ANATOMY BASED ANIMATION OF FACIAL EXPRESSIONS

TUĞBA ERKOÇ
B.S., Computer Engineering, IŞIK UNIVERSITY, 2010

Submitted to the Graduate School of Science and Engineering
in partial fulfillment of the requirements for the degree of
Master of Science
in
Computer Engineering

IŞIK UNIVERSITY
2013

IŞIK UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

ANATOMY BASED ANIMATION OF FACIAL EXPRESSIONS

TUĞBA ERKOÇ

APPROVED BY:

Assist. Prof. M. Taner Eskil (Thesis Supervisor)	Işık University	_____
Assoc. Prof. Olcay Taner Yıldız	Işık University	_____
Assist. Prof. Devrim Akça	Işık University	_____

APPROVAL DATE: ..../..../....

ANATOMY BASED ANIMATION OF FACIAL EXPRESSIONS

Abstract

This study presents a new physics based facial expression animation system based on anatomic data. Proposed system consists a face model and a new facial expression animation generation algorithm.

The proposed system is a Mass-Spring-Damper (MSD) system. It consists single layer face model HIGEM and a set of facial muscles which are placed anatomically correct places. Non-linear viscoelastic characteristics of the human skin is approximated with non-linear springs. The set of muscles triggers facial expressions.

HIGEM does not include a skull to support the facial mesh, so it tends to collapse under muscle forces. A new algorithm proposed to prevent this. This algorithm uses back and forward projections for re-defining the muscle forces. Another problem of MSD systems is individual element collapse under large forces. This is addressed with Edge Repulsion (ER) approach.

Dynamics of the face is modelled with a quasi-implicit Ordinary Differential Equation (ODE). The regions of the face affected by the facial muscles were computed and marked offline in previous works to speed up the animations. It requires offline re-calculation whenever the activation level of the muscles or facial mesh topology changes. Thus, a new generic stack based approach which works at runtime is proposed.

The proposed system is tested with eight facial expressions; happy, sad, angry, feared, surprised, disgusted, disgusted anger and happy surprise. Elapsed times for the facial expressions vary with number of contracting muscles, their region of influence, and the time value for each timestep.

Keywords: Facial expression animation,physics based approach, generic wireframe, mass-spring-damper system

YÜZ İFADELERİNİN ANATOMİYE DAYALI ANİMASYONU

Özet

Bu araştırma anatomik bilgiye dayalı yeni bir fizik bazlı yüz ifadesi animasyon sistemi sunar. Önerilen sistem bir yüz modeli ve yeni bir yüz ifadesi yaratma algoritmasından oluşur.

Önerilen sistem bir Kütle-Yay-Amortisör (KYA) sistemidir. Sistem tek katmanlı yüz modeli HIGEM ve bir grup anatomik olarak doğru yerlere yerleştirilmiş yüz kaslarından oluşmaktadır. İnsan derisinin vizkoelastik karakteristikleri doğrusal olmayan yaylar ile yakınsanmıştır. Kas grubu yüz ifadelerini tetikler.

HIGEM yüz ızgarasını destekleyen bir kafatası içermez. Bu yüzden kas kuvvetleri altında yıkılabilir. Bunu önlemek için yeni bir algoritma önerilmiştir. Bu algoritma geri ve ileri iz düşümleri kullanarak kas kuvvetlerini yeniden tanımlar. KYA sistemlerinin bir diğer problemi büyük kuvvetler altında bireysel parçaların yıkılmasıdır. Buna hitaben Kenar Geri Tepme yaklaşımı kullanılmıştır.

Yarı-örtülü Bayağı Diferansiyel Denklem (BDD) ile yüz dinamikleri modellenmiştir. Yüz kasları tarafından etkilenen yüz alanları daha önceki çalışmalarda ön hesaplama ile bulunup işaretlenmiştir. Kasların aktivasyon seviyesi ya da yüzün topolojisi değiştiğinde bu ön hesaplamalar yeniden yapılmalıdır. Bu yüzden, yeni bir çalışma zamanında çalışan jenerik yığın bazlı yaklaşım önerilmiştir.

Önerilen sistem sekiz yüz ifadesi ile test edilmiştir; mutlu, üzgün, sinirli, korkmuş, şaşırmış, iğrenmiş, iğrenmiş sinirli ve mutlu şaşırmış. Yüz ifadeleri için geçen süreler kasılan kasların etki alanlarına, sayısına ve her bir zaman adımı için ayarlanmış süreye göre değişir.

Keywords: Yüz ifadesi animasyonu, fizik bazlı yaklaşım, jenerik telkafes, kütle-yay-amortisör sistemi

Acknowledgements

This research project would not have been possible without the support of many people.

First of all, I would like to express my gratitude to my supervisor M. Taner Eskil for his invaluable guidance, assistance and support throughout this study.

I would like to acknowledge the researchers at PiLab for providing the face model HIGEM.

Finally, I would like to give special thanks to my family for their understanding and endless love, through the duration of my studies.

Without their support and encouragement, this study would have not been possible.

This research is part of project "Expression Recognition based on Facial Anatomy", grant number 109E061, supported by The Support Programme for Scientific and Technological Research Projects of The Scientific and Technological Research Council of Turkey (TÜBİTAK).

To my family. . .

Table of Contents

List of Tables

List of Figures

List of Abbreviations

2D	**2** **D**imension
3D	**3** **D**imension
AU	**A**ction **U**nits
DFFD	**D**irichlet **F**ree-**F**orm **D**eformations
ER	**E**dge **R**epulsion
FACS	**F**acial **A**ction **C**oding **S**ystem
FAP	**F**acial **A**ction **P**arameters
FAT	**F**acial **A**nimation **T**ables
FDP	**F**ace **D**efinition **P**arameters
FFD	**F**ree **F**orm **D**eformations
FFP	**F**acial **F**eature **P**oints
HCI	**H**uman **C**omputer **I**nteraction
MPEG	**M**oving **P**icture **E**xpert Group
MSD	**M**ass - **S**pring - **D**amper
ODE	**M**ass - **S**pring - **D**amper
PCA	**P**rincipal **C**omponent **A**nalysis
RBF	**R**adial **B**asis **F**unction
SU	**S**hape **U**nits

Chapter 1

Introduction

1.1 Facial Expressions

Facial expressions are involuntary actions that express our feelings to the outside world. Happiness, sadness, anger and disgust are only a couple of examples that we frequently display and observe. These facial expressions are triggered by our emotional states but produced by the muscles that are beneath the skin tissue of the face. All facial expressions are the results of the contraction of these muscles.

We depend on communication to survive since it is the only way to interact with our surroundings to provide physical and psychological needs for ourselves. To interact with the others, we developed different ways of communication during long history of evolution. As a result of this process, we have verbal and non-verbal means to interact with others.

Verbal communication is through the use of words. We use speech to convey our thoughts by using the words face to face or over the phone communications. Intonations are the only way to add emphasis to the things that we want to disclose in verbal communication. However, voice alone cannot show how we really feel about the situation that we are talking about. Mehrabian [1] points out an irrefutable example; it is possible to verbally express hatred and pass on exactly the opposite feeling. He goes one step further and claims that the feeling

or attitude conveyed by a speaker is 55% facial, 38% vocal and only 7% verbal [2]. Non-verbal communication means come into play when we want to convey our most sincere feelings.

One of the non-verbal communication methods is the body gestures. Hunched shoulders for dejection or crossing arms on chest for anxiety or defensiveness are both body gestures that we employ to emphasize the meaning of the words that we use to communicate. The second non-verbal communication method is the usage of facial expressions. Facial expressions are generally more honest than either words or body gestures in human communications. Words might not always reflect the real feelings we have for the topic of communication. Also, words do not always reflect the truth. Furthermore, humans can self-train to control the body gestures so that they could deceive others during communication. On the other hand, facial expressions are not easy to hide. They are triggered by emotions. Thus, they are often involuntary actions that are hard to self-train to hide deception. This is why humans always seek the face of the person they are communicating with. This way they know that whether they are welcome to continue communicating with the other person by the ability to recognize the emotions. Because of this attribute of facial expressions, they are the most important part of the non-verbal communications between humans.

1.1.1 Facial Expressions in Computer Applications

In the age of the computers, smartphones, tablets and animated movies one of the most popular fields of research is the communication of humans and computers. Human Computer Interaction (HCI) is the field which deals with the ways to provide an interface -whether it is on computer or smartphone screen- that is effortlessly perceived by humans to operate the medium with ease. To provide this kind of an interface, the nature of human communication is needed to be taken into consideration. Since the biggest part of human communications is

through non-verbal communications, implementing an interface which uses this attribute is convenient.

Addition of a graphical human face that is familiar to communicate with as an interface to different kind of gadgets might improve the ease of use. However, people would not feel comfortable to see a graphical face that is emotionless. Thus, realistic looking animation of these graphical faces needs to be implemented. Deployment of facial expressions could provide the realistic looking animations. Interactive kiosks could use this kind of interface with the implementation of an attendant that helps people wither with buying their movie tickets from ticket kiosks or getting directions from way-finding kiosks.

Since 2007, smartphone usage has been rapidly increasing around the world. Actually, smartphones are small computers with their running operating systems that people carry around every day. Thus, similar HCI principles are applicable to smartphones as in computers. There are quite a few applications developed for these phones ranging from music players to games and social networking applications. Most of them benefit from the simple design principles of HCI.

One special smart phone application carries out the job of a person. The type of this kind of application is personal assistant. There are already a couple of them available in the market. However, they all only use verbal communication technique as of today. One of them, Speaktoit Assistant [3], attempts to add a familiar look by adding 2-dimensional graphical human picture but it does not employ neither facial expressions nor body gestures. An example screen captured from Speaktoit Assistant can be observed in figure 1.1 Even without having a non-verbal communication means this kind of application is popular. Adding this realistic interface could result getting more attention from the smartphones users. Thus, personal assistant applications would benefit highly from adding an interface which includes facial expressions to make them more human like.

Games often include human characters that people use as their avatars. With the rise of the online role playing games, the look of these avatars has become

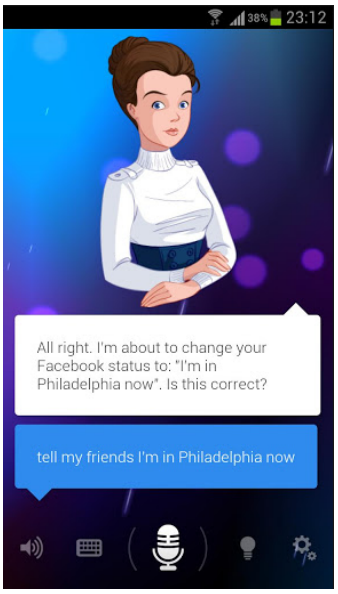

Figure 1.1: Speaktoit Assistant.

more important since it is the representation of the person who plays the game to the other players. This avatar is the only way to communicate with the others in game. Most of the players demands more realistic looking characters instead of cartoon like graphics. The more realistic the character looks like the more attractive the game is. Since game developers are aware of the demand, they provide lots of character customizations which focused mainly on facial features to the players as seen in figure 1.2 . Adding these facial features is not enough to create realistic look if your character cannot emote basic facial expressions like displaying happiness or sadness.

