

國立交通大學

多媒體工程研究所

碩士論文

以三維模式轉換技術作二維虛擬人臉之
自動產生及其應用

Automatic 2D Virtual Face Generation by 3D Model
Transformation Techniques and Applications

研究生：張依帆

指導教授：蔡文祥 教授

中華民國九十六年六月

以三維模式轉換技術作二維虛擬人臉之自動產生及其應用
Automatic 2D Virtual Face Generation by 3D Model Transformation
Techniques and Applications

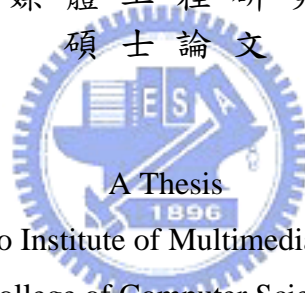
研 究 生：張依帆

Student: Yi-Fan Chang

指 導 教 授：蔡文祥

Advisor: Prof. Wen-Hsiang Tsai

國 立 交 通 大 學
多 媒 體 工 程 研 究 所
碩 士 論 文



Submitted to Institute of Multimedia Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

June 2007

Hsinchu, Taiwan, Republic of China

中華民國九十六年六月

以三維模式轉換技術作二維虛擬人臉之 自動產生及其應用

研究生：張依帆 指導教授：蔡文祥 博士

國立交通大學多媒體工程研究所

摘要



本論文提出了一套自動產生會說話的虛擬卡通臉系統。這個系統包含了四個階段：卡通人臉產生、語音分析、臉部表情與嘴形合成、動畫製作。配合本論文採用的人臉模型，系統會自動建構出一個三維人臉座標系統，並利用三維轉換技術產生不同角度的二維卡通人臉。同時我們以部份特徵點作為控制點來控制卡通人臉的表情，並藉由統計的方法來模擬說話時自然轉頭及不同表情的時間變化。接著，藉由分析輸入的語音及相對應的文字稿件，我們將語音以句子的形式作切割，再使用語音同步技術，配合提出的十二種基本嘴形來模擬會說話的卡通臉。最後，藉由一可編輯且具有開放性之可擴展標記語言(XML)，亦即 SVG，來達成繪圖及語音同步輸出之效果。利用上述方法，我們實作出兩種有趣的應用。從我們所獲得的良好實驗結果，證實了本論文所提出方法之可行性及應用性。

Automatic 2D Virtual Face Generation by 3D Model Transformation Techniques and Applications

Student: Yi-Fan Chang Advisor: Prof. Wen-Hsiang Tsai, Ph. D.

Institute of Multimedia Engineering, College of Computer Science
National Chiao Tung University

ABSTRACT

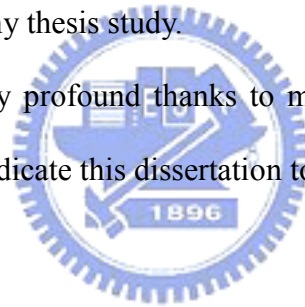
In this study, a system for automatic generation of talking cartoon faces is proposed, which includes four processes: cartoon face creation, speech analysis, facial expression and lip movement synthesis, and animation generation. A face model of 72 facial feature points is adopted. A method for construction of a 3D local coordinate system for the cartoon face is proposed, and a transformation between the global and the local coordinate systems by the use of a knowledge-based coordinate system transformation method is conducted. A 3D rotation technique is applied to the cartoon face model with some additional points to draw the face in different poses. A concept of assigning control points is applied to animate the cartoon face with different facial expressions. A statistical method is proposed to simulate the timing information of various facial expressions. For lip synchronization, a sentence utterance segmentation algorithm is proposed and a syllable alignment technique is applied. Twelve basic mouth shapes for Mandarin speaking are defined to synthesize lip movements. A frame interpolation method is utilized to generate the animation. Finally, an editable and opened vector-based XML language - Scalable Vector Graphics (SVG) is used for rendering and synchronizing the cartoon face with speech. Two kinds of interesting applications are implemented. Good experimental results show the feasibility and applicability of the proposed methods.

ACKNOWLEDGEMENTS

I am in hearty appreciation of the continuous guidance, discussions, support, and encouragement received from my advisor, Dr. Wen-Hsiang Tsai, not only in the development of this thesis, but also in every aspect of my personal growth.

Thanks are due to Mr. Chih-Jen Wu, Miss Kuan-Ting Chen, Mr. Kuan-Chieh Chen, Mr. Jian-Jhong Chen, Mr. Tsung-Chih Wang, and Mr. Shang-Huang Lai for their valuable discussions, suggestions, and encouragement. Appreciation is also given to the colleagues of the Computer Vision Laboratory in the Institute of Computer Science and Engineering at National Chiao Tung University for their suggestions and help during my thesis study.

Finally, I also extend my profound thanks to my family for their lasting love, care, and encouragement. I dedicate this dissertation to my beloved parents.



CONTENTS

ABSTRACT(in Chinese).....	i
ABSTRACT(in English).....	ii
ACKNOWLEDGEMENTS.....	iii
CONTENTS.....	iv
LIST OF FIGURES.....	vii
LIST OF TABLES.....	x

Chapter 1	Introduction.....	1
1.1	Motivation.....	1
1.2	Survey of Related Studies.....	3
1.3	Overview of Proposed Method.....	5
1.3.1	Definitions of Terms.....	5
1.3.2	Assumptions.....	7
1.3.3	Brief Descriptions of Proposed Method.....	7
1.4	Contributions.....	9
1.5	Thesis Organization.....	9
Chapter 2	Cartoon Face Generation and Modeling from Single Images.....	11
2.1	Introduction.....	11
2.2	Review of Adopted Cartoon Face Model.....	12
2.3	Construction of 3D Face Model Based on 2D Cartoon Face Model... ..	15
2.3.1	Basic Idea.....	15
2.3.2	Construction Process.....	16
2.4	Creation of Cartoon Face.....	21
2.4.1	Creation of Frontal Cartoon Face.....	21
2.4.2	Generation of Basic Facial Expressions.....	24
2.4.3	Creation of Oblique Cartoon Face.....	26
2.5	Experimental Results.....	32
Chapter 3	Speech Segmentation for Lip Synchronization.....	34
3.1	Introduction to Lip Synchronization for Talking Cartoon Faces.....	34
3.2	Segmentation of Sentence Utterances by Silence Feature.....	36

3.2.1	Review of Adopted Segmentation Method	36
3.2.2	Segmentation Process	37
3.3	Mandarin Syllable Segmentation	40
3.3.1	Review of Adopted Method	40
3.3.2	Segmentation Process	41
3.4	Experimental Results	41
Chapter 4	Animation of Facial Expressions	44
4.1	Introduction	44
4.2	Analysis of Facial Expression Data from Images of TV News Announcers	45
4.3	Review of Adopted Simulation Methods of Eye Blinks and Eyebrow Movements	47
4.4	Simulation of Eyebrow Movements	50
4.5	Simulation of Head Tilting and Turning	52
4.5.1	Simulation of Head Tilting	52
4.5.2	Simulation of Horizontal Head Turning	54
4.5.3	Simulation of Vertical Head Turning	54
Chapter 5	Talking Cartoon Face Generation	60
5.1	Introduction	60
5.2	Definitions of Basic Mouth Shapes	61
5.2.1	Review of Definition of Basic Mouth Shapes	63
5.2.2	New Definitions of Basic Mouth Shapes	67
5.3	Review of Adopted Time Intervals of Sentence Syllables for Mouth Shape Generation	73
5.4	Talking Cartoon Face Generation by Synthesizing Lip Movements ...	74
5.5	Experimental Results	75
Chapter 6	Talking Cartoon Face Generator Using Scalable Vector Graphics	78
6.1	Introduction	78
6.2	Overview of Scalable Vector Graphics (SVG)	78
6.3	Construction of a Talking Cartoon Face Generator Using SVG	80
6.3.1	Spatial Domain Process	80
6.3.2	Temporal Domain Process	84
6.4	Experimental Results	85

Chapter 7	Applications of Talking Cartoon Faces.....	88
7.1	Introduction to Implemented Applications	88
7.2	Application to Virtual Announcers	89
7.2.1	Introduction to Virtual Announcers.....	89
7.2.2	Process of Talking Face Creation.....	89
7.2.3	Experimental Results	90
7.3	Applications to Audio Books for E-Learning	91
7.3.1	Introduction to Audio Books.....	91
7.3.2	Process of Audio Book Generation.....	92
7.3.3	Experimental Results	93
Chapter 8	Conclusions and Suggestions for Future Works	94
8.1	Conclusions.....	94
8.2	Suggestions for Future Works.....	95
References		98



LIST OF FIGURES

Figure 1.1	Configuration of proposed system.....	8
Figure 2.1	Flowchart of hierarchical bi-level thresholding method in Chen and Tsai [1].....	13
Figure 2.2	A face model. (a) Proposed 72 feature points. (b) Proposed facial animation parameter units in Chen and Tsai [1].	14
Figure 2.3	An illustration of corner-cutting algorithm in Chen and Tsai [1].	14
Figure 2.4	Cubic Bezier curve in Chen and Tsai [1].	15
Figure 2.5	Two coordinate systems. The lines drawn in black color represent the global coordinate system, and those drawn in red color represent the local one.....	17
Figure 2.6	Points to help drawing.	17
Figure 2.7	Two orthogonal photographs. (a) Front view. (b) Side view.	20
Figure 2.8	An illustration of $arc(P_1, \dots, P_n)$	22
Figure 2.9	An illustration of the steps in the creation of the frontal cartoon face. (a) The creation of the contour of a face. (b) The creation of the ears. (c) The creation of the nose. (d) The creation of the eyebrows. (e) The creation of the eyes. (f) The creation of the mouth.....	24
Figure 2.10	An experimental result of the creation of a frontal cartoon face. (a) A male face model. (b) A female face model.	24
Figure 2.11	An experimental result of generation of an eye blinking effect.....	25
Figure 2.12	An experimental result of generation of a smiling effect.....	26
Figure 2.13	An experimental result of generation of an eyebrow raising effect.....	26
Figure 2.14	An illustration of a point rotated on the three Cartesian axes.....	27
Figure 2.15	An illustration of a point rotated on the Y axis.	28
Figure 2.16	An illustration of the focus and eyeballs. (a) Before rotation. (b) After rotation.	29
Figure 2.17	An illustration of the unreality of the hair contour.	30
Figure 2.18	An illustration of the shift of hair contour points. (a) Before rotation. (b) After rotation.....	31
Figure 2.19	An illustration of creation of oblique cartoon faces. (a) An oblique cartoon face with $\beta = 15$ degrees. (b) An oblique cartoon face with $\beta = -15$ degrees.	32
Figure 2.20	An example of experimental results for creation of cartoon faces in different poses with different facial expressions.....	33
Figure 3.1	A flowchart of proposed speech analysis process.....	35

Figure 3.2	An example of recorded video contents and corresponding actions in Lai and Tsai [4].	36
Figure 3.3	A flowchart of the sentence utterance segmentation process.	38
Figure 3.4	An example of selecting the first silent part in an input audio.	40
Figure 3.5	An example of sentence utterances segmentation results. The blue and green parts represent odd and even speaking parts, respectively.	40
Figure 3.6	A flowchart of the Mandarin syllable segmentation process.	42
Figure 3.7	An example of entire audio data of a transcript. The content of the transcript is “或許你已聽過，多補充抗氧化劑可以延緩老化。但真相為何？”	42
Figure 3.8	The result of syllable alignment of the audio in Figure 3.7.	42
Figure 3.9	An example of entire audio data of a transcript. The content of the transcript is “長期下來，傷害不斷累積，就可能造就出一個較老、較脆弱的身體。”	43
Figure 3.10	The result of syllable alignment of the audio in Figure 3.9.	43
Figure 4.1	An illustration of the definitions of t_s and t_e for eyebrow movements.	45
Figure 4.2	An illustration of the definitions of t_s and t_e for head movements.	46
Figure 4.3	A screen shot of the software VirtualDub.	46
Figure 4.4	An illustration of the probability function of eye blinks in Lin and Tsai [3]. The one in blue color is the probability function of the Gamma distribution with $\alpha = 2$ and $\theta = 1.48$. The other one in pink color is the probability function of eye blinks approximated from the analysis data of TV News announcers.	49
Figure 5.1	An illustration of basic components for definition of basic mouth shapes. (a) Control points of the mouth. (b) FAPUs of the mouth and the nose.	64
Figure 5.2	An illustration of basic mouth shapes of Mandarin initials in Chen and Tsai [1]. (a) Basic mouth shape m . (b) Basic mouth shape f . (c) Basic mouth shape h .	65
Figure 5.3	An illustration of basic mouth shapes of the Mandarin finals in Chen and Tsai [1]. (a) Basic mouth shape a . (b) Basic mouth shape i . (c) Basic mouth shape u . (d) Basic mouth shape e . (e) Basic mouth shape o . (f) Basic mouth shape n .	67
Figure 5.4	An illustration of basic mouth shapes of Mandarin initials. (a) Basic mouth shape m . (b) Basic mouth shape f . (c) Basic mouth shape h' . (d) Basic mouth shape h . (e) Basic mouth shape r . (f) Basic mouth shape z .	72
Figure 5.5	An illustration of basic mouth shapes of Mandarin finals. (a) Basic mouth shape a . (b) Basic mouth shape i . (c) Basic mouth shape u . (d) Basic	

	mouth shape <i>e</i> . (e) Basic mouth shape <i>o</i> . (f) Basic mouth shape <i>n</i>	72
Figure 5.6	An illustration of time intervals of a syllable of two basic mouth shapes in [1].	74
Figure 5.7	An illustration of time intervals of a syllable of three basic mouth shapes in [1].	74
Figure 5.8	An illustration of time intervals of a syllable of four basic mouth shapes in [1].	74
Figure 5.9	A concept of the use of key frames.	75
Figure 5.10	An overall illustration of the process of talking cartoon face generation.	75
Figure 5.11	An experimental result of the talking cartoon face speaking “願望.”	76
Figure 5.12	An experimental result of the talking cartoon face speaking “波濤.”	77
Figure 6.1	A result of an SVG source code.	80
Figure 6.2	A concept of layers in the special domain.	81
Figure 6.3	An example of the syntax <i>polyline</i> of SVG.	81
Figure 6.4	An example of the syntax <i>circle</i> of SVG.	82
Figure 6.5	An illustration of eye drawing.	82
Figure 6.6	An example of using the syntax <i>path</i> of SVG to draw the cubic Bezier curve.	83
Figure 6.7	An example of adding two layers of the background and the clothes for the cartoon face.	83
Figure 6.8	Another example of adding two layers of the background and the clothes for the cartoon face.	84
Figure 6.9	An experimental result of the talking cartoon face speaking “蜿蜒.”	86
Figure 6.10	An experimental result of the talking cartoon face speaking “光明.”	87
Figure 7.1	An illustration of the process of proposed system.	90
Figure 7.2	An example of a virtual announcer.	91
Figure 7.3	Another example of a virtual announcer.	91
Figure 7.4	An example of a virtual teacher.	93

LIST OF TABLES

Table 2.1	The values of the points in the z-direction.....	20
Table 3.1	Descriptions of audio features.	35
Table 4.1	Statistics of time intervals of eyebrow movements.	51
Table 4.2	Statistics of durations of eyebrow movements.	52
Table 4.3	Statistics of time intervals of head tilting.	53
Table 4.4	Statistics of durations of head tilting.	54
Table 4.5	Statistics of time intervals of horizontal head turning.	55
Table 4.6	Statistics of durations of horizontal head turning.	56
Table 4.7	The mean values, the standard deviation values, and the adopted intervals of uniform random variables for simulation of horizontal head turning.....	56
Table 4.8	Some examples of the relation between the vertical head turning and the pause time.	57
Table 4.9	Statistics of time intervals between the nod and the pause.....	57
Table 4.10	Statistics of durations of the nod.....	58
Table 4.11	Statistics of durations of the head raising after the nod.	58
Table 4.12	The mean value, the standard deviation value, and the adopted intervals of uniform random variables for simulation of vertical head turning.....	59
Table 5.1	Classification of initials according to the manners of articulation proposed in Yeh [15].....	61
Table 5.2	An illustration of 7 kinds of mouth shapes of initials proposed in Yeh [15].	62
Table 5.3	Three basic mouth shapes of Mandarin initials in Chen and Tsai [1].....	63
Table 5.4	A set of combinations with 7 basic mouth shapes of Mandarin finals in Chen and Tsai [1].....	64
Table 5.5	Five basic mouth shapes of Mandarin initials.....	68
Table 5.6	A set of combinations with 7 basic mouth shapes of Mandarin finals.....	68

Chapter 1

Introduction

1.1 Motivation

With the improvement on network and multimedia technologies, people are changing their habits of using computers. They start to get used to dealing with their works by means of their computers and obtain information through the Internet. Therefore, different types of multimedia, including text, image, audio, and video, are evolved. People can now read news and acquire new knowledge with a great diversity of forms on the Internet. However, the variety of multimedia types does not make computers friendlier to interact with. People are still unsatisfied because the computer still lacks human nature.



Due to this reason, more and more researchers devote themselves to improving the interaction between humans and computers. Several researchers report that the use of virtual talking faces, which are animations of human faces on computer screens, can increase the attention paid by users. People not only can get impressed by the virtual face and hence keep relevant information in mind, but also can have a good time interacting with the computer.

Although the topic of virtual talking faces has been studied for many years, generating realistic talking faces is still a challenging task because face movements are quite complicated to simulate, especially the deformation of muscles. To achieve the goal of generating realistic virtual faces, it requires some motion capture equipments, which are too expensive for common users. For some applications,

realism is not an essential property. Instead, some people are more concerned about how to make the talking faces livelier to express their words in a natural way.

For this reason, we use a 3D cartoon-like virtual face model which can be used to display proper lip movements synchronized with the speech, lifelike head movements, and emotional expressions. The style of non-photorealistic cartoon faces can be designed more freely than photorealistic ones. Shapes and textures of cartoon faces can be represented simply, so it is unnecessary to calculate the complex deformation of muscles. Also, the data size of the face model may be reduced because of the simpler representation of the cartoon face. Expressions can be easily modified by relocating the positions of predefined feature points, so it is full of variety and fun to use cartoon face models to display personalized faces instead of using photorealistic ones.

However, generating a three-minute animation requires at least 4320 frames. To animate the cartoon face without dealing with it frame by frame, we must apply some methodology to generate it automatically. One method to generate the animation of virtual faces automatically is to use the technique of real-time motion capturing which has been developed for many years. By putting some markers on one's face and tracking them continuously by sensors, and then mapping the tracked markers to the virtual face, we can extract facial features and create corresponding facial expressions on the virtual face. As we have mentioned above, this approach is too expensive and complicated for ordinary users.

To animate virtual faces more realistically but with less effort, we want to design a method to generate a virtual cartoon face speaking Mandarin, which just requires existing cartoon face models and segments of speech as input. We hope that we can achieve our goal by utilizing the techniques of 3D coordinate generation and

transformation, speech processing and statistical simulation, as well as creation of basic emotions and head movements.

Based on the research result of this study, it is also hoped that the generation of virtual announcers, virtual teachers, virtual storytellers, and so on, may become easier and more convenient for use in various applications.

1.2 Survey of Related Studies

Roughly speaking, there are two main phases to generate virtual talking faces. One is the creation of head models, including 2D and 3D models. The other is the generation of virtual face animations with speech synchronization.

For the first phase, the main issue is how to create a face model. In Chen and Tsai [1], cartoon face models are created from single front-view facial images by extraction of facial features points, and cartoon faces are generated by curve drawing techniques. In Zhang and Cohan [5], multiple image views of a particular face are utilized to morph the generic 3D face model into specific face structures. In Goto, Kshirsagar, and Thalmann [6], the method of automatic face cloning using two orthogonal photographs was proposed. It includes two steps: face model matching and texture generation. After these two steps are performed, a generic 3D face model is deformed to fit to the photographs. Zhang *et al.* [12] advanced a practical approach that also needs only two orthogonal photos for fast 3D modeling. They used radial basis functions (RBF) to deform a generic model with corresponding feature points and then performed texture mapping for realistic modeling. Chen *et al.* [7, 8] proposed a method to automatically generate a facial sketch of human portraits from input images. They used the non-parametric sampling method to learn the drawing

styles in the sketches illustrated by an artist. According to the learned styles, they can fit a flexible sketch model to the input images and then generate the corresponding sketches by an example-based method. The non-parametric sampling scheme and the example-based method are also adopted in the cartoon system PicToon, which is designed in [9]. The system can be used to generate a personalized cartoon face from an input image. By sketch generation and stroke rendering techniques, a stylized cartoon face is created.

In order to animate a virtual talking face, speech synchronization is an important issue to be concerned about. In [1] and [9], cartoon faces are animated by an audio-visual mapping between input speeches and the corresponding lip configuration. In Li *et al.* [10], cartoon faces are animated not only from input speeches, but also based on emotions derived from speech signals. In [3] and [4], methods of animating a photorealistic virtual face were studied. A frame generation algorithm was used for audio synchronization to generate a talking face.

Another approach to generating virtual talking face animations is to track the facial features in real-time and map the feature points to the control points on the face model. A method for real-time tracking was proposed in [11] by putting some markers on faces. To track facial features without markers, some image processing techniques are required. A system, designed in [6], can track many facial features in real-time, like eye, eyebrow, mouth, and jaw. Chen and Tsai [2] also proposed a method of eye-pair tracking, mouth tracking, and detection of head turning for real-time facial images. They designed a real-time virtual face animation system, which is combined with networks, to implement an application to multi-role avatar broadcasting and an application to web TV by ActiveX technique.

1.3 Overview of Proposed Method

An overview of the proposed approach is described in this section. First, some definitions of terms used in this study are described in Section 1.3.1. And some assumptions made for this study are listed in Section 1.3.2. Finally a brief description of the proposed method is outlined in Section 1.3.3.

1.3.1 Definitions of Terms

The definitions of some terms used in this study are listed as follows.

- (1) **Neutral Face:** MPEG-4 specifies some conditions for a head in its neutral state [13] as follows.

1. Gaze is in the direction of the Z-axis.
2. All face muscles are relaxed.
3. Eyelids are tangent to the iris.
4. The pupil is one third of the iris diameter.
5. The lips are in contact.
6. The line of the lips is horizontal and at the same height of lip corners.
7. The mouth is closed and the upper teeth touch the lower ones.
8. The tongue is flat and horizontal with the tip of the tongue touching the boundary between the upper and lower teeth.

In this thesis, a face with a normal expression is called a neutral face.

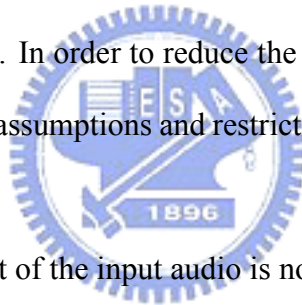
- (2) **Neutral Facial Image:** A neutral facial image is an image with a frontal and straight neutral face in it.
- (3) **Facial Features:** In the proposed system, we care about several features of the face, including hair, face, eyebrows, eyes, nose, mouth, and ears of each facial image.

- (4) **Facial Action Units (FAUs):** Facial Action Coding System (FACS) [14] defines 66 basic Facial Action Units (FAUs). The major part of FAUs represents primary movements of facial muscles in action such as raising eyebrows, blinking, and talking. Other FAUs represent head and eye movements.
- (5) **Facial Expression:** A facial expression is a facial aspect representative of feeling. Here, Facial expressions include emotions and lip movements. Facial expressions can be described as combinations of FAUs.
- (6) **FAPUs:** Facial animation parameter units (FAPUs) are the fractions of the distances between some facial features, like eye separation, mouth width, and so on.
- (7) **Face Model:** A face model is a 2D cartoon face model with 74 feature points and some FAPUs, including hair, eyebrows, eyes, noses, mouths, and so on.
- (8) **Face Model Control Points:** These points are some of the 74 feature points of the face model. They are used to control many features of the face model, like eyebrow raising, eye opening, lip movement, head tilting, and head turning.
- (9) **Phoneme:** A phoneme is a basic enunciation of a language. For example, ㄉ, ㄌ, ㄎ are phonemes in Mandarin.
- (10) **Syllable:** A syllable consists of phonemes. For example, ㄉㄤ, ㄍㄨ are syllables in Mandarin.
- (11) **Viseme:** A viseme is the visual counterpart of a phoneme.
- (12) **Speech Analyzer:** A speech analyzer receives a speech file and a script file as input, and applies speech recognition techniques to get the timing information of each syllable.
- (13) **Key Frame:** A key frame can be any frame with the timeline feature in the animation at which you can exactly control the look of a cartoon face.

- (14) **Hidden Markov Model (HMM)**: The HMM is used to characterize the spectral properties of the frames of a speech pattern.
- (15) **Transcript**: A transcript is a text file that contains the corresponding content of a speech.

1.3.2 Assumptions

In real situation, it is not easy to simulate a 3D rotation when lacking the depth information. Because in the proposed system, only 2D cartoon face models and audio data are required to input, the inappropriate properties of these data may influence the result. For example, noise in the audio may affect the result of syllable segmentation. And unusual distribution of the facial features will cause exceptions in the 3D coordinate generating process. In order to reduce the complexity of processing works in the proposed system, a few assumptions and restrictions are made in this study. They are described as follows.



- (1) The recording environment of the input audio is noiseless.
- (2) The speech is spoken at a steady speed and in a loud voice.
- (3) The face of the model always faces the camera, with the rotation angle of the face about the three Cartesian axes does not exceed $\pm 15^\circ$ when naturally speaking.
- (4) The face of the model has smooth facial features.

1.3.3 Brief Descriptions of Proposed Method

In the proposed system, four major parts are included: a cartoon face creator, a speech analyzer, an animation editor, and an animation and webpage generator. The cartoon face creator creates cartoon faces from neutral facial images or cartoon face models. The speech analyzer segments the speech file into sentences and then performs the speech-text alignment for lip synchronization. The animation editor

allows users to edit facial actions such as head movements and eyebrows raises in the animation. The animation and webpage generator renders cartoon faces and generates webpages with embedded animation. A configuration of proposed system is shown in Figure 1.1.

