0.30**	**Good**

The evaluation results in Table 4.8 reveal that the system usability averaged from all samples is in a good level ($\bar{X}$ = 2.77 from 3, S.D. = 0.30). The best criterion of the system is the affect ($\bar{X}$ = 2.91, S.D. = 0.19), which is followed by the helpfulness ($\bar{X}$ = 2.84, S.D. = 0.24), the control ($\bar{X}$ = 2.78, S.D. = 0.18), the learnability ($\bar{X}$ = 2.77, S.D. = 0.33), and the efficiency ($\bar{X}$ = 2.57, S.D. = 0.42), respectively. Figure 4.16 visualizes the results of overall system usability testing.

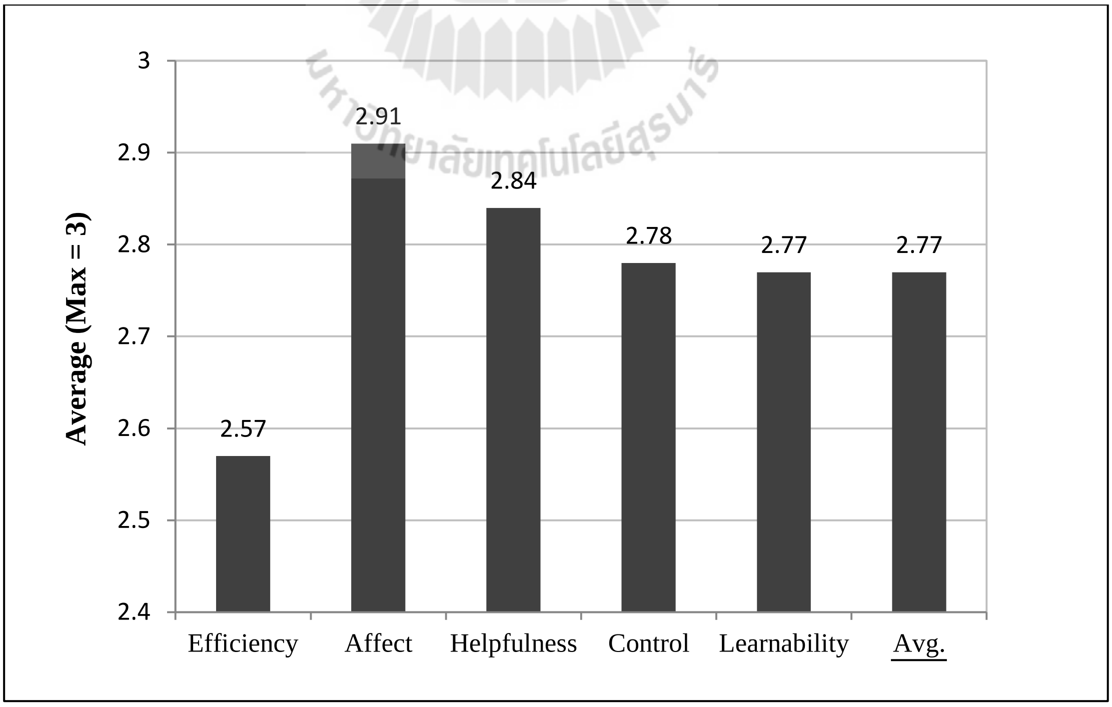


Figure 4.16 The results of overall system usability testing

Moreover, overall system usability testing can be examined based on five criteria as follows:

1) Efficiency: The results of the system efficiency analysis are presented in Table 4.9.

Table 4.9 The results of the system efficiency analysis

No.	Question	$\bar{X}$	S.D.	Level
1	This software responds too slowly to inputs. (−)	2.67	0.62	Good
2	I sometimes don't know what to do next with this software. (−)	2.13	0.83	Fair
3	I sometimes wonder if I am using the right function. (−)	2.47	0.74	Good
4	This software seems to disrupt the way I normally like to arrange my work. (−)	2.27	0.80	Fair
5	I think this software is inconsistent. (−)	2.73	0.59	Good
6	Tasks can be performed in a straight forward manner using this software. (+)	2.73	0.46	Good
7	It is obvious that user needs have been fully taken into consideration. (+)	2.93	0.26	Good
8	There are too many steps required to get something to work. (−)	3.00	0.00	Good
9	The software hasn't always done what I was expecting. (−)	2.13	0.74	Fair
10	This software occasionally behaves in a way which can't be understood. (−)	2.60	0.63	Good
Average		**2.57**	**0.67**	**Good**

The analysis results in Table 4.9 reveal that the system efficiency averaged from all samples is in a good level ($\bar{X}$ = 2.57 from 3, S.D. = 0.67). The highest score is in the question no. 8: “There are too many steps required to get something to work.” ($\bar{X}$ = 3.00), i.e., the system has appropriate steps required to get something to work. While the standard deviation is equal to zero (S.D. = 0.00) that means the evaluation of all samples is similar.

In addition, the good level is also achieved in six questions, which are arranged in descending order of average as follows: the question no. 7: “It is obvious that user needs have been fully taken into consideration.” ($\bar{X}$ = 2.93, S.D. = 0.26), the question no. 6: “Tasks can be performed in a straight forward manner using this software.” ($\bar{X}$ = 2.73, S.D. = 0.46) that is equal to the question no. 5: “I think this software is inconsistent.” ($\bar{X}$ = 2.73, S.D. = 0.59) followed by the question no. 1: “This software responds too slowly to inputs.” ($\bar{X}$ = 2.67, S.D. = 0.62), the question no. 10: “This software occasionally behaves in a way which can't be understood.” ($\bar{X}$ = 2.60, S.D. = 0.63), and the question no. 3: “I sometimes wonder if I am using the right function.” ($\bar{X}$ = 2.47, S.D. = 0.74).

A fair level is found in three questions, which are arranged in descending order of average as follows: the question no. 4: “This software seems to disrupt the way I normally like to arrange my work.” ($\bar{X}$ = 2.27, S.D. = 0.80), the question no. 2: “I sometimes don't know what to do next with this software.” ($\bar{X}$ = 2.13, S.D. = 0.83) that is equal to the question no. 9: “The software hasn't always done what I was expecting.” ($\bar{X}$ = 2.13, S.D. = 0.74).

2) Affect: The results of the system affect analysis are presented in Table 4.10.

The analysis results in Table 4.10 reveal that the system affect averaged from all samples is in a good level ($\bar{X}$ = 2.91 from 3, S.D. = 0.31). The highest score ($\bar{X}$ = 3.00) is in the question no. 1: “I would recommend this software to my colleagues.”, the question no. 3: “Working with this software is satisfying.”, the question no. 4: “Working with this software is mentally stimulating.”, the question no. 6: “Using this software is frustrating.”, i.e., the use of the system is not frustrating, the question no. 8: “I think this software has sometimes given me a headache.”, i.e., the system not makes an annoyance to the users, the question no. 9: “The software presents itself in a very attractive way.”, and the question no. 10: “This software is really very awkward.”, i.e., the system is very convenient. While the standard deviation is equal to zero (S.D. = 0.00) that means the evaluation of all samples is similar.

Table 4.10 The results of the system affect analysis

No.	Question	$\bar{X}$	S.D.	Level
1	I would recommend this software to my colleagues. (+)	3.00	0.00	Good
2	I enjoy the time I spend using this software. (+)	2.80	0.56	Good
3	Working with this software is satisfying. (+)	3.00	0.00	Good
4	Working with this software is mentally stimulating. (+)	3.00	0.00	Good
5	I would not like to use this software every day. (−)	2.40	0.51	Good
6	Using this software is frustrating. (−)	3.00	0.00	Good
7	There have been times in using this software when I have felt quite tense. (−)	2.87	0.35	Good
8	I think this software has sometimes given me a headache. (−)	3.00	0.00	Good
9	The software presents itself in a very attractive way. (+)	3.00	0.00	Good
10	This software is really very awkward. (−)	3.00	0.00	Good
Average		**2.91**	**0.31**	**Good**

In addition, the good level is also achieved in three questions, which are arranged in descending order of average as follows: the question no. 7: “There have been times in using this software when I have felt quite tense.” ($\bar{X}$ = 2.87, S.D. = 0.35), the question no. 2: “I enjoy the time I spend using this software.” ($\bar{X}$ = 2.80, S.D. = 0.56), and the question no. 5: “I would not like to use this software every day.” ($\bar{X}$ = 2.40, S.D. = 0.51).

3) Helpfulness: The results of the system helpfulness analysis are presented in Table 4.11.

Table 4.11 The results of the system helpfulness analysis

No.	Question	$\bar{X}$	S.D.	Level
1	The instructions and prompts are helpful. (+)	2.93	0.26	Good
2	I find that the help information given by this software is not very useful. (−)	3.00	0.00	Good
3	The way that system information is presented is clear and understandable. (+)	2.93	0.26	Good
4	There is never enough information on the screen when it's needed. (−)	2.47	0.74	Good
5	I can understand and act on the information provided by this software. (+)	2.93	0.26	Good
6	The software has helped me overcome any problems I have had in using it. (+)	2.93	0.26	Good
7	The organization of the menus seems quite logical. (+)	3.00	0.00	Good
8	Error messages are not adequate. (−)	2.60	0.51	Good
9	Either the amount or quality of the help information varies across the system. (+)	2.80	0.41	Good
10	It is easy to see at a glance what the options are at each stage. (+)	2.80	0.56	Good
Average		**2.84**	**0.42**	**Good**

The analysis results in Table 4.11 reveal that the system helpfulness averaged from all samples is in a good level ($\overline{X}$ = 2.84 from 3, S.D. = 0.42). The highest score ($\overline{X}$ = 3.00) is in the question no. 2: “I find that the help information given by this software is not very useful.”, i.e., the help information given by this software is very useful, and the question no. 7: “The organization of the menus seems quite logical.”. While the standard deviation is equal to zero (S.D. = 0.00) that means the evaluation of all samples is similar.