Figure 1.2: Characters from games AKB: Reloaded, Skyrim and Final Fantasy XIV: Realm Reborn respectively.

Beginning with the millennium, we witnessed the rise of the animated movies.

Whether it is partly or fully animated, they were generally dealing with the events of daily life of people. Thus, creation of realistic animations is required to please the viewer. To maintain realistic look, it is essential for animated characters to exhibit facial expressions and overall body language as real people would do. This fact is recognized by the animation teams and different techniques are adopted to create authentic facial expressions. The production process of authentic looking animations often involves motion capture where an actor wears special clothes which can capture the movements. The captured data is used for creation of realistic animations including facial expressions and body movements. Avatar and Beowulf are just two examples that display beautifully animated facial expressions.

Figure 1.3: Avatar character Neytiri smiling.

Robotics is a research area that is working on development of the machines that can think and act like humans. In the recent years, the development has resulted into creation of human like robots which resembles humans with their arms, legs, head and a torso that connects them. To appeal people, fake eyes and mouth are also added. Even though adding those details, humanoid robots still look like aliens. What they are missing is personality which can be remedied by deployment of an animated face that is able to perform facial expressions.

Figure 1.4: Geminoid HI - 2.

1.2 Methods for Generating Facial Expressions

There are several techniques to create facial expressions. These techniques can be categorized into two major categories, geometric and image manipulations. Face models are deformed by usage of geometric calculations to animate the face in geometric manipulation category. Examples of geometric manipulation techniques are key-framing, interpolations, parametrizations and facial muscle based models. Image manipulations are applied onto images of the face to be animated. Image manipulation techniques are image morphing, texture manipulations, image blending and vascular expressions.

1.2.1 Geometry Based Modeling Tecniques

1.2.1.1 Interpolation Techniques

Easiest way to animate an object is using key-frames. The technique where key frames are used is called *key-framing*. In this technique, first and second key frames define the states of the object in starting and final positions of the object. With the help of interpolation, the intermediate positions of the object can be calculated and the respective frames can be created. This process is called *tweening*. Since the animation is created via the help of interpolations, same

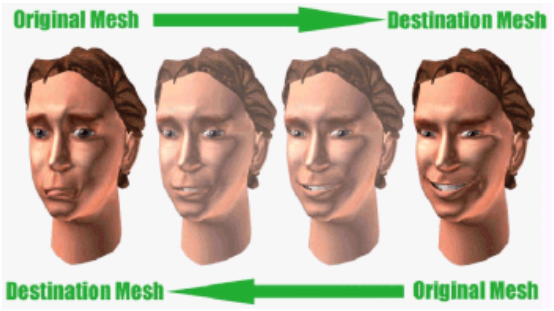

Figure 1.5: Key-framing example.

primitive facial expressions is obtained by using this technique as *interpolations*. An example of key-framing can be seen in figure 1.5 .

Interpolations are the functions that are defining the smooth movement between two different key-frames. This technique can be used for creating primitive facial expressions rapidly. Primitive means that only one part of the face can be animated with this technique. However, multiple parts of the face need to be animated to give a realistic animation. Thus, most of the facial expressions cannot be animated via usage of interpolation.

1.2.1.2 Parameterization Techniques

To overcome the primitive facial expressions problem, parametrization technique can be chosen. Ideally, this technique allows creation of specific facial expressions via usage of independent parameters on any face [4]. However, when more than one parameter is affecting a vertex, the effect of these parameters cannot be combined and this leads to unrealistic facial expression generation. Thus, this technique is only useful when the parameters are affecting different parts of the face. Since, parameters should not collide; each face that is animated needs to have different parameters. One generic parametrization is not possible to deform any face. Instead of dealing with parametrization and its obstacles, image morphing or muscle based simulation techniques can be chosen to animate faces. Following is an example for parameterization technique.

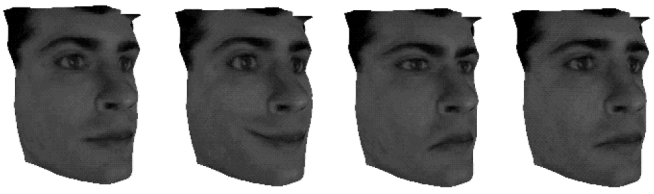

Figure 1.6: Parameterization example from Hoch et al.'s work [31].

1.2.1.3 Image Morphing

Image morphing technique resembles key-framing technique by having a starting image and target image. Starting image is current look of the face and the target image is the facial expression that we want to achieve. By the help of cross dissolving and warping, a metamorphosis between the two images occurs. First, corresponding points on the face are chosen. These points are the common points between the two images. Then, these corresponding points are used to warp the images on to a cylinder or a plane. After warping process is completed, cross dissolving is applied. The starting image slowly fades out while the target image slowly appears. To generate realistic looking animation, the starting image, target image and correspondence points are needed to be chosen carefully [5]. Also, user intervention is needed for the color balance of the areas which affected by warping. Both images needed to have the same point of view; else choosing the correspondence points becomes difficult.

Figure 1.7: Morphing between a woman and a tiger.

1.2.2 Physics Based Modeling Techniques

There are also physics based muscle modelling techniques to animate faces. Physic based muscle modelling techniques define the facial skin, facial muscles and the bones with mathematical equations. Facial Action Coding System (FACS) of Ekman [6] is generally the basis for this technique. FACS defines the movement of the facial muscles, tongue and mandible via usage of 44 different action units (AU). Each facial expression is a group of these action units and Ekman defines six basic facial expressions with action units. As an example, sad expression is a combination of AU1, AU2, AU4, AU15 and AU23. There are three different approaches in physics based muscle modelling. Spring mesh model, vector representations and layered spring meshes.

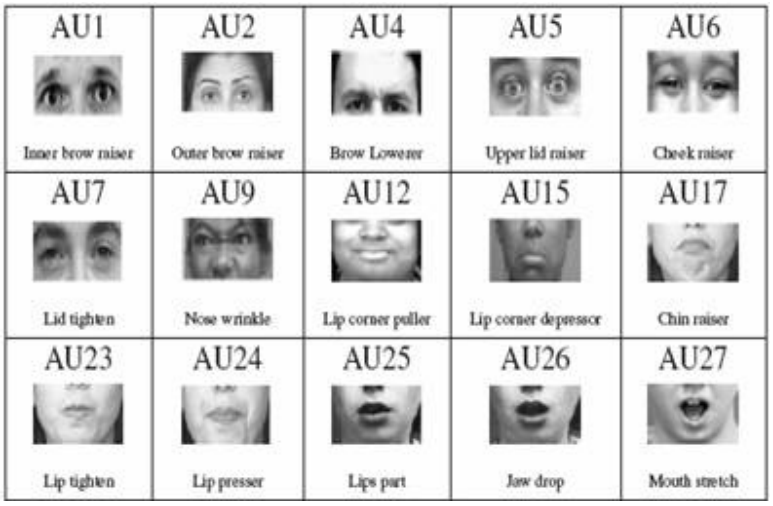

Figure 1.8: A small set of Action Units (AUs) presented in FACs.

Spring mesh model requires an elastic face mesh. Facial expressions are attained by applying muscle forces to muscle arcs on this elastic mesh and deforming the mesh with the exerted muscle force. In this area Platt's work is the pioneer. Platt [7]'s model suggest a face model where 38 regional muscle blocks connected by a network of springs. This model generates the facial expressions by exerting muscle forces on the muscle blocks in accordance to action units of FACS.

Vector representations are the product of the work of Waters [8]. This model is based on vectorized muscles which affect predefined regions of the face. The

affected regions of the face by vector muscles are called muscle's area of influence. The influence area is conic shaped due to the cosine functions and falloff parameters. Each muscle has an attachment and an insertion points. There are also sphincter muscles that enable the deformation of mouth and are modelled as parametric ellipse. This muscle deforms around the centroid of the ellipse. Waters uses the linear and sphincter muscles to simulate the six basic facial expressions defined by FACS. However, the downside of this model is that muscles cannot be placed anatomically correct places. User interference is required to find the correct positions by trial and error. This approach is used by Pixar in an animated short film in 1988 to simulate the facial expressions of a baby via usage of 47 Waters' muscles. The baby can be observed in figure 1.9 .

Figure 1.9: Screen captured from short film Tin Toy.

Layered spring mesh is the product of Waters and Terzopoulos' work [9]. They offer a three layered face model which defines facial skin, the fat layer underneath the skin and the muscles attached to the skull respectively. Every node on each layer is connected via springs to each layer. This connection helps to dissipate the muscle forces to the three layers which results in realistic looking facial expressions. However, volumetric deformation on the three dimensional lattice requires huge amount of calculations. Lee et al.[10] propose another three layered model where the model consists of a biological tissue that can be non-linearly deformed, muscle layer and skull. This model is defined as a triangle prism which is divided into different surfaces on the layers. These layers are named as epidermal, fascia and skull surfaces. Epidermal and fascia surfaces simulate the elastic structure

of the face while the springs that are providing the muscle force effect connect the fascia and skull. This model can provide realistic animations like the previous one. However, the down side is the requirement of vast calculations and modifications for each specific face.

Even though the realistic animations generated by the muscle based techniques, simulation calculations, the requirement to adjust the parameters to fit to the specific faces and the creation of a face model are difficult processes. Pseudo muscles overcome the down sides of the muscle based models by deforming the mesh without the need of the underlying complicated anatomic layers of the face. Deformation is only performed on the facial mesh by using free form deformation or splines.

In free form deformation, there is a volumetric object which resides inside a three dimensional cubical lattice. This lattice has control points that are manipulated to deform the object. When weights are assigned to these control points and these weights are manipulated, the object is deformed without moving the control points. This approach is called rational free form deformation. Kalra et al.[11] use this method to simulate the effect of the regional influence of the muscles. According to this method, region of influence of the muscle is determined and a parallelepiped control volume is inserted here. To deform the face, this control volume is stretched, compressed or expanded. This is achieved by changing the weights and the position of the control points interactively.