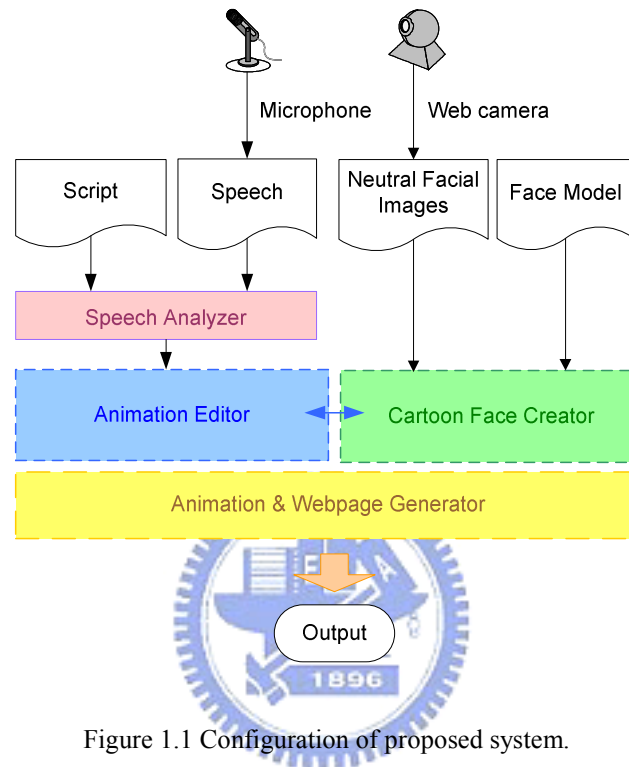


Figure 1.1 Configuration of proposed system.

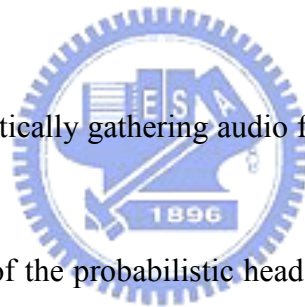
We use a web camera to capture a single neutral facial image, and then utilize the proposed cartoon face creator to create a personal cartoon face. The positions of feature points and the values of some FAPUs can be saved as a face model, which can be loaded as input by the cartoon face creator to create a personal cartoon face, too. After the personal cartoon face is created, users may use a speech file and a script file as inputs of the speech analyzer, which gets the timing information of each syllable in the speech file. Then the animation editor automatically synthesizes lip movements according to the timing information. Users may specify facial actions or generate them automatically by the animation editor. Finally, the animation and webpage generator will output an animation file of a talking cartoon face and a webpage file

with the animation embedded in it.

1.4 Contributions

Some major contributions of the study are listed as follows.

- (1) A complete system for automatically creating personal talking cartoon faces is proposed.
- (2) A method for construction of 3D cartoon face models based on 2D cartoon face models is proposed.
- (3) A method for simulation of head tilting and turning using 3D rotation techniques is proposed.
- (4) Some methods for automatically gathering audio features for speech segmentation are proposed.
- (5) A method for simulation of the probabilistic head movements and basic emotions is proposed.
- (6) Several new applications are proposed and implemented by using the proposed system.



1.5 Thesis Organization

The remainder of the thesis is organized as follows. In Chapter 2, the proposed method of construction of 3D cartoon face models based on 2D cartoon face models and a method of creation of virtual cartoon faces are described. In Chapter 3, the proposed method of speech segmentation for lip synchronization is described. In

Chapter 4, the proposed method of simulating facial expressions and head movements is described. And then, some animation issues such as lip movements and smoothing of talking cartoon facial animation are discussed and solved in Chapter 5. Up to Chapter 5, talking cartoon faces are generated.

A final integration using an open standard language SVG (Scalable Vector Graphics) to generate web-based animations is described in Chapter 6. Some examples of applications using the proposed system are presented in Chapter 7. Finally, conclusions and some suggestions for future works are included in Chapter 8.



Chapter 2

Cartoon Face Generation and Modeling from Single Images

2.1 Introduction

To animate a cartoon face much livelier, many issues are of great concern to us, such as lip movements, eye blinks, eyebrow movements, and head movements. Especially for simulation of head movements, including head tilting and head turning, a 2D face model is not enough to synthesize proper head poses of cartoon faces. Due to this reason, a 3D face model must be constructed to handle this problem. In the proposed system, one of the four major parts shown in Figure 1.1, which is named cartoon face creator, is designed to create personal cartoon faces, integrating the technique of 3D face model construction. In the creation process, three main steps are included. The first step is to assign facial feature points to a 2D face model. It can be done in two ways. One is to detect facial features of an input neutral facial image, generate corresponding feature points, and map them to the feature points in the predefined face model. The other is to directly assign the feature points according to the input 2D face data. In this study, we adopt both ways in constructing our face model.

The second step is to construct the local coordinate system of the face model for applying 3D rotation techniques. By creating a transformation between the global and the local coordinate systems and assigning the position of the feature points in the

third dimension, namely, the Cartesian z -coordinate, this step can be done, and then essential head movements can be simulated. The last step is to define basic facial expression parameters for use in face animation.

In this chapter, some techniques are proposed to achieve the purpose mentioned above. First, a review of Chen and Tsai [1] about constructing a 2D face model from single images is presented in Section 2.2. Second, a technique to construct a 3D face model based on the 2D face model is proposed in Section 2.3. In Section 2.4, a technique is proposed to create the cartoon face with different expressions in different poses.

2.2 Review of Adopted Cartoon Face Model



Chen and Tsai [1] proposed an automatic method for generation of personal cartoon faces from a neutral facial image. In their method, three main steps are carried out: extraction of facial feature regions, extraction of facial feature points, and creation of a face model. In the first step, a hierarchical bi-level thresholding method is used to extract the background, hair, and face regions in a given face image. A flowchart of the hierarchical bi-level thresholding method is shown in Figure 2.1.

Then, by finding all probable pairs of eye regions according to a set of rules related to the region's heights, widths, etc., and filtering these regions according to the symmetry of the two regions in each pair, an optimal eye-pair can be detected. Taking the positions of the detected optimal eye-pair as a reference, the facial feature regions can be extracted by a knowledge-based edge detection technique.

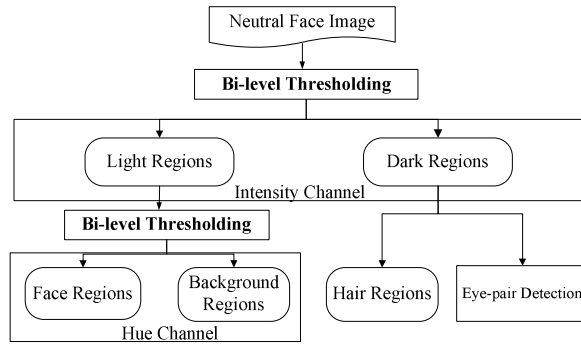


Figure 2.1 Flowchart of hierarchical bi-level thresholding method in Chen and Tsai [1].

Before extracting facial feature points, a face model with facial feature points must be defined first. Because the 84 feature points and the *facial animation parameter units* (FAPUs) of the face model specified in the MPEG-4 standard are not suitable for cartoon face drawing, Chen and Tsai [1] defined a face model with 72 feature points by adding or eliminating some feature points of the face model in MPEG-4. Some FAPUs were also specified according to the MPEG-4 standard, and then a new adoptable face model was set up. An illustration of the proposed face model is shown in Figure 2.2. In order to control the facial expression of the cartoon face, some feature points were assigned to be control points which are listed as follows:

1. *eyebrow Control Points*: there are 8 control points in both eyebrows, namely, 4.2, 4.4, 4.4a, 4.6, 4.1, 4.3, 4.3a, and 4.5.
2. *Eye Control Points*: there are 4 control points in eyes, namely, 3.1, 3.3, 3.2, and 3.4.
3. *Mouth Control Points*: there are 4 control points in the mouth, namely, 8.9, 8.4, 8.3, and 8.2, by which other mouth feature points are computed.
4. *Jaw Control Point*: there is 1 control point in the jaw, namely, 2.1, which is automatically computed by the position of the control point 8.2 and the value of the facial animation parameter *JawH*.

These control points in this study are the so-called *face model control points*. After setting up the face model with 72 feature points, the corresponding feature points in a given facial image can be extracted from the previously mentioned facial feature regions.

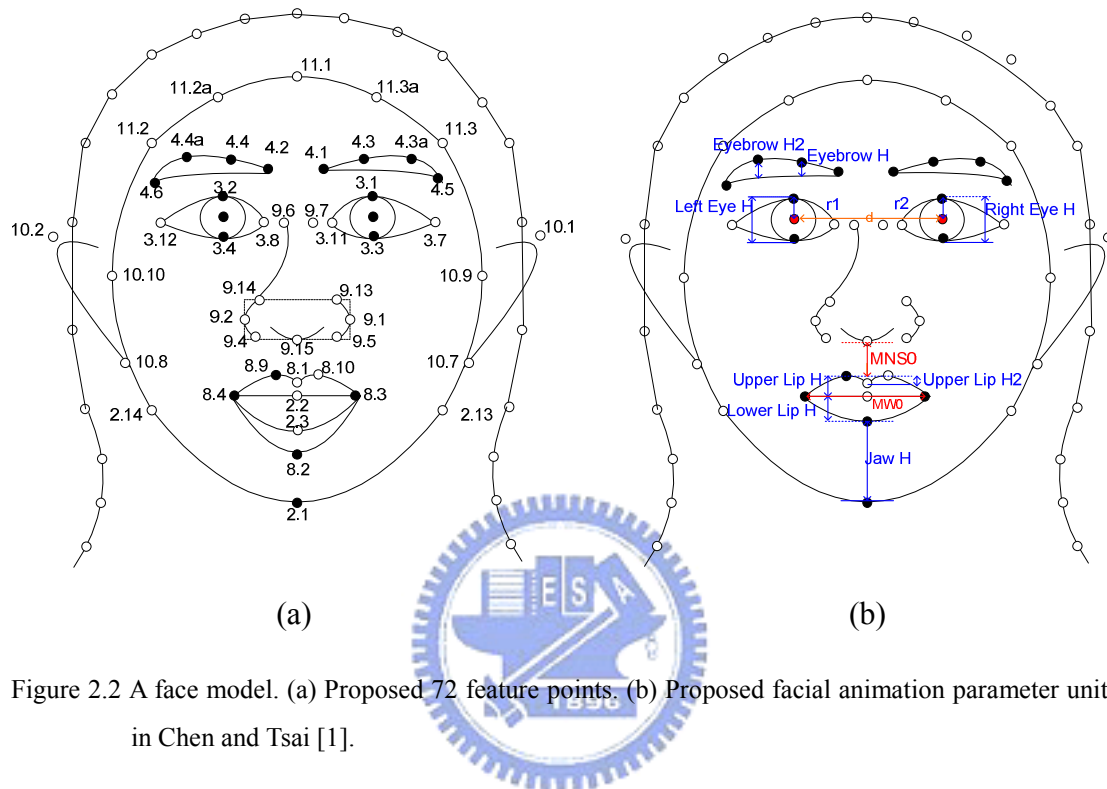


Figure 2.2 A face model. (a) Proposed 72 feature points. (b) Proposed facial animation parameter units in Chen and Tsai [1].

Finally, two curve drawing methods are applied to create cartoon faces. One is the corner-cutting subdivision method, in which a subdivision curve is generated by repeatedly cutting off corners of a polygon until a certain condition is reached, as shown in Figure 2.3. The other is the cubic Bezier curve approximation method, which is used to produce smooth curves with a simple polynomial equation, as shown in Figure 2.4.

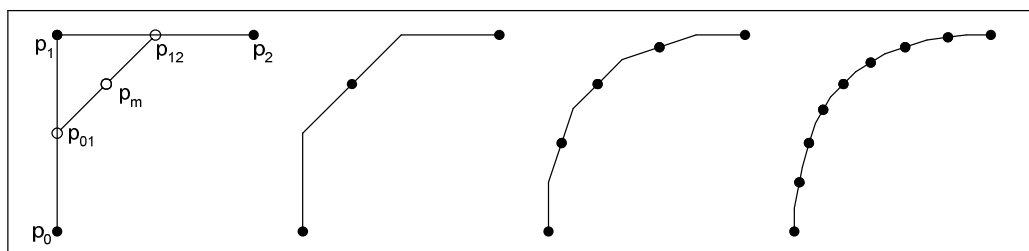


Figure 2.3 An illustration of corner-cutting algorithm in Chen and Tsai [1].

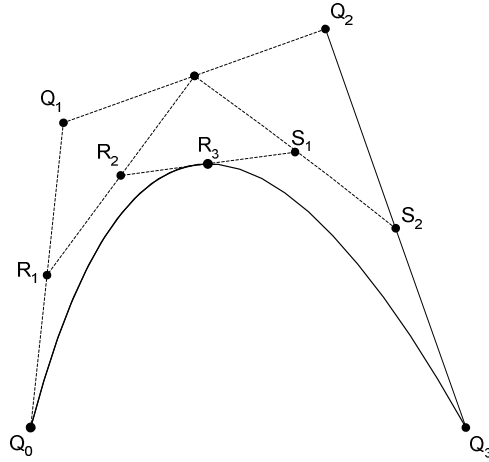


Figure 2.4 Cubic Bezier curve in Chen and Tsai [1].

2.3 Construction of 3D Face Model Based on 2D Cartoon Face Model

In this section, the basic idea of constructing a 3D face model based on the above-mentioned 2D cartoon face model is introduced in Section 2.3.1. And the detail of the construction process is described in Section 2.3.2.

2.3.1 Basic Idea

Based on the face model mentioned in Section 2.2, a method is proposed to construct a 3D face model. The method can be divided into two steps: the first is to construct a local coordinate system from the global one, and the second is to assign the position of the feature points in the Cartesian z -direction. The basic idea for constructing a local coordinate system is to define a rotation origin and transform the points of the global coordinate system into those of the local one based on the rotation origin. The basic idea for assigning the position of the feature points in the Cartesian z -direction is to do the assignment based on a proposed generic model. Although a

generic model cannot represent all cases of human faces, it is practical enough in the application of generating virtual talking faces, because in real cases, one usually does not roll his/her head violently when giving a speech. With the assumption that heads are rotated slightly when speaking, a little inaccuracy of the depth information in a face model would not affect the result much, so we can then easily generate different head poses of the face model by a 3D rotation technique after the 3D face model is constructed.

2.3.2 Construction Process

The first step to construct a 3D face model is to construct a local coordinate system. As mentioned above, to achieve this goal, the first issue is to define a rotation origin. The ideal position of the rotation origin is the center of the neck, so we propose a knowledge-based method to define its position according to the position of the eyes. Some definitions of the terms used in this section are listed first as follows:

- $Eyeball_{Left/Right} \cdot x$ is the x position of the center of the left/right eyeball circle;
- $Eyeball_{Left/Right} \cdot y$ is the y position of the center of the left/right eyeball circle;
- $EyeMid$ is the position (x, y) of the center between $Eyeball_{Left}$ and $Eyeball_{Right}$.

The green dot shown in Figure 2.5 represents the point $EyeMid$. After computing the position of $EyeMid$ and making use of the FAPU d in the face model which denotes the Euclidean distance between two eyeballs, we can set the rotation origin and create a the transformation between the global coordinate system and the local one, as shown in Figure 2.5.

However, before we start the transformation, some additional points (as shown in Figure 2.6) must be defined to help drawing the cartoon face in different poses, as we

will describe later in Section 2.4. These points will be also transformed into the local coordinate system, so we must set up their positions before the transformation is started. And the detailed method for setting up the additional points and conducting the transformation is expressed as an algorithm in the following.

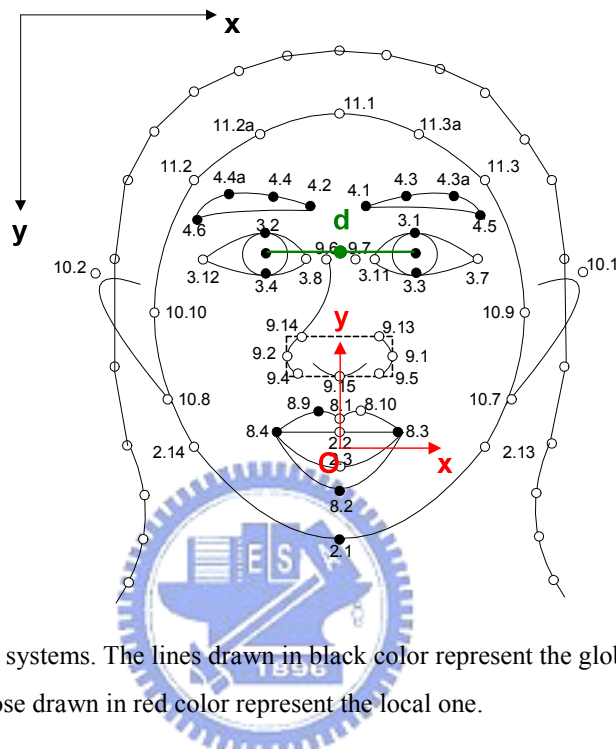


Figure 2.5 Two coordinate systems. The lines drawn in black color represent the global coordinate system, and those drawn in red color represent the local one.

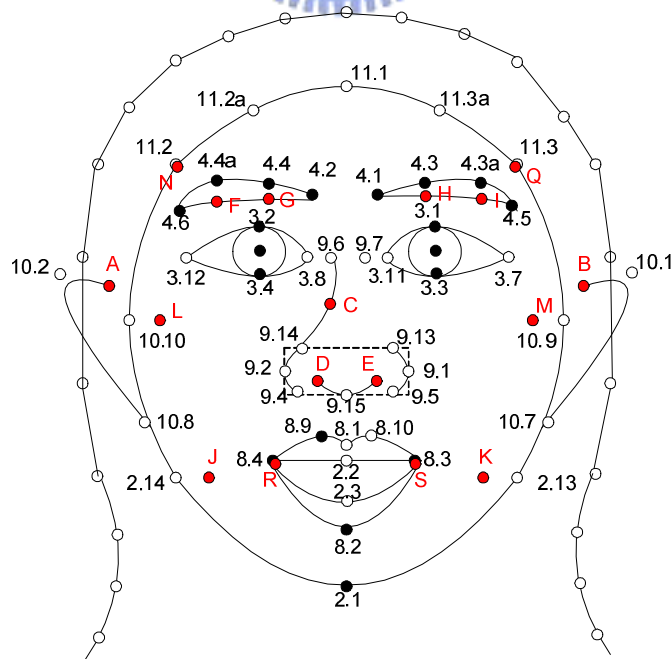


Figure 2.6 Points to help drawing.

Algorithm 2.1. *Knowledge-based coordinate system transformation.*

Input: One point *EyeMid*, some FAPUs, including *d*, *EyebrowH*, and *EyebrowH2*, and 72 model points in the global coordinate system.

Output: A rotation origin *O*, 17 additional points, and 72 model points in the local coordinate system.

Steps:

1. Let W_{ear} denote the distance between the *EyeMid* and an ear in the face model.
2. Let x_p and y_p denote the x -position and the y -position of a point P in the face model.
3. Speculate a rotation origin $O(x_o, y_o)$ to represent the center of the neck with

$$x_o = \text{EyeMid}.x;$$

$$y_o = \text{EyeMid}.y + d \times 1.3;$$

4. Set the additional points $A(x_a, y_a)$, $B(x_b, y_b)$, $C(x_c, y_c)$, $D(x_d, y_d)$, $E(x_e, y_e)$, $F(x_f, y_f)$, $G(x_g, y_g)$, $H(x_h, y_h)$, $I(x_i, y_i)$, $J(x_j, y_j)$, $K(x_k, y_k)$, $L(x_l, y_l)$, $M(x_m, y_m)$, $N(x_n, y_n)$, $Q(x_q, y_q)$, $R(x_r, y_r)$, and $S(x_s, y_s)$ in the following way:

$$x_a = \text{EyeMid}.x - W_{\text{ear}}, \quad y_a = (4 \times y_{10.2} + y_{10.8})/5;$$

$$x_b = \text{EyeMid}.x + W_{\text{ear}}, \quad y_b = (4 \times y_{10.1} + y_{10.7})/5;$$

$$x_c = (x_{9.13} + x_{9.14})/2, \quad y_c = (y_{9.14} + \text{EyeMid}.y)/2.;$$

$$x_d = (5 \times x_{9.4} + x_{9.5})/6, \quad y_d = y_{9.4} - (y_{9.4} - y_{9.14})/3;$$

$$x_e = (5 \times x_{9.5} + x_{9.4})/6, \quad y_e = y_{9.5} - (y_{9.5} - y_{9.13})/3;$$

$$x_f = x_{4.4a}, \quad y_f = y_{4.4a} + \text{EyebrowH2}/1.3;$$

$$x_g = x_{4.4}, \quad y_g = y_{4.4} + \text{EyebrowH}/1.3;$$

$$x_h = x_{4.3}, \quad y_h = y_{4.3} + \text{EyebrowH}/1.3;$$

$$x_i = x_{4.3a}, \quad y_i = y_{4.3a} + \text{EyebrowH2}/1.3;$$

$$x_j = (5 \times x_{2.14} + x_{2.1})/6, \quad y_j = y_{2.14};$$

$$x_k = (5 \times x_{2.13} + x_{2.1})/6, \quad y_k = y_{2.13};$$

$$x_l = (x_{3.12} + x_{10.10})/2, \quad y_l = y_{10.10};$$

$$x_m = (x_{3.7} + x_{10.9})/2, \quad y_m = y_{10.9};$$

$$x_n = x_{11.2}, \quad y_n = y_{11.2};$$

$$x_q = x_{11.3}, \quad y_q = y_{11.3};$$

$$x_r = x_{8.4}, \quad y_r = y_{8.4};$$

$$x_s = x_{8.5}, \quad y_s = y_{8.5}.$$

5. For each of the additional points and the 72 model points $P(x_p, y_p)$, set the point (x_p, y_p) in the following way:

$$x_p = x_o - x_p;$$

$$y_p = y_o - y_p.$$



The second step to construct a 3D face model is to assign the position of the points in the Cartesian z -direction. A generic model is proposed as the reference for the assignment. To generate the generic model, two orthogonal photographs are used, as shown in Figure 2.7. By calculating the Euclidean distance d between two eyeballs and the distance d' between the y -position of *EyeMid* and the y -position of the feature point 2.2 in the front-view image, d' can be expressed as a constant multiple of d . Here it is shown as $1.03d$ in the experiment in Figure 2.7(a). Similarly in the side-view image, the distance between *EyeMid* and the point 2.2 in the y -direction is set to the constant multiple of d , as shown in Figure 2.7(b). By marking all of the viewable points, including the rotation origin and some of the additional points mentioned above, and computing the distance in the z -direction between the origin and each of the points in the image, the positions of the points in the z -direction can

be computed as a constant multiple of d , too. For those points which are not viewable, based on the symmetry of the human face, their positions can be also assigned. After some adjustments and experiments, the values of the points in the z -direction adopted in this study is listed in Table 2.1.

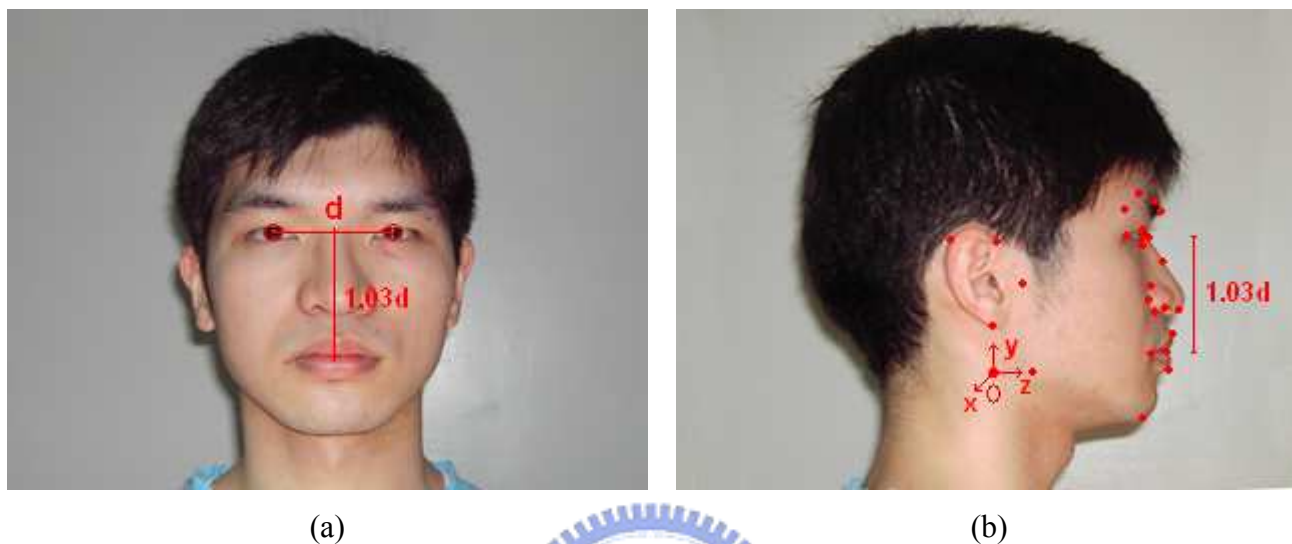


Figure 2.7 Two orthogonal photographs. (a) Front view. (b) Side view.

Table 2.1 The values of the points in the z -direction.

Category	Point	Value	Category	Point	Value
Hair	All hair points	$-0.58d$	Forehead	11.1	$1.37d$
Eyes	3.2, 3.4, 3.1, 3.3, <i>Eyeball_{Left}, Eyeball_{Right}</i>	$1.33d$		11.2a, 11.3a	$0.90d$
	3.12, 3.7	$1.19d$		11.2, 11.3	$0.42d$
	3.8, 3.11	$1.25d$		N, Q	$0.85d$
Eyebrows	4.6, 4.5	$1.17d$	Ears	10.2, 10.1	$-0.37d$
	4.4a, 4.3a, F, I	$1.30d$		10.8, 10.7	$0d$
	4.4, 4.3, G, H	$1.44d$		A, B	$0.04d$
	4.2, 4.1	$1.49d$	Nose	9.6, 9.7, 9.2, 9.1	$1.39d$
Mouth	2.2, 2.3	$1.55d$		9.14, 9.13	$1.40d$
	8.4, 8.3, R, S	$1.40d$		9.4, 9.5	$1.44d$
	8.1, 8.9, 8.10	$1.60d$		9.15	$1.65d$
	8.2	$1.56d$		C	$1.51d$
Jaw	2.14, 2.13	$0.36d$		D, E	$1.53d$
	2.1	$1.33d$	Cheek	10.9, 10.10	$0.27d$
	J, K	$1.18d$		L, M	$1.11d$

After the two steps are done, a 3D face model is constructed. We consider d as a length unit in the face model, and we can easily change the scale and the position of a 3D face model by changing its origin and the reference d value. The scheme is useful for normalization between different faces in different scales and positions. For example, if there is a face model whose d value is a certain constant c , and we want to scale it to a larger size with the value d being another constant c' , we can just apply the geometric ratio principle to multiply the position of each point and each FAPU by a factor of c'/c .