In addition, the good level is also achieved in eight questions, which are arranged in descending order of average as follows: the question no. 1: “The instructions and prompts are helpful.” ($\overline{X}$ = 2.93, S.D. = 0.26) that is equal to the question no. 3: “The way that system information is presented is clear and understandable.” ($\overline{X}$ = 2.93, S.D. = 0.26), the question no. 5: “I can understand and act on the information provided by this software.” ($\overline{X}$ = 2.93, S.D. = 0.26), and the question no. 6: “The software has helped me overcome any problems I have had in using it.” ($\overline{X}$ = 2.93, S.D. = 0.26) followed by the question no. 9: “Either the amount or quality of the help information varies across the system.” ($\overline{X}$ = 2.80, S.D. = 0.41), the question no. 10: “It is easy to see at a glance what the options are at each stage.” ($\overline{X}$ = 2.80, S.D. = 0.56), the question no. 8: “Error messages are not adequate.” ($\overline{X}$ = 2.60, S.D. = 0.51), and the question no. 4: “There is never enough information on the screen when it's needed.” ($\overline{X}$ = 2.47, S.D. = 0.74).

4) Control: The results of the system control analysis are presented in Table 4.12.

Table 4.12 The results of the system control analysis

No.	Question	$\bar{X}$	S.D.	Level
1	This software has at some time stopped unexpectedly.(−)	2.87	0.35	Good
2	If this software stops it is not easy to restart it. (−)	2.73	0.46	Good
3	I feel safer if I use only a few familiar functions. (+)	2.93	0.26	Good
4	I feel in command of this software when I am using it. (+)	2.73	0.59	Good
5	This software is awkward when I want to do something which is not standard. (−)	2.73	0.46	Good
6	The speed of this software is fast enough. (+)	2.73	0.46	Good
7	The software allows the user to be economical of keystrokes. (+)	2.73	0.70	Good
8	It is easy to make the software do exactly what you want. (+)	2.93	0.26	Good
9	It is relatively easy to move from one part of a task to another. (+)	2.67	0.62	Good
10	Getting data files in and out of the system is not easy. (−)	2.73	0.46	Good
Average		**2.78**	**0.48**	**Good**

The analysis results in Table 4.12 reveal that the system control averaged from all samples is in a good level ($\bar{X}$ = 2.78 from 3, S.D. = 0.48). The good level is achieved in all questions, which are arranged in descending order of average as follows: the question no. 3: “I feel safer if I use only a few familiar functions.” ($\bar{X}$ = 2.93, S.D. = 0.26) that is equal to the question no. 8: “It is easy to make the software do exactly what you want.” ($\bar{X}$ = 2.93, S.D. = 0.26) followed by the question no. 1: “This software has at some time stopped unexpectedly.” ($\bar{X}$ = 2.87, S.D. = 0.35), the question no. 2: “If this software stops it is not easy to restart it.” ($\bar{X}$ = 2.73, S.D. = 0.46) that is equal to the question no. 4: “I feel in command of this software when I am using it.” ($\bar{X}$ = 2.73, S.D. = 0.59), the question no. 5: “This software is awkward when I want to do something which is not standard.” ($\bar{X}$ = 2.73, S.D. = 0.46), the question no. 6: “The speed of this software is fast enough.” ($\bar{X}$ = 2.73, S.D. = 0.46), and the question no. 7: “The software allows the user to be economical of keystrokes.” ($\bar{X}$ = 2.73, S.D. = 0.70) followed by the question no. 10: “Getting data files in and out of the system is not easy.” ($\bar{X}$ = 2.73, S.D. = 0.46), and the question no. 9: “It is relatively easy to move from one part of a task to another.” ($\bar{X}$ = 2.67, S.D. = 0.62).

5) Learnability: The results of the system learnability analysis are presented in Table 4.13.

Table 4.13 The results of the system learnability analysis

No.	Question	$\bar{X}$	S.D.	Level
1	Learning to operate this software initially is full of problems. (−)	2.93	0.26	Good
2	It takes too long to learn the software functions. (−)	2.87	0.35	Good
3	The software documentation is very informative. (−)	2.87	0.35	Good
4	I prefer to stick to the functions that I know best. (−)	2.33	0.82	Fair
5	There is too much to read before you can use the software. (−)	2.93	0.26	Good
6	I keep having to go back to look at the guides. (−)	2.20	0.94	Fair
7	Learning how to use new functions is difficult. (−)	3.00	0.00	Good
8	I will never learn to use all that is offered in this software. (−)	2.87	0.35	Good
9	It is easy to forget how to do things with this software. (−)	2.87	0.35	Good
10	I have to look for assistance most times when I use this software. (−)	2.87	0.35	Good
Average		**2.77**	**0.53**	**Good**

The analysis results in Table 4.13 reveal that the system learnability averaged from all samples is in a good level ($\bar{X}$ = 2.77 from 5, S.D. = 0.53). The highest score is in the question no. 7: “Learning how to use new functions is difficult.” ($\bar{X}$ = 3.00), i.e., the learning how to use new functions is not difficult. While the standard deviation is equal to zero (S.D. = 0.00) that means the evaluation of all samples is similar.

In addition, the good level is achieved in seven questions, which are arranged in descending order of average as follows: the question no. 1: “Learning to operate this software initially is full of problems.” ($\bar{X}$ = 2.93, S.D. = 0.26) that is equal to the question no. 5: “There is too much to read before you can use the software.” ($\bar{X}$ = 2.93, S.D. = 0.26) followed by the question no. 2: “It takes too long to learn the software functions.” ($\bar{X}$ = 2.87, S.D. = 0.35) that is equal to the question no. 3: “The software documentation is very informative.” ($\bar{X}$ = 2.87, S.D. = 0.35), the question no. 8: “I will never learn to use all that is offered in this software.” ($\bar{X}$ = 2.87, S.D. = 0.35), the question no. 9: “It is easy to forget how to do things with this software.” ($\bar{X}$ = 2.87, S.D. = 0.35), and the question no. 10: “I have to look for assistance most times when I use this software.” ($\bar{X}$ = 2.87, S.D. = 0.35).

A fair level is found in two questions, which are arranged in descending order of average as follows: the question no. 4: “I prefer to stick to the functions that I know best.” ($\bar{X}$ = 2.33, S.D. = 0.82), and the question no. 6: “I keep having to go back to look at the guides.” ($\bar{X}$ = 2.20, S.D. = 0.94).

4.3.1.2 System Usability Testing as Individual Sample Group: This evaluation compares the system usability testing results in each sample group. The results of comparison are shown in Table 4.14.

Table 4.14 The comparison of the system usability testing results in each sample group

Criterion	Group ID*					
	1		2		3	
	$\bar{X}$	Level	$\bar{X}$	Level	$\bar{X}$	Level
Efficiency	2.58	Good	2.50	Good	2.62	Good
Affect	2.92	Good	2.90	Good	2.90	Good
Helpfulness	2.94	Good	2.72	Good	2.86	Good
Control	2.76	Good	2.72	Good	2.86	Good
Learnability	2.90	Good	2.66	Good	2.76	Good
Average	**2.82**	**Good**	**2.70**	**Good**	**2.80**	**Good**

** Group ID: 1 = the experts of ukulele playing with basic computer skills*
2 = the experts of system design and development
3 = the persons who are interested in ukulele playing with basic computer skills

The comparison in Table 4.14 reveals that the system testing results of all sample groups are similar, i.e., the system usability averaged from all samples of each group is in a good level. The usability testing results of each group are arranged in descending order of average as follows: group 1: the experts of ukulele playing with basic computer skills ($\bar{X}$ = 2.82), group 3: the persons who are interested in ukulele playing with basic computer skills ($\bar{X}$ = 2.80), and group 2: the experts of system design and development ($\bar{X}$ = 2.70).

In group 1: the experts of ukulele playing with basic computer skills, the results of each criterion are arranged in descending order of average as follows: the helpfulness ($\bar{X}$ = 2.94), the affect ($\bar{X}$ = 2.92), the learnability ($\bar{X}$ = 2.90), the control ($\bar{X}$ = 2.76), and the efficiency ($\bar{X}$ = 2.58).

In group 2: the experts of system design and development, the results of each criterion are arranged in descending order of average as follows: the affect ($\bar{X}$ = 2.90), the helpfulness ($\bar{X}$ = 2.72) that is equal to the control ($\bar{X}$ = 2.72) followed by the learnability ($\bar{X}$ = 2.66), and the efficiency ($\bar{X}$ = 2.50).

In group 3: the persons who are interested in ukulele playing with basic computer skills, the results of each criterion are arranged in descending order of average as follows: the affect ($\bar{X}$ = 2.90), the helpfulness ($\bar{X}$ = 2.86) that is equal to the control ($\bar{X}$ = 2.86) followed by the learnability ($\bar{X}$ = 2.76), and the efficiency ($\bar{X}$ = 2.62).

Figure 4.17 visualizes the results of the system usability testing analysis in each sample group.