1.2.3 Anatomy Based and Rapid Simulations of Facial Expressions

In this research, an anatomical approach to facial expression simulation is presented. The facial model consists of a single layered mesh where nodes are connected through a network of springs which simulates the elastic property of the skin. The underlying fat tissue or the skull is not modelled but handled algorithmically for generating rapid simulations. The proposed facial animation system ensures that the facial muscles deform the facial mesh without the need

of the underlying structures as in pseudo muscles. However, the calculations for muscle forces are performed in a manner as if Water's vector muscles are used. The muscles have insertion and attachment points and an area of influence which is controlled by angular and length parameters. The muscles are placed to the anatomically correct places on the facial mesh by using the artistic work of Goldfinger [12]. Facial expressions are generated by activating a muscle or a group of muscles and deforming the mesh according to the influence areas of these muscles.

Our anatomy based approach provides realistic look since the anatomic data of the face and the underlying muscle placements are used. There is no need for computationally complex trial and error procedures while searching for a good position to place the muscles since the anatomical data provides the correct placements. When the insertion and attachment points of the muscles are placed at anatomically correct places, the influence area of the muscles are automatically defined. We do not need to predefine the influence area as in other methods. Since the model is a single layer mesh, our approach is faster than the methods that employ three layered models and volumetric models.

The usage of anatomical data for the creation of anatomic models enable us to mimic the facial expressions by analyzing the facial expressions of humans and applying the results to the anatomic model. Other methods require a skull to generate restrictive forces to prevent skin mesh modes to move to positions which would result in model collapse in three dimensional space. In this research, muscle forces are first calculated in two dimensional space and then projected back to three dimensional space to enable the simulation of a model sans skull. The muscle forces are applied in a way that the three dimensional facial mesh deforms as if there is an underlying skull via the method introduced in this research. Also, a stack based muscle calculation is employed to perform calculations only on the nodes which reside in the activated muscles' influence area. This enables to animate the facial expressions efficiently.

Chapter 2

Literature Review

Animation of facial expressions is a challenging subject which requires construction of a face model and a technique to deform this model. This matter is tackled by many researchers with different approaches on creation and deformation of face models.

2.1 Face Models

To demonstrate facial expressions, one needs to have a model of the face. The challenge here is that human face has a complex structure. This complex structure can be modelled by approximating the surface of the human face via polygons or by use of parametric surfaces. If polygons are used to approximate the surface of human face, number of polygons used in the face model, in other words resolution of the model becomes important. Using a very large number of polygons in the construction of the face model provides more details and realism.

However, the animation's performance depends on the number of polygons in the face model since polygons are manipulated to provide the facial expressions. The time spent on calculations gets longer as the number of polygons increases. Thus, the desired detail level of the face model has to be balanced with the desired duration for the facial expression generation. There are approaches that employ varying resolutions to generate faster animations and retain the realism. The

smooth regions of the face are modelled with smaller number of polygons whereas the resolution is kept high around the areas of more detail, such as mouth and nose.

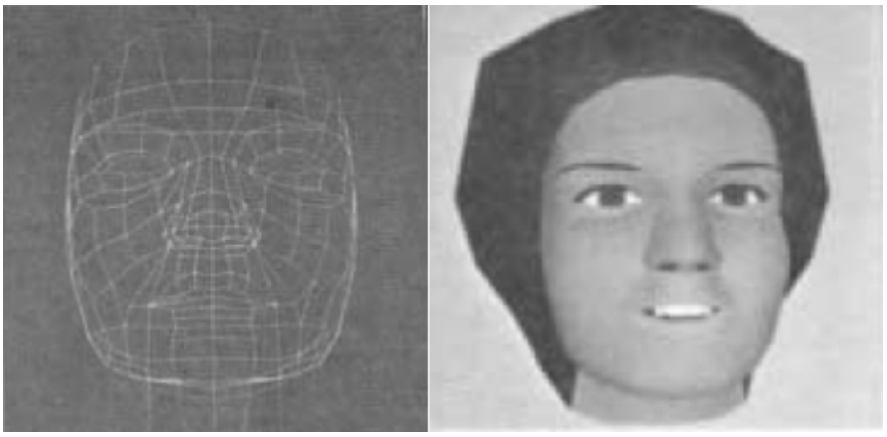

Figure 2.1: Parke's face model. On left: Mesh of polygons used to represent the face, on right: Face model with texture.

Parke [13] created a realistic face model by approximating the face surface with a polygonal mesh which contains 400 vertices and 250 polygons which can be seen in figure 2.1. In this work, Parke only creates one half of the face model since the human face is nearly symmetric.

CANDIDE face models are examples of parameterized facial masks which provide low complexity animations with its simple structure. The original CANDIDE model was created by Rydfalk [14] for model-based coding of human faces. Rydfalk's facial model contains 75 vertices and 100 triangles.

An improved version of original CANDIDE model which is called CANDIDE-1 was used more frequently in the literature. This version is composed of 79 vertices and 108 surfaces. CANDIDE-1 also adds 11 Action Units (AUs) to the original CANDIDE model.

CANDIDE-1 was further improved by Welsh [15] with the addition of more vertices to entirely cover the frontal head with hair, teeth and shoulders. This version, called CANDIDE-2, includes 160 vertices, 238 triangles and 6 AUs.

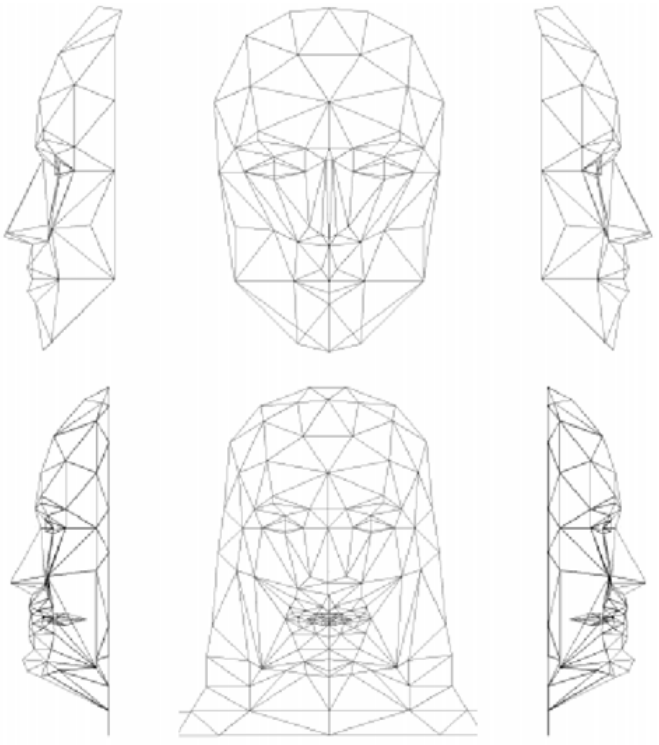

Figure 2.2: Top: CANDIDE-1 with 79 vertices and 108 surfaces.Bottom: CANDIDE-2 with 160 vertices and 238 surfaces.

The most recent CANDIDE model is CANDIDE-3 [16] which has 113 vertices and 168 surfaces. This version improves the rough modelling around the eyes and mouth with introduction of more vertices in these regions. A number of Facial Feature Points (FFPs) which are standardized to be used in facial animation by MPEG-4 [17] were missing in the CANDIDE model and added in CANDIDE-3 model.

Figure 2.3: CANDIDE-3 with 113 verticesand 168 surfaces.

With CANDIDE-3 model, AUs that were defined in CANDIDE-1 model are extended to include new vertices which would help creation of more complex facial animations. Also, in CANDIDE-3, facial animation can be performed both by AUs and Facial Animation Parameters (FAPs) [17]. Another addition in this version is 12 Shape Units (SUs) which identifies the static shape of the face. SUs identify the shape of a face, therefore they are specific to an individual subject and are constant during animation. With the addition of SUs, CANDIDE model can be reshaped to have a wider variation of facial meshes. A recent work [18] suggests an improved CANDIDE-3 model via adding blink detection which was not available in the CANDIDE-3 model in their 3D face reconstruction algorithm.

Figure 2.4: Greta.

Greta [19] is a face model which complies with the MPEG-4 specifications. This face model offers a great level of detail via usage of large number of polygons. Greta contains 15000 polygons which are distributed more densely around eyes, mouth and nose areas for better communication of the expressed emotions. Also, forehead is specifically modelled to generate horizontal wrinkles when the eyebrows are raised.

Another 3D morphable face model was proposed in [20].It is created by using Principal Component Analysis (PCA) over texture and shape data gathered from a facial image database. This image database was created by scanning the 101 Korean faces by use of a 3dMD face scanning system.

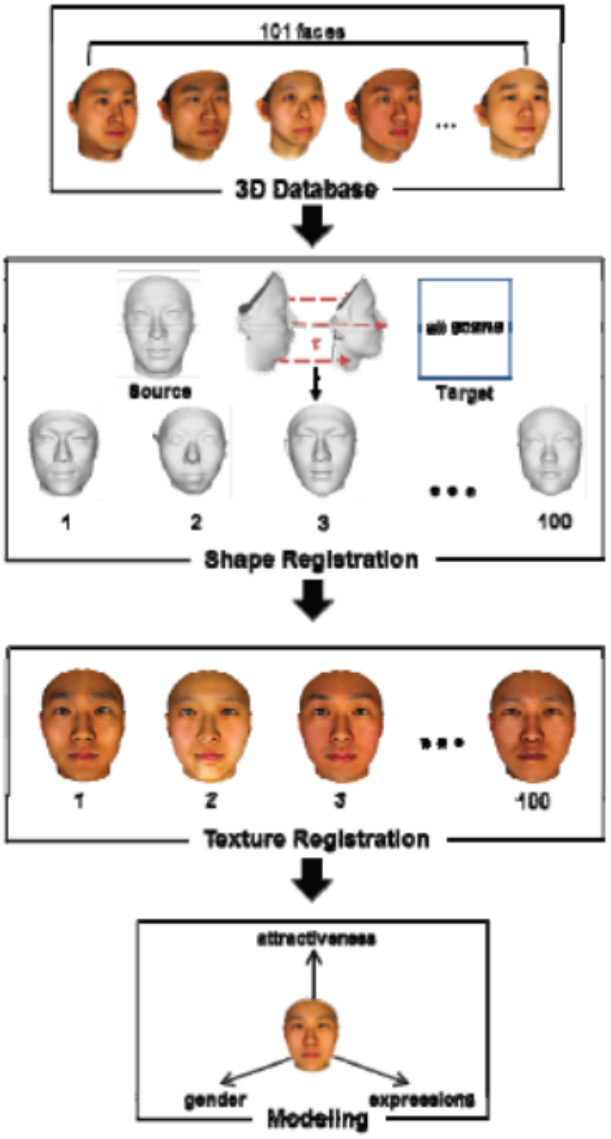

Figure 2.5: Overview of building the 3D morphable face model in [20].