2.4 Creation of Cartoon Face

As mentioned in Section 2.2, the cartoon face is created by the corner-cutting subdivision and the cubic Bezier curve approximation methods. In this section, two types of cartoon faces are introduced, one being the frontal cartoon face and the other the oblique cartoon face. It is hoped that by the two types of cartoon faces, a head-turning talking cartoon face can be represented smoothly.

2.4.1 Creation of Frontal Cartoon Face

A frontal cartoon face is drawn by the 72 feature points and some of the additional points mentioned previously. Let $O(x_o, y_o)$ denote the position of the rotation origin in the face model. Before the creation process, for each of the additional points and the 72 model points $P(x_p, y_p)$, the position of P must be transformed into the global coordinate system in the following way:

$$x_p = x_p + x_o; \quad y_p = y_o - y_p.$$

After the transformation, the cartoon face can be drawn in the global coordinate system. The detail of the proposed frontal face creation method is described in the following as an algorithm.

Algorithm 2.2. *Creation of frontal cartoon face.*

Input: 72 feature points, 17 additional points, and some FAPUs, including the radii of the eyeballs r_1 and r_2 in the face model.

Output: an image of the frontal cartoon face.

Steps:

1. Let $arc(P_1, \dots, P_n)$ denote a curve composed by the points P_1, \dots, P_n .

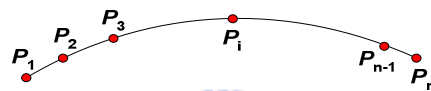


Figure 2.8 An illustration of $arc(P_1, \dots, P_n)$.

2. Draw the contour of the hair by a polygon composed by 23 hair feature points.
3. Draw the contour of the face, including the forehead, cheek, and jaw, by the cubic Bezier curves $arc(11.2, 11.2a, 11.1, 11.3a, 11.3)$, $arc(11.3, 10.9, 2.13)$, $arc(2.13, 2.1, 2.14)$, and $arc(2.14, 10.10, 11.2)$.
4. Draw the contour of the left ear by the cubic Bezier curves $arc(10.8, 10.2, A)$.
5. Draw the contour of the right ear in a similar way.
6. Draw the contour of the nose by the cubic Bezier curves $arc(9.6, C, 9.14)$, $arc(9.14, 9.2, 9.4)$, $arc(9.13, 9.1, 9.5)$, and $arc(D, 9.15, E)$.
7. Draw the contour of the left eyebrow by the corner-cutting subdivision curves $arc(4.6, 4.4a, 4.4, 4.2)$ and $arc(4.2, G, F, 4.6)$.
8. Draw the contour of the right eyebrow in a similar way.

9. Draw the contour of the left eye by the cubic Bezier curves $arc(3.12, 3.2, 3.8)$, $arc(3.8, 3.4, 3.12)$.
10. Draw the contour of the right eye in a similar way.
11. Draw a circle with the radius r_1 and the center at $Eyeball_{Left}$ representative of the left eyeball.
12. Draw a circle of the right eyeball in a similar way.
13. Draw the contour of the mouth by the cubic Bezier curves $arc(8.1, 8.9, 8.4)$, $arc(8.4, 8.2, 8.3)$, $arc(8.3, 8.10, 8.1)$, $arc(R, 2.2, S)$, and $arc(S, 2.3, R)$.
14. Fill the predefined colors into their corresponding parts.

An illustration of the steps in the creation of the frontal cartoon face is shown in Figure 2.9. An experimental result of the creation of a frontal cartoon face is shown in Figure 2.10.

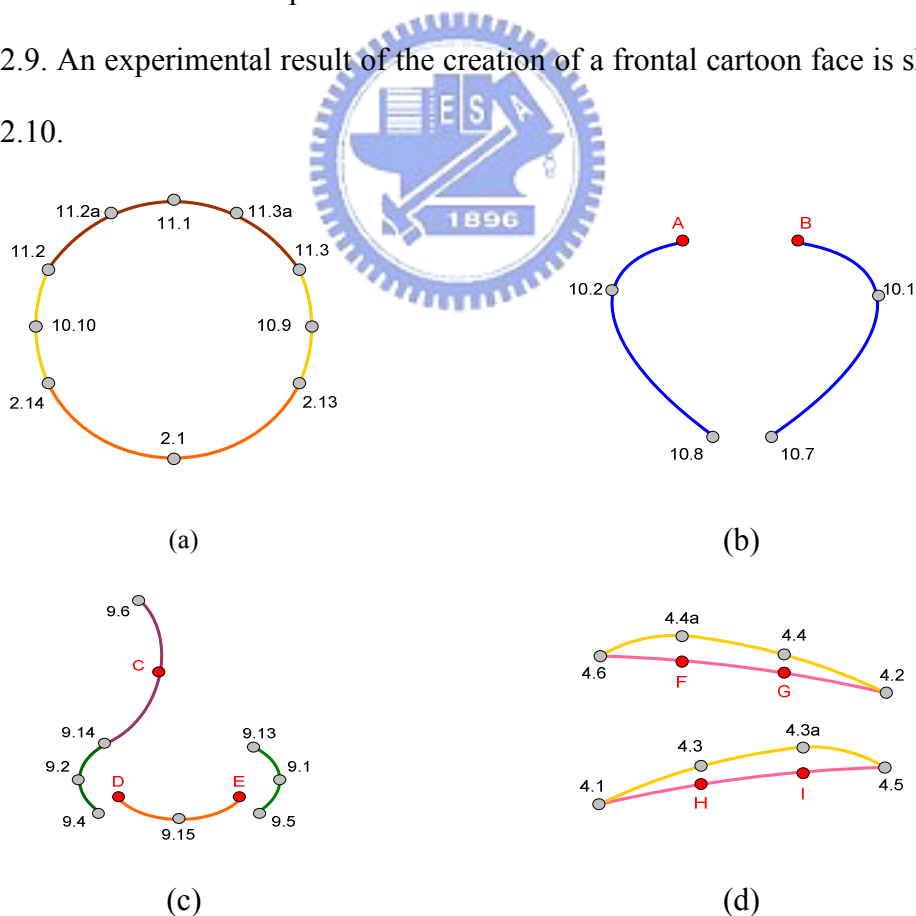


Figure 2.9 An illustration of the steps in the creation of the frontal cartoon face. (a) The creation of the contour of a face. (b) The creation of the ears. (c) The creation of the nose. (d) The creation of the eyebrows. (e) The creation of the eyes. (f) The creation of the mouth.

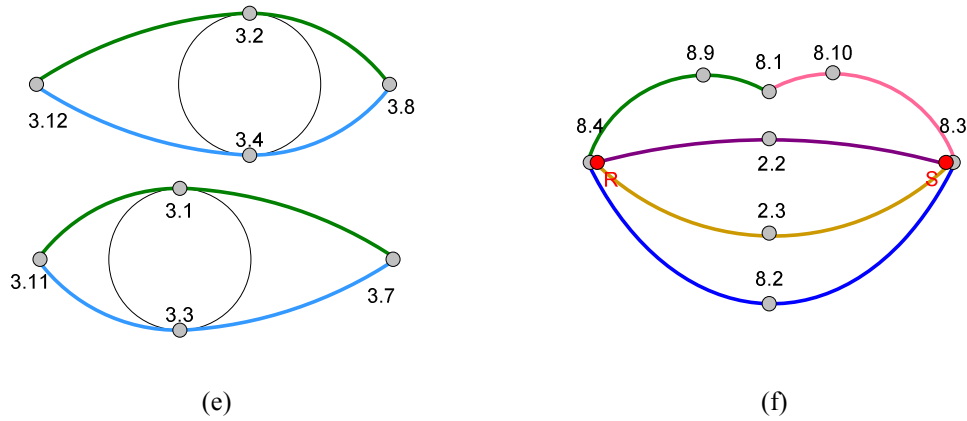


Figure 2.9 An illustration of the steps in the creation of the frontal cartoon face. (a) The creation of the contour of a face. (b) The creation of the ears. (c) The creation of the nose. (d) The creation of the eyebrows. (e) The creation of the eyes. (f) The creation of the mouth. (continued)

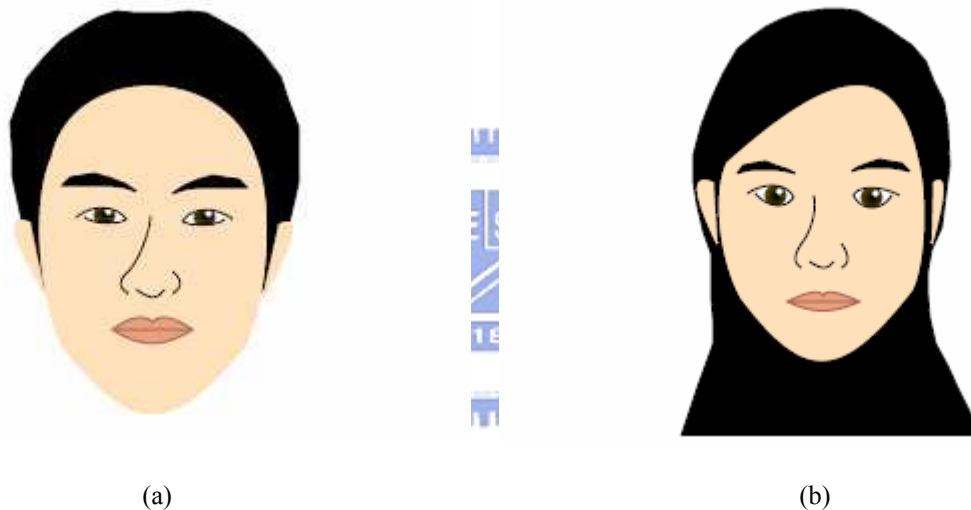


Figure 2.10 An experimental result of the creation of a frontal cartoon face. (a) A male face model. (b) A female face model.

2.4.2 Generation of Basic Facial Expressions

After a frontal cartoon face is created, we are concerned about how to generate some basic facial expressions to make the face livelier. Facial Action Coding System (FACS) [14] defines some basic Facial Action Units (FAUs), which represents primary movements of facial muscles in actions such as raising eyebrows, blinking, talking, etc. The FACS has been useful for describing most important facial actions, so some of the FAUs defined in it are considered to be suitable in the study for

synthesis of facial expressions. For example, the FAU 12, whose description is lip corner puller, can be viewed as a smile. And the FAUs 1 and 2, which respectively represent the inner and outer eyebrow raisings, are the basic facial expressions that frequently happen when one is making a speech. By taking the FAUs as references, we decide to define three basic facial expressions: eye blinking, smiling, and eyebrow raising.

For eye blinking, by changing the value of the FAPU *LeftEyeH* and *RightEyeH*, and setting up the positions of four model eye points 3.2, 3.4, 3.1, and 3.3 according to these two FAPUs, we can easily generate an eye blinking effect. An experimental result is shown in Figure 2.11.

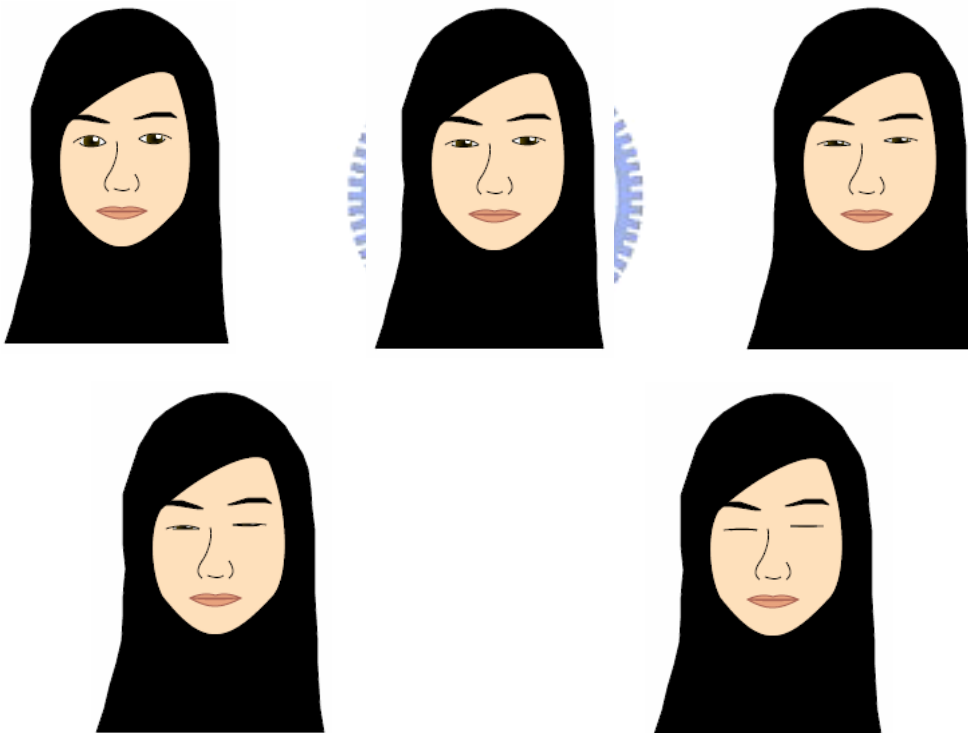


Figure 2.11 An experimental result of generation of an eye blinking effect.

Similarly, by changing the positions of two model mouth points 8.4 and 8.3 according to the FAPU *UpperLipH*, a smiling effect can be created. In the meanwhile, by modifying the positions of two model eye points 3.4 and 3.3 based on the FAPUs *LeftEyeH* and *RightEyeH*, a squinting effect can be combined into the cartoon face to

make the smiling more vivid. An experimental result is shown in Figure 2.12.

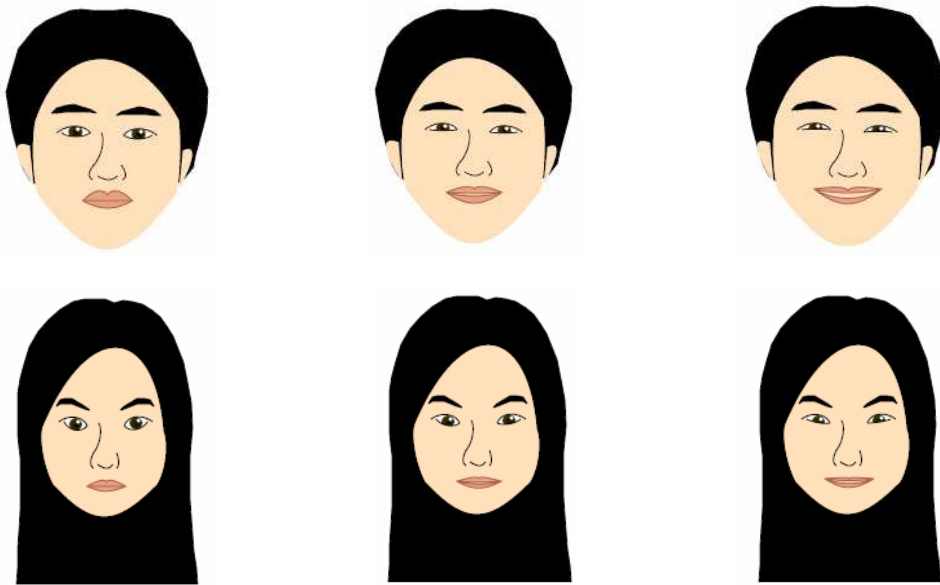


Figure 2.12 An experimental result of generation of a smiling effect.

For eyebrow raising, all of the 8 model eyebrow points and 4 additional eyebrow points are involved. By regulating the positions of these points according to the FAPU *EyebrowH*, an eyebrow raising effect can be generated. An experimental result is shown in Figure 2.13.

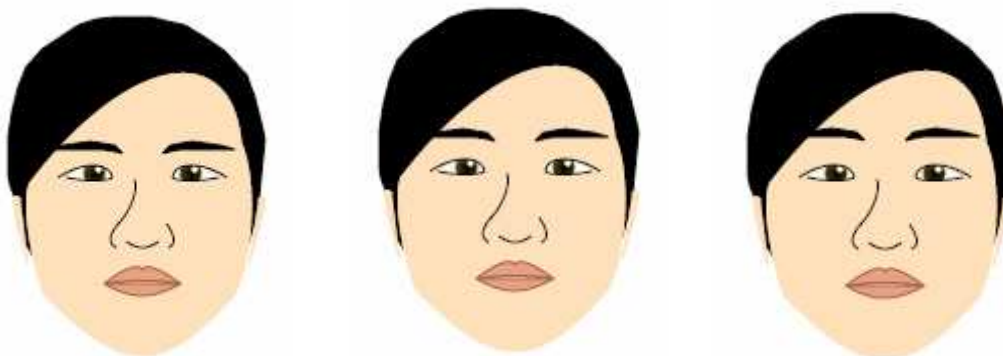


Figure 2.13 An experimental result of generation of an eyebrow raising effect.

2.4.3 Creation of Oblique Cartoon Face

The basic idea of creation of an oblique cartoon face is to rotate the 3D face model on the three Cartesian axes in the local coordinate system. After rotation, the

3D points are projected to the X-Y plane and transformed into the global local system. Then the cartoon face can be illustrated by the previously-mentioned corner-cutting subdivision and cubic Bezier curve approximation methods. In this section, a review of a 3D rotation technique is presented in Section 2.4.3.1. A simulation of eyeballs gazing at a fixed target while the head is turning is described in Section 2.4.3.2. At last, the creation process, including some methods to solve the additional problems while drawing, is described in Section 2.4.3.3.

2.4.3.1 Review of 3D Rotation Technique

Suppose that a point in a 3D space, which is denoted by (x, y, z) , is rotated on the three Cartesian axes respectively, as shown in Figure 2.14

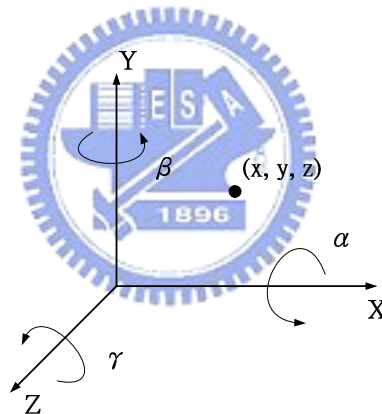


Figure 2.14 An illustration of a point rotated on the three Cartesian axes.

We define positive angles to be representative of counter-clockwise rotations, and negative ones representative of clockwise rotations. Two basic trigonometric equations as follows are used as the 3D rotation formula:

$$\sin(\theta + \beta) = \sin \theta \times \cos \beta + \cos \theta \times \sin \beta ;$$

$$\cos(\theta + \beta) = \cos \theta \times \cos \beta - \sin \theta \times \sin \beta .$$

Suppose the point is first rotated on the Y axis, so the y coordinate will not be changed. It is assumed that after projecting the point onto the X - Z plane, the distance

between the point and the origin is L , as shown in Figure 2.15.

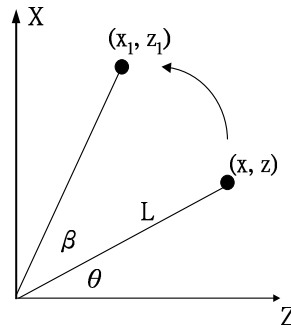


Figure 2.15 An illustration of a point rotated on the Y axis.

Then the equations above can be transformed to be as follows:

$$\frac{x_1}{L} = \frac{x}{L} \times \cos \beta + \frac{z}{L} \times \sin \beta ;$$

$$\frac{z_1}{L} = \frac{z}{L} \times \cos \beta - \frac{x}{L} \times \sin \beta .$$

After canceling L , the formula for the point rotated on the Y -axis can be derived to be as follows:

$$\begin{cases} x_1 = x \times \cos \beta + z \times \sin \beta ; \\ y_1 = y ; \\ z_1 = -x \times \sin \beta + z \times \cos \beta . \end{cases}$$

Similarly, the formula for the point rotated on the X - and Z -axes are derived as follows, respectively:

$$\begin{cases} x_2 = x_1 ; \\ y_2 = y_1 \times \cos \alpha - z_1 \times \sin \alpha ; \\ z_2 = y_1 \times \sin \alpha - z_1 \times \cos \alpha . \end{cases}$$

$$\begin{cases} x_3 = x_2 \times \cos \gamma - y_2 \times \sin \gamma ; \\ y_3 = x_2 \times \sin \gamma + y_2 \times \cos \gamma ; \\ z_3 = z_2 . \end{cases}$$

Finally, projecting the point (x_3, y_3, z_3) to the X - Y plane, we can get the new position of the point after the rotation is performed.

2.4.3.2 Simulation of Eyeballs Gazing at a Fixed Target

The basic idea to simulate the eyeballs gazing at a fixed target is to set up a point representative of the focus of the eyes in the local coordinate system of the face model. By speculating the radius of the eyeball, the position of the eyeball center can be computed by the position of the pupil and the focus. Then for every rotation performed in the creation process, the new position of the eyeball center is also calculated. And the new position of the pupil can be computed by the position of the eyeball center and the focus. In this study, the speculated radius of the eyeball is set to be $0.3d$, and the position of the focus is $(EyeMid.x, EyeMid.y, 15d)$. An illustration of the focus and eyeballs is shown in Figure 2.16.

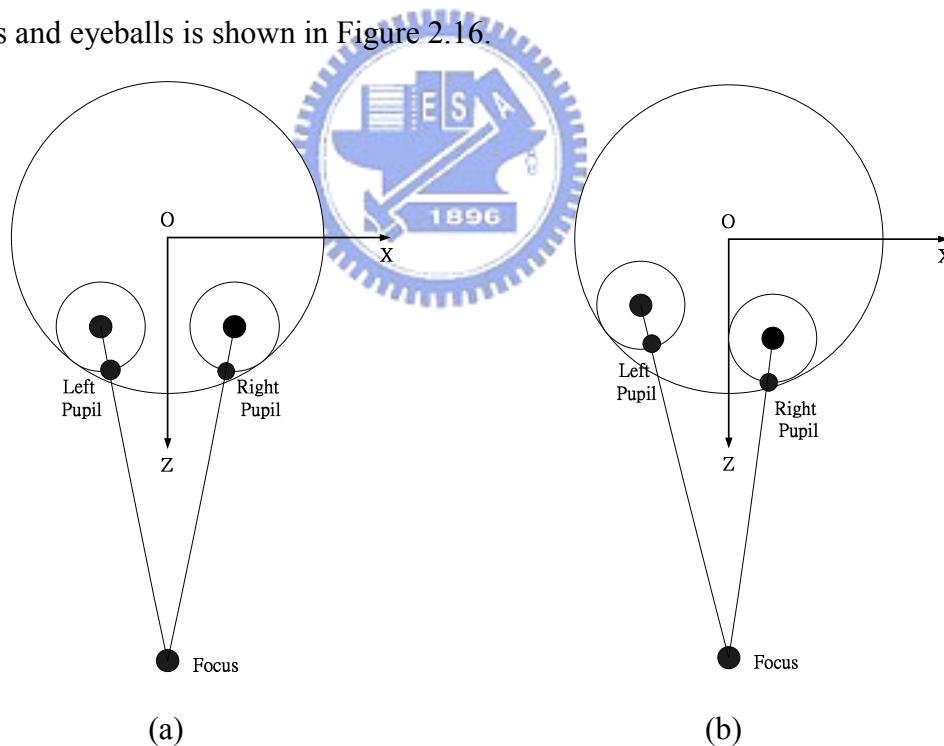


Figure 2.16 An illustration of the focus and eyeballs. (a) Before rotation. (b) After rotation.

2.4.3.3 Creation Process

An oblique cartoon face is drawn by the 72 feature points and some of the

additional points mentioned previously. The creation process is similar to the one of the frontal cartoon face, but the difference is that it must be done with some additional steps, including the rotation step. Furthermore, there are some problems after applying the rotation technique. One of the problems is that the face contour will become deformed, because some of the face contour points will be hidden and not viewable after the head is turned, and they cannot represent the face contour point any more. Therefore, we must use some other points instead of them. Another problem is that the depth of the hair contour points is defined in a flat plane, which would look unreal after the rotation, as shown in Figure 2.17. We propose a method to solve these problems, which is to change the depth of some of these points before the rotation according to the rotation direction and the angle. The detail of the proposed oblique face creation method is described in the following algorithm.

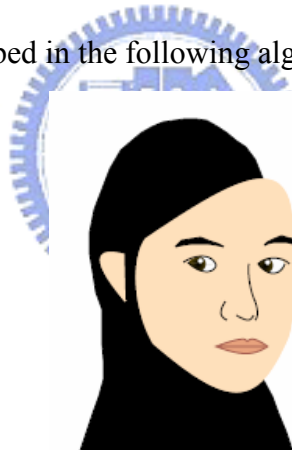


Figure 2.17 An illustration of the unreality of the hair contour.

Algorithm 2.3. *Creation of oblique cartoon face.*

Input: 72 feature points, 17 additional points, a rotation origin $O(x_0, y_0)$ in the global coordinate system, some FAPUs, including the radii of the eyeballs r_1 and r_2 in the face model, and 3 rotation angles α , β , and γ in degrees around X -, Y -, and Z -axes.

Output: an image of an oblique cartoon face.

Steps:

1. If β is larger than 0, for each of the hair point $P_{\text{hair}}(x_{\text{ph}}, y_{\text{ph}}, z_{\text{ph}})$ in the right half of the face model where $y_{\text{ph}} \geq \text{EyeMid}.y$, add a constant multiple of d to z_{ph} according to the value of x_{ph} and β .

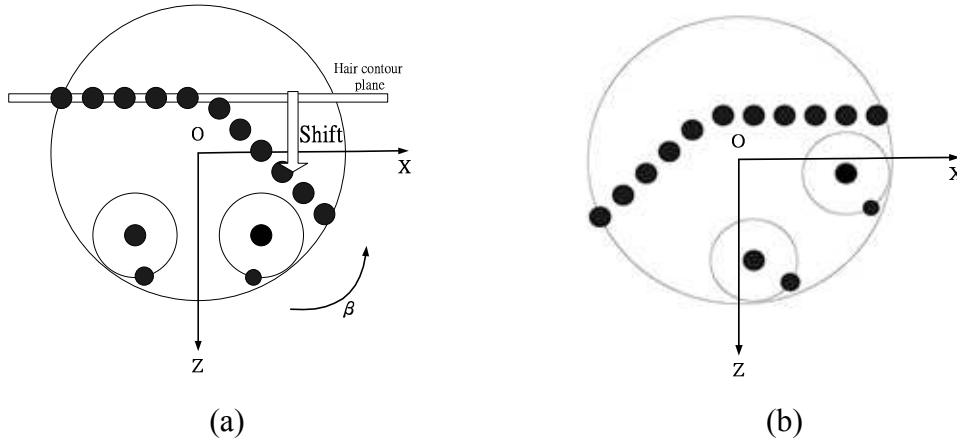


Figure 2.18 An illustration of the shift of hair contour points. (a) Before rotation. (b) After rotation.