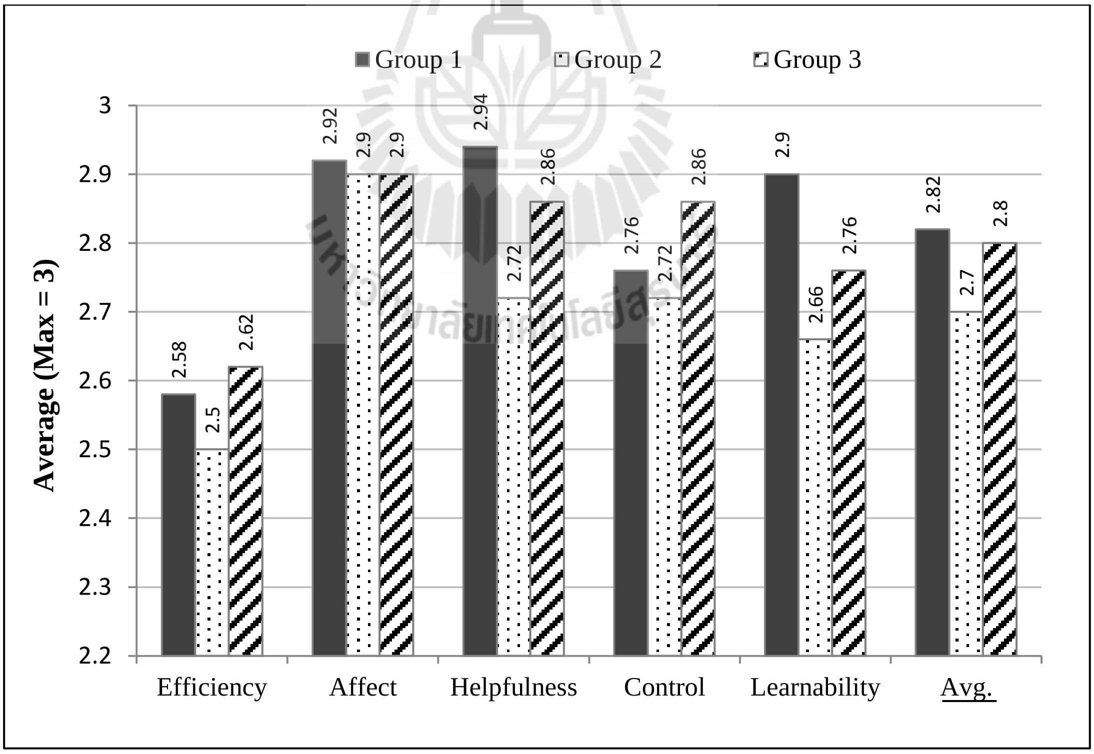


Figure 4.17 The comparison of the system usability testing results in each sample group

As mentioned above, the average of the system usability testing results of all sample groups is similar. However, those similarities cannot refer to a statistically significant difference. Therefore, the significant difference analysis of the average results in each sample group is implemented by using Post-Hoc test in ANOVA statistic (Huck, Cormier, and Bounds, 1974) as shown in Table 4.15.

Table 4.15 The results of the significant difference analysis of the system usability testing in each sample group by Post-Hoc test in ANOVA

Criterion	**Group ID***		**Sig.**	**Group ID***		**Sig.**	**Group ID***		**Sig.**
	1	**2**		**1**	**3**		**2**	**3**	
	$\bar{X}$	$\bar{X}$		$\bar{X}$	$\bar{X}$		$\bar{X}$	$\bar{X}$	
Efficiency	2.58	2.50	**0.680**	2.58	2.62	**0.836**	2.50	2.62	**0.537**
Affect	2.92	2.90	**0.826**	2.92	2.90	**0.826**	2.90	2.90	**1.000**
Helpfulness	2.94	2.72	**0.039****	2.94	2.86	**0.436**	2.72	2.86	**0.178**
Control	2.76	2.72	**0.608**	2.76	2.86	**0.206**	2.72	2.86	**0.081**
Learnability	2.90	2.66	**0.112**	2.90	2.76	**0.347**	2.66	2.76	**0.500**
Avg.	**2.82**	**2.70**	**0.049****	**2.82**	**2.80**	**0.741**	**2.70**	**2.80**	**0.100**

** Group ID: 1 = the experts of ukulele playing with basic computer skills*
2 = the experts of system design and development
3 = the persons who are interested in ukulele playing with basic computer skills

*** Significant difference (Sig. < 0.05)*

Table 4.15 presents the results of the significant difference analysis of the system usability testing in each sample group by Post-Hoc test in ANOVA. The results show there is a significant difference between the sample group 1 and 2 at the significance level of 0.05 in the term of overall system usability and the criterion of helpfulness. The results can be described that the experts of ukulele playing (group 1) requires the system helpfulness and the other capability less than the experts of system design and development (group 2) because they have an experience and expertise in ukulele playing. However, the results of the significant difference analysis in the other sample group pair are not obtained the significant difference.

The overall results of the usability testing can be summarized that the UkeMe3D system is achieved in the good level with all sample groups and these results can be claimed that all sample groups are in the same opinion direction.

4.3.2 The Usability Testing Results using the Thinking-Aloud Protocol

This section presents the results which are recorded from the think out loud data or speeches of the samples during system testing. The thinking-aloud protocol receives more than 60 difference opinions of all samples. These opinions are summarized with a short description and separated into five topics based on the criteria of the system usability as follows:

4.3.2.1 Efficiency: The opinions in the term of the system efficiency can be concluded as shown in Table 4.16.

In Table 4.16, most opinions refer to the user emotion that the system is empowering the function to be completed in an effective. The performance of song recognition is often positively mentioned. Moreover, the GUI with the 3D animation is acclaimed that it is very exciting and attractive. In contrast, the negative opinions refer to the system improvement, such as the new function of audio controller, the increasing of realistic animation, etc.

Table 4.16 The opinions in the term of the system efficiency

No.	Opinion	Opinion Type*
1	The 3D musician looks like a real human.	+
2	The 3D ukulele neck is transparent, and it makes a new viewpoint easy to observe the finger position.	+
3	The changing chord animation is consistent with ukulele sound.	+
4	The Graphical User Interface (GUI) is very impressive.	+
5	The left and the right hand animation is consistent with ukulele song sound.	+
6	The posture of touching chord of 3D musician is realistic. Users can imitate touching chord style.	+
7	The sound feature extraction is the highlight of the system.	+
8	The system is easy to use and it does not want the technical ability from users.	+
9	The system is fast loading and processing.	+
10	The system presents itself in a very attractive way.	+
11	The system shows the correct information of chord and strumming pattern.	+
12	In some ukulele songs, the strumming animation of the right hand is not suitable with the sound.	–
13	In the back view of the left hand, the point lights should be minimized because they hide the fingertips.	–
14	The font family used in the GUI should use only one family.	–
15	The string vibration may create more realistic animation.	–
16	The strumming animation should enlarge the arrows for referring to the strumming direction.	–
17	The system should allow users to import MP3 ukulele song with a singer's voice.	–
18	The system should allow users to increase the bass sound.	–

** Opinion types: (+) = Positive opinion, (-) = Negative opinion*

4.3.2.2 Affect: The opinions in the term of the system affect can be concluded as shown in Table 4.17.

Table 4.17 The opinions in the term of the system affect

No.	Opinion	Opinion Type*
1	An automatic 3D animation is very interesting and innovative.	+
2	The good performance of the system makes users want to use frequently.	+
3	The method of 3D view controlling is very exciting. It is difference from the traditional learning system.	+
4	The system helps users to practice the ukulele in the easy way.	+
5	Users can use the system instead of other resources for analyzing chord and strumming information.	+
6	Users enjoy the time during using this system.	+
7	In front of the 3D stage of the output screen, it should be decorated with other 3D objects, such as seats or viewers.	–
8	The 3D model should be changed posture and facial expression for increasing a user friendly.	–
9	The 3D musician should sit on the sofa to make more relaxing.	–

** Opinion types: (+) = Positive opinion, (-) = Negative opinion*

In Table 4.17, most opinions refer to user feeling and user satisfaction interacting with the system. The UkeMe3D helps users to practice the ukulele in the easy way and they feel pleasant during using the system. Furthermore, the performance of 3D view controller is dissimilar from the existing systems about string instrument playing. In contrast, the negative opinions refer to the system improvement, such as the additional decoration in 3D stage, the changing posture and facial expression of 3D musician, etc.

4.3.2.3 Helpfulness: The opinions in the term of the system helpfulness can be concluded as shown in Table 4.18.

Table 4.18 The opinions in the term of the system helpfulness

No.	Opinion	Opinion Type*
1	A loading bar is advantageous because it visualizes the progression of the system operation.	+
2	An introductory description is easy to conceive for the first reading.	+
3	The instructions and prompts are helpful.	+
4	The method that system information is presented is clear and understandable.	+
5	The system notification and confirmation are displayed in the proper situation.	+
6	A cursor image should be changed from arrow to hand on the buttons.	–
7	The system should add intensive introductory information for the first time when users open the system.	–
8	In the introduction page, it should present the basic skill information of ukulele playing because it is necessary for beginners.	–
9	Some buttons should show a performance label when touching by cursor.	–
10	Some progress information on loading page is incomprehensible, such as “extracting bar”, “extracting beat”, etc.	–

** Opinion types: (+) = Positive opinion, (-) = Negative opinion*

In Table 4.18, most opinions refer to the suitability of the help information such as the easy reading of the introductory description, the comfort of the system notification and confirmation, the attention of the loading bar description during the system is loaded, etc. In contrast, the negative opinions refer to the system improvement, such as the addition of the ukulele basic skill information on the home screen for beginners, etc.

4.3.2.4 Control: The opinions in the term of the system control can be concluded as shown in Table 4.19.

Table 4.19 The opinions in the term of the system control

No.	Opinion	Opinion Type*
1	It is easy to control the system to do exactly what users want.	+
2	The 3D view controlling is freedom and suitable to use	+
3	The audio controller is active and good performance.	+
4	The menu organization is logical.	+
5	The speed of the system is fast enough.	+
6	The system allows users to control the 3D view in any angle, which this mode cannot do in the other video tutorials such as YouTube.	+
7	The users can work all functions on the output screen and it is relatively easy to change to another screen.	+
8	There are some intricateness to control 3D view for the first time.	–
9	The audio controller should be allowed to command the song by pressing spacebar.	–
10	The exit button should be placed on the right top screen corner.	–
11	The file selection feature is not allowed to double click on the filename.	–
12	The output screen should be added a drop down dialogue box for changing an ukulele song without returning to the home screen.	–
13	The song selection page should be added the audio controller.	–
14	The system display resolution is not full on wide screen.	–

** Opinion types: (+) = Positive opinion, (-) = Negative opinion*

In Table 4.19, the opinions in this part refer to the satisfaction of the users for controlling the system. The users rather satisfy in the way for controlling the system in aspects of 3D view controller and audio controller. The speed of the system is fast enough and the regularity of menu structure makes users to control the system in the simple way. In contrast, the negative opinions refer to the system improvement, such as the additional function of the file selection and audio controller.