In this research, we are introducing a new face model that conforms to the anatomical structure of the human face. This new face model is more suitable for precise modelling of human face than CANDIDE or Greta models since it is a generic and anatomically accurate wireframe model which is customizable to any human subject.

2.2 Waters Muscle Model

Waters'[8] muscle model defines muscles that are controlled by a small number of parameters. This is a generic model that is proposed for creation of primary facial expressions. Two muscle types are provided in this model, linear and sphincter muscles. Linear muscles simulate the pull and the sphincter muscles simulate the squeeze. Each muscle has an attachment point that connects the muscle with the facial bones and an insertion point that connects muscle with the skin of the face. Attachment points are immobile while the insertion points are mobile.

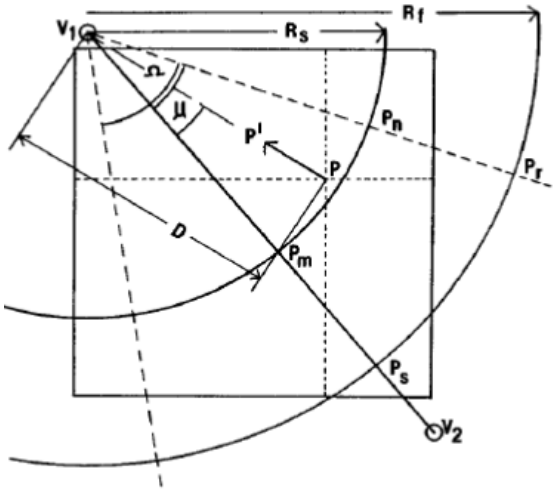

Figure 2.6: Waters'[8] linear muscle model.

Each muscle has a region of influence. When a muscle contracts, muscle force is only exerted on the vertices which lie in the region of influence. However, the exerted muscle force magnitude varies in the region of influence. The magnitude of the muscle force is zero at the attachment point while the maximum value of muscle force is observed at the insertion point. Thus, the exerted muscle force on the vertices in the region of influence increases from attachment point to insertion point. Same falloff is observed in the angular displacement too. Muscle forces tend to get smaller in the magnitude while the angular displacement increases. The direction of the muscle forces are always from the vertices which are under the influence of the contracting muscle to attachment point.

2.2.1 Linear Muscles

In figure 2.6, linear muscle and its influence area is depicted where V_1 is the attachment point, P_m is the insertion point, R_s is falloff radius start and R_f is the falloff radius finish, D is the distance of the point $P(x, y)$ from V_1, points P_n and P_r are used to denote the subregion of region of influence, Ω is zone of influence and μ is the angle between V_1V_2 and V_1P. Any point $P(x, y)$ in the region of influence, which is inside zone $V_1P_rP_s$, is subject to muscle force and shall be displaced towards attachment point V_1. The displaced point $P(x, y)$ becomes $P(x', y')$ where

$$\begin{aligned} x' &\propto f(K.A.R.x) \\ y' &\propto f(K.A.R.y) \end{aligned} \tag{2.1}$$

The parameters K, A and R stand for muscle spring constant, maximum zone of influence and distance from V_1 to point P respectively.

Angular displacement A is defined as:

$$A = \cos((\frac{\mu}{\pi}).(\frac{\pi}{2})) \tag{2.2}$$

where μ is the angle between V_1V_2 and V_1P.

Radial displacement R is defined as:

$$R = \cos((\frac{1-D}{R_s})(\frac{\pi}{2})) \tag{2.3}$$

for P inside $V_1P_nP_m$ and

$$R = \cos(\frac{D - R_s}{R_f - R_s}\frac{\pi}{2}) \tag{2.4}$$

for P inside $P_n P_r P_s P_m$.

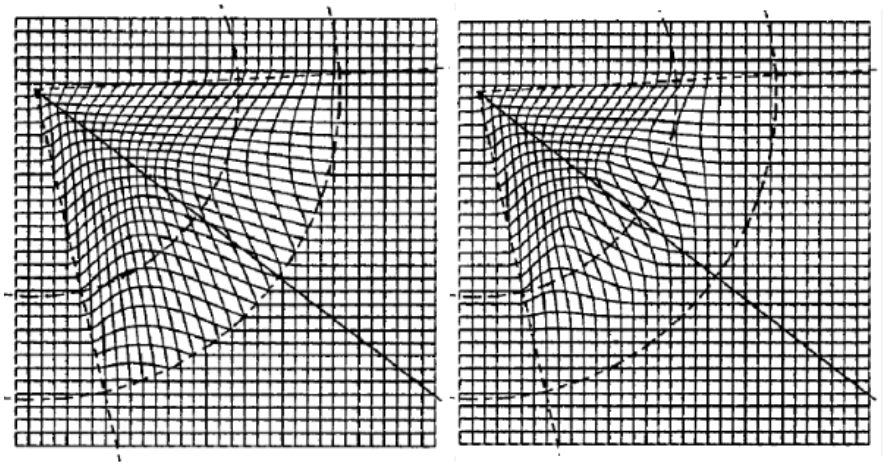

Figure 2.7: A three dimensional muscle vector lying in the x y plane. (A)Zone of influence $\Omega = 35.0$, falloff start $R_s = 7.0$, falloff finish $R_f = 14.0$, muscle spring constant $k = 0.75$, elasticity 1.0 (B) Same muscle with elasticity rose to a power 10.0.

2.2.2 Sphincter Muscles

Waters defines sphincter muscles as muscles that contract around a single point like a string bag. This squeeze can be described as occurring uniformly around the contracting point. Thus, angular displacement is removed from equation (2.1) and the muscle force generated by sphincter muscles becomes

$$\begin{aligned} x' &\propto f(K.R.x) \\ y' &\propto f(K.R.y) \end{aligned} \tag{2.5}$$

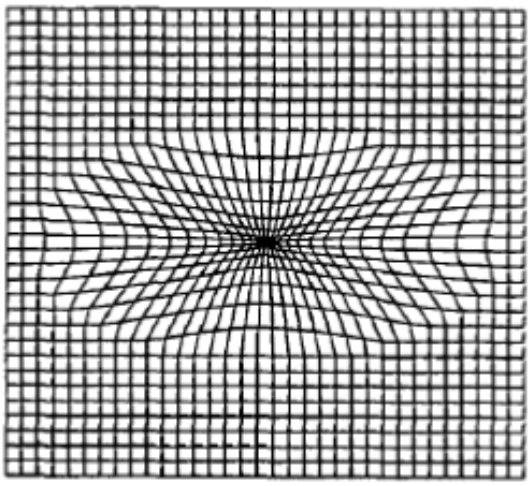

Figure 2.8: Waters' sphincter muscle contracts

However, since the sphincter muscles do not affect the vertices in a regular fashion when they contract, elliptical shapes are chosen to represents the sphincter muscles. This is achieved by the addition of vertical and horizontal axes.

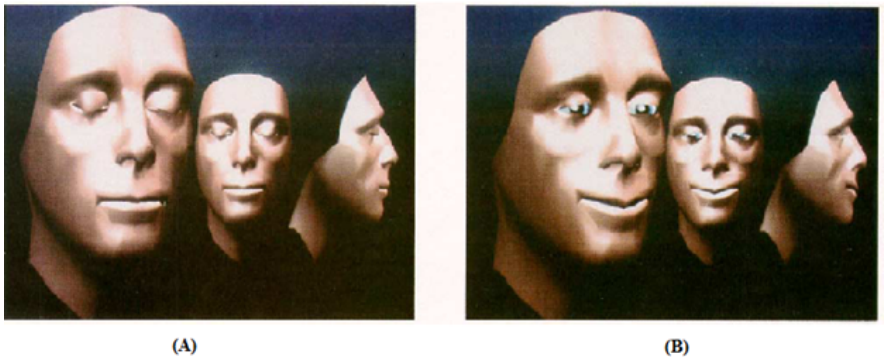

Figure 2.9: Results of [8] (A) Neutral face with muscles relaxed, (B) Happiness

2.3 Model Animation Techniques

When a face model is constructed, a suitable deformation technique has to be chosen depending on the speed and quality requirements of the animation. The techniques proposed in the literature can be grouped into two major categories, geometry based approaches and physics based approaches [21].

2.3.1 Geometry Based Approaches

Geometry based approaches try to generate facial expressions by utilization of geometric functions. Proposed techniques include key frame interpolation, morphing, parameterizations, pseudo-muscle usage and MPEG-4.

2.3.1.1 Key Frame Interpolations

Key frame interpolation is the most straightforward technique to generate facial expressions. This technique requires two images. One image depicts the initial or starting state of the animation and the other one depicts the desired final state of the animation. Intermediate frames are generated by the help of interpolation functions.

Parke [13] proposes cosine interpolation to generate intermediate frames. Facial motion is not linear and its acceleration or deceleration is modeled more realistically with a cosine curve. Parke performed these actions on the vertices of polygons.

Typical key framing in the recent years is performed on images. Pighin et al. [22] generate facial expressions via linear interpolation of key frames with user-controlled cubic Bézier curves to achieve photorealistic 3D expression morphing between the given images. Mendi et al. [23] propose key frame interpolation to animate phonemes. Zhou et al. [24] perform Radial Basis Function (RBF) interpolations to deform the facial model.

2.3.1.2 Morphing

Morphing is performed on two images or models by generation of a metamorphosis between the two target images. Warping and simultaneous cross dissolving are performed over the correspondence points carefully selected by the user for generating realistic facial expressions. Beier and Neely [25] perform 2D morphing

Figure 2.10: Emotion key frames rendered on the mobile emulator [23] Anger, disgust, joy, fear, surprise and sadness.

with manually selected correspondence points. Pighin et al. [26] use a combination of 2D morphing and 3D transformations to generate facial expressions that are viewpoint independent. Noh and Neumann [27] propose a new approach on producing facial expression via cloning expressions from an existing model to a new model by using RBF morphing. Zamith et al. [28] propose an approach similar to [25] with real-time feature based parallel morphing on GPU.

The primary disadvantage of morphing techniques is the requirement of the user interference for correspondence point mapping. However, a recent research proposes a process for automatic correspondence point mapping with the help of critical point filters [29].