2. If β is smaller than 0, shift the hair points in a similar way.
3. For each of the additional points and the 72 model points $P(x_p, y_p, z_p)$, apply the rotation technique in Section 2.4.3.1 to get a new point $P'(x_{p'}, y_{p'})$ on the X - Y plane.
4. For each of the points $P'(x_{p'}, y_{p'})$, transform the position of P' into the global coordinate system in the following way:

$$x_{p'} = x_p + x_o, \quad y_{p'} = y_o - y_p.$$

5. If β is larger than 10 degrees, replace the model points 11.3, 10.9, and 2.13 by Q, M, and K, respectively.
6. If β is smaller than -10 degrees, replace the model points 11.2, 10.10, and 2.14 by N, L, and J, respectively.
7. Apply Algorithm 2.2 to create the desired oblique cartoon face.
8. In Step 7, if β is larger than 0, draw the contour of the nose by the cubic Bezier curves $\text{arc}(9.7, C, 9.13)$, $\text{arc}(9.14, 9.2, 9.4)$, $\text{arc}(9.13, 9.1, 9.5)$,

and $\text{arc}(D, 9.15, E)$.

An illustration of the creation of oblique cartoon faces is shown in Figure 2.19

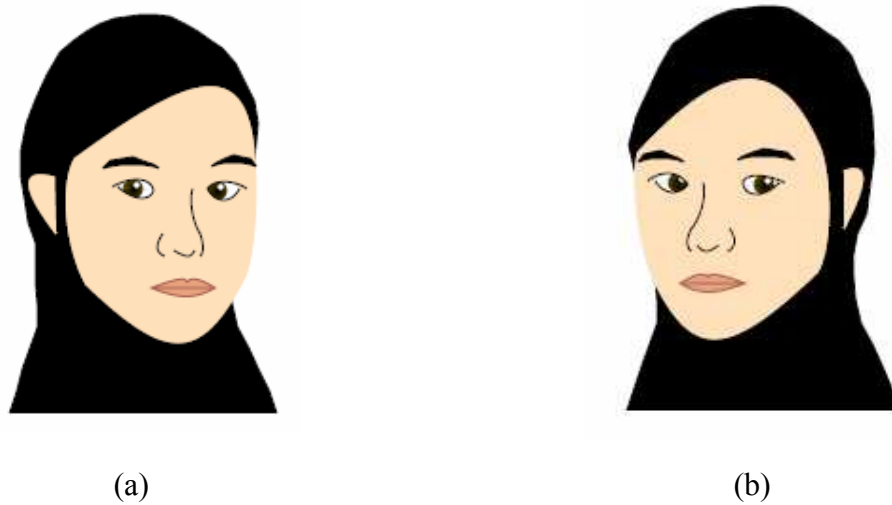


Figure 2.19 An illustration of creation of oblique cartoon faces. (a) An oblique cartoon face with $\beta = 15$ degrees. (b) An oblique cartoon face with $\beta = -15$ degrees.

2.5 Experimental Results

Some experimental results of creating cartoon faces in different poses and with different facial expressions are shown in Figure 2.20.

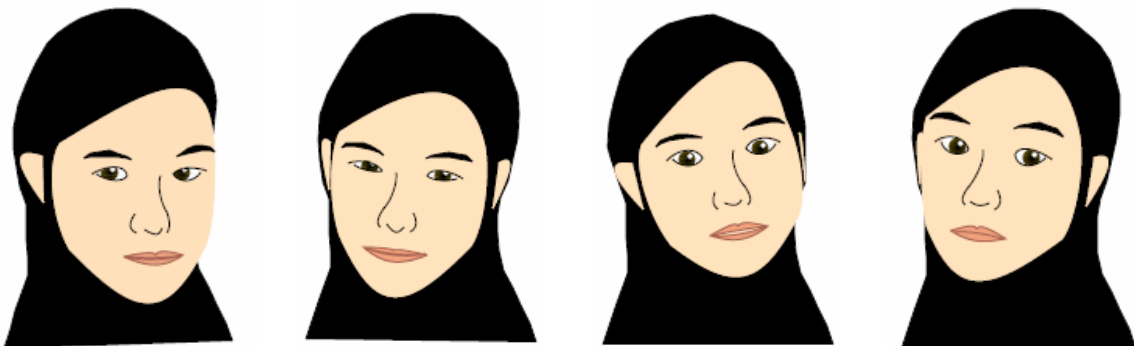


Figure 2.20 An example of experimental results for creation of cartoon faces in different poses with different facial expressions.

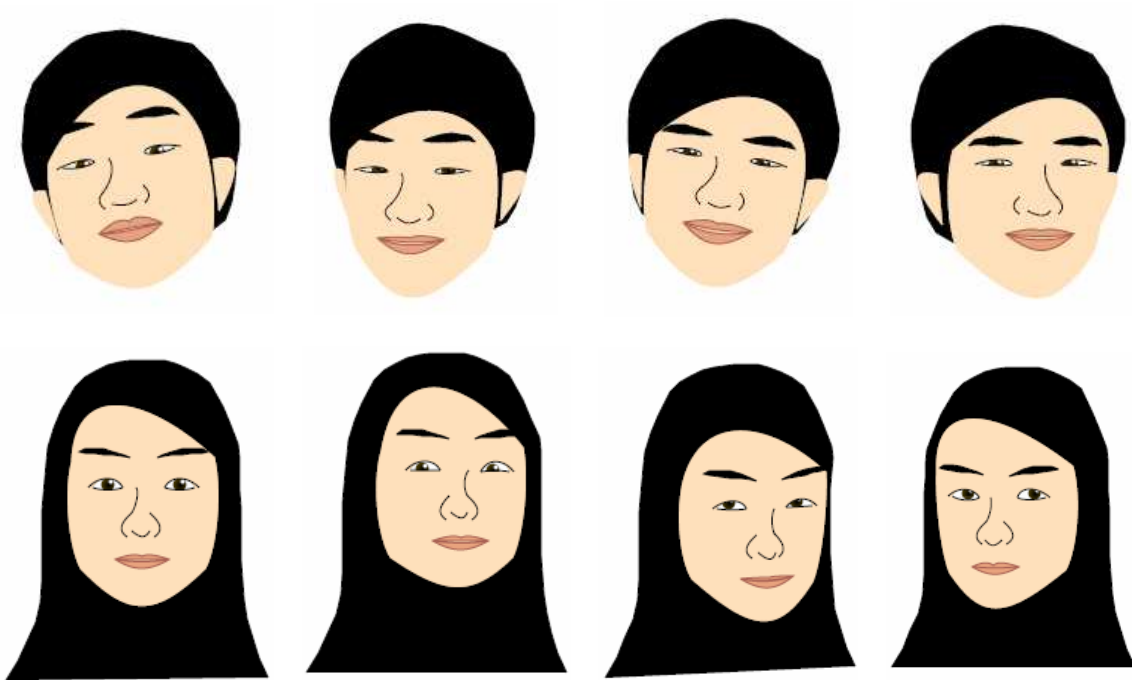
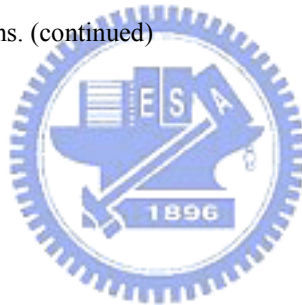


Figure 2.20 An example of experimental results for creation of cartoon faces in different poses with different facial expressions. (continued)



Chapter 3

Speech Segmentation for Lip Synchronization

3.1 Introduction to Lip Synchronization for Talking Cartoon Faces

The main purpose of this study is to establish a system which can be used to generate a speech-driven synchronized talking cartoon face. In Chapter 2, we have described how we construct a cartoon face model, which can be used to animate a moving face by changing the positions of its control points according to some of its FAPUs. The next issue is to control the lip movement by analyzing the speech track and gathering the timing information, namely, the duration, of each syllable in the speech. In the proposed system, one of the four major parts shown in Figure 1.1, which is named speech analyzer, is designed to achieve this goal. The speech analyzer receives a speech file and a script file, which is called a *transcript* in this study, and applies speech recognition techniques to get the timing information of each syllable. A flowchart of the proposed speech analyzer is shown in Figure 3.1.

A transcript is usually composed of many sentences. Although it is feasible to directly get the timing information of each syllable from the speech of the entire transcript without segmentation of the sentence utterances, it will take too much time to do so if the input audio is long. Therefore, by segmenting the entire audio into sentence utterances as the first step and then processing each segmented shorter

sentence utterance piece sequentially to extract the duration of each syllable in the speech, the overall processing speed can be accelerated. Some audio features mentioned above are listed in Table 3.1.

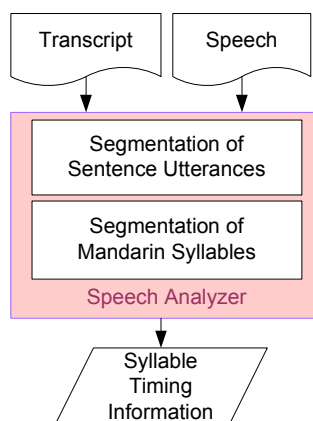


Figure 3.1 A flowchart of proposed speech analysis process.

Table 3.1 Descriptions of audio features.

Feature	Description	Example
Speech of Transcript	A speech that contains the audio data of the entire transcript including many sentences.	或許你已聽過，多補充抗氧化劑可以延緩老化。但真相為何？
Speech of Sentence Utterance	A speech that contains the audio data of a single sentence utterance including several syllables.	或許你已聽過。
Speech of Syllable	A speech that contains the audio data of a single syllable.	ㄉㄨˇ、ㄊㄩˇ、ㄓ 一、一、去一ㄥ、ㄍ ㄨˇ

In Section 3.2 , a method for segmentation of sentence utterances is proposed. In Section 3.3, the process of Mandarin syllable segmentation is described.

3.2 Segmentation of Sentence Utterances by Silence Feature

3.2.1 Review of Adopted Segmentation Method

In Lai and Tsai [4], a video recording process was designed to extract necessary feature information from a human model to generate a virtual face animation. In the recording process, the model should keep his/her head facing straightly to the camera, shake his/her head slightly for a predefined period of time while keeping silent, and read aloud the sentences on the transcript one after another, each followed by a predefined period of silent pause. An example of diagrams of recorded video contents and corresponding taken actions is shown in Figure 3.2.

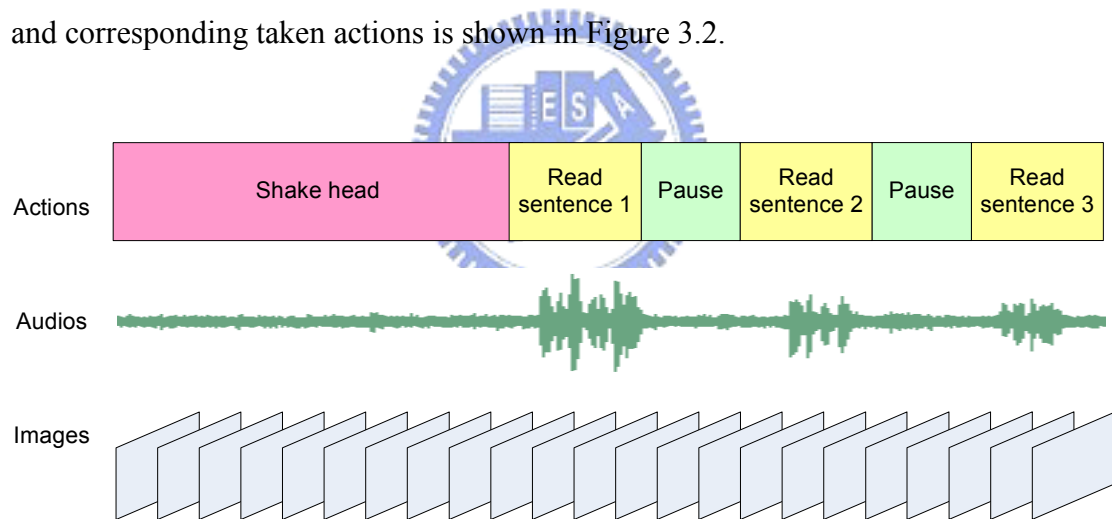
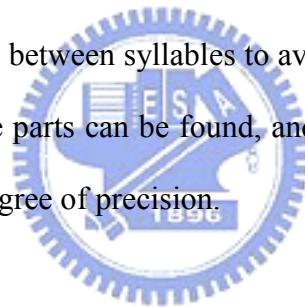


Figure 3.2 An example of recorded video contents and corresponding actions in Lai and Tsai [4].

The model keeps silent for predefined periods of time in two situations, as we can see in Figure 3.2. One is the initial silence where the model keeps silent while shaking his/her head, and the other is the intermediate silence where the model keeps silent in pauses between sentences. In order to segment the speech into sentence utterances automatically, the silence features are used in their method to detect the

silence parts between sentences and perform automatic segmentation of sentence utterances based on these detected silence parts. Due to the environment noise, the volume of these silence parts usually is not zero. Therefore, the positions of silences cannot be detected by simply searching zero-volume zones. So in their method, the maximum volume of the environment noise is measured first in the initial silence period and is then used as a threshold value to determine the intermediate silence parts by searching for the audio parts whose volumes are smaller than the threshold value. Lai and Tsai also defined another threshold to be representative of the minimum length of the intermediate silence. Because short pauses between syllables in a sentence may be viewed as silences, the minimum duration of pauses between sentences, i.e. the minimum length of intermediate silences, should be designed to be much longer than the duration between syllables to avoid incorrect detections. In such ways, the intermediate silence parts can be found, and the sentence utterances can be segmented, to a rather high degree of precision.



3.2.2 Segmentation Process

In Lai and Tsai's method, speech can be segmented into sentence utterances by silence features. However, the process of measuring the environment noise is not suitable for real cases because the duration of the initial silence period is unknown. Even in some cases, there is no initial silence period before the speaker starts to talk. Hence, the maximum volume of the environment noise cannot always be measured by adopting their method. Moreover, the duration of pauses between sentences is different for different speakers. So the silence features, including the maximum volume of the environment noise and the duration of pauses between sentences, must be learned in another way.

In the proposed system, an interface is designed to let users select the pause

between the first sentence and the second one from the input audio. Then the silence features can be learned according to the selected part of audio. A flowchart of the proposed sentence segmentation process is shown in Figure 3.3. The entire process of sentence utterance segmentation is described as an algorithm in the following.

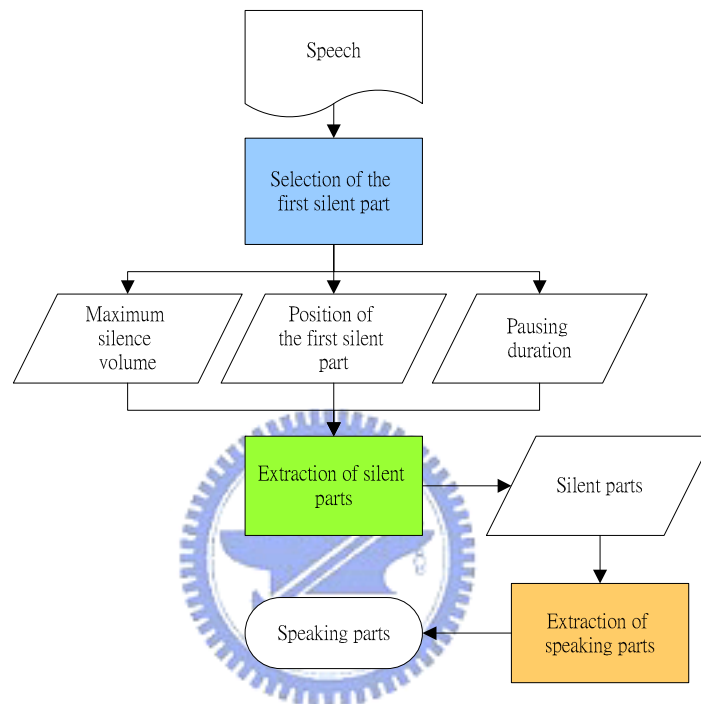


Figure 3.3 A flowchart of the sentence utterance segmentation process.

Algorithm 3.1. *Segmentation of sentence utterances.*

Input: A speech file $S_{transcript}$ of the entire transcript.

Output: Several audio parts of sentence utterances $S_{sentence1}$, $S_{sentence2}$, etc.

Steps:

1. Select the start time t_s and the end time t_e of the first intermediate silence in $S_{transcript}$ by hand.
2. Find the maximum volume V appearing in the audio part within the selected first intermediate silence.

3. Set the minimum duration of the intermediate silence D_{pause} as $(t_e - t_s) \times c_1$, where c_1 is a constant between 0.5 and 1.
4. Set the maximum volume of environment noise V_{noise} as $V \times c_2$, where c_2 is a constant between 1 and 1.5.
5. Start from t_s to find all continuous audio parts $S_{silence}$ whose volume are smaller than V_{noise} and last longer than D_{pause} .
6. Find a continuous audio part $S_{sentence}$, called a *speaking part*, which is not occupied by any $S_{silence}$.
7. Repeat Step 6 until all speaking parts are extracted.
8. Break $S_{transcript}$ into audio parts of the speaking parts found in Step 7.

Since we assume that the speech is spoken at a steady speed, the durations of the other intermediate silences are considered to be close to the first one. Therefore, c_1 in Step 3 is chosen to be 0.95. Furthermore, since we assume that the speech is spoken in a loud voice and the recording environment of the input audio is noiseless, the volume of speaking parts is considered to be much larger than that of environment noise. To avoid misses of detecting silent parts, c_2 in Step 4 is chosen to be a larger value 1.45. An example of selecting the first silent part in an input audio is shown in Figure 3.4. The red part represents the selected silence period between the first sentence and the second one. An example of experimental results of the proposed segmentation algorithm is shown in Figure 3.5. The blue and green parts represent odd and even sentences, respectively. As we can see, speaking parts of the input audio are extracted correctly.

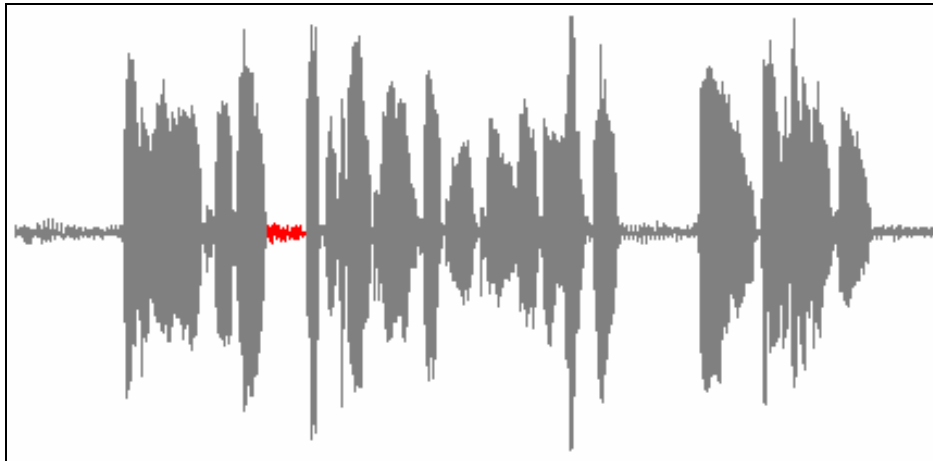


Figure 3.4 An example of selecting the first silent part in an input audio.

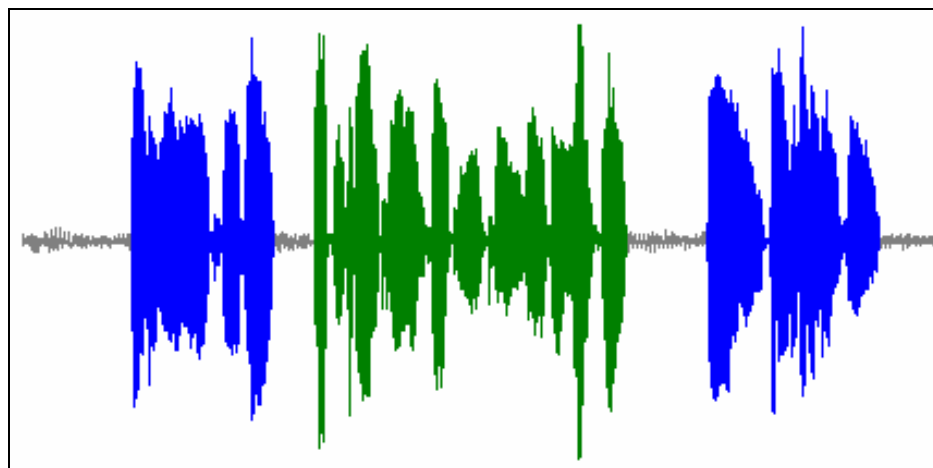


Figure 3.5 An example of sentence utterances segmentation results. The blue and green parts represent odd and even speaking parts, respectively.

3.3 Mandarin Syllable Segmentation

3.3.1 Review of Adopted Method

After the speech of the entire transcript is segmented into sentence utterances, the timing information of each syllable in a sentence can be extracted by speech recognition techniques. One of speech recognition techniques, called speech alignment, produces recognition results with higher accuracy because the syllables

spoken in input speeches are known in advance. In this study, a speech alignment technique using the Hidden Markov Model (HMM) is adopted to extract the timing information of syllables.

The HMM was widely used for speech recognition. It is a kind of statistical methods which is useful for characterizing the spectral properties of the frames of a speech pattern. In Lin and Tsai [3], a sub-syllable model was adopted together with the HMM for recognition of Mandarin syllables. After the construction of the sub-syllable model, the Viterbi search is used to segment the utterance. Then the timing information of each syllable in the input audio can be extracted.

3.3.2 Segmentation Process

In the proposed system, an entire transcript is segmented into sentences by punctuation marks. For each sentence, the Mandarin characters are transformed into their corresponding syllables according to a pre-constructed database. If a character has multiple pronunciations, its correct syllable is selected by hand. Then each sentence utterance is aligned with its corresponding syllables. Finally, the timing information for each sentences utterance is combined into a global timeline. A flowchart of the Mandarin syllable segmentation process is shown in Figure 3.6.

3.4 Experimental Results

Some experimental results of applying the proposed method for extracting the timing information of Mandarin syllables in the speech of an entire transcript are shown here. Two examples of entire audios of a transcript are shown in Figure 3.7 and Figure 3.9, and their corresponding results of syllable alignment are shown in Figure

3.8 and Figure 3.10. Durations of syllables are shown in blue and green colors.

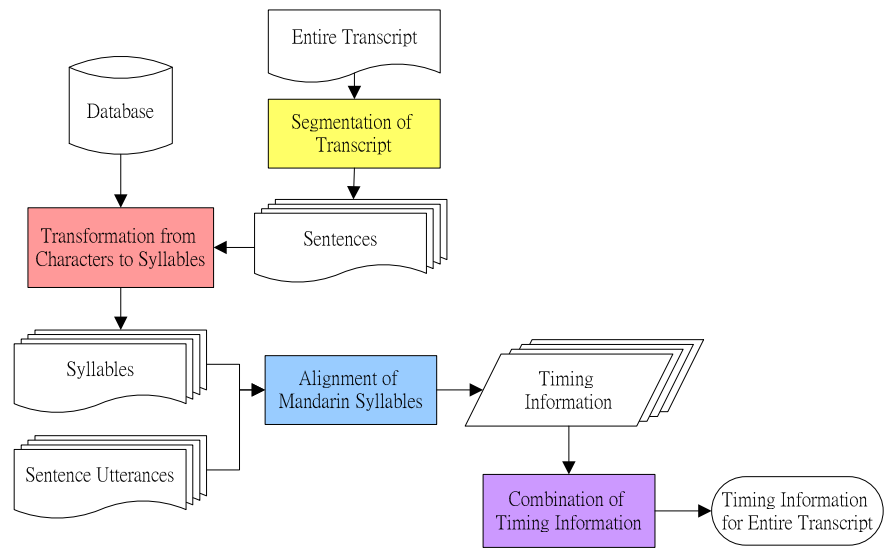


Figure 3.6 A flowchart of the Mandarin syllable segmentation process.

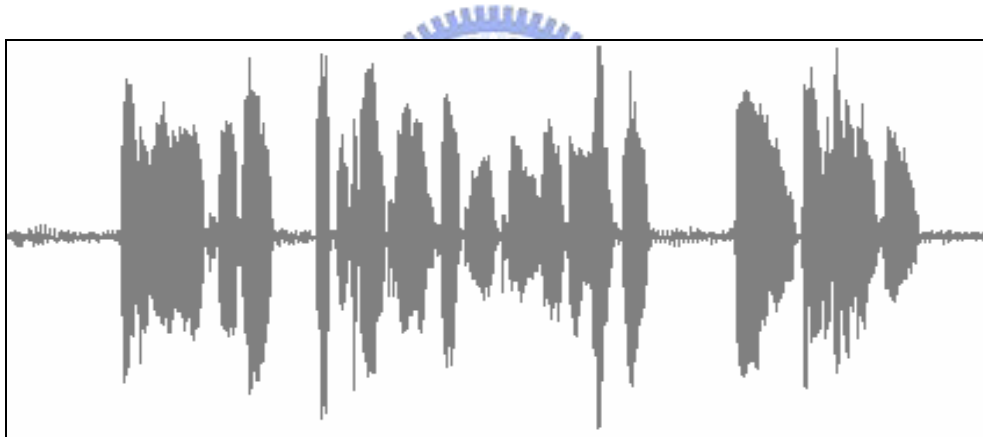


Figure 3.7 An example of entire audio data of a transcript. The content of the transcript is “或許你已聽過，多補充抗氧化劑可以延緩老化。但真相為何？”。

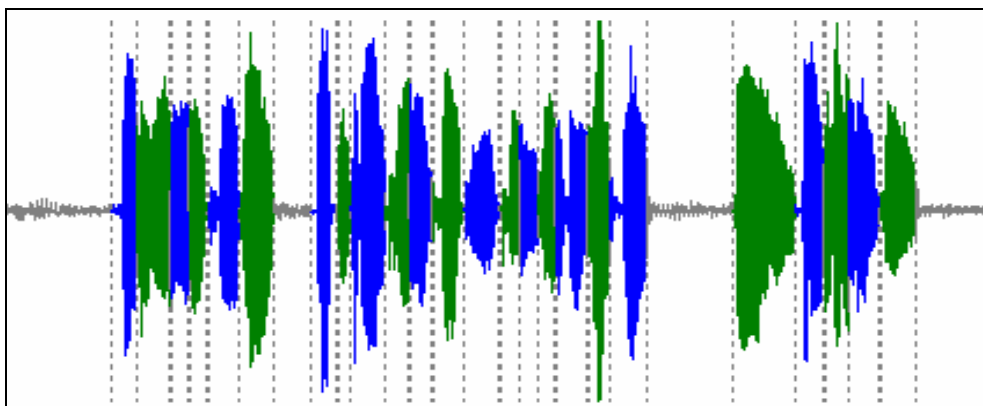


Figure 3.8 The result of syllable alignment of the audio in Figure 3.7.