4.3.2.5 Learnability: The opinions in the term of the system learnability can be concluded as shown in Table 4.20.

Table 4.20 The opinions in the term of the system learnability

No.	Opinion	Opinion Type*
1	It takes short time to learn the system functions.	+
2	Learning how to use the system is not difficult.	+
3	Learning the ukulele by this software is satisfying.	+
4	The system improves the visibility of ukulele learning in many views. It helps users to understand faster.	+
5	The system shows the animation clearly and users can learn quickly.	+
6	The system tutorial is informal but very helpful for learning.	+
7	There is not much information to read before users can use the system.	+
8	There is suitable for beginners to learn ukulele.	+
9	After the song selection step, some users do not know that the next step is pressing the “Uke Me!” button to process.	–
10	In the future, the system can be applied to another type of instrument, such as guitar, violin, and traditional Thai musical instruments.	–
11	Some users have to look for aid sometime when using the system.	–

** Opinion types: (+) = Positive opinion, (-) = Negative opinion*

In Table 4.20, the opinions in this part refer to the effort for learning the operation of the system. The users satisfy in learning to use the system. They take a short time to learn the system functions and they think that UkeMe3D is suitable for beginners to learn the ukulele. In contrast, the negative opinions refer to the system improvement, such as the development of another type of musical instrument for creating a new learning style.

As discussed above, in the thinking-aloud protocol operation, the most samples mention about the system performance and user satisfaction. The operation receives more than 60 difference opinions of all samples. When considering on the quantity between positive and negative opinions, most opinions are about the benefits of the system, such as the advantages for learning, the novelty of the use, and the impressive GUI, etc. These consequences can be referred to the results of the post-study questionnaire, which obtain a good level of system usability by average. Furthermore, the operation receives the suggestions about the limitations of the system. All suggestions have a benefit to update the system performance in the future.

CHAPTER 5

CONCLUSIONS AND RESEARCH RECOMMENDATIONS

This chapter offers a summary of the study. It starts with describing a summary of the research findings. Then the limitation of the study is presented. After that, this chapter explains the application of the study, followed by recommendations for further study.

5.1 Summary of the Research Findings

This research aims to design and develop an automatic 3D animation builder for displaying ukulele playing. It begins with surveying the capability of the existing systems about string instrument playing and reviewing previous research related to music extraction and 3D animation builders for displaying musical instrument playing. The results of studying present the problem statements that are necessary for designing and developing the proposed system. The summary of problem statements consists of two parts. First, most of the existing systems concentrate on basic skills for ukulele playing but they cannot import any audio files, and secondly many systems display an animation in 2D format and users can view from the front view only.

Therefore, this research proposed to develop an animation builder for displaying ukulele playing which can import the audio files and presents the animation in 3D format. Many techniques are applied for developing the 3D animation builder such as machine learning for recognition in music, ukulele playing information recognition, motion capture system, and automatic 3D animation construction.

The development of 3D animation builder consists of two main processes which are information recognition and 3D animation construction.

The first process of development is recognizing information related to the ukulele playing information consisting of three sub-stages.

1) Strumming pattern recognition is the method for analyzing a proper strumming pattern of each query song. The main step of strumming pattern recognition is strumming type prediction and strumming pattern summarization. The model of strumming type prediction is constructed by three machine learning approaches , which are decision tree learning, neural network learning and hidden Markov model. The best prediction model will be applied to the process of strumming pattern summarization.

2) Chord recognition is performed by a Non-Negative Least Squares (NNLS) Chroma and Chordino (Mauch and Dixon, 2010), which is an open source Vamp plugin library for harmony and chord extraction. This recognition is essential for building the chord touching animation. Moreover, the received chord information will be used to recognize chord changing time, which is discussed in the next step.

3) Chord changing time recognition is operated to calculate the appropriate time to create a smooth and realistic ukulele playing. At the beginning, the recognition is based on two presumptions: a) chord changing time seems to reverse with song tempo; and b) the same types of connected chords should take

the same chord changing time in any ukulele song. The results of both presumptions show that the chord changing time is not related to song tempo and the type of connected chords. However, the new solution is performed by estimating the same chord changing time in every song.

The second process is the development of 3D animation builder consisting of three sub-stages. The first sub-stage is building a strumming animation, which is operated by a motion capture system for capturing the musician's right hand motion data in three-dimensional coordinates. The second sub-stage is building a chord touching animation, which is operated by the researcher via the Unity3D software. All ukulele chords will be aligned to the most realistic of the musician's left hand touching. Finally, the third sub-stage is building a chord changing animation, which is operated by using the recognized chord changing time for generating a proper chord changing animation.

The output of three sub-stages is integrated into the new system named UkeMe3D, which allows users to import MP3 ukulele songs for recognizing ukulele playing information and building 3D ukulele playing animation. Users can learn how to play the ukulele and control the viewpoint from any angle.

After the system is developed, usability testing is implemented in three groups of users who are related to ukulele playing with basic computer skills. The first group is experts of ukulele playing. The second group is experts of system design and development. The third group is persons who are interested in ukulele playing. Each group comprises five persons. All users will be tested by two testing techniques consisting of a post-study system usability questionnaire and a thinking-aloud protocol. The system usability testing analysis is based on five criteria: efficiency, affect, helpfulness, control and learnability.

The research findings can be summarized, as follows:

5.1.1 In order to evaluate the process of strumming pattern recognition, the two sub-stages, which are strumming type prediction and strumming pattern summarization, are validated.

The preparatory evaluation of strumming type prediction models uses 10 ukulele songs with different strumming patterns as supplied test set, and employs Precision, Recall and F-measure (Galitsky, 2013) as performance measures. The evaluation results reveal that the decision tree learning-based predictor produces the best accuracy followed by neuron network learning-based predictor and hidden Markov model-based predictor, respectively. The average correctness of decision tree learning-based predictor of 10 test songs is 89.98 per cent of F-measure. Then, the decision tree learning-based predictor is applied in the next process of strumming pattern summarization.

The strumming pattern summarization, which is the specific solution of this research consists of strumming type maximizing and pattern matching. The results of the strumming pattern summarization show that all of the test songs are summarized with the correct strumming patterns.

5.1.2 In case of chord changing time recognition, the two basic presumptions are rejected, i.e., the chord changing time is not related to song tempo and the type of connected chords. However, the new solution is achieved by using the attributes of the first presumption experiment to estimate the chord changing time. The recognized chord changing time is 0.28 seconds or 18 frames of 3D animation based on 60 frames per second. This result is used to create the chord changing animation in the 3D animation construction.

5.1.3 The evaluation results of the post-study system usability questionnaire reveal that the system usability averaged from all samples is in a good level. The best criterion of the system is the affect, which is followed by the helpfulness, the control, the learnability, and the efficiency, respectively.

When considering the significant difference analysis of the system usability testing in each sample group, the results show the significance difference between the sample group 1 (the experts of ukulele playing) and 2 (the experts of system design and development) at the significance level of 0.05 in terms of overall system usability and the criterion of helpfulness. However, the results of the significant difference analysis in the other sample group's pairs do not obtain the significant difference.

In the thinking-aloud protocol operation, most of the samples mention about the system performance and user satisfaction. The operation receives more than 60 difference opinions from all the samples. When considering the quantity of positive and negative opinions, most opinions are about the benefits of the system, especially the performance of strumming pattern and chord recognition. Moreover, the performance of 3D view controlling is dissimilar from the existing string instrument playing systems and this proposed system is suitable for beginners to learn the ukulele. These consequences can refer to the finding results of the post-study questionnaire, which obtain a good level of system usability by average.

5.2 The Limitation of the Study

This section presents the limitation of the development of the automatic 3D animation builder for displaying ukulele playing. The details of limitation are described as follows:

5.2.1 In general, the ukulele has four standard sizes: soprano, concert, tenor, and baritone. The baritone is lower pitched instrument, which differs from the other sizes and it is suitable to the experts of ukulele playing. Consequently, the ukulele playing information recognition of this research is examined on the soprano, concert, and tenor waveform only, which give the high pitched sound level and they are more popular than the baritone.

5.2.2 The designed system lacks the capability of noise removal, such as vocal and the other instrument sounds. Therefore, this proposed UkeMe3D system supports only pure ukulele songs. In other words, the waveform should not combine the singer's voice or other instrument sounds because they may decrease the recognition performance.

5.2.3 The playing style of ukulele can be divided into two types: strumming style and finger style. Strumming style is using the dominant hand strums on all of the strings in the form of the strumming pattern. Nevertheless, the finger style is plucking the strings directly with the dominant fingers and it is not based on the strumming pattern. The UkeMe3D system can be performed with the best performance in the ukulele song with the strumming style playing. Hence, the ukulele songs with the finger style playing cannot be recognized in this system.

5.2.4 The UkeMe3D system is developed to support computer devices with Windows and Mac operating systems. To use the system, the users are required to install the UkeMe3D software on their computer devices. Moreover, the UkeMe3D system is necessary to connect the Internet during the operation for running the extraction scripts. Lacking testing with the different Internet speeds and other platforms, such as Web Player, iOS, Android and Xbox 360, could be the limitation of this study.

5.3 The Application of the Study

The benefit of this research could be useful in the music recognition study. This research investigates the best procedure to perceive ukulele playing information, especially the strumming pattern recognition. The recognition process based on machine learning approaches shows the concept of data preparation and attributes extraction, including the model construction. In addition, the way to recognize the proper time for building chord changing animation is examined in this research. These received methodologies can be applied to other musical instruments.