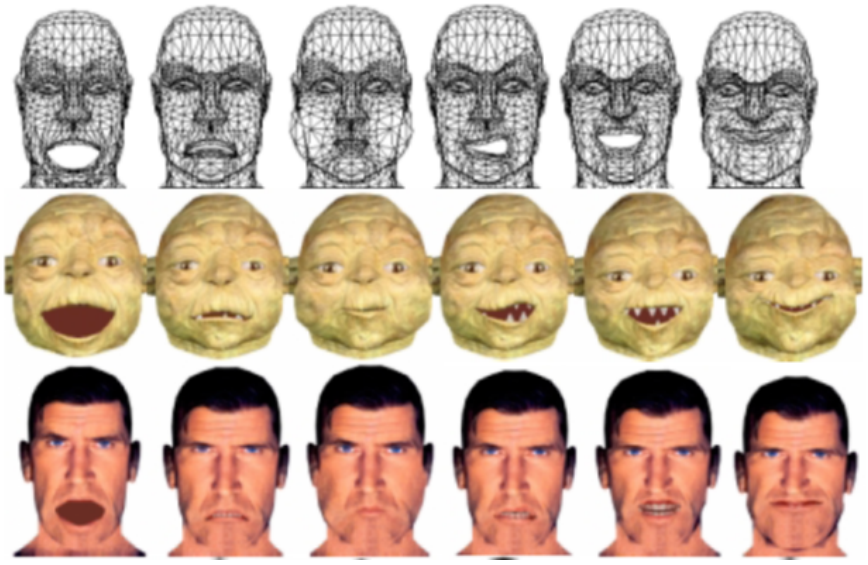

Figure 2.11: Facial expression cloning [27] on different models.

2.3.1.3 Parametrizations

Parametrizations allow control of the face model's shape and generated facial expressions [30]. More control over generated facial expressions is granted to the user compared to interpolations at a low cost with a couple of parameters [4]. However, each parameter affects disjoint set of vertices and conflicting parameters cannot be blended to generate realistic deformations on the face model.

This has been addressed by Waters [8] by a new muscle model which defines a region on the skin to be affected by muscle contraction. Hoch et al [31] uses a parametrization technique to animate faces using AUs. A recent research [32] proposes generation of facial expressions on a DNURBs surface, generated by physical scan data, by moving the control parameters with a real-time performance.

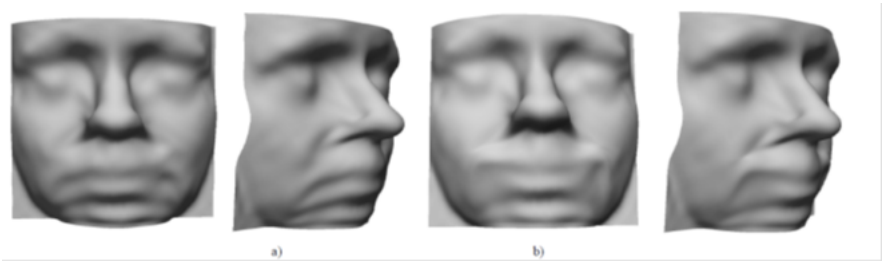

Figure 2.12: Results of [32]. a) Original scan, b) Simulation of smiling.

2.3.1.4 Pseudo-muscles

Pseudo-muscle approach is proposed to resolve the unrealistic facial expression generation by other geometric techniques by deforming the face model in a muscle like fashion without modelling the physical layers of the face [4]. One of the techniques used for pseudo muscle approach is Free Form Deformations (FFD) which is performed by manipulating the control points situated in a cubic lattice to deform volumetric objects [33]. Kalra et al. [11] provide an extension to FFD by including radial basis functions to deformation formula. This allows addition of

weights to the control points in parallelepiped grid. Thus, more degree of freedom is added to deformation by manipulating the weights on the control points.

Lee et al. [34] propose a 3D model which is deformed to generate facial expression with Dirichlet Free-Form Deformations (DFFD). Other techniques for pseudo muscles include usage of splines [35–37] for smooth and flexible deformations and wires [38]. On the other hand, pseudo muscle approach cannot generate precise skin and muscle behavior like formation of furrows, bulges, wrinkles due to ignoring the underlying structure of the human face [39].

Figure 2.13: Animated faces given expression parameters [34].

2.3.1.5 MPEG-4

MPEG-4 is proposed by Moving Pictures Expert Group (MPEG) as a standard for reducing the abundance of information required in multimedia communications and files. MPEG-4 standard includes 84 parameters to describe the face model [40]. These defined parameters are called Face Definition Parameters (FDPs), Face Animation Parameters (FAPs), Facial Animation Unit Parameters and Facial Animation Tables (FATs). There are 68 FAPs defined to specify the deformation of the face. The Facial Animation Unit Parameters aims to interpret the FAPs. FATs define how to deform the faces.MPEG-4 provides facial expression

generation via the defined FAPs. Each FAP is associated with predefined vertices on the facial model. These vertices can be moved according to the maximal values defined for them on FATs.

Goto et al. [41] gathers information by tracking facial features and transfers the data to generate facial animation with usage of FAPs. Dai et al. [42] animates 2D wireframe face model with FAPs to generate basic facial actions. Patel and Zaveri [43] propose a 3D face model generated from a single image that is animated with a combination of FAPs and morphing techniques. Sheu et al. [44] introduces an interactive approach which applies interpolations and Delaunay triangulations on control points based on FAPs for each feature in the face. A similar approach is present in [45].

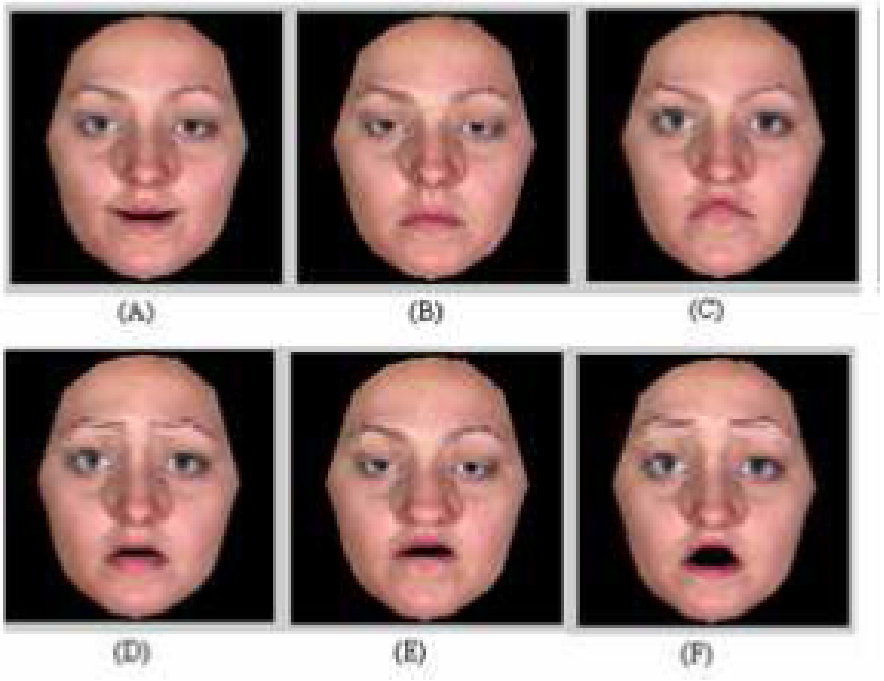

Figure 2.14: Simulation results of [43] for basic facial expressions. (A) Happy, (B) Sad, (C) Angry, (D) Fear, (E) Disgust, (F) Surprise.

2.3.1.6 Disadvantages of Geometric Approaches

Geometric approaches provide fast generation of facial expression animations by use of interpolations, parametrizations and free form deformations. These methods incorporate well-defined geometrical equations that are available to display the effect of muscle force on the face model. Thus, geometric approaches were popular among researchers due to the availability of equations that can simulate the deformation of the skin in a fast fashion. However, these approaches require intensive tuning on the parameters to provide realistic facial expressions and can still fail to produce natural results. Also, complex facial expression generation is poor compared to the physics based approaches due to the parametrization limitations. Another drawback from geometric approaches is that it is not possible to create generic face models or generic parametrizations while physics based approaches provide generic facial meshes.

In parametrization method, whenever more than one muscle are affecting the same group of vertices, blending the effect of the muscles are not possible. This leads to unnatural deformation of the skin. Also, the parameter set depends on the topology of the facial mesh and this prevents preparation of a generic parametrization.

A group of AUs are defined by FACS to describe basic facial expressions and to overcome the issues of the parametrization approaches. However, AU usage is providing good results only on basic facial expressions. More complicated facial expressions are not described well with AUs like the movement around the mouth area. Also, AU usage depends on the face model like any other parametrization approaches.

MPEG-4 is another parametrization method proposed to generate facial expression animations. As the other parameter based approaches, this method needs intervention of the user to tune the parameters like FDPs. Due to these limitations on construction and animation of the face model, geometric approaches are not very successful in generating realistic facial expressions.

These limitations lead the researchers on experimenting with the idea of physics based approaches. Physics based approach promises more realistic animation generation than geometric ones with the construction of physics based face models. These models comply with physics of the human facial skin and underlying muscle and bone structures. Thus, the animations reflect more realistic results. Therefore, physics based approaches are more suitable to generate natural facial expressions over the geometric approaches.

2.3.2 Physics Based Approaches

Physics based approach to facial expression animation deals with the underlying facial structures of the human face to achieve convincing and realistic facial expressions. Since this approach deals with the underlying structures of the face, a face model which reflects this structure is needed for deforming the face. The face model needs to reflect the elasticity of the skin, the contraction of the muscles under this skin and support of the skull. Once a face model which can reflect these characteristics is at hand, it is possible to propose a suitable algorithm that produces facial expressions through the physical interactions of muscles, skin and skull.

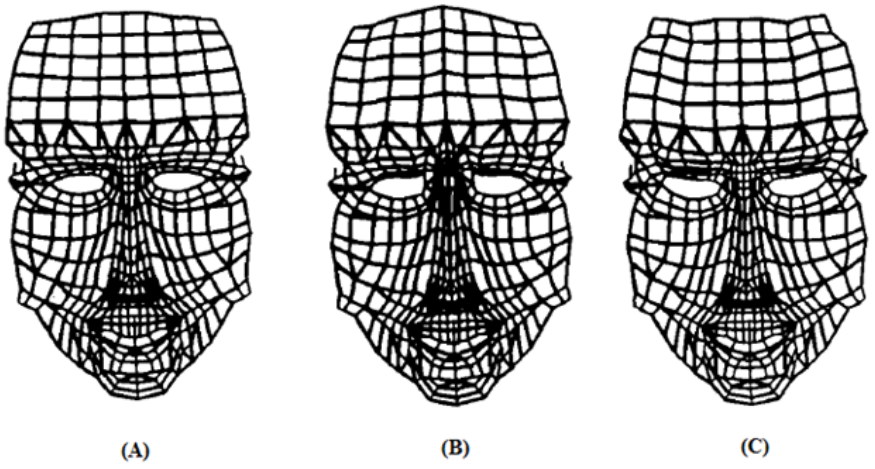

Figure 2.15: Expression examples from [7] (A) No expression, (B) AU-1, (C) AU-2.