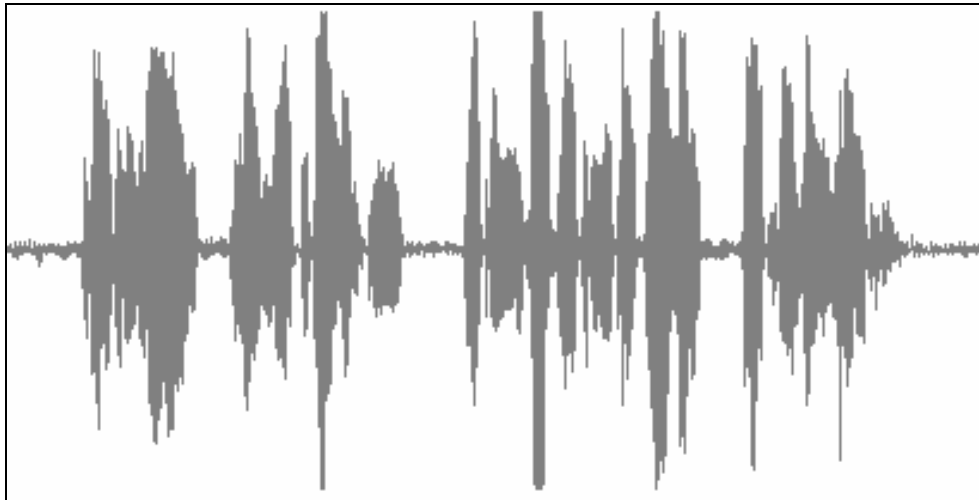


Figure 3.9 An example of entire audio data of a transcript. The content of the transcript is “長期下來，傷害不斷累積，就可能造就出一個較老、較脆弱的身體。”.

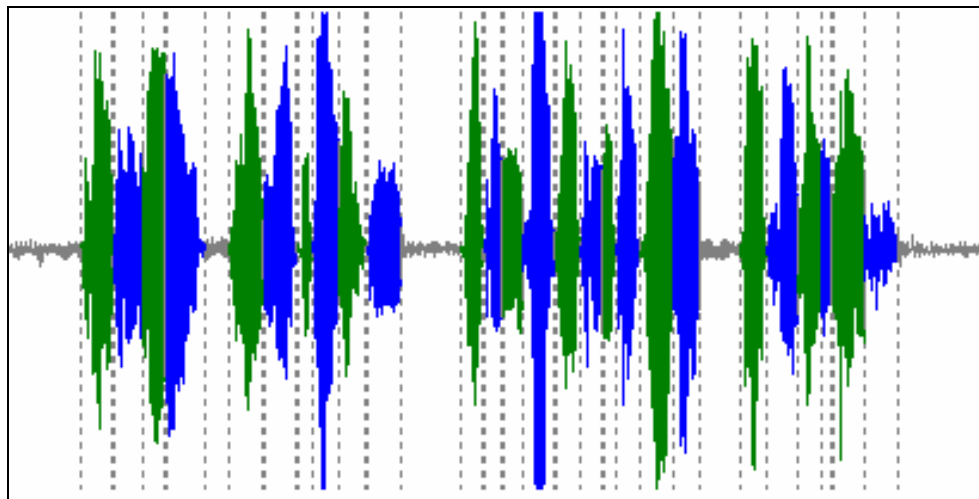


Figure 3.10 The result of syllable alignment of the audio in Figure 3.9.

Chapter 4

Animation of Facial Expressions

4.1 Introduction

In Chapter 2, we have described how to generate some basic facial expressions and head movements by controlling the control points of the face model. And in Chapter 3, we have presented how to get the timing information of syllables in the speech. However, the timing information of syllables is helpful just for the control of mouth movements. In order to control movements of the other facial parts, including eye blinks, eyebrow raises, and head movements, the timing information of the facial parts must be simulated. The cartoon face can so be animated more realistically. In the study, a statistical method is proposed to model the probabilistic functions of these facial behaviors. We tried to achieve this goal by analyzing the facial expression data from a lot of images of some TV news announcers, and finding appropriate functions to model the time intervals of the facial behaviors.

In Section 4.2, a method for analysis of facial expression data from images of TV news announcers is described. In Section 3.2, a review of some simulation methods of eye blinks and eyebrow movements adopted in this study are described. In Section 4.4 and Section 4.5, simulation methods of eyebrow movements and head movements are proposed, respectively.

4.2 Analysis of Facial Expression Data from Images of TV News Announcers

In order to simulate the behaviors of a human's face, we have tried to measure the timing information of eyebrow movements and head movements by analyzing the facial expressions of news announcers on TV news programs. For each behavior which we observed, we use two time stamps t_s and t_e to represent its start time and end time. Then we define two parameters to describe the timing information of the behavior. One is the *time interval* between two consecutive behaviors, which denotes the time interval between the end time of the first behavior and the start time of the second one. The other parameter is the *duration* of the behavior, which denotes the time interval between the start time and the end time of the behavior.

For eyebrow movements, because an eyebrow raising is usually followed by a corresponding eyebrow lowering, we define t_s as the start time of the eyebrow raising and t_e as the end time of the eyebrow lowering. For head movements, we define t_s as the time when the head starts moving and t_e as the time when the head stops moving. An illustration of the definitions of t_s and t_e for eyebrow movements and head movements are shown in Figure 4.1 and Figure 4.2, respectively.

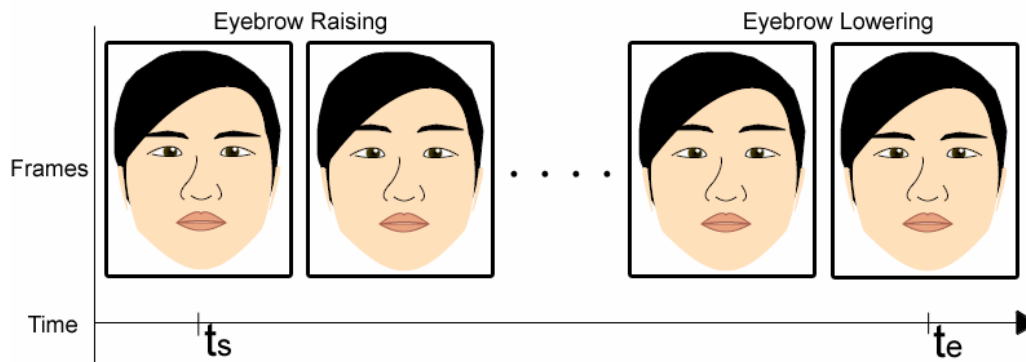


Figure 4.1 An illustration of the definitions of t_s and t_e for eyebrow movements.

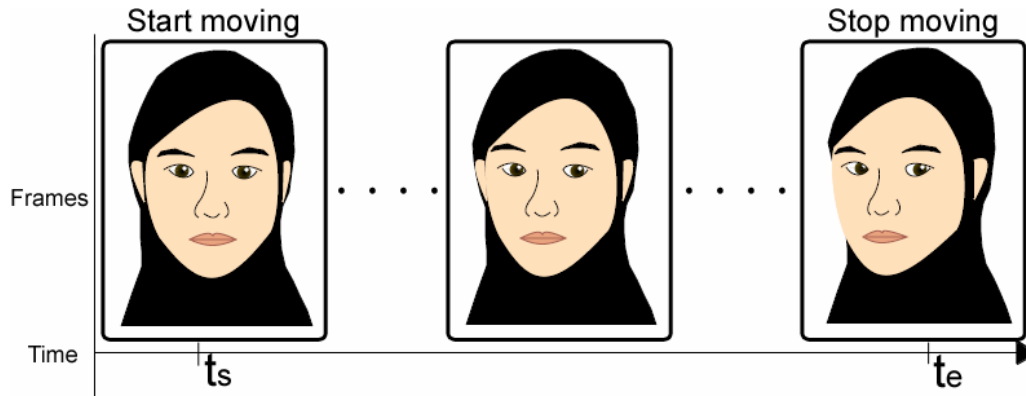


Figure 4.2 An illustration of the definitions of t_s and t_e for head movements.

Based on these definitions, we started to record the time intervals and durations according to the movements of TV News announcers. Because these movements in the video display were too fast to be observed, we segmented the video into frames. In this study, a software named VirtualDub is utilized to do this work. The video was segmented into 30 frames per second, and we can obtain the timing information of eyebrow and head movements easily by observing the images frame by frame. A screen shot of the software is shown in Figure 4.3. As we can see, the video frame and its corresponding timing information can be easily extracted.



Figure 4.3 A screen shot of the software VirtualDub.

4.3 Review of Adopted Simulation Methods of Eye Blinks and Eyebrow Movements

Lin and Tsai [3] proposed a simulation method of eye blinks by the Gamma distribution, and a simulation method of eyebrow movements by the uniform distribution. They indicated that eye blinks satisfy the three conditions defined in the Poisson process. The definition of the Poisson process is described in the following. Let the number of events which occur in a given continuous interval be counted. Let λ be the average number of events in a unit of time. We have an approximate Poisson process with parameter $\lambda > 0$ if the following conditions are satisfied.

1. The numbers of events occurring in two non-overlapping intervals are independent.
2. The probability that exactly one event occurs in a sufficiently short interval of length h is proportional to the length, say λh .
3. When h is small, the probability that two or more events occur is essentially zero.

In the Poisson process with mean λ , let X denote the number of events in the time interval $[0, t]$. Then the probability function of x arrivals is:

$$\Pr(X = x) = \frac{e^{-\lambda t} (\lambda t)^x}{x!}.$$

Let T denote the waiting time until the α^{th} event occurs, then the distribution of T can be derived as follows:

$$\begin{aligned}
P(T \leq t) &= 1 - P(T > t) = 1 - P(\text{less than } \alpha \text{ events occur in } [0, t]) \\
&= 1 - \sum_{x=0}^{\alpha-1} \frac{e^{-\lambda t} (\lambda t)^x}{x!}, \quad t > 0 \\
f(t) &= \frac{d}{dt} P(T \leq t) = \frac{\lambda (\lambda t)^{\alpha-1} e^{-\lambda t}}{(\alpha-1)!} = \frac{t^{\alpha-1} e^{-t/\theta}}{\Gamma(\alpha) \theta^\alpha}, \quad t > 0
\end{aligned}$$

where $\theta = 1/\lambda$ and $\Gamma(\alpha)$, called the gamma function, is defined as:

$$\Gamma(\alpha) = \int_0^{\infty} e^{-y} y^{\alpha-1} dy, \quad \alpha > 0.$$

Then the random variable T is said to have a gamma distribution with parameters α and θ . Since eye blinks conform to the Poisson process, Lin and Tsai used the Gamma distribution to simulate eye blinks. Let t be the time interval between two consecutive eye blinks. By the definition mentioned above, α is 2 and θ is the average time interval between eye blinks. In their experiment, the average time interval between eye blinks was set to be 1.48 second. An illustration of the probability function of eye blinks $f(t)$ is shown in Figure 4.4. It can be observed that the blue curve, which denotes the probability function of the Gamma distribution with $\alpha = 2$ and $\theta = 1.48$, and the pink curve, which denotes the probability function generated from the analysis data of TV news announcers, are close. Therefore, the utilization of the Gamma distribution is verified to be able to simulate the real human's eye blinks.

To apply the gamma distribution, Lin and Tsai used an inverse transformation method to simulate a random variable which has the distribution. The inverse transformation method states that if X is a continuous random variable with a strictly increasing cumulative distribution function (cdf) F , and if $Y = F(X)$, then Y has a uniform distribution in $[0, 1]$. Therefore, given a continuous uniform variable U on $[0, 1]$ and an invertible distribution function F , the random variable $X = F^{-1}(U)$ has distribution F . Suppose that $F(x) = 1 - e^{-\lambda x}$, which is the cdf of an exponential distribution function with a rate λ . Let $1 - e^{-\lambda x} = u$, then $x = F^{-1}(u) = -(1/\lambda) \log(1-u)$. If U

is a uniform variable on $(0, 1)$, then $F^{-1}(U) = -(1/\lambda)\log(1-U)$ has distribution F , i.e. $F^{-1}(U)$ is exponentially distributed with a rate λ . Since $1-U$ is also uniformly distributed in $(0, 1)$, it follows that $-(1/\lambda)\log U$ is exponential with a rate λ . By the fact that the sum of n independent exponential random variables, which are all with a rate λ , is gamma distributed with parameters n and $1/\lambda$, the following formula was adopted to generate such a random variable:

$$X = -\sum_{i=1}^n \frac{1}{\lambda} \log U_i = -\frac{1}{\lambda} \log\left(\prod_{i=1}^n U_i\right)$$

where U_1, \dots, U_n are independent uniform random variables in $(0, 1)$.

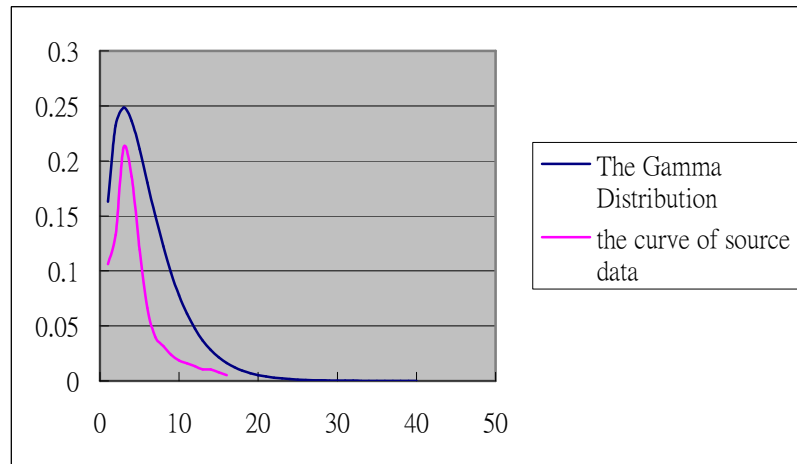


Figure 4.4 An illustration of the probability function of eye blinks in Lin and Tsai [3]. The one in blue color is the probability function of the Gamma distribution with $\alpha = 2$ and $\theta = 1.48$. The other one in pink color is the probability function of eye blinks approximated from the analysis data of TV News announcers.

For the simulation of eyebrow movements, Lin and Tsai indicated that the time interval of eyebrow movements is decided with the content of the speech. So they considered that eyebrow movements occur randomly, and used a uniform distribution function to perform the simulation. In their experiment, the maximum time interval of eyebrow movement was set as an integer 24. And the probability density function is defined by $f(x) = 1/24$, where x denotes the time interval of eyebrow movements and

is selected randomly as an integer from 1 to 24. They used a random number generator to apply the uniform distribution.

4.4 Simulation of Eyebrow Movements

In Lin and Tsai's method, only one parameter, namely, the time interval, is involved to simulate the eyebrow movements. As we mentioned in Section 4.2, another parameter, namely, the duration, should be considered, because in real cases, one may keep his/her eyebrows in a raising state for a long time. Besides, the time interval adopted in their method was chosen to be an integer, which is not precise enough to simulate the time interval of a real human's eyebrow movements. Moreover, the maximum time interval was set as 24 and was not able to show the statistical meaning. In this section, a collection of statistics of the time interval and the duration defined in Section 4.2 is provided, and a method to simulate eyebrow movements is proposed.

We have observed eyebrow movements from the images of about ten different TV news announcers. According to the observation, some examples of statistical data of time intervals and durations of eyebrow movements are listed in Table 4.1 and Table 4.2. We have found that the time intervals and the durations of eyebrow movements are different for different TV news announcers, and correlated to the content of the speech. For example, one may raise his/her eyebrows while he/she is saying something compelling or interesting. Hence, we consider that time intervals and durations of eyebrow movements are distributed randomly and can be simulated by the uniform distribution. According to the analyzed data, we compute the mean value and the standard deviation value of the time intervals and the durations. To

avoid some special cases ('outliers') which are extremely large or small and may affect the simulation result, using the mean value and the standard deviation value is better than directly choosing a number. In the experiment, the mean value of the time interval is 6.72, and the standard deviation value is 5.82, so we randomly generate a variable on $[6.72 - 5.82, 6.72 + 5.82]$ to represent the time interval between two consecutive eyebrow movements. The maximum number of significant digits to the right of the decimal point is set to 2, so the interval $[0.90, 12.54]$ is divided into 1165 discrete floating point numbers where every number has a probability of $1/1165$ to be generated as the time interval of eyebrow movements. In a similar way, the mean value of the duration is 0.75, and the standard deviation value is 0.24, so we randomly generate a variable on $[0.51, 0.99]$ to represent the duration of eyebrow movements.

Table 4.1 Statistics of time intervals of eyebrow movements.

CTN News		ETtoday News		Other News	
	Time interval of eyebrow movements (second)		Time interval of eyebrow movements (second)		Time interval of eyebrow movements (second)
1	0.433	1	5.267	1	2.367
2	0.7	2	7.233	2	4.333
3	1.367	3	8.167	3	1.9
4	1.933	4	13.333	4	3.267
5	1.333	5	12.333	5	2.2
6	0.6	6	3.333	6	3.067
7	0.6	7	6.167	7	2.833
8	1.1	8	7.2	8	1.133
9	3.767	9	6.833	9	7.233

Table 4.2 Statistics of durations of eyebrow movements.

CTN News		ETtoday News		Other News	
	Duration of eyebrow movements (second)		Duration of eyebrow movements (second)		Duration of eyebrow movements (second)
1	1.067	1	0.667	1	0.433
2	0.633	2	0.6	2	0.3
3	0.5	3	0.567	3	0.6
4	0.8	4	0.967	4	1.2
5	0.767	5	0.533	5	0.8
6	1.333	6	0.7	6	1.167
7	0.633	7	0.733	7	1.067
8	0.7	8	0.7	8	0.633
9	0.767	9	0.567	9	0.5

4.5 Simulation of Head Tilting and Turning

4.5.1 Simulation of Head Tilting

Similar to the simulation of eyebrow movements, the timing information of head tilting is related to the content of the speech and varies with the habit of different TV news announcers. Some examples of statistical data of time intervals and durations of head tilting are listed in Table 4.3 and Table 4.4. In the experiment, the mean value of the time interval is 1.60, and the standard deviation value is 1.72, so we randomly

generate a variable in $[0, 3.32]$ to represent the time interval between two consecutive head tilting. However, the speaking speed of the TV news announcer is much faster than the normal condition, so the value of the simulated time interval for head tilting is not suitable for the speech in the normal pace. So we adjust the time interval in $[0, 3.32]$ to the double interval $[0, 6.64]$ to represent the time interval of head tilting with a normal pace of speech. Similarly, the mean value of the duration is 0.35, and the standard deviation value is 0.15, and we can adjust the interval $[0.2, 0.5]$ to get a better simulation in $[0.4, 1.0]$.

Table 4.3 Statistics of time intervals of head tilting.

CTN News				ETtoday News				Other News			
	Time interval of head tilting (second)		Time interval of head tilting (second)		Time interval of head tilting (second)		Time interval of head tilting (second)		Time interval of head tilting (second)		Time interval of head tilting (second)
1	2.4	10	2.2	1	0.3	10	0.367	1	0.2	10	4.2
2	1.467	11	0.266	2	1.233	11	4.2	2	5.433	11	2.233
3	0.567	12	1.366	3	1.233	12	2.233	3	0.1	12	0.667
4	0.333	13	0.9	4	0.3	13	0.667	4	0.2	13	2.033
5	0.4	14	0.767	5	2.933	14	2.033	5	4.233	14	0.7
6	0.3	15	0.633	6	0.667	15	0.7	6	1.033	15	1.6
7	0.333	16	1.767	7	0.5	16	1.6	7	0.133	16	0.667
8	2.767	17	0.733	8	2.967	17	1.1	8	1.667	17	4.233
9	0.733	18	0.4	9	4.5	18	0.433	9	7.633	18	2.467

Table 4.4 Statistics of durations of head tilting.

CTN News				ETtoday News				Other News			
	Duration of head tilting (second)		Duration of head tilting (second)		Duration of head tilting (second)		Duration of head tilting (second)		Duration of head tilting (second)		Duration of head tilting (second)
1	0.367	10	0.3	1	0.2	10	0.567	1	0.2	10	0.4
2	0.6	11	0.267	2	0.233	11	0.133	2	0.5	11	0.333
3	0.233	12	0.267	3	0.5	12	0.5	3	0.5	12	0.233
4	0.233	13	0.333	4	0.4	13	0.667	4	0.5	13	0.3
5	0.233	14	0.233	5	0.267	14	0.433	5	0.2	14	0.433
6	0.233	15	0.2	6	0.267	15	0.767	6	0.267	15	0.3
7	0.6	16	0.333	7	0.5	16	0.633	7	0.267	16	0.267
8	0.2	17	0.333	8	0.3	17	0.3	8	0.3	17	0.333
9	0.4	18	0.267	9	0.867	18	0.333	9	0.4	18	0.467

4.5.2 Simulation of Horizontal Head Turning

Horizontal head turning is also simulated by the uniform distribution. Some examples of statistical data of time intervals and durations of horizontal head turning are listed in Table 4.5 and Table 4.6. The mean value, the standard deviation value, and the adopted intervals of the uniform random variable are shown in Table 4.7.

4.5.3 Simulation of Vertical Head Turning

In our experiment, we have found that the time interval of vertical head turning is partially related to the time of pauses in the speech. The TV news announcer usually nods when his/her speech is coming to a pause, and breaths during the pause time with his/her head raised. Some examples of the time stamp of vertical head

turning and its corresponding pause recorded in our experiment are listed in Table 4.8.

Table 4.5 Statistics of time intervals of horizontal head turning.

CTN News		ETtoday News		Other News	
	Time interval of horizontal head turning (second)		Time interval of horizontal head turning (second)		Time interval of horizontal head turning (second)
1	2.9	1	0.7	1	0.633
2	6.934	2	0.233	2	0.167
3	1.567	3	1.433	3	1.267
4	3	4	2.433	4	1.9
5	1.8	5	0.833	5	1.333
6	0.667	6	0.867	6	0.067
7	7.1	7	1.6	7	2.1
8	2.5	8	0.7	8	1.2
9	3.367	9	2.5	9	1.6

Due to this finding, we collect some statistics of the time interval between the end time of the head nod (i.e. the head turning in a vertical down direction) and the start time of the pause. We also collect the duration of the nod and the duration of the head raising after the nod. Some examples of the three statistics are listed in Table 4.9, Table 4.10, and Table 4.11, respectively. Then, as we can see in Table 4.12, by computing the mean value and the standard deviation value, the time interval between the nod and the pause can be simulated as a random variable t_1 in $[0.07, 0.35]$. The duration of the nod and the head raising after the nod can be simulated and adjusted as d_1 in $[0.34, 0.94]$ and d_2 in $[0.34, 0.70]$ for a normal pace of speech which is mentioned above in Section 4.5.1. By finding all silent parts $S_{silence}$ in a speech based

on the method described in Algorithm 3.1, and using the symbol $t_{silence}$ to represent the start time of the pause for each $S_{silence}$, the occurring time of the nod can be simulated as $[t_{silence} - t_1 - d_1, t_{silence} - t_1]$, and the occurring time of the head raising after the nod can be simulated as $[t_{silence}, t_{silence} + d_2]$.

Table 4.6 Statistics of durations of horizontal head turning.

CTN News		ETtoday News		Other News	
	Duration of horizontal head turning (second)		Duration of horizontal head turning (second)		Duration of horizontal head turning (second)
1	0.233	1	0.167	1	0.367
2	0.333	2	0.633	2	0.267
3	0.2	3	0.433	3	0.3
4	0.1	4	0.2	4	0.333
5	0.267	5	0.267	5	0.1
6	0.4	6	0.367	6	0.267
7	0.5	7	0.267	7	0.4
8	0.3	8	0.2	8	0.433
9	0.433	9	0.5	9	0.3

Table 4.7 The mean values, the standard deviation values, and the adopted intervals of uniform random variables for simulation of horizontal head turning.

	Mean	Standard deviation	Result of the interval	Adjustment of the interval
Time interval of horizontal head turning (second)	2.00	1.96	[0.04, 3.96]	[0.08, 7.92]
Duration of horizontal head turning (second)	0.33	0.13	[0.2, 0.46]	[0.4, 0.92]

Table 4.8 Some examples of the relation between the vertical head turning and the pause time.

Direction of vertical head turning	Start time (second)	End time (second)	Pause time (second)
Down	2.533	2.767	2.7
Up	2.867	2.967	
Down	4.033	4.133	4.567
Up	4.533	4.8	
Down	8	8.333	8.533
Up	8.4	8.567	

Table 4.9 Statistics of time intervals between the nod and the pause.

CTN News		ETtoday News		Other News	
	Time interval between the nod and the pause (second)		Time interval between the nod and the pause (second)		Time interval between the nod and the pause (second)
1	0.433	1	0.2	1	0.2
2	0.2	2	0.233	2	0.3
3	0.267	3	0.067	3	0.233
4	0.133	4	-0.033	4	0.333
5	0.333	5	-0.1	5	0.267
6	0.033	6	0.333	6	0.367
7	0.333	7	0.1	7	0
8	0.1	8	0.033	8	0.167
9	0.167	9	0.367	9	0.3

Table 4.10 Statistics of durations of the nod.

CTN News		ETtoday News		Other News	
	Duration of the nod (second)		Duration of the nod (second)		Duration of the nod (second)
1	0.233	1	0.167	1	0.333
2	0.1	2	0.367	2	0.567
3	0.467	3	0.5	3	0.2
4	0.333	4	0.467	4	0.367
5	0.167	5	0.467	5	0.467
6	0.533	6	0.5	6	0.267
7	0.5	7	0.3	7	0.233
8	0.233	8	0.333	8	0.3
9	0.3	9	0.167	9	0.267

Table 4.11 Statistics of durations of the head raising after the nod.

CTN News		ETtoday News		Other News	
	Duration of the head raising (second)		Duration of the head raising (second)		Duration of the head raising (second)
1	0.1	1	0.2	1	0.167
2	0.267	2	0.433	2	0.2
3	0.233	3	0.167	3	0.267
4	0.267	4	0.233	4	0.333
5	0.167	5	0.4	5	0.267
6	0.2	6	0.3	6	0.233
7	0.333	7	0.2	7	0.333
8	0.233	8	0.267	8	0.167
9	0.267	9	0.2	9	0.367

Table 4.12 The mean value, the standard deviation value, and the adopted intervals of uniform random variables for simulation of vertical head turning.