Furthermore, this research demonstrates many benefits to the musical instrument learning, especially the ukulele. The users can apply the recognized ukulele playing information for supporting their self-studying swiftly and easily by using the proposed system. For the beginners, strumming patterns and chords are important information for practicing the ukulele. The beginners can prepare the favorite ukulele songs for importing into the UkeMe3D system. After that, the system will extract information of the songs and display 3D animation with the corrective and attractive output. The output from the system helps the beginners to practice and understand ukulele playing effectively.

5.4 Recommendations for Further Study

There are some improvements for studying that could be made in the near future as explained below:

5.4.1 Evaluating the system in the terms of e-learning: the system usability testing of this research is proceeded by two testing techniques, consisting of a post-study system usability questionnaire and a thinking-aloud protocol. Both testing techniques emphasize user's perception of the system usability, such as a psychological term for emotional feeling, the level of self-studying satisfaction, the user's effort for learning the handling of the system, etc. These evaluations are not considered in learning achievement because they do not have a process of comparison of the user's skill level between before and after learning. Therefore, the learning achievement can be evaluated in the near future.

5.4.2 Attaching the function of vocal removal and other musical instrument sound removal: many songs are integrated from various musical instruments, such as piano, drum, and bass, including singer's voice. The capability of the other sounds removal reduces the limitation of the system which supports only pure ukulele song. Moreover, it helps expand the diversity of song types for users and improving the system performance. However, the existing research about the noise removal are studied, such as "Karaoke system with spatial acoustics estimation for vocal or instrumental removal" (Xiang and Dubnov, 2005), "A study on vocal separation from mixtured music" (Kim and Park, 2011), etc.

5.4.3 Supporting ukulele songs in the finger style: The finger style playing is interested in the ukulele specialists because it makes sound more melodious than the strumming style. The finger style is often played at the song intro before strumming. For the ukulele specialists, the finger style is applied to the whole song without strumming, and it is rather hard to play because it uses the dominant fingers for plucking the strings directly. However, the finger style song recognition is very interesting to study and it could help the players to practice the advanced ukulele playing style.

5.4.4 Applying the methodology to other types of musical instruments: some types of musical instrument lack lessons or systems supporting for self-studying. In addition, some types of musical instrument lack experts who can teach how to play those instruments, such as some types of Thai traditional musical instruments. Consequently, the methodology of this research can be utilized to the other types of musical instruments. The research methodology is divided into two main processes consisting of the music instrument playing information recognition and the 3D animation construction. Nevertheless, the details of the recognition's overall process depend on types of musical instruments, playing format, and waveform structure, which must be adapted to the finest specific procedure for the recognition.

REFERENCES

Alten, S. (2013). **Audio in media.** London: Wadsworth.

Amanda, H. (2014). **Advantages of Using 3D CGI Programs in Filmmaking** [On-line]. Available: http://beefycow.com/advantages-of-using-3d-cgi-programs-in-filmmaking/

Animation Arena. (2012). **Introduction to 3D Animation** [On-line]. Available: http://www.animationarena.com/introduction-to-3d-animation.html

Barthélemy, J., and Bonardi, A. (2001). Figured Bass and Tonality Recognition. In **Proceeding of 2nd Annual International Symposium on Music Information Retrieval 2001** (pp. 129-136). Indiana, USA.

Beverly, C. (2011). **Ukulele Boom Generates Global Sales for Hawaii Companies** [On-line]. Available: http://www.hawaiibusiness.com/ukulele-boom-generates-global-sales-for-hawaii-companies/

Bishop, C. M. (2006). **Pattern recognition and machine learning.** New York: Springer.

Chai, W., and Vercoe, B. (2001). Folk music classification using hidden Markov models. In **Proceedings of International Conference on Artificial Intelligence** (n.d.). (n.p.).

Chew, E. (2002). The spiral array: An algorithm for determining key boundaries. In **Proceeding of the 2nd International Conference on Music and Artificial Intelligence** (pp. 18-31). Edinburgh, Scotland: Springer Berlin Heidelberg.

Chuan, C. H. (2011). Harmonic style-based song retrieval using N-gram. In **Proceeding of the 1st ACM International Conference on Multimedia Retrieval** (p. 17). New York, USA: ACM.

Davies, M. E., and Plumbley, M. D. (2007). Context-dependent beat tracking of musical audio. Audio, Speech, and Language Processing. **IEEE Transactions**. 15(3): 1009-1020.

Dixon, S. (2001). Automatic extraction of tempo and beat from expressive performances. **Journal of New Music Research**. 30(1): 39-58.

Doszkocs, T. E., Reggia, J., and Lin, X. (1990). Connectionist models and information retrieval. **Annual Review of Information Science and Technology (ARIST)**. 25: 209-262.

Duda, R. O., Hart, P. E., and Stork, D. G. (2012). **Pattern classification.** USA: John Wiley & Sons.

Dymarski, P. (2011). **Hidden Markov Models, theory and applications.** India: InTech Open Access Publishers.

Eargle, J. M. (2013). **Music, Sound, and Technology**. New York, USA: Springer Science & Business Media.

El-Sawah, A., Georganas, N. D., and Petriu, E. M. (2006). Finger inverse kinematics using error model analysis for gesture enabled navigation in virtual environments. In **Proceeding of the 2006 Haptic Audio Visual Environments and Their Applications** (pp. 34-39). Ottawa, Canada: IEEE.

Fausett, L. (1994). **Fundamentals of neural networks: Architectures, algorithms, and applications.** New Jersey: Prentice-Hall.

Feng, Y., Zhuang, Y., and Pan, Y. (2003). Popular music retrieval by detecting mood. In **Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval** (pp. 375-376). New York, USA: ACM.

Galitsky, B. (2013). Machine learning of syntactic parse trees for search and classification of text. **Engineering Applications of Artificial Intelligence.** 26(3): 1072-1091.

Gardner, M. W., and Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron) a review of applications in the atmospheric sciences. **Atmospheric Environment.** 32(14): 2627-2636.

Gillies, M. (2010). **HMMWeka**. [On-line]. Available:http://doc.gold.ac.uk/ ~mas02mg/ software/ hmmweka/#documentation

Guitar Alchemist. (2014). **Guitar Alchemist**. [On-line]. Available: http://www. guitaralchemist.com

Guitar Encyclopedia. (2014). **Guitar Encyclopedia for the Electric and Acoustic Guitar.** [On-line]. Available: http://www.mussidamusic.com

Guitar Scales. (2013). **Guitar Scales.** [On-line]. Available: http://guitar-scales.softpile.com

Gulati, S., and Rao, P. (2010). Rhythm pattern representations for tempo detection in music. In **Proceedings of the First International Conference on Intelligent Interactive Technologies and Multimedia** (pp. 241-244). Allahabad, India: ACM.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA Data Mining Software: an Update. **ACM SIGKDD Explorations Newsletter.** 11(1): 10-18.

Harden, M. J., and Brinkman, S. (1999). **Voices of wisdom: Hawaiian elders speak.** Kula, Hawaii: Aka Press.

Harte, C., Sandler, M., and Gasser, M. (2006). Detecting harmonic change in musical audio. In **Proceeding of the 1st ACM Workshop on Audio and Music Computing Multimedia** (pp. 21-26). New York, USA: ACM.

Haykin, S. (1999). **Neural networks: A comprehensive foundation.** New Jersey: Prentice Hall.

Henery, R. J. (1994). **Classification** [On-line]. Available: http://www1.maths.leeds.ac.uk/~charles/statlog/whole.pdf

Huck, S. W., Cormier, W. H., and Bounds, W. G. (1974). In **Reading Statistics and Research** (pp. 74-102). New York, USA: Harper & Row.

Hu, D., and Saul, L. K. (2009). A Probabilistic Topic Model for Unsupervised Learning of Musical Key-Profiles. In **Proceeding of the Conference of the International Society for Music Information Retrieval** (pp. 441-446). Taiwan: Taipei.

Humphrey, C. (2012). **10 Most Popular Musical Instruments You should Learn to Play** [On-line]. Available: http://1stangel.co.uk/blog/2012/10-most-popular-musical-instruments-you-should-learn-to-play

iPerform 3D. (2014). **iPerform 3D: Rotate Zoom Rock**. [On-line]. Available: http://www.iperform3d.com

Irwin, G. W., Warwick, K., and Hunt, K. J. (1995). **Neural Network Applications in Control**. London: IEE.

Jensen, K., and Arnspang, J. (1999). Binary decision tree classification of musical sounds. In **Proceedings of the 1999 International Computer Music Conference** (n.d.). (n.p.).

Jones, A. N. (2013). **A Cultural Study of Ukulele in the Music Classroom**. M.S. Thesis, The Pennsylvania State University.

Kaiga, T., Yukawa, T., Mitobe, K., Miura, T., Tamamoto, H., and Yoshimura, N. (2007). Magnetic Motion Capture System Measuring Movements of Hands and a Body Simultaneously. In **Proceeding of ACM SIGGRAPH 2007 Posters** (p. 79). California, USA: ACM.

Kaliakatsos-Papakostas, M. A., Floros, A., and Vrahatis, M. N. (2013). A clustering strategy for the key segmentation of musical audio. **Computer Music Journal**. 37(1): 52-69.

Kim, H. T., and Park, J. S. (2011). A Study on Vocal Separation from Mixtured Music. **Journal of Information and Communication Convergence Engineering**. 9(2): 161-165.

Kirakowski, J., and Corbett, M. (1993). SUMI: The Software Usability Measurement Inventory. **British Journal of Educational Technology**. 24(3): 210-212.

Kummong, S. (2013). **Teach yourself the ukulele.** Bangkok: Thanbandit publishing.

Kurihara, K., Hoshino, S. I., Yamane, K., and Nakamura, Y. (2002). Optical motion capture system with pan-tilt camera tracking and realtime data processing. In **Proceeding of the International Conference on Robotics & Automation** (pp. 1241-1248). Washington, DC, USA: IEEE.