A three layer muscle based facial model is described by Platt and Badler [7] which employs FACS for facial deformation. The three layers proposed on this model is outermost skin layer which is defined as a set of movable 3D points, unmovable skull layer at the bottom layer and muscles as elastic arcs between outermost and bottom layers. Platt and Badler's work is extended by adding anatomy based muscles to the face model instead of using arcs as muscles by Essa [46]. This model is animated via estimating the muscle control parameters from video sequences with optimal optical flow method and applying these parameters to the model.

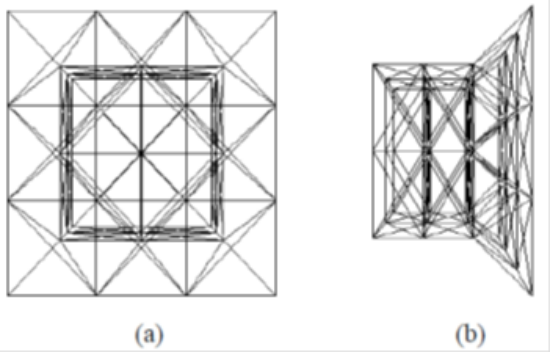

Figure 2.16: Three layer facial tissue of [9] (a) Top view, (b) Side view.

Another three layered face model approach was suggested by Waters and Terzopoulos [9]. This model employs deformable lattice which is a discrete deformable model constructed by point masses connected by springs. The three layers in this model represent the epidermis, dermis and fascia/skull layers of the human face. Each layer has different resistance to deformation. Topmost layer is resistant to deformation to some extent while the dermis layer is highly deformable to reflect the fatty tissue under the skin of the face. The muscles in this model are situated at the bottom layer and Waters' [8] muscle model is deployed. FACS is incorporated to generate the facial expressions. To simulate the dynamics of a deformable lattice, a numerical approach is used to mobilize this mass-spring system.

Lee et al. [10] proposes a simplified version of [9] which can produce realistic facial expressions. The face model is created by adapting a generic face mesh to high

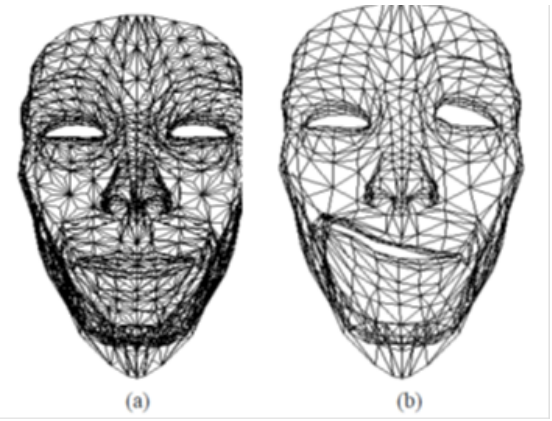

Figure 2.17: (a) Undeformed skin layer. (b) Deformed under the influence of AU1 and AU12 (only epidermis is displayed for clarity).

resolution laser scanned range data. This presented technique for model creation allows the face mesh to be conformed to individuals. Wu et al. [47] suggests a physical approach on creation of wrinkles with a dynamic model. To do this viscoelastic property of the human skin is modelled along with the anatomic muscles and fatty tissue as Hookean springs. Facial skin is animated through the tension created by the muscles at the insertion points which are assigned on the skin according to the anatomic data.

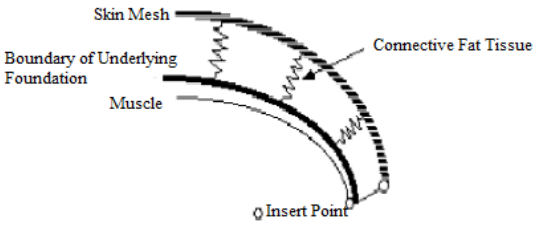

Figure 2.18: Simplified facial structure from [47].

The study of the biomechanics of the human skin in [48] suggests that skin has a non-linear stress-strain relationship. Skin applies small resistance when it is under low stress while it applies significantly more resistance to stretch when it is under great stress. This viscoelastic property of the skin is modelled in [49] with biphasic springs. However, approximation of stress-strain relationship of the skin with biphasic springs results in inaccurate and non-smooth deformation. This issue is addressed by Zhang et al. in [50] with introduction of non-linear springs

in MSD system. It is argued that the viscoelastic property of the facial skin can be reflected with the characteristics of non-linear springs due to its stress-strain relationship resembles the one observed in facial skin. Thus an anatomy based mass-spring system which is modelled by non-linear springs to reflect the dynamics of skin and muscles without bone structure was proposed.

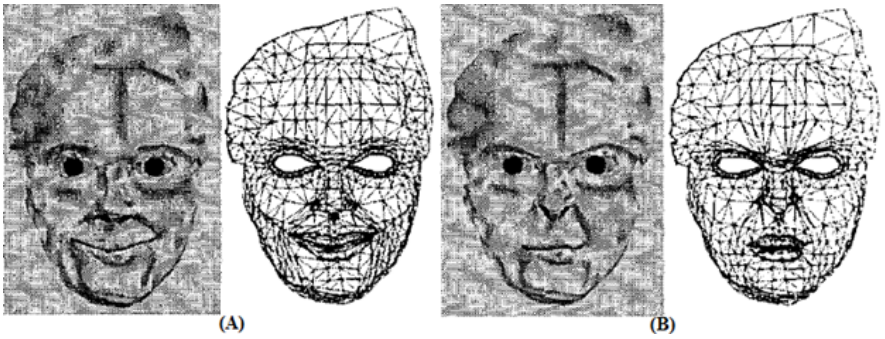

Figure 2.19: Synthesized expressions (A) Happiness, (B) Anger [50].

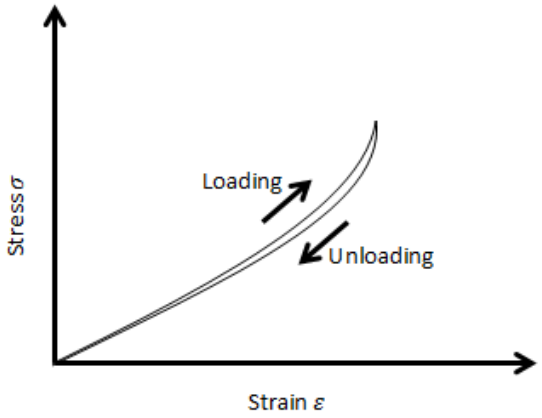

Figure 2.20: Stress-strain relationship of the skin [48].

Non-linear springs are employed to prevent the collapse of the spring-mesh system. It was shown that a linear spring-mesh system collapses under large compressive forces [51]. This model incorporates the muscle types introduced by Waters in [8]. FACS is employed for facial expression generation. The model is deformed based on Lagrangian dynamics when muscles contract. A new hierarchical model is presented in [51] to improve the results of Zhang et al. This new model includes a skull structure which helps with articulation of the jaw. A new force is introduced for preventing the collapse of mass-spring system. Non-linear dynamics of the

skin is provided through structural springs via non-linear functions to simulate the strain-stress relationship of the springs. Improvements on the muscle mapping and skull modelling are presented in [52]. A further extension [21] is added to [52] by adding adaptive refinement process which is applied on highly deformed edges and highly stretched elements for refinement. The results of this work can be seen in figure 2.21.

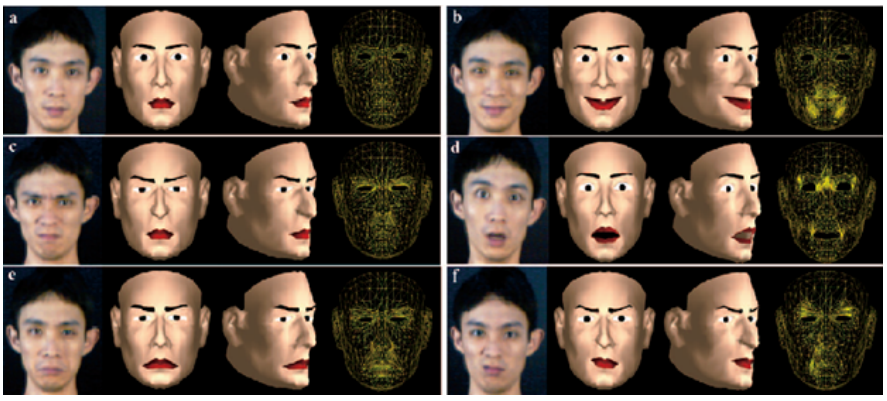

Figure 2.21: Synthesized expressions compared with the actual ones: (a) neutral face, (b) happiness, (c) anger, (d) surprise, (e) sadness, and (f) disgust.

Bui et al [53] proposes a simple muscle based 3D triangle polygon mesh for face model generated from 3D scanner data while extending muscle model of Waters [8] via adding wrinkles and bulges. The proposed face model incorporates the idea to add more polygons on the most expressive parts of the face from Greta [19]. Since the sphincter muscles defined by Waters [8] do not generate realistic deformations, pseudo-muscles are used for eyes and mouth regions. Also, use of vector muscles instead of sheet muscles for realistic deformation in the forehead area is proposed. Wrinkles are generated by assigning wrinkle amplitudes to each vertex.

2.3.2.1 Advantages and Disadvantages of Physical Approaches

Physics based approaches are more sophisticated than the geometrical approaches since they deal with deformation of facial models that are generated based on physical structures of the human face. Physics based approach tries to generate

Figure 2.22: Generated wrinkles [52].

the facial expressions as close as possible to the real human face would. However, to achieve this, actual facial tissue dynamics needs to be modelled. This leads to high computational cost due to the high complexity of the face models.

On the other hand, geometric approaches promise fast generation of facial expressions without any emphasis on the physics of the human skin and its underlying structures. This is achieved by the usage of parametrization techniques.

However, the parametrizations require intensive tuning in both creation and deformation of the face model to generate natural facial expressions. Physical approaches do not require intensive tuning since they are based on physical characteristics of skin and the underlying structures. Deformation of the regions affected by multiple muscles exhibits realistic look in contrast to geometric approaches where deformation blending is not possible due to parametrization limitations. This means complex facial expression generations is superior to geometric approaches. Physics-based approaches provide generic facial meshes while it is not possible to create generic face models in geometric approaches. Therefore, physics based approaches are more suitable to generate natural facial expressions over the geometric approaches.