	Mean	Standard deviation	Result of the interval	Adjustment of the interval
Time interval between the nod and the pause (second)	0.21	0.14	[0.07, 0.35]	
Duration of the nod (second)	0.32	0.15	[0.17, 0.47]	[0.34, 0.94]
Duration of the head raising after the nod (second)	0.26	0.09	[0.17, 0.35]	[0.34, 0.70]



Chapter 5

Talking Cartoon Face Generation

5.1 Introduction

Up to now, we have known how to create cartoon faces with different facial expressions in different poses, how to gather the timing information of syllables in the speech, and how to simulate the facial expressions for facial animations. According to the timing information of syllables mentioned in Chapter 3, the time about when to make the cartoon face speak is identified. However, the information about how to make a speaking motion for the cartoon face is still unknown. In this chapter, we propose an automatic method for talking cartoon face generation to solve this problem. In our method, twelve *basic mouth shapes* are defined, and syllables are translated into combinations of the basic shapes. By assigning basic mouth shapes into proper key frames in an animation and applying an interpolation technique to generate other frames among key frames, the animation of talking cartoon faces can be created.

Definitions of basic mouth shapes are presented in Section 5.2. A review of the adopted time intervals between the mouth shapes within a syllable is described in Section 5.3. Finally, the process of talking cartoon face generation by synthesizing lip movements is illustrated in Section 5.4.

5.2 Definitions of Basic Mouth Shapes

In Mandarin, characters are produced by monosyllables with 21 initials, 38 finals, and 5 tones. The total number of different syllables for Mandarin speaking is 1345 and can be reduced to 411 by ignoring the differences in tones. In Yeh [15], the 21 kinds of initials were classified into 7 classes according to the manners of articulation. The result of the classification is shown in Table 5.1, and the corresponding mouth shapes are shown in Table 5.2. Based on the classification, the 411 visemes corresponding to the 411 syllables can be reduced to 167 postures.


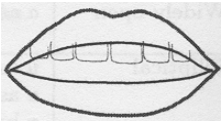
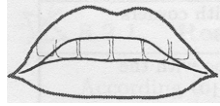

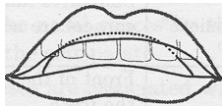
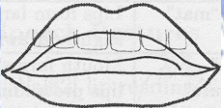
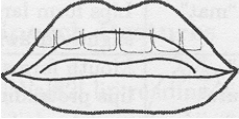
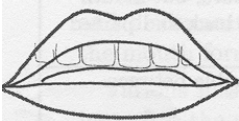
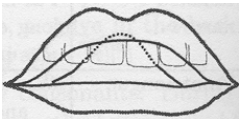
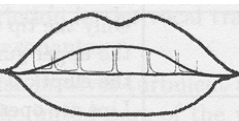
Table 5.1 Classification of initials according to the manners of articulation proposed in Yeh [15].

Place of Articulation		Manner of Articulation	Members of the Classes
Upper Obstruction	Lower Obstruction		
Upper Lip	Lower Lip	Bilabial	ㄅ、ㄆ、ㄇ
Upper Teeth	Lower Lip	Labiodental	ㄆ
Tooth Ridge	Tongue Tip	Tongue Tip (Apical)	ㄉ、ㄊ、ㄋ、ㄌ
Soft Palate	Velar	Velar	ㄍ、ㄎ、ㄍ
Palate	Front Palatal	Palatal	ㄑ、ㄓ、ㄔ
Palate	Retroflex	Retroflex	ㄑ、ㄓ、ㄔ
Behind Lower Teeth	Tongue Tip	Blade-Alveolar	ㄆ、ㄇ、ㄌ

Since the face model adopted in this study is a cartoon-like one, not a photorealistic one, the postures can be further classified into a small number of groups based on the similarity between mouth shapes of pronunciations in Mandarin initials and finals. In this section, twelve basic mouth shapes are defined based on the nine basic mouth shapes proposed in Chen and Tsai [1]. In Section 5.2.1, a review of the

definitions of basic mouth shapes in [1] is given. After some adjustments of their definitions, new definitions of basic mouth shapes are proposed in this study, as shown in Section 5.2.2.

Table 5.2 An illustration of 7 kinds of mouth shapes of initials proposed in Yeh [15].

Mouth Shapes of Pronunciations		Members of the Classes
 <i>Lip shut</i>	 <i>Relaxed narrow</i>	ㄅ、ㄆ、ㄇ
 Protrusion	 Relaxed narrow	ㄉ
 Tongue to gum	 Medium oval	ㄏ、ㄏ、ㄏ、ㄏ
 Medium oval		ㄍ、ㄍ、ㄍ
 Front tongue		ㄌ、ㄌ、ㄌ
 Tip to gum		ㄒ、ㄒ、ㄒ、ㄒ
 Tightened narrow		ㄆ、ㄆ、ㄆ

5.2.1 Review of Definition of Basic Mouth Shapes

Chen and Tsai [1] proposed a method to reduce the above-mentioned 7 kinds of mouth shapes for Mandarin initials to 3 basic initial mouth shapes, as shown in Table 5.3. And based on the Taiwan Tongyoung Romanization, which contains a transcription of the Mandarin phonetics into a set of English alphabets, mouth shapes for Mandarin finals were also reduced to a set of combinations with 7 basic mouth shapes, as shown in Table 5.4. According to the reduction result, any given syllable which consists of phonemes can be translated into a combination of basic mouth shape symbols. For example, the phoneme “ㄩ” can be translated into “u,” a syllable “ㄅㄟ” into “mei,” etc.

Table 5.3 Three basic mouth shapes of Mandarin initials in Chen and Tsai [1].

Mouth Shape Symbols	Members of the classes
<i>m</i>	ㄇ、ㄇ、ㄇ
<i>f</i>	ㄈ
<i>h'</i>	ㄏ、ㄏ、ㄏ、ㄏ、ㄏ、ㄏ、ㄏ、ㄏ、ㄏ、 ㄏ、ㄏ、ㄏ、ㄏ、ㄏ、ㄏ

Since the pre-posture of the open mouth shape is sometimes missed, *h'* is eliminated in their method if there is a symbol (including *a*, *i*, *u*, *e*, *o*) behind it. For example, the syllable “ㄏㄟ” is translated into “*h'i*,” where the “*i*” is right behind the “*h'*,” and the final transcription of “ㄏㄟ” will be adjusted to be “*i*” automatically.

After the classification of the Mandarin initials and finals, they defined the mouth shape using the concept of control points, which is mentioned previously in Chapter 2, as shown in Figure 5.1. Each basic mouth shape was defined by a set of

mouth control points {8.9, 8.2, 8.4, 8.3}. The details of the definitions are listed as follows. Illustrations of basic mouth shapes of the Mandarin initials and finals are shown in Figure 5.2 and Figure 5.3, respectively.

Table 5.4 A set of combinations with 7 basic mouth shapes of Mandarin finals in Chen and Tsai [1].

Mouth Shape Symbols	Members of the classes	Combinations of Mouth Shapes	Members of the classes
<i>a</i>	ㄚ	<i>au</i>	ㄨㄚ
<i>e</i>	ㄝ	<i>ou</i>	ㄨㄛ
<i>i</i>	ㄟ	<i>en</i>	ㄣ
<i>o</i>	ㄛ	<i>an</i>	ㄢ
<i>u</i>	ㄨ, ㄩ	<i>ei</i>	ㄟ
<i>n</i>	ㄣ	<i>ai</i>	ㄞ
<i>h</i>	ㄣ, ㄨ, ㄛ		

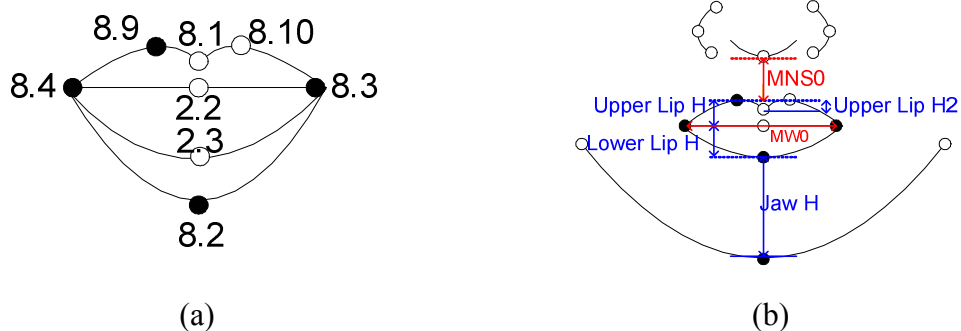


Figure 5.1 An illustration of basic components for definition of basic mouth shapes. (a) Control points of the mouth. (b) FAPUs of the mouth and the nose.

A. Basic mouth shapes of the Mandarin initials.

1. Basic mouth shape *m*: *m* is the initial mouth shape in the neutral state:

$$m = \{8.9_{\text{Netural}}, 8.2_{\text{Netural}}, 8.4_{\text{Netural}}, 8.3_{\text{Netural}}\}.$$

2. Basic mouth shape f : A part of the lower lip is hidden by the upper lip:

$$f = \{8.9_{\text{Netural}}, dn, 8.4_{\text{Netural}}, 8.3_{\text{Netural}}\},$$

$$\text{where } dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} - 0.5 \times \text{LowerLipH}).$$

3. Basic mouth shape h' and h :

$$h'/h = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.3 \times \text{MNSO});$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.3 \times \text{MNSO});$$

$$lf = (8.4_{\text{Netural}.x}, 8.4_{\text{Netural}.y} - 1);$$

$$rt = (8.3_{\text{Netural}.x}, 8.3_{\text{Netural}.y} - 1).$$

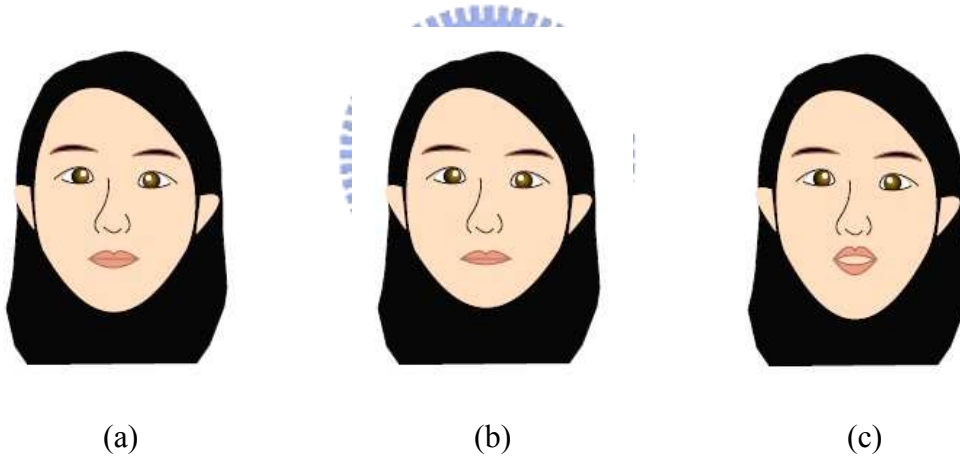


Figure 5.2 An illustration of basic mouth shapes of Mandarin initials in Chen and Tsai [1]. (a) Basic mouth shape m . (b) Basic mouth shape f . (c) Basic mouth shape h .

B. Basic mouth shape of the Mandarin finals.

1. Basic mouth shape a :

$$a = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.3 \times \text{MNSO});$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.8 \times \text{MNSO});$$

$$lf = (8.4_{\text{Netural}.x} + \text{MW0}/8, 8.4_{\text{Netural}.y});$$

$$rt = (8.3_{\text{Netural}.x} - MW0/8, 8.3_{\text{Netural}.y}).$$

2. Basic mouth shape *i*:

$$i = \{8.9, dn, lf, rt\},$$

$$\text{where } dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + MNS0/6);$$

$$lf = (8.4_{\text{Netural}.x} - MW0/12, 8.4_{\text{Netural}.y} - 3);$$

$$rt = (8.3_{\text{Netural}.x} + MW0/12, 8.3_{\text{Netural}.y} - 3).$$

3. Basic mouth shape *u*:

$$u = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.3 \times MNS0);$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.3 \times MNS0);$$

$$lf = (8.4_{\text{Netural}.x}, 8.4_{\text{Netural}.y} - 1);$$

$$rt = (8.3_{\text{Netural}.x}, 8.3_{\text{Netural}.y} - 1).$$

4. Basic mouth shape *e* :

$$e = \{8.9, dn, lf, rt\},$$

$$\text{where } dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + MNS0/2);$$

$$lf = (8.4_{\text{Netural}.x} - MW0/12, 8.4_{\text{Netural}.y} - 3);$$

$$rt = (8.3_{\text{Netural}.x} + MW0/12, 8.3_{\text{Netural}.y} - 3).$$

5. Basic mouth shape *o* :

$$o = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.3 \times MNS0);$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.5 \times MNS0);$$

$$lf = (8.4_{\text{Netural}.x} + 0.2 \times MW0, (up.y + dn.y)/2);$$

$$rt = (8.3_{\text{Netural}.x} - 0.2 \times MW0, lf.y).$$

6. Basic mouth shape *n* :

$$n = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.25 \times MNS0);$$

$$dn = (8.2_{\text{Netural-}x}, 8.2_{\text{Netural-}y} + 0.5 \times MNS0);$$

lf = equals to the last former mouth shape;

rt = equals to the last former mouth shape.

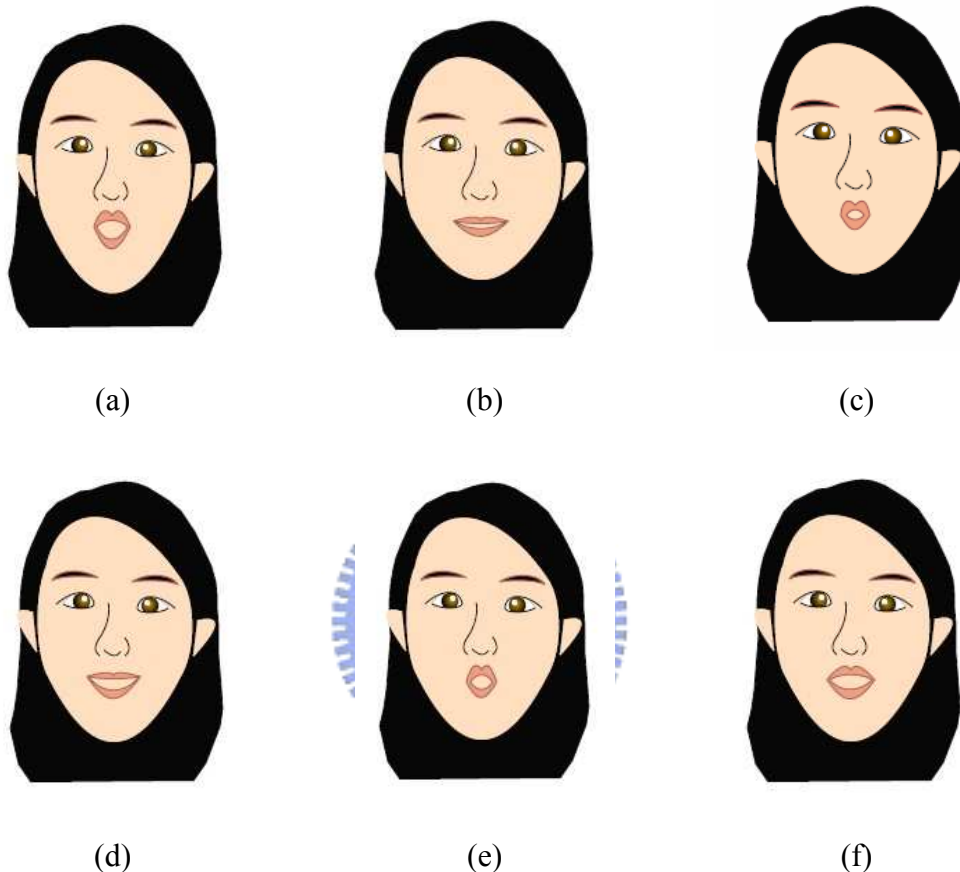


Figure 5.3 An illustration of basic mouth shapes of the Mandarin finals in Chen and Tsai [1]. (a) Basic mouth shape a . (b) Basic mouth shape i . (c) Basic mouth shape u . (d) Basic mouth shape e . (e) Basic mouth shape o . (f) Basic mouth shape n .

5.2.2 New Definitions of Basic Mouth Shapes

Since we consider that the classification of the Mandarin initials and finals in Chen and Tsai [1] was too simple to represent the differences between mouth shapes, we define twelve basic mouth shapes instead. For example, the mouth shape of the phoneme “ $ㄗ$ ” is not exactly the same as that of the phoneme “ $ㄨ$,” where the lip corners for “ $ㄗ$ ” are closer than for “ $ㄨ$.” And the basic mouth shape “ h ” has a little

difference from “h’.” For example, the syllable “*h’*ㄜ” is translated into “*h’h*,” but the posture of the former is narrower than the latter. For the Mandarin finals, the elements “ㄣ” and “ㄨ” exchange their classes since we consider that “ㄣ” is pronounced with wider lip corners than “ㄨ,” “ㄜ,” and “ㄝ.” The new classifications for the Mandarin initials and finals are listed in Table 5.5 and Table 5.6, respectively.

Table 5.5 Six basic mouth shapes of Mandarin initials.

Mouth Shape Symbols	Members of the classes
<i>m</i>	ㄇ、ㄨ、ㄇ
<i>f</i>	ㄈ
<i>h’</i>	ㄏ、ㄏ、ㄏ、ㄏ、ㄏ、ㄏ
<i>h</i>	ㄏ、ㄏ、ㄏ
<i>r</i>	ㄖ、ㄖ、ㄖ、ㄖ
<i>z</i>	ㄗ、ㄗ、ㄗ

Table 5.6 A set of combinations with 7 basic mouth shapes of Mandarin finals.

Mouth Shape Symbols	Members of the classes	Combinations of Mouth Shapes	Members of the classes
<i>a</i>	ㄚ	<i>au</i>	ㄨ
<i>e</i>	ㄝ	<i>ou</i>	ㄨ
<i>i</i>	ㄟ	<i>an</i>	ㄨ
<i>o</i>	ㄛ	<i>ah</i>	ㄨ
<i>u</i>	ㄨ、ㄨ	<i>ei</i>	ㄨ
<i>n</i>	ㄣ	<i>ai</i>	ㄨ
<i>h</i>	ㄨ、ㄝ、ㄜ		

Similarly to the method in [1], we eliminate h' and h if there is a symbol (including i , u , e , and o) behind them. However, they are not eliminated if the symbol “ a ” is behind them because the pre-posture is not missed in this case. Additionally, due to the habit of the pronunciation for Mandarin speaking, the mouth shape of “ ㄜ ” will be changed from “ an ” to “ en ” if there is a Mandarin final (including “ $-$ ” and “ ㄩ ”) before it, and the mouth shape of “ ㄛ ” will be changed from “ o ” to “ uo ” if there is a symbol m before it. Finally, the details of the definitions are listed as follows.

1. Basic mouth shape m and f have the same definitions as mentioned above in Section 5.2.1.

2. Basic mouth shape h' :

$$h' = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.25 \times MNS0);$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.3 \times MNS0);$$

$$lf = (8.4_{\text{Netural}.x}, 8.4_{\text{Netural}.y} - 0.1 \times MNS0);$$

$$rt = (8.3_{\text{Netural}.x}, 8.3_{\text{Netural}.y} - 0.1 \times MNS0).$$

3. Basic mouth shape h :

$$h = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.25 \times MNS0);$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.5 \times MNS0);$$

$$lf = (8.4_{\text{Netural}.x} + MW0/12, 8.4_{\text{Netural}.y} + 0.1 \times MNS0);$$

$$rt = (8.3_{\text{Netural}.x} - MW0/12, 8.3_{\text{Netural}.y} + 0.1 \times MNS0).$$

4. Basic mouth shape r :

$$r = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.25 \times MNS0);$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.25 \times MNS0);$$

$$lf = (8.4_{\text{Netural}\cdot x} + 0.15 \times MWO, 8.4_{\text{Netural}\cdot y});$$

$$rt = (8.3_{\text{Netural}\cdot x} - 0.15 \times MWO, 8.3_{\text{Netural}\cdot y}).$$

5. Basic mouth shape z :

$$z = \{up, dn, 8.4_{\text{Netural}}, 8.3_{\text{Netural}}\},$$

$$\text{where } up = (8.9_{\text{Netural}\cdot x}, 8.9_{\text{Netural}\cdot y} - 0.15 \times MNSO);$$

$$dn = (8.2_{\text{Netural}\cdot x}, 8.2_{\text{Netural}\cdot y} + 0.15 \times MNSO);$$

6. Basic mouth shape a :

$$a = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}\cdot x}, 8.9_{\text{Netural}\cdot y} - 0.3 \times MNSO);$$

$$dn = (8.2_{\text{Netural}\cdot x}, 8.2_{\text{Netural}\cdot y} + 0.8 \times MNSO);$$

$$lf = (8.4_{\text{Netural}\cdot x} + 0.8 \times MWO, (up.y + dn.y)/2 - UpperLipH/3);$$

$$rt = (8.3_{\text{Netural}\cdot x} - 0.8 \times MWO, lf.y).$$

7. Basic mouth shape i :

$$i = \{8.9_{\text{Netural}}, dn, lf, rt\},$$

$$\text{where } dn = (8.2_{\text{Netural}\cdot x}, 8.2_{\text{Netural}\cdot y} + MNSO/6);$$

$$lf = (8.4_{\text{Netural}\cdot x} - MWO/12, 8.4_{\text{Netural}\cdot y} - UpperLipH/2);$$

$$rt = (8.3_{\text{Netural}\cdot x} + MWO/12, 8.3_{\text{Netural}\cdot y} - UpperLipH/2).$$

8. Basic mouth shape u :

$$u = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}\cdot x}, 8.9_{\text{Netural}\cdot y} - 0.3 \times MNSO);$$

$$dn = (8.2_{\text{Netural}\cdot x}, 8.2_{\text{Netural}\cdot y} + 0.3 \times MNSO);$$

$$lf = (8.4_{\text{Netural}\cdot x} + 0.2 \times MWO, 8.4_{\text{Netural}\cdot y});$$

$$rt = (8.3_{\text{Netural}\cdot x} - 0.2 \times MWO, 8.3_{\text{Netural}\cdot y}).$$

9. Basic mouth shape e :

$$e = \{8.9_{\text{Netural}}, dn, lf, rt\},$$

$$\text{where } dn = (8.2_{\text{Netural}\cdot x}, 8.2_{\text{Netural}\cdot y} + 0.5 \times MNSO);$$

$$lf = (8.4_{\text{Netural}.x} - MW0/12, 8.4_{\text{Netural}.y} - 0.5 \times \text{UpperLipH});$$

$$rt = (8.3_{\text{Netural}.x} + MW0/12, 8.3_{\text{Netural}.y} - 0.5 \times \text{UpperLipH}).$$

10. Basic mouth shape o :

$$o = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.25 \times MNS0);$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.65 \times MNS0);$$

$$lf = (8.4_{\text{Netural}.x} + 0.2 \times MW0, (up.y + dn.y)/2);$$

$$rt = (8.3_{\text{Netural}.x} - 0.2 \times MW0, lf.y).$$

11. Basic mouth shape n :

$$n = \{up, dn, lf, rt\},$$

$$\text{where } up = (8.9_{\text{Netural}.x}, 8.9_{\text{Netural}.y} - 0.3 \times MNS0);$$

$$dn = (8.2_{\text{Netural}.x}, 8.2_{\text{Netural}.y} + 0.4 \times MNS0);$$

$$lf = (8.4_{\text{Netural}.x}, 8.4_{\text{Netural}.y} - \text{UpperLipH}/3);$$

$$rt = (8.3_{\text{Netural}.x}, 8.3_{\text{Netural}.y} - \text{UpperLipH}/3).$$

The jaw point 2.1 is moving simultaneously with the mouth point 8.2, where 2.1 = $(8.2.x, 8.2.y + \text{JawH})$. Illustrations of the basic mouth shapes of the Mandarin initials and finals defined above are shown in Figure 5.4 and Figure 5.5, respectively.

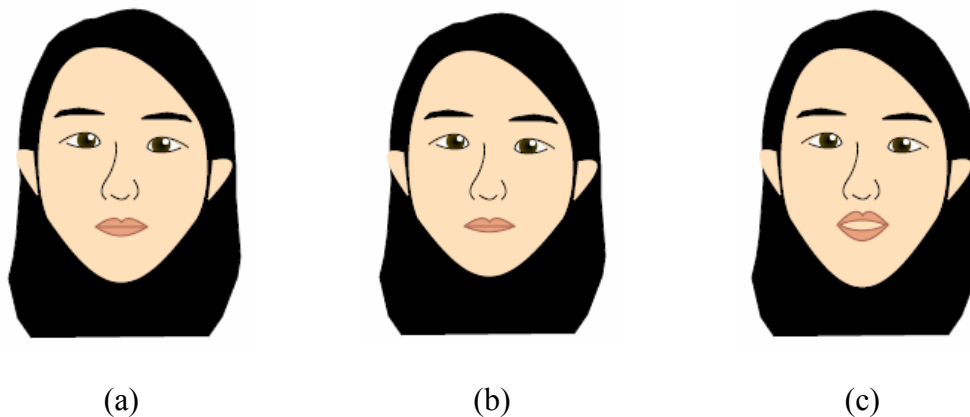


Figure 5.4 An illustration of basic mouth shapes of Mandarin initials. (a) Basic mouth shape m . (b) Basic mouth shape f . (c) Basic mouth shape h' . (d) Basic mouth shape h . (e) Basic mouth shape r . (f) Basic mouth shape z .