Laybourne, K. (1998). **The animation book: A complete guide to animated filmmaking -from flip-books to sound cartoons to 3D animation**. New York, USA: Three Rivers Press.

Lee, C. C., Mower, E., Busso, C., Lee, S., and Narayanan, S. (2011). Emotion recognition using a hierarchical binary decision tree approach. **Speech Communication**. 53(9): 1162-1171.

Lee, K., and Slaney, M. (2006). Automatic Chord Recognition from Audio Using a HMM with Supervised Learning. In **Proceeding of the 7th International Conference on Music Information Retrieval** (pp. 133-137). Victoria, Canada: ACM.

Leinecker, R. (1994). **Making noise: Creating sound on your PC.** New York: A Division of MIS Press.

Liem, C., Müller, M., Eck, D., Tzanetakis, G., and Hanjalic, A. (2011). The need for music information retrieval with user-centered and multimodal strategies. In **Proceeding of the 1st International ACM Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies** (pp. 1-6). New York, USA: ACM.

Lin, C. C., and Liu, D. S. M. (2006). An intelligent virtual piano tutor. In **Proceeding of the 2006 ACM International Conference on Virtual Reality Continuum and its Applications** (pp. 353-356). New York, USA: ACM.

Loy, G. (1985). Musicians make a standard: the MIDI phenomenon. **Computer Music Journal:** 8-26.

Martino, D. M., Hernández, G., Fiori, M., and Fernández, A. (2013). A new framework for optimal classifier design. **Pattern Recognition.** 46(8): 2249-2255.

Mauch, M., and Dixon, S. (2010). Approximate Note Transcription for the Improved Identification of Difficult Chords. In **Proceeding of 2010 International Society for Music Information Retrieval** (pp. 135-140). (n.d.).

McKinney, M. F., and Moelants, D. (2004). Extracting the perceptual tempo from music. In **Proceedings of 5th International Conference on Music Information Retrieval** (n.d.). Barcelona, Spain.

McQueen, B. (n.d.). **Ukulele sizes - Soprano, Concert, Tenor & Baritone** [On-line]. Available: http://www.ukuleletricks.com/ukulele-sizes-soprano-concert-tenor-baritone

McVicar, M., and De Bie, T. (2010). Enhancing chord recognition accuracy using web resources. In **Proceeding of 3rd International Workshop on Machine Learning and Music** (pp. 41-44). New York, USA: ACM.

Menache, A. (2011). **Understanding motion capture for computer animation**. Massachusetts, USA: Morgan Kaufmann.

Meta Motion. (n.d.). **Magnetic Motion Capture Systems** [On-line]. Available: http://www.metamotion.com/motion-capture/magnetic-motion-capture-1.htm

Mitobe, K., Kaiga, T., Yukawa, T., Miura, T., Tamamoto, H., Rodgers, A., and Yoshimura, N. (2006). Development of a motion capture system for a hand using a magnetic three dimensional position sensor. In **Proceeding of ACM SIGGRAPH 2006** (p. 102). New York, USA: ACM.

Morman, J., and Rabiner, L. (2006). A system for the automatic segmentation and classification of chord sequences. In **Proceeding of the 1st ACM Workshop on Audio and Music Computing Multimedia** (pp. 1-10). New York, USA: ACM.

Murphy, K. P. (2002). **Dynamic bayesian networks: representation, inference and learning**. Doctoral Dissertation, University of California, Berkeley.

Ng, K. C., Weyde, T., Larkin, O., Neubarth, K., Koerselman, T., and Ong, B. (2007). 3D augmented mirror: a multimodal interface for string instrument learning and teaching with gesture support. In **Proceeding of the 9th International Conference on Multimodal Interfaces** (pp. 339-345). New York, USA: ACM.

Nielsen, J. (1992). Evaluating the thinking-aloud technique for use by computer scientists. In **Proceeding of the Advances in Human-Computer Interaction** (vol. 3) (pp. 69-82). Ablex Publishing Corp.

Nielsen, J. (1994). Estimating the number of subjects needed for a thinking aloud test. **International Journal of Human-Computer Studies.** 41(3): 385-397.

Nielsen, J. (1997). **Discount usability for the web.** [On-line]. Available: http://www.nngroup.com/articles/web-discount-usability

Nielsen, J. (2012). **Thinking aloud: The #1 usability tool.** [On-line]. Available: http://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool

Nørgård, P. M., Ravn, O., Poulsen, N. K., and Hansen, L. K. (2000). **Neural Networks for Modelling and Control of Dynamic Systems-A Practitioner's Handbook.** London: Springer-London.

Noureldin, A., El-Shafie, A., and Taha, M. R. (2007). Optimizing neuro-fuzzy modules for data fusion of vehicular navigation systems using temporal cross-validation. **Engineering Applications of Artificial Intelligence**. 20(1): 49-61.

Paiement, J. F., Eck, D., and Bengio, S. (2005). A probabilistic model for chord progressions. In **Proceeding of the the 6th International Conference on Music Information Retrieval** (n.d.). London, United Kingdom.

Pang-Ning, T., Steinbach, M., and Kumar, V. (2006). **Introduction to data mining.** Pearson education, USA: Boston.

Pardo, B., and Birmingham, W. P. (2002). Algorithms for chordal analysis. **Computer Music Journal.** 26(2): 27-49.

Parent, R. (2008). **Computer animation: Algolithms and techniques**. Massachusetts, USA: Morgan Kaufmann.

Patil, T. R., and Sherekar, S. S. (2013). Performance analysis of Naive Bayes and J48 classification algorithm for data classification. **International Journal of Computer Science and Applications**. 6(2): 256-261.

Pauws, S. (2004). Musical key extraction from audio. In **Proceeding of the International Society for Music Information Retrieval** (n.d.). (n.p.).

Perez Carrillo, A., and Wanderley, M. M. (2012). Learning and extraction of violin instrumental controls from audio signal. In **Proceeding of the Second International ACM Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies** (pp. 25-30). New York, USA: ACM.

Pérez-García, T., Pérez-Sancho, C., and Iñesta, J. M. (2010). Harmonic and instrumental information fusion for musical genre classification. In **Proceedings of 3rd International Workshop on Machine Learning and Music** (pp. 49-52). Firenze, Italy: ACM.

Perlmutter, A. (2011). Make Room for Ukuleles**. Teaching Music.** 19(1): 56.

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. **Proceedings of The IEEE**.77(2): 257-286.

Raphael, C., and Stoddard, J. (2003). Harmonic analysis with probabilistic graphical models. In **Proceedings of the 9th International Conference on Music Information Retrieval** (n.d.). Maryland, USA.

Ripley, B. D. (1996). **Pattern recognition and neural networks.** Cambridge: University Press.

Rowan, G. (2011). **Motion Capture Explained** [On-line]. Available: http://computerstories.net/motion-capture-explained

Ruck, D. W., Rogers, S. K., and Kabrisky, M. (1990). Feature selection using a multilayer perceptron. **Journal of Neural Network Computing**. 2(2): 40-48.

Samarasinghe, S. (2006). **Neural networks for applied sciences and engineering: from fundamentals to complex pattern recognition.** New York, USA: CRC Press.

Sanders, A. L. (2013). **3D Animation** [On-line]. Available: http://animation.about. com/od/glossaryofterms/g/3danim_def.htm

Sapp, C. S. (2015). **Mz Power Curve: Measures the power over time of an audio signal** [On-line]. Available: http://sv.mazurka.org.uk/MzPowerCurve

Sauer, D., and Yang, Y. H. (2009). Music-driven character animation. **ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)**. 5(4): 27.

Schmidt-Jones, C. (2008). **The basic elements of music**. Texas: Connexions.

Scholz, R., and Ramalho, G. (2008). COCHONUT: Recognizing Complex Chords from MIDI Guitar Sequences. In **Proceedings of the 9th International Conference on Music Information Retrieval** (pp. 27-32). Kobe, Japan.

Shao, X., Xu, C., and Kankanhalli, M. S. (2004). Unsupervised classification of music genre using hidden markov model. In **IEEE International Conference on Multimedia** (pp. 2023-2026). (n.p.).

Sheh, A., and Ellis, D. P. (2003). Chord segmentation and recognition using EM-trained hidden Markov models. **ISMIR 2003**. 185-191.

Shenoy, A., and Wang, Y. (2005). Key, chord, and rhythm tracking of popular music recordings. **Computer Music Journal**. 29(3): 75-86.

Singhal, S., and Wu, L. (1988). Training Multilayer Perceptrons with the Extende Kalman Algorithm. In **NIPS** (pp. 133-140). (n.p.).

Smith, T. M. (2012). **Sound engineering explained.** Oxford: Focal Press.

Sparga, M. (2008). **What is 3D Animation?** [On-line]. Available: http://www.helium.com/items/986482-what-is-3d-animation

Steve, C. (2002). **Speech Recognition** [On-line]. Available: http://web.science.mq. edu. au /~cassidy/comp449/html/index.html

Swangoblues. (2014). **Optical Motion Capture System** [On-line]. Available: http://www.swangoblues.com/chrisarts/FineArtImages/Mocap.jpg

Tiesel, J. P., and Loviscach, J. (2006). A mobile low-cost motion capture system based on accelerometers. **ISVC 2006**. 36(1): 437-446.

Ukuguides. (2012). **The Ukulele Buying Guide** [On-line]. Available: https://ukuguides.com/guides/the-ukulele-buying-guide

Ukulele Beatles Fun!. (2003). **Ukulele Beatles Fun!: self-improvement for free.** [On-line]. Available: http://www.beatlesite.info

Ukulele Chord Finder. (2011). **Ukulele Chord Finder.** [On-line]. Available: http://www.ukulele.nl/chordfinder.php

Ukulele-Tabs. (2013). **Ukulele Chord Charts for Soprano, Concert, Tenor, Baritone** [On-line]. Available: http://www.ukulele-tabs.com/ukulele-chords-chart.html

Vince, J. (1992). **3-D Computer Animation.** British: Addison-Wesley Longman Publishing.