This research aims to propose improvement to the existing physics based approaches for creation of realistic facial expression animation while retaining anatomical consistency. To achieve this, a physics-based face model and an algorithm

to deform this face model are required. The face model that is used in this research is a single layer face model called HIGEM proposed in [54]. To properly deform HIGEM in 3D space, a new algorithm is proposed which does not require an underlying skull structure to avoid mesh collapse. Our model embodies all major muscles of the human face. Thus, it enables us to incorporate a muscle force model in our animations. This force model generates regions of influences and force distributions dynamically during the animation process.

Chapter 3

Proposed Work

Realistic animation of facial expressions with computers has been a topic of interest since Parke's work[13]. Generation of facial expressions requires a face model and a deformation algorithm. Thus, each study since Parke's pioneering work, proposed new techniques or improved the techniques proposed by others. This improvement can be through the introduction of a new face model or an animation technique. These works can be grouped into two groups with respect to the approach used in the study. These are geometric approaches and physical approaches. Geometric approaches are based on the geometric equations which could simulate facial deformation. This approach ignores the underlying structure of the face and the mechanics of the facial skin. On the other hand, physical approaches are based on the physics of the facial skin and underlying muscle and bone structures to deform the face model.

This research proposes a new physics-based animation of a generic face model without the need of an underlying bone structure for realistic facial expression generation. The face model used in this study is a generic wireframe face model proposed by [54]. This wireframe is a Mass-Spring-Damper (MSD) system and its polygons correspond to facial skin. The vertices of the wireframe represent the mass of the facial skin, providing momentum for the skin. A set of facial muscles are placed onto this facial model in accordance with the anatomic structure of the human face. This face model is deformed with the help of the muscle

forces exerted on the facial mesh. These muscle forces are calculated and applied according to the facial expression animation algorithm proposed in this study. The generated facial expressions can either be stored as a sequence of animation frames or as video.

3.1 The Integrated Physics Based Face Model

3.1.1 Single Layer Facial Mesh

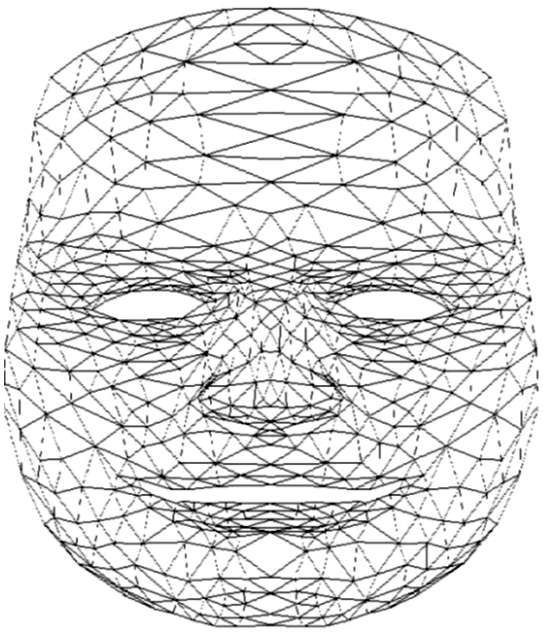

Figure 3.1: Single layer facial mesh.

The 3D single layer face model used in this study designed in [54] as a generic wireframe which conforms to the anatomic properties of the human face in 3D space. The model contains 612 nodes and 1128 triangles called faces as in CANDIDE models mentioned in [14–16]. Each face and node is numbered in the model. The model consists of a face mesh and facial muscles. The face mesh incorporates a classic MSD system which contains point masses and springs which are connecting each point mass to its neighboring masses.

3.1.2 Muscle Model

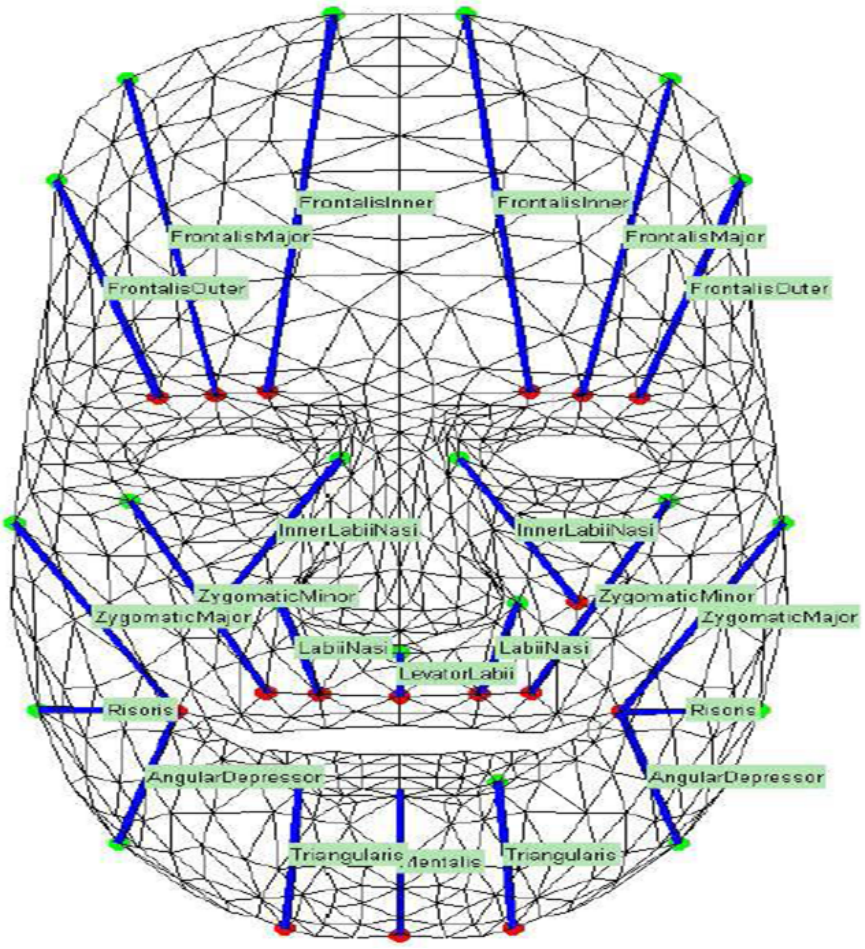

Figure 3.2: Muscles on the single layer face model.

The single layered face model includes major muscles of the human face. Placements of the facial muscles are performed by complying with the anatomic maps suggested by Goldfinger in [12]. The muscles are all implemented as linear muscles and modeled after Zhang et al.'s [21] linear muscle model which is an extension to [8]. Each muscle has an insertion point which connects the muscle to skin and an attachment point which connects the muscle to skull assigned on the vertices of the model. Since the proposed model is single layered with no skull structure, the attachment points of the muscles are assigned to proper vertices in the MSD system in accordance with [12]. Insertion and attachment points help to determine the direction of the muscle force exerted on the MSD system by each muscle.

The contraction of the linear muscle applies muscle forces in a region of influence. In this region of influence, the maximum force is applied at the insertion point while the minimum force (i.e. zero force) can be observed at the attachment point. This means that the muscle force on the vertices is decreased from insertion point to attachment point. The same decrease in muscle force is observed when angle increases with the axis created between the attachment and insertion points and the mass point in the influence area. Due to this distribution of muscle force in influence area, the point masses which lie in this region have different muscle forces on them according to their positions.

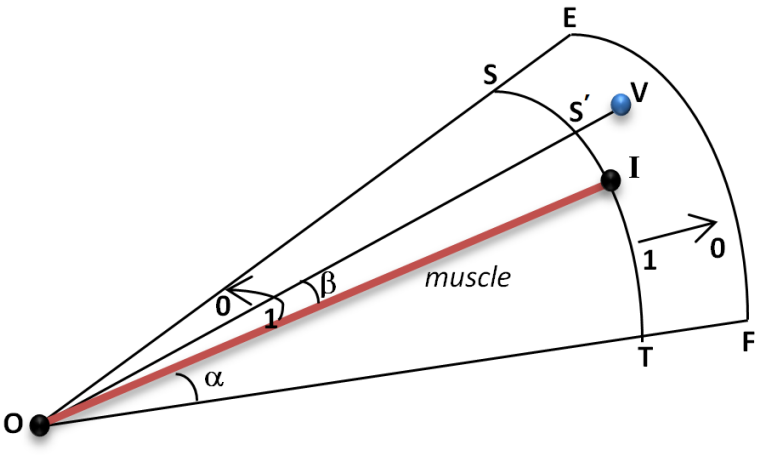

Figure 3.3: The linear muscle model.

The muscle forces applied on the masses inside the region of influence can be calculated with the following equation.

$$\vec{T}_{in} = \delta f_n \frac{O_n - X_i}{|O_n - X_i|} \tag{3.1}$$

where

$$\cos\beta = \vec{OI} \cdot \vec{OX} / (|\vec{OI}| \cdot |\vec{OX}|)$$

$$\delta_A = \begin{cases} \frac{cos\beta - cos\alpha}{1 - cos\alpha} & \text{if} \quad cos(\beta) \geq cos(\alpha) \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

$$\delta_R = \begin{cases} cos\left(\frac{\pi}{2}\frac{r - |\vec{OX}|}{r}\right) & \text{if} \quad |\vec{OX}| \leq r \\ cos\left(\frac{\pi}{2}\frac{|\vec{OX}| - r}{r_{max} - r}\right) & \text{if} \quad r < |\vec{OX}| \leq r_{max} \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

$$\delta = \delta_A \cdot \delta_R \tag{3.4}$$

The deformation of the mesh due to the contraction of the linear muscle model adopted in this study can be observed in the following figure.

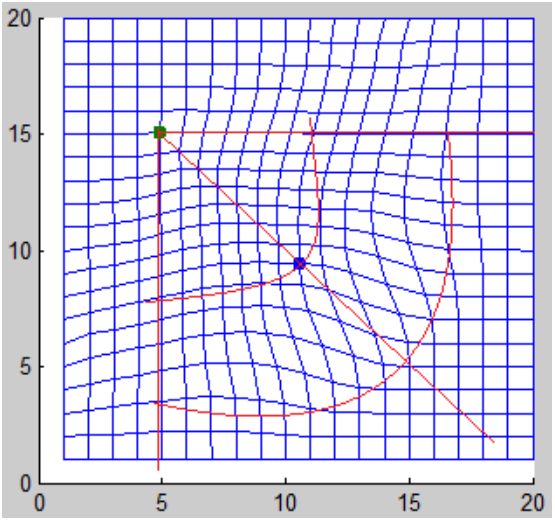

Figure 3.4: Deformation due to the contraction of the linear muscle adopted in this study.

3.1.3 Spring Forces, Damping and Acceleration

When a muscle contracts, the point masses in its region of influence are affected by the generated muscle forces. The affected point masses tend to move from their initial positions under the influence of muscle forces. Thus, the springs which are connected to these point masses are stretched. This stretch occurs due to the spring forces generated. In general MSD systems are modeled using Hooke's law. According to Hooke's Law, stress is directly proportional to strain. This is formulated as

$$F = -kx \tag{3.5}$$

where k is the stiffness of the spring, x is the displacement of the spring's end from the point of equilibrium and F is the restoring force exerted by the spring.