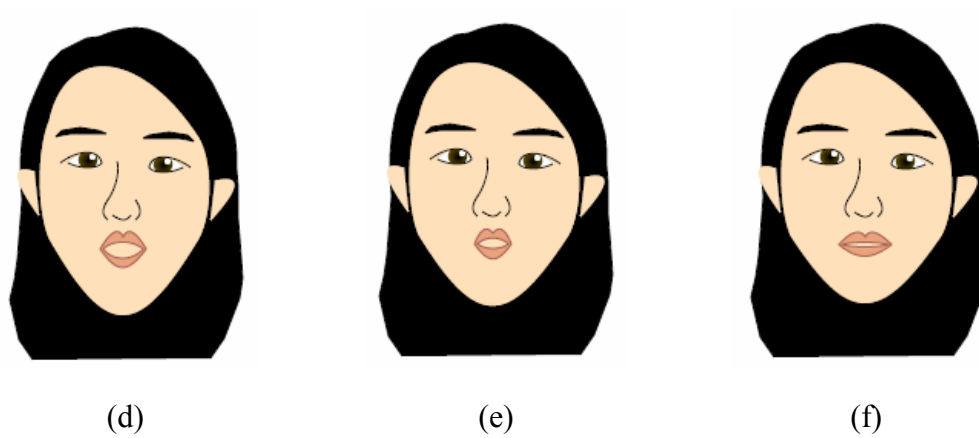


Figure 5.4 An illustration of basic mouth shapes of Mandarin initials. (a) Basic mouth shape *m*. (b) Basic mouth shape *f*. (c) Basic mouth shape *h'*. (d) Basic mouth shape *h*. (e) Basic mouth shape *r*. (f) Basic mouth shape *z*. (continued)

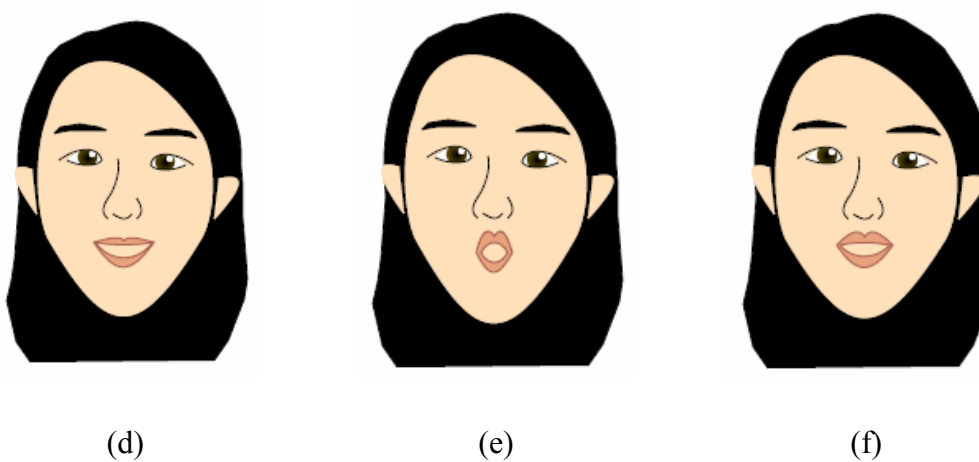
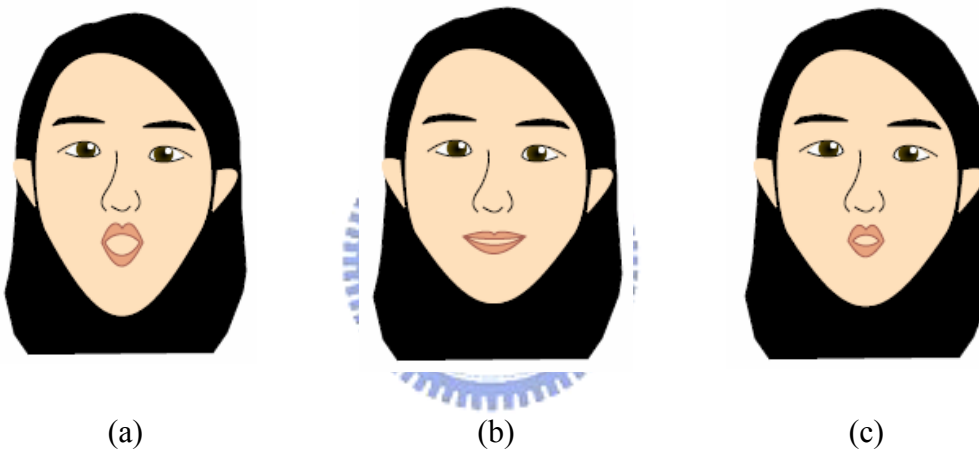


Figure 5.5 An illustration of basic mouth shapes of Mandarin finals. (a) Basic mouth shape *a*. (b) Basic mouth shape *i*. (c) Basic mouth shape *u*. (d) Basic mouth shape *e*. (e) Basic mouth shape *o*. (f) Basic mouth shape *n*.

5.3 Review of Adopted Time Intervals of Sentence Syllables for Mouth Shape Generation

After defining the basic mouth shapes, since a syllable may be combined with four basic mouth shapes at most in this study, the next issue is to decide when to show each mouth shape during the pronunciation of a syllable. Chen and Tsai [1] proposed a method for analysis of time intervals between mouth shapes of syllables. They recorded an audio with several continuous syllables, selected a part of a syllable, and found the time interval between two pronunciations of the basic mouths by careful listening. After repeatedly applying this method for all the Mandarin initials and finals, they got a statistics result for three conditions. The first condition is a syllable with two basic mouth shapes. They assumed that the second mouth shape is given at the time with a proportion 80.95% of the total time of the syllable. An example of “ㄓ” is shown in Figure 5.6. The second condition is a syllable with three basic mouth shapes. They assumed that the second mouth shape is given at the time with a proportion 43.37%, and the third mouth shape is given at the time with a proportion 39.75% of the total time of the syllable. An example of “ㄓㄣ” is shown in Figure 5.7. The last condition is a syllable with four basic mouth shapes. They assumed that the second mouth shape is given at the time with a proportion 24.5%, the third mouth shape is given at the time with a proportion 21.15%, and the fourth mouth shape is given at the time with a proportion of 25.88% of the total time of the syllable. An example of “ㄓㄣㄣ” is shown in Figure 5.8.



Figure 5.6 An illustration of time intervals of a syllable of two basic mouth shapes in [1].

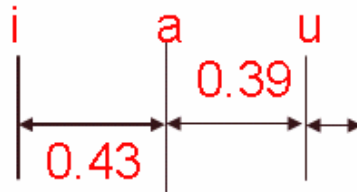


Figure 5.7 An illustration of time intervals of a syllable of three basic mouth shapes in [1].

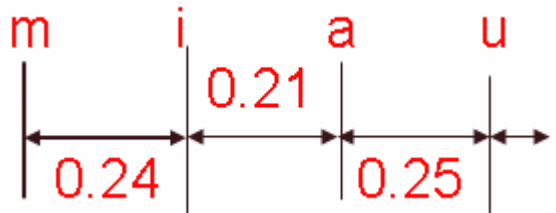


Figure 5.8 An illustration of time intervals of a syllable of four basic mouth shapes in [1].

5.4 Talking Cartoon Face Generation by Synthesizing Lip Movements

In this study, a timeline is used as a basic structure for animation. By obtaining the time information of syllables, which is done in the speech analyzer of the proposed system, arranging the timing of facial expressions, assigning basic mouth shapes for each syllable, setting up the positions of the control points in corresponding key frames in the timeline, applying an interpolation technique to generate the remaining frames among key frames, and synchronizing the frames with a speech file,

the animation of the talking cartoon face can be created. The concept of the use of key frames is shown in Figure 5.9. An overall illustration of the process mentioned above is shown in Figure 5.10.

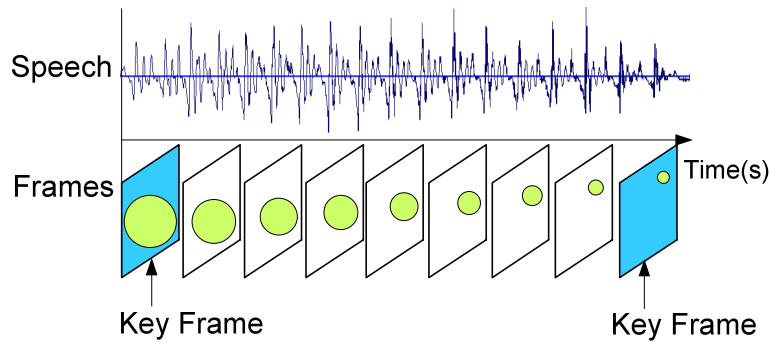


Figure 5.9 A concept of the use of key frames.

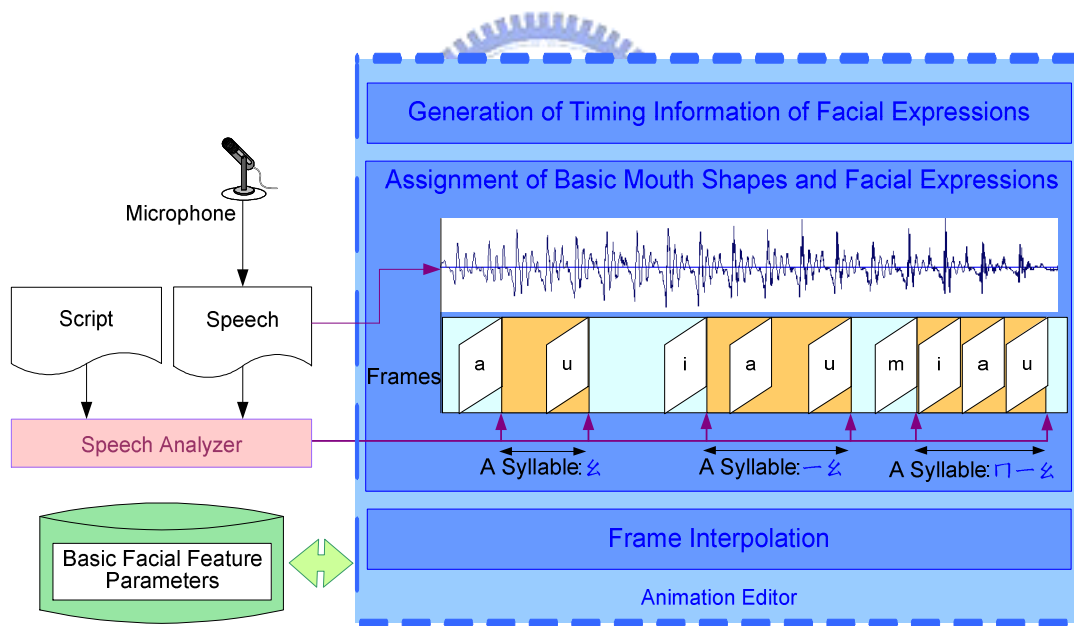


Figure 5.10 An overall illustration of the process of talking cartoon face generation.

5.5 Experimental Results

An experimental result of generating a talking cartoon face synchronized with a

speech file of saying two Mandarin words “願望” is shown in Figure 5.11. Another experimental result of saying two Mandarin words “波濤” is shown in Figure 5.12.

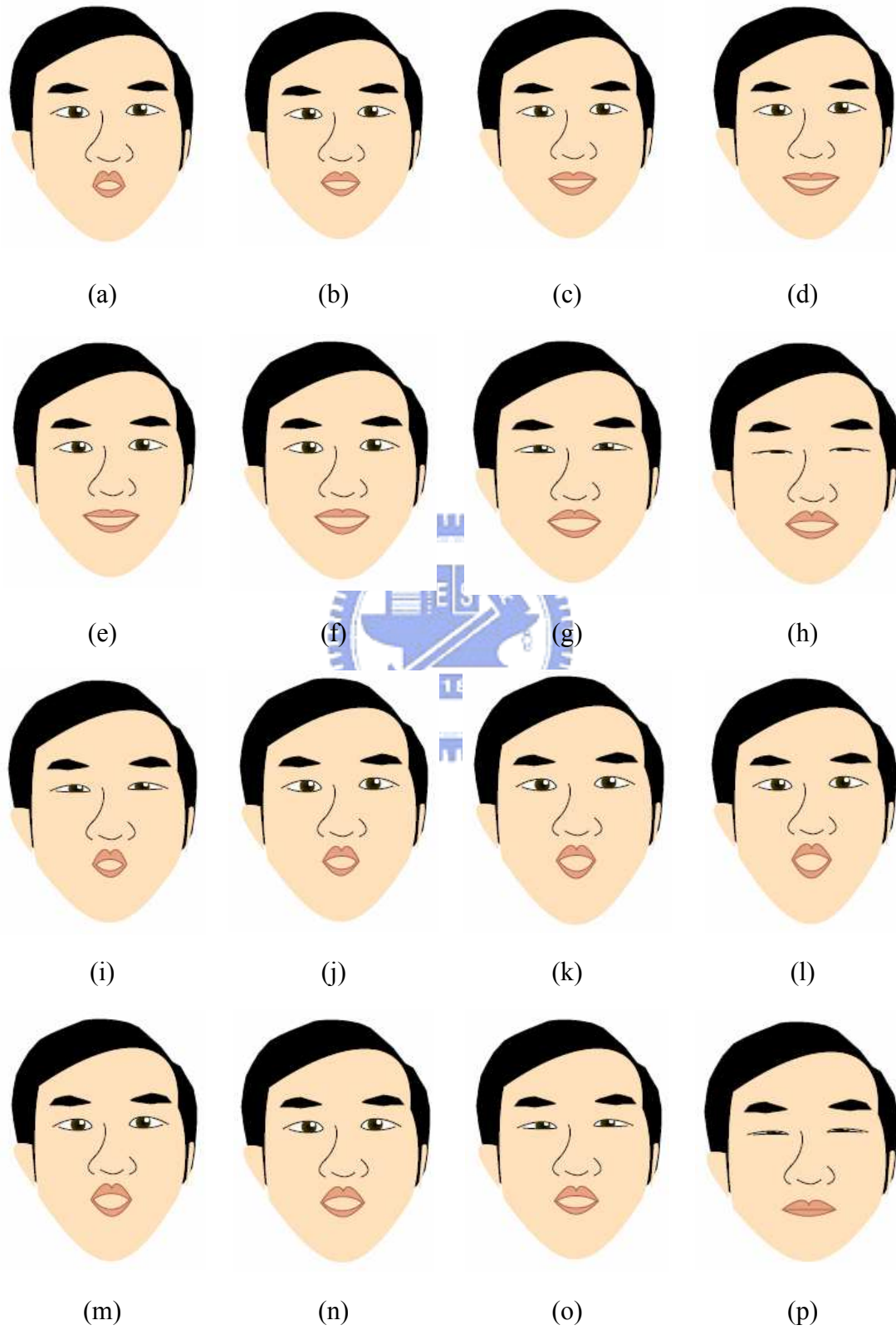


Figure 5.11 An experimental result of the talking cartoon face speaking “願望.”

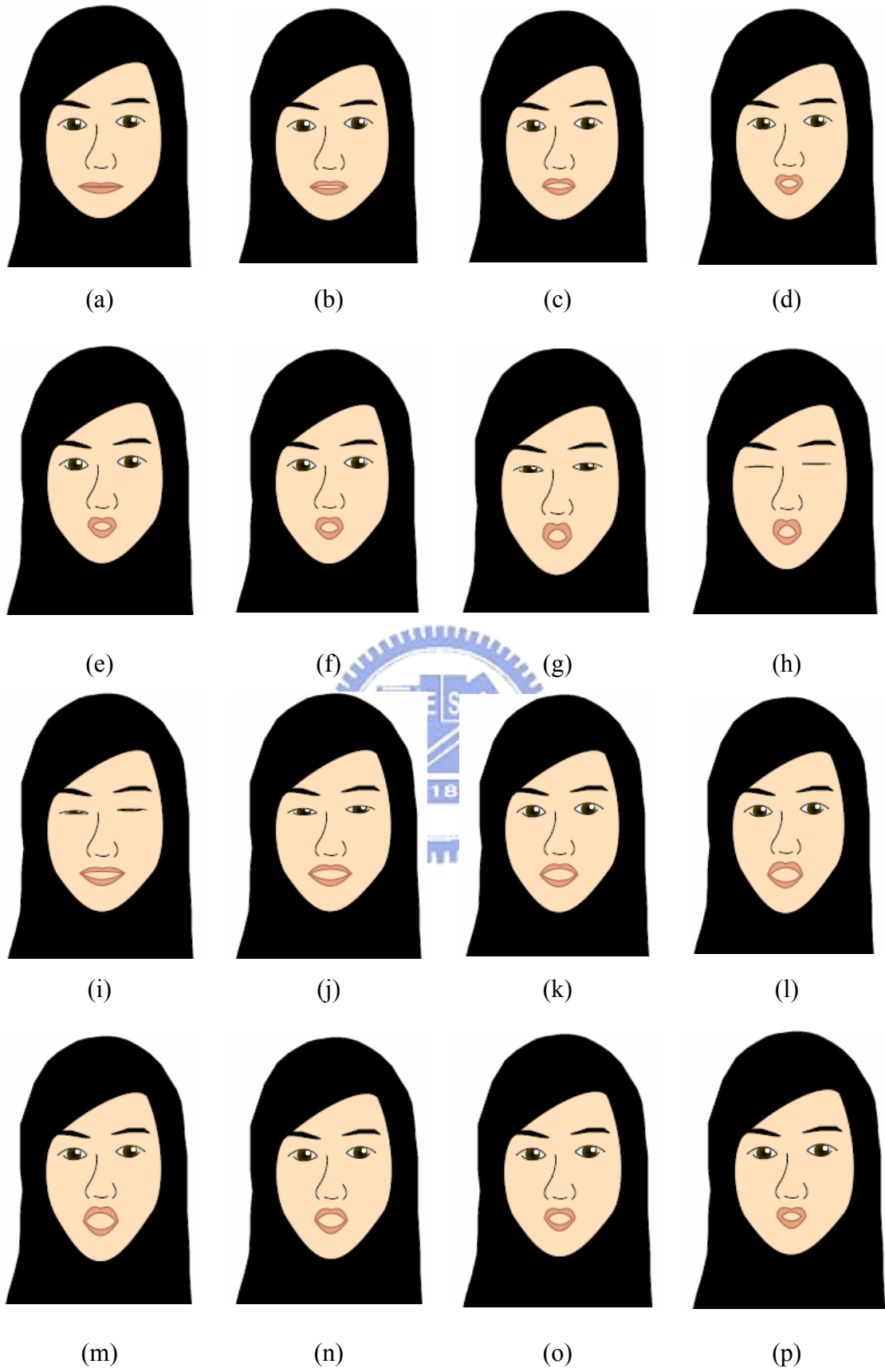


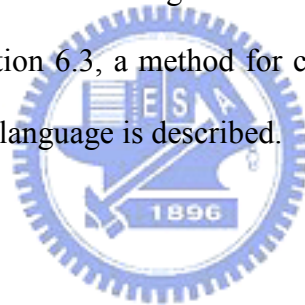
Figure 5.12 An experimental result of the talking cartoon face speaking “波濤.”

Chapter 6

Talking Cartoon Face Generator Using Scalable Vector Graphics

6.1 Introduction

Scalable Vector Graphics (SVG) is an XML markup language which is an open standard created by the World Wide Web Consortium (W3C). In the proposed system, it is used for rendering the generated talking cartoon face. In Section 6.2, an overview of SVG is given. And in Section 6.3, a method for construction of a talking cartoon face generator using the SVG language is described.



6.2 Overview of Scalable Vector Graphics (SVG)

SVG is an editable and vector-based XML language for designing two-dimensional graphics for the Web. Unlike traditional graphic formats, SVG uses a series of text-based XML syntax to describe graphics, so it can be created with less data for easier sharing. SVG allows three types of graphic objects: vector graphic shapes, raster graphics images, and text. Graphical objects can be easily grouped, styled, transformed, and combined into other previously rendered objects. SVG also introduces many powerful features such as scripting, alpha mask, clipping paths,

gradients, filter effect, etc. Since SVG is XML-based, it has the capability for unparalleled dynamic interactivity. It can respond to user actions with highlighting, tool tips, special effects, audio, and animation. It can be combined with other languages, such as HTML, JavaScript, SMIL, and so on. Since SVG is a vector-based language, users can magnify an SVG image up to a large scale without losing detail or clarity. Due to these reasons, SVG is suitable for applications including creation of virtual cartoon faces.

For the proposed system, there are still some other advantages of using SVG. First, the two curve drawing methods applied in the proposed system are supported in SVG. This feature is helpful for us to draw a cartoon face in a simple way. Second, the source codes of SVG files are viewable, editable, and searchable. Hence, SVG images can be easily created without using specific software. Furthermore, SVG is an open standard and widely supported by software developers and large companies such as Adobe. There have already been some tools used for creating, playing, and converting the SVG file to certain other file formats. These features help us easily create and play cartoon face animations by applying the syntax defined in the SVG specification.

An example of SVG showing a rectangle and two lines of styled text is shown in the following:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="400px" height="100px">
  <rect style="fill:#8080CC; stroke:#000000; stroke-width:2"
    width="320" height="80" x="40" y="10" />
  <text x="50" y="35" style="fill:#EEFFCC; stroke:pink; font-size:20">
    Talking Cartoon Face Generator
  </text>
  <text x="50" y="70" style="fill:#800080; stroke:cyan; font-size:20">
```

```
Using Scalable Vector Graphics
</text>
</svg>
```

The corresponding result of the above source code can be viewed directly by a web browser, as illustrated in Figure 6.1.



Figure 6.1 A result of an SVG source code.

6.3 Construction of a Talking Cartoon Face Generator Using SVG

In the proposed system, a talking cartoon face generator is designed to generate a desired talking cartoon face animation using SVG. Two kinds of views are considered for the generation process. In Section 6.3.1, the way from the view in the spatial domain is described. And in Section 6.3.2, the way from the view in the temporal domain is described.

6.3.1 Spatial Domain Process

Although SVG is for two-dimensional graphics, graphical objects still have a rendering priority according to their depth information. Therefore, a concept of layers is involved from the view of the spatial domain, as shown in Figure 6.2.

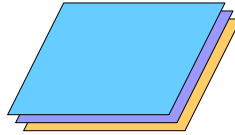


Figure 6.2 A concept of layers in the special domain.

The syntaxes in the SVG specification, including *path* (including cubic Bezier curves), *circle*, *polyline*, and so on, are utilized for drawing the cartoon face. By assigning the locations of feature points as parameters to corresponding syntaxes, a cartoon face can be rendered. In the proposed system, “*polyline*,” which describes a set of connected straight line segments, is used as an implementation of the corner-cutting subdivision drawing technique mentioned previously in Chapter 2 to draw the hair contour and the eyebrows with the hair and eyebrow points as parameters. An example of the syntax *polyline* is shown as follows.

```
<polyline points="10,175  
30,175 30,125 50,125 50,175  
70,175 70,75 90,75 90,175  
110,175 110,25 130,25 130,175  
150,175"  
style="fill:yellow; stroke:blue; stroke-width:10" />
```

Lines and shapes can be easily drawn by assigning demanded points. Even the style of the *polyline* can be specified by assigning the parameters of the style. The corresponding result of the above syntax is shown in Figure 6.3.

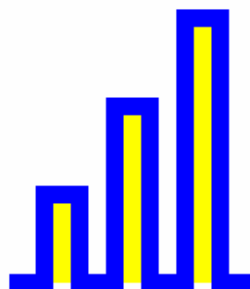


Figure 6.3 An example of the syntax *polyline* of SVG.

The syntax “*circle*” defines a circle based on a center point and a radius. It is used to draw eyeballs. Since during eye blinks, the eyeball will be hidden, an effect of *clipping paths* is utilized to solve the problem. The *clipping path* restricts the region to which paint can be applied. The basic concept is that any part of the drawing outside the region bounded by the clipping path is not drawn. An example of the syntax *circle* is shown as follows, and the corresponding result is shown in Figure 6.4.

```
<circle cx="150" cy="150" r="70"  
style="fill:#F8D2EF; stroke:purple; stroke-width:10" />
```

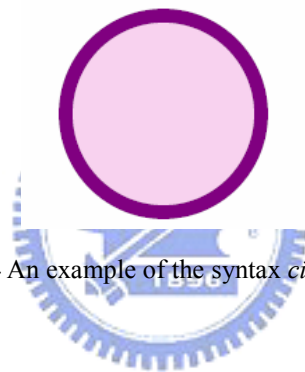


Figure 6.4 An example of the syntax *circle* of SVG.

An illustration of the rendering result of eyes is shown in Figure 6.5, integrating the special effect of *clipping paths* and *gradients*, which consist of continuously smooth color transitions along a vector from one color to another.



Figure 6.5 An illustration of eye drawing.

The remaining facial parts are drawn by applying the cubic Bezier curve in this study. The syntax of the cubic Bezier curve is a part of the syntax *path*. The syntax *path* contains a *d* = “(path data)” attribute, which contains the *moveto*, *line*, *curve*

(including the cubic and quadratic Bezier curves), *arc* and *closepath* instructions. An example of using the syntax *path* to draw the cubic Bezier curve is shown as follows, where the syntax *M* and *C* are representative the instruction *moveto* and *curve*, respectively. The corresponding result is shown in Figure 6.6.

```
<path d="M100,200 C100,100 220,100 250,200"  
      style="fill:none; stroke:green; stroke-width:2" />
```



Figure 6.6 An example of using the syntax *path* of SVG to draw the cubic Bezier curve.

By applying the above mentioned syntaxes and specifying the shape and the style of each component, cartoon faces can be drawn. Involving the concept of layers, four abstract layers are adopted in the proposed system: the background layer, the hair layer, the clothes layer, and the face layer. By rendering each component sequentially according to the layer which it belongs to, cartoon faces can be animated with a more colorful background, as shown in Figure 6.7 and Figure 6.8.



Figure 6.7 An example of adding two layers of the background and the clothes for the cartoon face.

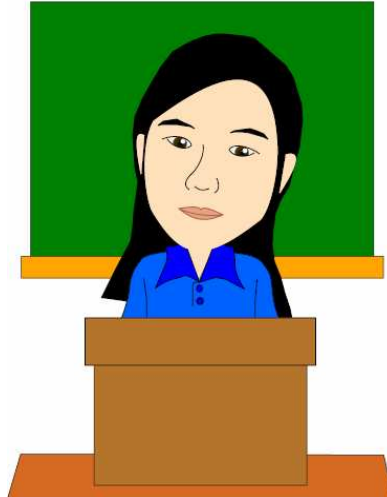


Figure 6.8 Another example of adding two layers of the background and the clothes for the cartoon face.

6.3.2 Temporal Domain Process

To synchronize the speech with the animation, we utilize some syntaxes of SVG for audio playing and frame simulation. For audio playing, since we use the Adobe SVG Viewer as the plug-in of the IE browser to view SVG files, the additional syntax “*a:audio*” defined in Adobe SVG Viewer extension namespaces is applied. An example of using the syntax “*a:audio*” to play the speech file is shown in the following.

```
<a:audio xlink:href="LifeScience.wav" begin="0s" />
```

As we can see, the time when the audio file begins to play can be specified. For frame simulation, a frame sequence can be simulated as an animation in the following way.