Virtual Ukulele Online. (2014). **Virtual Ukulele Soprano Online.** [On-line]. Available: http://www.ecarddesignanimation.com/home/virtual_ukulele_soprano.php

Wang, J., Wu, Q., Deng, H., and Yan, Q. (2008). Real-time speech/music classification with a hierarchical oblique decision tree. In **IEEE International Conference on Acoustics, Speech and Signal Processing 2008** (pp. 2033-2036). China: Beijing.

Wang, Y. S., and Wechsler, H. (2012). Musical keys and chords recognition using unsupervised learning with infinite Gaussian mixture. In **Proceeding of the 2nd ACM International Conference on Multimedia Retrieval** (p. 30). New York: USA: ACM.

Webster, C. (2005). **Animation: The machine of motion.** Massachusetts, USA: Focal Press.

Witten, I. H., Frank, E., and Hall, M. A. (2011). **Data Mining: Practical machine learning tools and techniques – Third edition.** Morgan Kaufmann, USA: Burlington.

Witten, I. H., and Frank, E. (2005). **Data Mining: Practical machine learning tools and techniques.** Morgan Kaufmann, USA: California.

Xia, G., Dannenberg, R., Tay, J., and Veloso, M. (2012). Autonomous robot dancing driven by beats and emotions of music. In **Proceeding of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1** (pp. 205-212). Richland, South Carolina: International Foundation for Autonomous Agents and Multiagent Systems.

Xiang, P., and Dubnov, S. (2005). Karaoke system with spatial acoustics estimation for vocal or instrumental removal. In **International Computer Music Conference (ICMC)** (n.d.). (n.p.).

Xiao, Y., and Shengfeng, Q. (2009). Generating 3D Architectural Models Based on Hand Motion and Gesture. **Computers in Industry.** 60: 6667- 6685.

Xiong, W., Lu, H., Fang, C., and Wang, Z. (2010). Research and implementation of instant collaborative development for 3D animation production. In **Proceeding of 2010 2nd International Workshop on Intelligent Systems and Applications** (pp. 1-4). Wuhan, China: IEEE.

Yang, Y. H., and Chen, H. H. (2012). Machine recognition of music emotion: a review. **ACM Transactions on Intelligent Systems and Technology (TIST)**. 3(3): 40.

Yasui, B. K. (2008). The Jumping Flea: Tales of the Modern Ukulele. **Journal of the Society for American Music.** 2(1): 134-135.

Yin, J., Wang, Y., and Hsu, D. (2005). Digital violin tutor: an integrated system for beginning violin learners. In **Proceeding of the 13th Annual ACM International Conference on Multimedia** (pp. 976-985). New York, USA: ACM.

Zarkami, R. (2011). Application of classification trees-J48 to model the presence of roach (Rutilus rutilus) in rivers. **Caspian Journal of Environmental Science (CJES)**. 9: 189-198.

Zhang, X., Biswas, D. S., and Fan, G. (2010). A software pipeline for 3D animation generation using mocap data and commercial shape models. In **Proceeding of the ACM International Conference on Image and Video Retrieval** (pp. 350-357). New York, USA: ACM.

Zhu, Y., Manders, C. M., Farbiz, F., and Rahardja, S. (2009). Music feature analysis for synchronization with dance animation. In **TENCON 2009-2009 IEEE Region 10 Conference** (pp. 1-5). Singapore: IEEE.

Zhu, Y., Ramakrishnan, A. S., Hamann, B., and Neff, M. (2012). A system for automatic animation of piano performances. **Computer Animation and Virtual Worlds.** 24(5): 445-457.

APPENDIX A

The Questions in the Post-Study System Usability Questionnaire

The post-study system usability questionnaire is applied from the concept of Software Usability Measurement Inventory or SUMI (Kirakowski and Corbett, 1993). The questionnaire consists of 50 questions which are shifted by five criteria: 1) Efficiency, 2) Affect, 3) Helpfulness, 4) Control and 5) Learnability). Moreover, the questions are mixed between positive and negative questions. Table A.1 presents the details of the questions in the post-study system usability questionnaire.

Table A.1 The questions in the post-study system usability questionnaire

No.	Question	Question Type*	Criteria No.**
1	This software responds too slowly to inputs.	-	1
2	I would recommend this software to my colleagues.	+	2
3	The instructions and prompts are helpful.	+	3
4	This software has at some time stopped unexpectedly.	-	4
5	Learning to operate this software initially is full of problems.	-	5
6	I sometimes don't know what to do next with this software.	-	1
7	I enjoy the time I spend using this software.	+	2
8	I find that the help information given by this software is not very useful.	-	3
9	If this software stops it is not easy to restart it.	-	4
10	It takes too long to learn the software functions.	-	5
11	I sometimes wonder if I am using the right function.	-	1
12	Working with this software is satisfying.	+	2

* *Question types: (+) = Positive question, (−) = Negative question*
** *Criteria No.: 1 = Efficiency, 2 = Affect, 3 = Helpfulness, 4 = Control and 5 = Learnability*

Table A.1 The questions in the post-study system usability questionnaire (cont.)

No.	Question	Question Type*	Criteria No.**
13	The way that system information is presented is clear and understandable.	+	3
14	I feel safer if I use only a few familiar functions.	+	4
15	The software documentation is very informative.	-	5
16	This software seems to disrupt the way I normally like to arrange my work.	-	1
17	Working with this software is mentally stimulating.	+	2
18	There is never enough information on the screen when it's needed.	-	3
19	I feel in command of this software when I am using it.	+	4
20	I prefer to stick to the functions that I know best.	-	5
21	I think this software is inconsistent.	-	1
22	I would not like to use this software every day.	-	2
23	I can understand and act on the information provided by this software.	+	3
24	This software is awkward when I want to do something which is not standard.	-	4
25	There is too much to read before you can use the software.	-	5
26	Tasks can be performed in a straight forward manner using this software.	+	1
27	Using this software is frustrating.	-	2
28	The software has helped me overcome any problems I have had in using it.	+	3
29	The speed of this software is fast enough.	+	4

* *Question types: (+) = Positive question, (−) = Negative question*
** *Criteria No.: 1 = Efficiency, 2 = Affect, 3 = Helpfulness, 4 = Control and 5 = Learnability*

Table A.1 The questions in the post-study system usability questionnaire (cont.)

No.	Question	Question Type*	Criteria No.**
30	I keep having to go back to look at the guides.	-	5
31	It is obvious that user needs have been fully taken into consideration.	+	1
32	There have been times in using this software when I have felt quite tense.	-	2
33	The organization of the menus seems quite logical.	+	3
34	The software allows the user to be economical of keystrokes.	+	4
35	Learning how to use new functions is difficult.	-	5
36	There are too many steps required to get something to work.	-	1
37	I think this software has sometimes given me a headache.	-	2
38	Error messages are not adequate.	-	3
39	It is easy to make the software do exactly what you want.	+	4
40	I will never learn to use all that is offered in this software.	-	5
41	The software hasn't always done what I was expecting.	-	1
42	The software presents itself in a very attractive way.	+	2
43	Either the amount or quality of the help information varies across the system.	+	3
44	It is relatively easy to move from one part of a task to another.	+	4

* *Question types: (+) = Positive question, (−) = Negative question*
** *Criteria No.: 1 = Efficiency, 2 = Affect, 3 = Helpfulness, 4 = Control and 5 = Learnability*

Table A.1 The questions in the post-study system usability questionnaire (cont.)

No.	Question	Question Type*	Criteria No.**
45	It is easy to forget how to do things with this software.	-	5
46	This software occasionally behaves in a way which can't be understood.	-	1
47	This software is really very awkward.	-	2
48	It is easy to see at a glance what the options are at each stage.	+	3
49	Getting data files in and out of the system is not easy.	-	4
50	I have to look for assistance most times when I use this software.	-	5

* *Question types: (+) = Positive question, (−) = Negative question*
** *Criteria No.: 1 = Efficiency, 2 = Affect, 3 = Helpfulness, 4 = Control and 5 = Learnability*

APPENDIX B

The Post-Study System Usability Questionnaire (Thai Version)

แบบสอบถามเพื่อการวิจัย

เรื่อง การใช้งานได้ของระบบสร้างภาพเคลื่อนไหว 3 มิติอัตโนมัติเพื่อแสดงผลการเล่นอูคูเลเล่

แบบสอบถามนี้เป็นส่วนหนึ่งของการศึกษาในหลักสูตรปริญญาดุษฎีบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ สำนักวิชาเทคโนโลยีสังคม มหาวิทยาลัยเทคโนโลยีสุรนารี เรื่อง การพัฒนาระบบสร้างภาพเคลื่อนไหว 3 มิติอัตโนมัติเพื่อแสดงผลการเล่นอูคูเลเล่ (The Development of an Automatic 3D Animation Builder for Displaying Ukulele Playing) โดยงานวิจัยดังกล่าวได้พัฒนาระบบที่มีชื่อว่า “UkeMe3D” ซึ่งมีความสามารถในการนำแฟ้มเพลงอูคูเลเล่มาสกัดคุณสมบัติทางด้านเสียงเพื่อแสดงภาพเคลื่อนไหวการเล่นอูคูเลเล่ในรูปแบบ 3 มิติ

แบบสอบถามนี้เป็นการประเมินความสามารถในการใช้งานได้ของระบบดังกล่าวภายหลังจากที่ท่านได้ทดลองใช้งานเป็นที่เรียบร้อยแล้ว ผู้วิจัยจึงใคร่ขอความร่วมมือจากท่านในการตอบแบบสอบถามตามความเป็นจริง โดยข้อมูลทั้งหมดที่ท่านตอบแบบสอบถามในครั้งนี้ จะถูกนำไปใช้เพื่อประโยชน์ในเชิงวิชาการและจะถูกเก็บไว้เป็นความลับโดยไม่มีผลกระทบใด ๆ ต่อท่าน