```
<g visibility="hidden">
  <set attributeName="visibility" from="hidden" to="visible"
    begin="0.00s" dur="0.03s"/>
  <!--Components of a frame-->
</g>
```


As seen above, the syntax “g” can be used to group a set of components which belong to the same frame, and the properties of visibility and time can be set up by using the syntax “set” to simulate the frame according to the demanded frame rate.

By applying the above mentioned syntaxes and specifying the properties of visibility and time for each group of components, cartoon faces can be animated with the speech synchronized.

6.4 Experimental Results

Combing the techniques proposed in the previous chapters, some experimental results of talking cartoon faces with facial expressions and head movements rendered by SVG are shown in this section. An experimental result of generating a talking cartoon face synchronized with a speech file of saying two Mandarin words “蜿蜒” is shown in Figure 6.9. Another experimental result of saying two Mandarin words “光明” is shown in Figure 6.10.

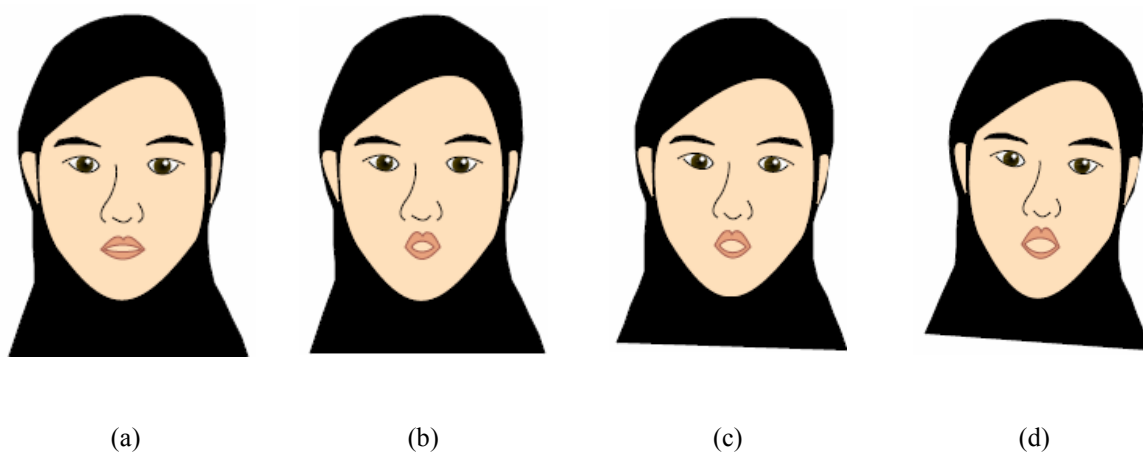
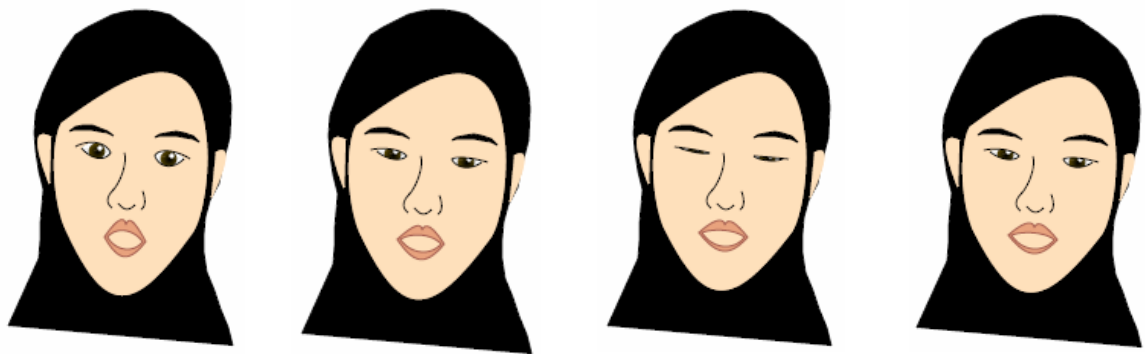


Figure 6.9 An experimental result of the talking cartoon face speaking “蜿蜒.”

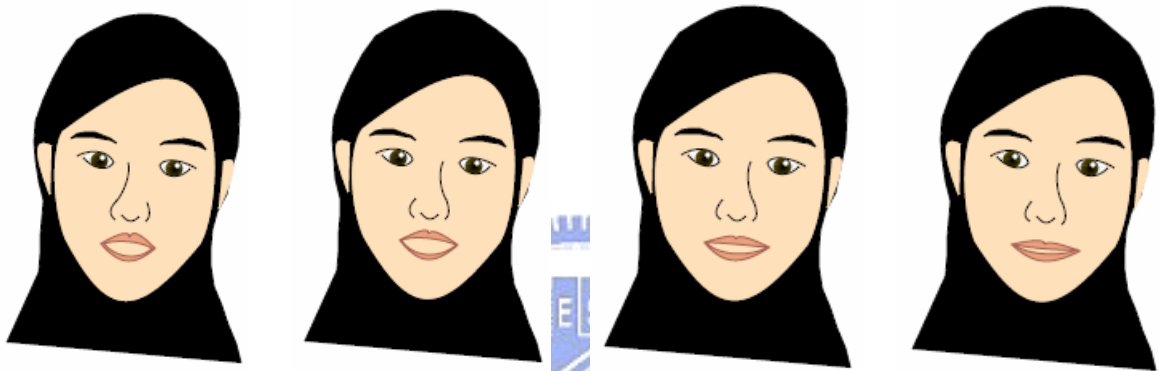


(e)

(f)

(g)

(h)

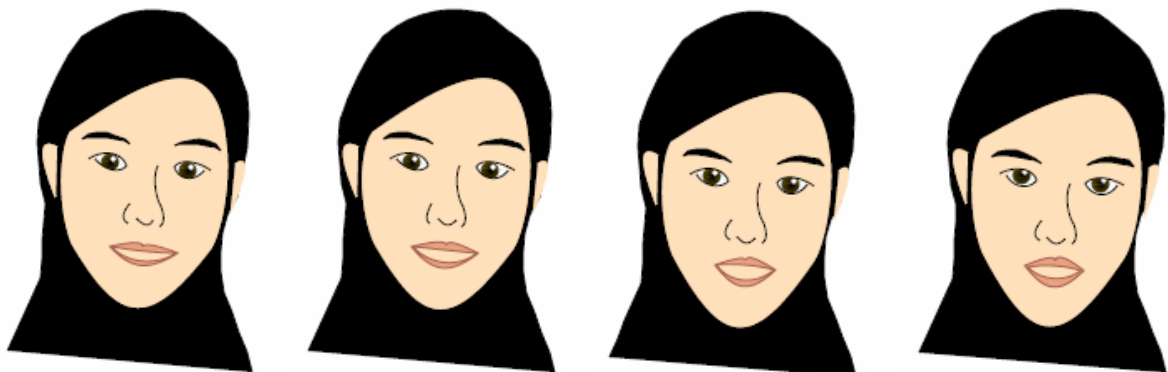


(i)

(j)

(k)

(l)



(m)

(n)

(o)

(p)

Figure 6.9 An experimental result of the talking cartoon face speaking “蜿蜒.” (continued)

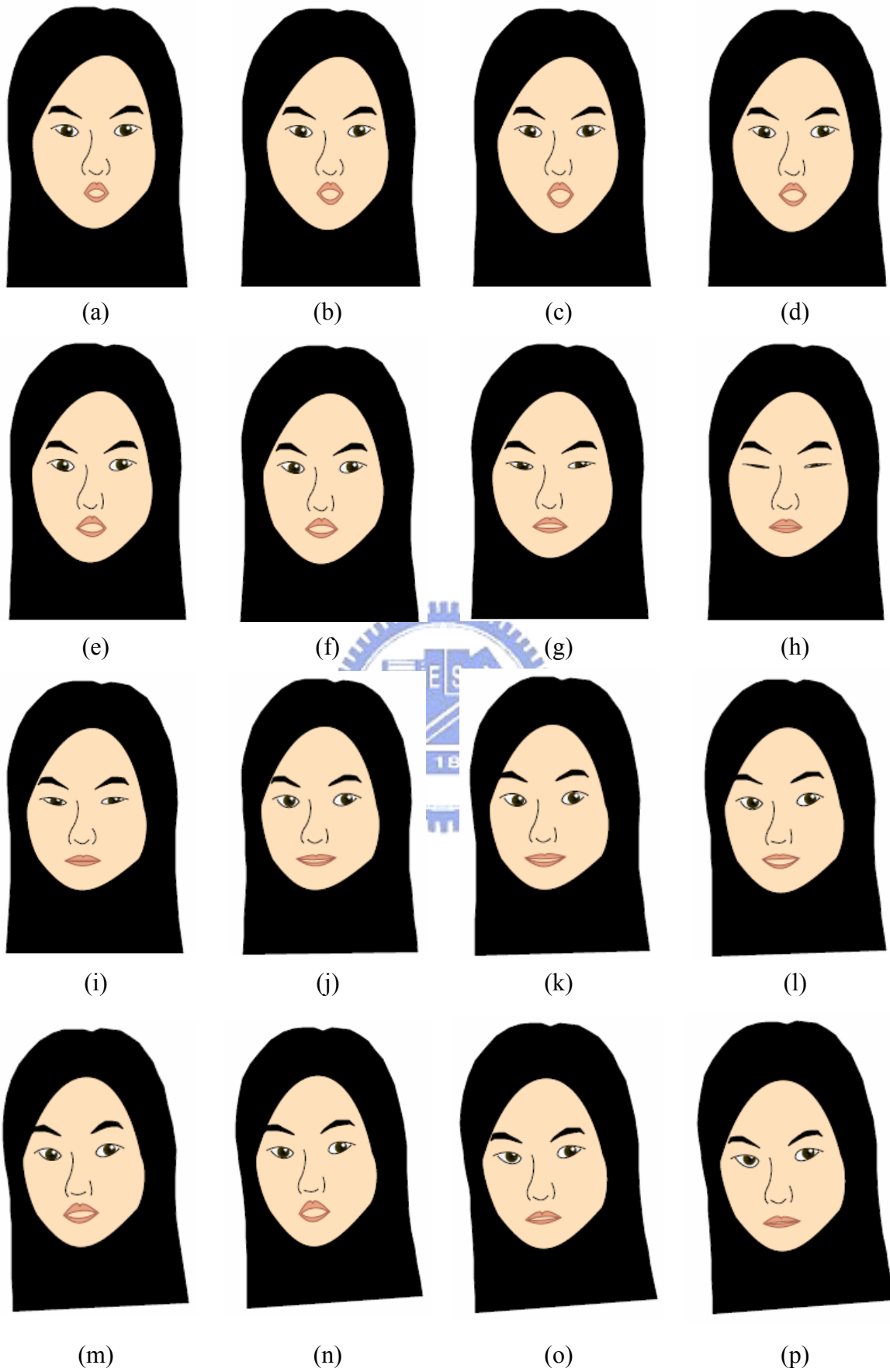


Figure 6.10 An experimental result of the talking cartoon face speaking “光明.”

Chapter 7

Applications of Talking Cartoon Faces

7.1 Introduction to Implemented Applications

Talking cartoon faces can be integrated into many applications. For example, they can be used as virtual guides to guide visitors in libraries and museums. When rambling around while visiting a museum, visitors may listen to the narration of exhibitions provided by the virtual guide, or watch an introduction film with a virtual guide speaking beside it. Talking cartoon faces can also be used as software agents to help users operate the software. They can, with more vivid presentations, give users some tips on how to make use of some specific functions provided in the software.

In this study, some applications of talking cartoon faces are implemented. In Section 6.2, an application to virtual announcers is described. Talking cartoon faces are used to report news, just like news announcers on the TV. In Section 7.3, an application to audio books is presented. Unlike traditional audio books, animations of talking cartoon faces are shown together with the text content and synchronized with the speech. This application can be used in the e-learning area. Thereby, students not only can get impressed by the talking cartoon face and hence keep the content of study materials in mind, but also can have fun.

7.2 Application to Virtual Announcers

7.2.1 Introduction to Virtual Announcers

Virtual announcers are virtual talking faces that can report news. Since real news announcers may take a leave sometimes due to some reasons, virtual announcers can be used in place of them. Even though the absentee did not record news releases in advance, we still can use other one's voice instead to generate a talking cartoon face of his/her appearance by inputting a neutral facial image of him/her to the proposed system. Therefore, virtual announcers can be easily created to temporarily take place of real announcers if the need arises. Moreover, unlike real announcers, virtual announcers can be animated throughout 24 hours without weariness. Although the content of the speech may be the same, the pose and the facial expression of the virtual announcer can be changed momentarily. Therefore, it is considered that using a virtual announcer is a more interesting way to report the same news than replaying the same video clips of the news hourly, as some TV news channels used to doing in their news programs.

7.2.2 Process of Talking Face Creation

The detail of the process of the proposed system is shown in Figure 7.1. First, a 3D cartoon face model is constructed by the cartoon face creator, which is described in Chapter 2. Second, the timing information of a speech file, which is an audio of news here, is extracted by the speech analyzer, as introduced in Chapter 3. Third, basic mouth shapes and facial expressions are assigned in the timeline based on the techniques proposed in Chapter 4 and Chapter 5. Then, a frame interpolation technique is performed by the animation editor. After the above mentioned steps are

done, the frames and the audio are synchronized in an SVG file by the animation and webpage generator. Additionally, by adding two layers, namely, the background layer and the clothes layer, for the cartoon face, an animation of a virtual announcer can be created.

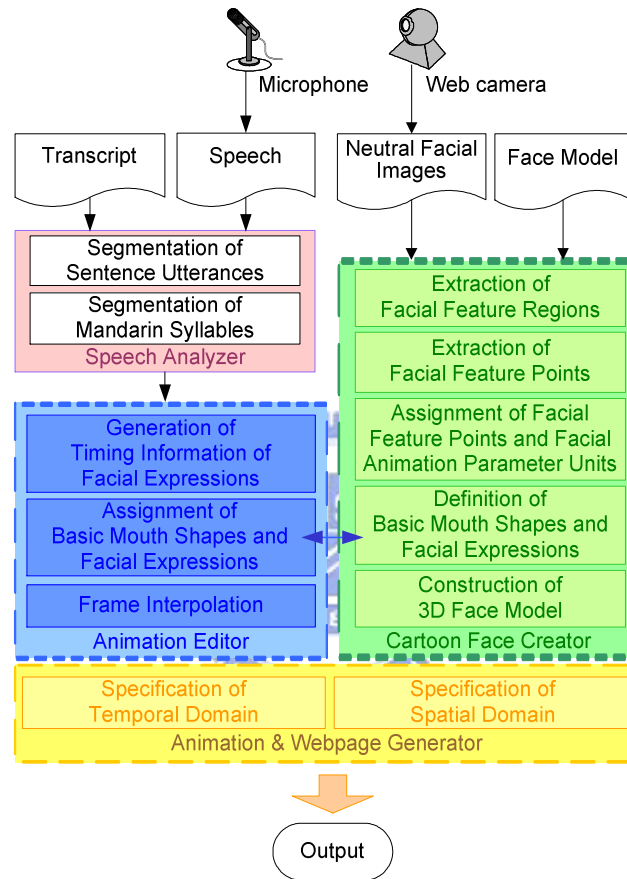


Figure 7.1 An illustration of the process of proposed system.

7.2.3 Experimental Results

Two examples of experimental results are shown in Figure 7.2 and Figure 7.3. Since the location of the rotation origin can be changed to other positions, by applying the transformation between the global and the local coordinate systems, as mentioned in Chapter 2, the location of cartoon faces can be easily changed.

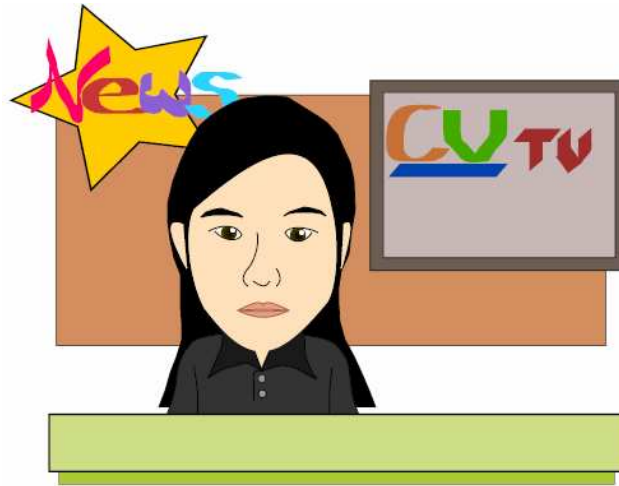


Figure 7.2 An example of a virtual announcer.

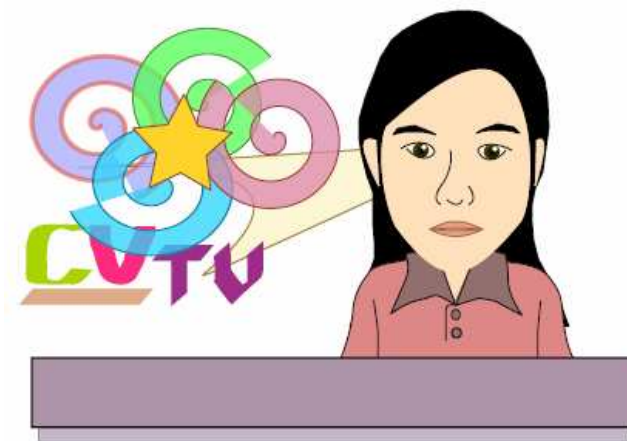


Figure 7.3 Another example of a virtual announcer.

7.3 Applications to Audio Books for E-Learning

7.3.1 Introduction to Audio Books

An audio book is a recording of the contents of a book read aloud. Audio books are usually circulated as CDs, cassette tapes, or other digital formats. They are used to

tell stories, or used for other education purposes. Unlike traditional books, one may listen to an audio book while doing other works. With voice reading, one may learn the pronunciation of words which he/she may not learn correctly by just reading the book. The voice also can make a deeper impression on the reader than the text. However, only a text and an audio version of a book is not enough. Sometimes doing other works may reduce one's attention and hence he/she cannot concentrate on the content of the audio book. In this study, we implement an application to audio books, which combines an animation of talking cartoon faces together with the text content and the audio. By watching the animation, listening to the speech, and reading the text if necessary, students may be attracted by the talking cartoon face and hence get a better learning effect.

7.3.2 Process of Audio Book Generation

The process of audio book generation is similar to the one of virtual announcers. The only difference is that after the generation of the SVG file, the animation must be embedded into an HTML document to be shown together with the text content and some study materials.

There are three ways to embed an SVG file into a web browser. One is to use the `<embed>` tag. The Adobe SVG Viewer, which is adopted in this study as a plug-in of the IE browser, recommends users to use the `<embed>` tag when embedding an SVG file in HTML pages. The syntax is listed as follows.

```
<embed src="animation.svg" width="300" height="100"  
      type="image/svg+xml"  
      pluginspage="http://www.adobe.com/svg/viewer/install/" />
```

The second way is to use the `<object>` tag. The syntax is shown as follows.


```
<object data="animation.svg" width="300" height="100"
type="image/svg+xml"
codebase="http://www.adobe.com/svg/viewer/install/" />
```

The third way is to use the `<iframe>` tag. The syntax is shown as follows.

```
<iframe src="animation.svg" width="300" height="100">
</iframe>
```

7.3.3 Experimental Results

An example of experimental results is shown in Figure 7.4. By inputting an SVG file, the title and the content of an audio book, and some related links to the proposed system, an HTML document was automatically created.



Figure 7.4 An example of a virtual teacher.

Chapter 8

Conclusions and Suggestions for Future Works

8.1 Conclusions

In this study, a system for automatic 2D virtual face generation by 3D model transformation techniques has been implemented. We have presented a way to automatically transform a 2D cartoon face model into a 3D one, and animate it by statistical approximation and lip movement synthesis. The system consists of four major components, including a cartoon face creator, a speech analyzer, an animation editor, and an animation and webpage generator.

The cartoon face creator is designed to include the functions of (1) assigning feature points to a 2D face model according to an input neutral facial image or an input 2D face data set; (2) constructing a 3D local coordinate system and create a transformation between the global and the local coordinate systems by the use of a knowledge-based coordinate system transformation method proposed in this study; and (3) defining basic facial expression parameters for use in facial animation. A face model of 72 facial feature points is used in this study. For the purpose of applying a 3D rotation technique and two curve drawing methods, some additional points are also computed.

Next, the speech analyzer is designed to perform the functions of (1) segmenting a speech file into sentence utterances; and (2) processing each segmented shorter

sentence utterance piece sequentially to extract the duration of each syllable in the speech. A method for segmentation of sentence utterances has also been proposed.

The animation editor is designed to conduct the functions of (1) generating the timing information of facial expressions automatically according to a statistical method proposed in this study; (2) translating syllables into combinations of 12 pre-defined basic mouth shapes; (3) assigning basic mouth shapes and facial expressions in the timeline as key frames; and (4) applying a frame interpolation technique to generate the remaining frames among key frames.

Finally, the animation and webpage generator is designed to perform the functions of (1) rendering and synchronizing the cartoon face with speech by the use of an editable and opened vector-based XML language, namely, Scalable Vector Graphics (SVG); (2) implementing an application to virtual announcers; and (3) embedding the SVG animation into an HTML document to generate an audio book for e-learning. The outlook of the cartoon face can be specified by adding backgrounds and clothes into the animation in this component.

Experimental results shown in the previous chapters have proven the feasibility and applicability of the proposed methods.

8.2 Suggestions for Future Works

Several suggestions for future researches are listed as follows.

- (1) Improvement on facial feature detection --- In order to fit more application environments for creation of face models from neutral facial images, the performance of the facial feature detection must be improved. Besides, for precise construction of 3D face models, assigning the position of the feature

points in the Cartesian z -direction may be combined with facial feature detection techniques for side-view photographs.

- (2) Rendering cartoon faces with more types --- More face types should be supported to render talking cartoon faces with higher qualities and lovelier appearances. Not only human faces, but also some face types for animals and nonhuman objects need be supported.
- (3) Improvement on speech recognition --- In order to generate the talking cartoon face with less input, some speech recognition techniques such as Speech-to-Text (STT) can be integrated into the proposed system. Then a cartoon face can be animated without knowing the transcript of the speech in advance.
- (4) Integration of more facial expressions --- With more facial expressions integrated, generated talking cartoon faces will become more interesting and lifelike.
- (5) Integration of gestures and body actions --- Same as integration of more facial expressions, talking cartoon faces with gestures and body actions are more vivid and amusing.
- (6) Integration of virtual face-painting or hair-designing --- With the integration of virtual face-painting or hair-designing, the proposed system can be used at beauty salons for customers to choose favorite make-up and hair styles that they want to be with.
- (7) Simulating facial expressions and head movements with different statistical models --- Since different TV news announcers have different habits when reporting news, more types of statistical models can be used to present different reporting styles for different TV news announcers.
- (8) Simulating hair movements dynamically according to the gravity --- When rotating the cartoon face, the positions of hair contour control points can be

dynamically computed according to the gravity to make the hair more realistic.



References

- [1] Y. L. Chen and W. H. Tsai, "Automatic Generation of Talking Cartoon Faces from Image Sequences," *Proceedings of 2004 Conference on Computer Vision, Graphics and Image Processing*, Hualien, Taiwan, Republic of China, August 2004.
- [2] Y. L. Chen and W. H. Tsai, "Automatic Real-time Generation of Talking Cartoon Faces from Image Sequences in Complicated Backgrounds and Applications," *Proceedings of 2006 International Computer Symposium (ICS 2006) - International Workshop on Image Processing, Computer Graphics, and Multimedia Technologies*, Taipei, Taiwan, Republic of China, Dec. 4-6, 2006.
- [3] Y. C. Lin, "A Study on Virtual Talking Head Animation by 2D Image Analysis and Voice Synchronization Techniques," *M. S. Thesis*, Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan, Republic of China, June 2002.
- [4] C. J. Lai and W. H. Tsai, "A Study on Automatic Construction of Virtual Talking Faces and Applications," *Proceedings of 2004 Conference on Computer Vision, Graphics and Image Processing*, Hualien, Taiwan, Republic of China, August 2004.
- [5] C. Zhang and F. S. Cohan, "3-D Face Structure Extraction and Recognition From Images Using 3-D Morphing and Distance Mapping," *IEEE Transactions on Image Processing*, Vol. 11, No. 11, pp. 1249-1259, Nov. 2002.
- [6] T. Goto, S. Kshirsagar, and N. Magnenat-Thalmann, "Automatic Face Cloning and Animation Using Real-Time Facial Feature Tracking and Speech Acquisition," *IEEE Signal Processing Magazine*, Vol. 18, No. 3, pp. 17-25, May

2001.

- [7] H. Chen, Y. Q. Xu, H. Y. Shum, S. C. Zhu, and N. N. Zheng, "Example-based Facial Sketch Generation with Non-parametric Sampling," *Proceedings of 8th IEEE International Conference on Computer Vision*, Vol. 2, pp. 433-438, July 2001.
- [8] H. Chen, Z. Liu, C. Rose, Y. Xu, H. Y. Shum, and D. Salesin, "Example-Based Composite Sketching of Human Portraits," *Proceedings of the 3rd International Symposium on Non-photorealistic Animation and Rendering*, Annecy, France, pp. 95-153, 2004.
- [9] H. Chen, N. N. Zheng, L. Liang, Y. Li, Y. Q. Xu, and H. Y. Shum, "PicToon: A Personalized Image-based Cartoon System," *Proceedings of the 10th ACM international conference on Multimedia*, Juan-les-Pins, France, pp. 171-178, 2002.
- [10] Y. Li, F. Yu, Y. Q. Xu, E. Chang, and H. Y. Shum, "Speech Driven Cartoon Animation with Emotions," *Proceedings of the 9th ACM international conference on Multimedia*, Ottawa, Canada, pp. 365-371, 2001.
- [11] D. Burford and E. Blake, "Real-time Facial Animation for Avatars in Collaborative Virtual Environments," *Proceedings of South African Telecommunications Networks and Applications Conference '99*, pp. 178-183, 1999.
- [12] M. Zhang, L. Ma, X. Zeng, and Y. Wang, "Imaged-Based 3D Face Modeling," *International Conference on Computer Graphics, Imaging and Visualization*, pp. 165-168, July 26-29 2004.
- [13] J. Ostermann, "Animation of Synthetic Faces in MPEG-4," *Proceedings of the Computer Animation*, pp. 49-55, June 08-10, 1998.
- [14] P. Ekman and W. V. Friesen, *Facial Action Coding System: Investigator's Guide*,

Consulting Psychologists Press Inc., Palo Alto, California, 1978.

- [15] T. M. Yeh, *Drills and Exercises in Mandarin Pronunciation*, National Taiwan Normal University, ROC, May 1982.