ขอขอบพระคุณทุกท่านที่สละเวลาในการตอบแบบสอบถามในครั้งนี้

นายธวัชพงษ์ พิทักษ์
สาขาวิชาเทคโนโลยีสารสนเทศ
สำนักวิชาเทคโนโลยีสังคม มหาวิทยาลัยเทคโนโลยีสุรนารี

<u>คำชี้แจง</u>

แบบสอบถามฉบับนี้ มีทั้งหมด 4 หน้า แบ่งออกเป็น 3 ส่วน ดังนี้

ส่วนที่ 1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม

ส่วนที่ 2 การประเมินความสามารถในการใช้งานได้ของระบบ UkeMe3D

ส่วนที่ 3 ข้อเสนอแนะอื่น ๆ

ส่วนที่ 1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม

<u>คำชี้แจง</u> โปรดทำเครื่องหมาย ✓ ในช่องว่าง ☐ หน้าข้อความที่ตรงกับข้อมูลของท่าน

1. **เพศ** ☐ ชาย ☐ หญิง
2. **อายุ** ☐ ต่ำกว่า 16 ปี ☐ 16 – 25 ปี ☐ 26 – 35 ปี
   ☐ 36 – 45 ปี ☐ 46 ปีขึ้นไป
3. **ระดับการศึกษา**
   ☐ ประถมศึกษา ☐ มัธยมศึกษาตอนต้น ☐ มัธยมศึกษาตอนปลาย / ปวช.
   ☐ อนุปริญญา / ปวส. ☐ ปริญญาตรี ☐ ปริญญาโท
   ☐ ปริญญาเอก ☐ อื่น ๆ โปรดระบุ ....

ส่วนที่ 2 การประเมินความสามารถในการใช้งานได้ของระบบ UkeMe3D

<u>คำชี้แจง</u> พิจารณาข้อความในแต่ละข้อ แล้วทำเครื่องหมาย ✓ ในช่องระดับความคิดเห็นที่ตรงกับความเห็นของท่านมากที่สุด โดยคำถามแบ่งเป็น 5 ส่วน ๆ ละ 10 ข้อ รวม 50 ข้อ

ข้อที่	ข้อความพิจารณา	ระดับความคิดเห็น		
		เห็นด้วย	*ไม่แน่ใจ*	*ไม่เห็นด้วย*
1	ระบบนี้มีการตอบสนองที่ช้าเกินไป			
2	คุณจะแนะนำระบบนี้ให้กับคนรู้จัก			
3	คำแนะนำและการแจ้งเตือนในระบบนี้ มีประโยชน์ต่อคุณ			
4	ในบางครั้งระบบหยุดการทำงานโดยไม่คาดคิด			
5	การเรียนรู้ที่จะใช้งานระบบนี้ในครั้งแรก เต็มไปด้วยปัญหา			
6	บางครั้งในการใช้งาน คุณไม่ทราบว่าต้องทำอย่างไรต่อ			
7	คุณรู้สึกสนุกและมีส่วนร่วมในขณะที่ใช้ระบบนี้			
8	คุณพบว่าข้อมูลความช่วยเหลือที่ระบบแสดง ไม่มีประโยชน์อย่างมาก			
9	เป็นเรื่องยากที่จะเริ่มต้นใหม่ หากระบบนี้หยุดการทำงาน			
10	ต้องใช้เวลานานเกินไปที่จะเรียนรู้การใช้งานระบบ			
11	บางครั้งคุณสงสัยว่า คุณใช้งานระบบได้อย่างถูกต้องหรือไม่			
12	การใช้งานระบบนี้เป็นที่น่าพอใจ			

ข้อที่	ข้อความพิจารณา	ระดับความคิดเห็น		
		เห็นด้วย	*ไม่แน่ใจ*	*ไม่เห็นด้วย*
13	รูปแบบการแสดงข้อมูลมีความชัดเจนและเข้าใจได้			
14	คุณจะรู้สึกปลอดภัยมากขึ้น ถ้าในการใช้งานระบบ ไม่ต้องใช้คำสั่งอะไรมาก			
15	คำแนะนำประกอบการใช้งานระบบมีเนื้อหามากเกินไป			
16	ระบบนี้ส่งผลกระทบต่อการทำงานแบบเดิมของคุณในปัจจุบัน			
17	การทำงานของระบบนี้ช่วยกระตุ้นความสนใจให้กับคุณ			
18	ระบบแสดงข้อมูลไม่เพียงพอกับความต้องการ			
19	คุณเข้าใจในคำสั่งของระบบ			
20	คุณมักจะยึดติดอยู่กับอุปกรณ์หรือเครื่องมือที่คุณรู้จักดีอยู่แล้ว□□			
21	คุณคิดว่าการทำงานของระบบนี้ไม่สอดคล้องกัน			
22	คุณคิดว่าคุณไม่ชอบที่จะใช้ระบบนี้ทุกวัน			
23	คุณมีความเข้าใจและใช้ระบบตามข้อมูลที่ระบบจัดหาไว้ให้			
24	ระบบนี้ทำให้คุณรู้สึกอึดอัดใจเมื่อคุณต้องการทำบางสิ่งบางอย่างที่ไม่ปกติ			
25	มีข้อมูลจำนวนมากที่ต้องอ่าน ก่อนที่คุณจะใช้ระบบนี้เป็น			
26	คุณสามารถทำในสิ่งที่ต้องการได้อย่างตรงไปตรงมา			
27	การใช้ระบบนี้เป็นที่น่าผิดหวัง□			
28	ระบบนี้ช่วยให้คุณเอาชนะปัญหาบางอย่างได้			
29	ระบบนี้มีความรวดเร็วเพียงพอในการประมวลผล			
30	บางครั้งคุณต้องกลับไปดูคำแนะนำ			
31	ระบบสามารถทำงานได้ตรงกับความต้องการของคุณ			
32	บางช่วงเวลาขณะที่ใช้ระบบนี้ คุณค่อนข้างเครียด □ □			
33	โครงสร้างของเมนูหรือหัวข้อรายการจัดเรียงได้อย่างสมเหตุสมผล			
34	ในการใช้งานระบบ คุณไม่จำเป็นต้องพิมพ์ข้อความจำนวนมาก			
35	การเรียนรู้วิธีใช้งานฟังก์ชั่นการทำงานต่าง ๆ ของระบบเป็นเรื่องยาก			
36	มีขั้นตอนมากเกินไปในการทำงานบางอย่าง□			

ข้อที่	ข้อความพิจารณา	ระดับความคิดเห็น		
		เห็นด้วย	*ไม่แน่ใจ*	*ไม่เห็นด้วย*
37	คุณคิดว่าระบบนี้ทำให้คุณรู้สึกปวดหัวในบางครั้ง□			
38	ข้อความแจ้งเตือนเพื่อป้องกันข้อผิดพลาด มีไม่เพียงพอ			
39	ระบบนี้ทำให้คุณได้รับสิ่งที่ต้องการโดยง่าย□			
40	คุณไม่ได้เรียนรู้ทุกสิ่งทุกอย่างที่นำเสนอในระบบนี้			
41	ระบบไม่สามารถทำบางสิ่งบางอย่างที่คุณคาดหวังได้			
42	สิ่งที่ระบบนำเสนอมีความน่าสนใจเป็นอย่างมาก□ □			
43	ข้อมูลให้ความช่วยเหลือมีคุณภาพและกระจายอยู่ทั่วทั้งระบบ			
44	เป็นเรื่องง่ายที่จะสลับหน้าจอการทำงานไปยังส่วนอื่นของระบบ			
45	เป็นเรื่องง่ายที่คุณจะลืมวิธีการใช้งานระบบนี้□			
46	ในบางครั้ง ระบบนี้แสดงอาการบางอย่างที่คุณไม่เข้าใจ □			
47	การใช้งานระบบนี้เป็นที่น่าอึดอัดใจอย่างมาก □			
48	เป็นเรื่องง่ายที่จะเห็นวิธีการหรือตัวเลือกในการใช้งานแต่ละขั้นตอนได้อย่างรวดเร็ว□			
49	การนำเข้าและการแสดงผลข้อมูลในระบบเป็นเรื่องยาก□			
50	คุณต้องการความช่วยเหลือหลายครั้งเมื่อใช้ระบบนี้□ □			

ส่วนที่ 3 ข้อเสนอแนะอื่นๆ

<u>คำชี้แจง</u> โปรดแสดงความเห็นหรือข้อเสนอแนะของท่าน

1. คุณคิดว่าคุณจะใช้ระบบนี้เพื่อวัตถุประสงค์ใด

....

....

....

....

....

....

....

....

2. สิ่งใดที่คุณคิดว่าเป็นสิ่งที่ดีที่สุดของระบบนี้ / เพราะอะไร

....

....

....

....

....

....

....

....

3. สิ่งใดที่คุณคิดว่าเป็นสิ่งที่ควรปรับปรุงหรือเพิ่มเติมในระบบนี้ / เพราะอะไร

....

....

....

....

....

....

....

....

✧ *ขอขอบพระคุณที่สละเวลาและให้ความร่วมมือในการตอบแบบสอบถามนี้* ✧

CURRICULUM VITAE

Mr. Thawatphong Phithak was born on October 19, 1986 in Nakhon Ratchasima Province, Thailand. He received Bachelor and Master of Information Science from Suranaree University of Technology, Thailand in 2007 and 2009, respectively. In 2012, he has got a scholarship from Suranaree University of Technology research to pursue his doctoral degree in Information Technology Program at Suranaree University of Technology. His major research interests focus on feature extraction, classification and 3D animation. He is working on the design and development of an automatic 3D animation builder.