Hooke's Law represents the linear spring elasticity. However, the facial skin represents non-linear behavior as discussed in Literature Review section. Thus, the spring force equation shall reflect non-linearity even though it is based on Hooke's Law. This is achieved by addition of a non-linearity factor to the equation (3.5). The resulting equation is,

$$\vec{S}_{ij} = -(1 + (|\vec{X}_{ij}| - d_{ij})^2)^{\gamma} \times k_0(|\vec{X}_{ij}| - d_{ij})\frac{\vec{X}_{ij}}{|\vec{X}_{ij}|} \tag{3.6}$$

where X_i is an arbitrary skin point, X_j is one of X_i's neighbors, γ is non-linearity factor, d_{ij} is the rest length of spring and $\vec{X}_{ij} = X_i - X_j$.

Under large forces, it is possible for a point mass to move past the opposite edge as discussed in [55]. To avoid this kind of behavior under strong forces, a new force that provides resistance to the collapse needs to be exerted to the affected

point mass in the opposing direction. This force is based on the perpendicular distance of the point that is on the opposite edge and always in away from the edge. The approach introduced in [55] is employed in [49], [21], [52] and [56] with success. Thus, this study proposes Edge Repulsion (ER) approach for prevention of face collapse. For this ER approach, equation (3.7) is proposed.

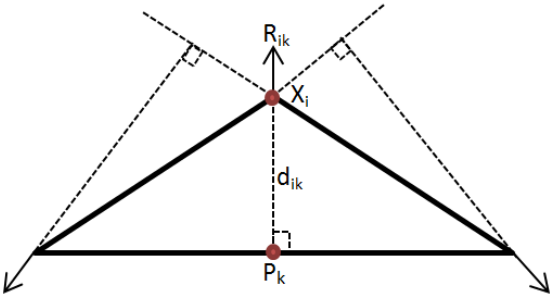

Figure 3.5: Edge Repulsion Forces.

$$\vec{R}_{ik} = -(1 + (|X_i P_k| - d_{ik})^2)^{\zeta} \times k_0(|X_i P_k| - d_{ik}) \frac{X_i - P_k}{|X_i P_k|} \tag{3.7}$$

where $\vec{R}_{ik}$ is the force applied to the affected point mass to prevent collapse, X_i is the affected point mass, P_k is the point projection of X_i onto the opposite edge, $X_i P_k = X_i - P_k$, d_{ik} is the rest length of the ER spring and ζ is non-linearity factor.

The total spring forces applied on the point masses are then calculated with the equation (3.8).

$$F_{Vis} = \sum_{j \in N_i} \vec{S}_{ij} + \sum_{k \in E_i} \vec{R}_{ik} \tag{3.8}$$

where N_i is the set of neighboring mass points and E_i is the set of edge projection points.

In Physics, damping is an effect that reduces the amplitude of oscillations in an oscillatory system and defined with equation (3.9).

$$F_c = -c\frac{dx}{dt} \tag{3.9}$$

where c is the viscous damping coefficient.

Since human skin has a viscoelastic structure, viscous damping behavior is observed. The effect of the forces applied on the point masses shall dissipate through the facial mesh to model damping. This dissipation shall eventually lead the facial mesh to rest. Thus, the rate of the dissipation of the kinetic energy generated by the MSD system is controlled by a damping factor. This damping factor is dependent on the velocity of the point masses. The viscous damping is modelled with equation (3.10).

$$\vec{D}_i = -c\frac{d(X_i)}{dt} \tag{3.10}$$

When the muscle forces and spring forces are applied on the facial mesh, the point masses gain acceleration. This is due to the Newton's second law of motion. Whenever a mass is under force, it tends to accelerate. Thus, acceleration of the point masses is modelled in accordance with the second law of Newton's laws of motion.

$$\mathbf{M}\frac{d^2x(t)}{dt^2} \tag{3.11}$$

where $\mathbf{M}$ is the matrix of point masses.

The motion of the point masses of the facial mesh under muscle forces is then modelled with the integrated equation (3.12).

$$\begin{aligned} \vec{F_i} &= \sum_{n \in M} \vec{T_{in}} + \sum_{j \in N_i} \vec{S_{ij}} + \sum_{k \in E_i} \vec{R_{ik}} + \vec{D_i} \\ &= m \frac{d^2(X_i)}{dt^2} \end{aligned} \tag{3.12}$$

In this second order ordinary differential equation (ODE), $\vec{F_i}$ represents the integrated forces, M is the set of muscles and m is the point mass of the vertex. Equation (3.12) can be solved for the rate of change of velocity, in other words acceleration as follows;

$$\vec{V_i}(t) = \frac{dX_i(t)}{dt}$$

$$\frac{d\vec{V_i}(t)}{dt} = \frac{\sum_{n \in M} \vec{T_{in}} + \sum_{j \in N_i} \vec{S_{ij}} + \sum_{k \in E_i} \vec{R_{ik}} + \vec{D_i}}{m} \tag{3.13}$$

This first order ODEs are then solved by a quasi-implicit method by estimating the new positions of the point masses in the next time step for the calculations.

$${X_i}^{t+1} = X_i^t + \Delta t \vec{V_i}^t \tag{3.14}$$

When the estimated displacements of the point masses are integrated to (3.13), the equation (3.15) is obtained.

$$\vec{V}_i^{t+1} = \vec{V}_i^t + \frac{\Delta t}{m}\Big(\sum_{n\in M} \vec{T}_{in}^{t+1} + \sum_{j\in N_i} \vec{S}_{ij}^{t+1} + \sum_{k\in E_i} \vec{R}_{ik}^{t+1} - c\vec{V}_i^{t+1}\Big) \tag{3.15}$$

where

$$X_i{}^{t+1} = X_i^t + \Delta t \vec{V}_i^{t+1} \tag{3.16}$$

$$\begin{aligned}
g_{ij}^{t+1} =& -\left(1 + (|\vec{X}_{ij}^{t+1}| - d_{ij})^2\right)^{\gamma} \times k_0 \frac{(|\vec{X}_{ij}^{t+1}| - d_{ij})}{|\vec{X}_{ij}^{t+1}|} \\
\vec{S}_{ij}^{t+1} =& g_{ij}^{t+1} \vec{X}_{ij}^{t+1} \\
h_{ik}^{t+1} =& -\left(1 + (|X_i^{t+1} P_k^{t+1}| - d_{ik})^2\right)^{\zeta} \times k_0 \frac{(|X_i^{t+1} P_k^{t+1}| - d_{ik})}{|X_i^{t+1} P_k^{t+1}|} \\
\vec{R}_{ik}^{t+1} =& h_{ik}^{t+1} (X_i^{t+1} - P_k^{t+1})
\end{aligned} \tag{3.17}$$

$$\Delta X_{ij}^{t+1} = X_i^{t+1} - X_j^{t+1},$$

$$\Delta X_i^{t+1} P_k^{t+1} = X_i^{t+1} - P_k^{t+1} \tag{3.18}$$

Using Equations (3.15), (3.16) and (3.17) we can represent the next velocity as,

$$\vec{V}_i^{t+1} = \vec{V}_i^t + \frac{\Delta t}{m}\Big(\sum_{n\in M} \vec{T}_{in}^{t+1} + \sum_{j\in N_i} g_{ij}^{t+1}(X_i^t + \Delta t\vec{V}_i^{t+1} - X_j^{t+1}) + \sum_{k\in E_i} h_{ik}^{t+1}(X_i^t + \Delta t\vec{V}_i^{t+1} - P_k^{t+1}) - c\vec{V}_i^{t+1}\Big) \tag{3.19}$$

This implicit equation is then solved for $\vec{V}_i^{t+1}$ to obtain explicit equation (3.20).

$$\vec{V}_i^{t+1} = \frac{m\vec{V}_i^t + \Delta t \sum_{n\in M} \vec{T}_{in}^{t+1} + \sum_{j\in N_i} g_{ij}^{t+1}(X_i^t - X_j^{t+1}) + \sum_{k\in E_i} h_{ik}^{t+1}(X_i^t - P_k^{t+1})}{m + \Delta tc - \Delta t^2(\sum_{j\in N_i} g_{ij}^{t+1} + \sum_{k\in E_i} h_{ik}^{t+1})} \tag{3.20}$$

Also the new positions of the point masses shall be computed with the newly found velocities by usage of the equation (3.14).

3.2 UV Conversions

3.2.1 Skull Penetration Problem

In physics based face models, three layer face models are chosen to provide the most realistic face models. In these three layer structure models, facial skin and muscle layers are directly affecting the motion of the facial skin points by contractions and the movement of the parts of the skin as a response to the muscle contractions. The skull layer in this three layer structure serves for two purposes. First is to provide a base for the muscles to attach and the second purpose is

prevention of collapse of the skin layer into the skull layer. In 3D space, skull penetration can easily happen due to the direction of the muscle forces applied on the point masses in the facial skin mesh. Human face is not a flat surface but a bumpy one. Thus, the combined spring and muscle forces can point into the skull. This is shown in 3.6 with force f_{input} applied onto point mass X. This issue was addressed by Zhang et al. in [21] by addition of a stabilizing force called $f_{constraint}$. This force helps prevention of the skull penetration by cancelling the force f_{normal} which is the component of f_{input}. However, this kind of approach adds more load into the already complex system by addition of new calculations.

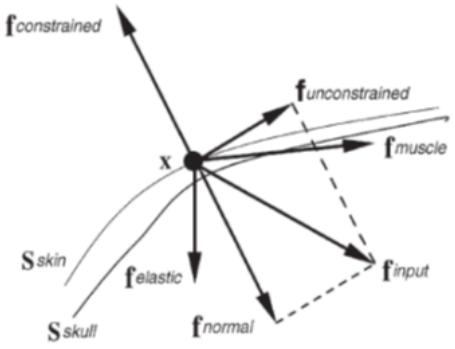

Figure 3.6: Skull penetration prevention technique introduced by [21].

In this study, employed face model HIGEM has a single layer structure and a skull structure is not present. Thus, prevention of skull penetration is not possible with the approach proposed by [21]. Both attachment and insertion points are on the same mesh since no skull structure is present. In 3D space, the direction of the calculated muscle force may lead to collapse of the face model due to the absence of skull. In 3.7, this kind of a muscle force is depicted. In time with large enough muscle force, the insertion point on the facial mesh starts moving in the direction of muscle force and eventually folds into the attachment point. This folding of the facial mesh onto itself shall be prevented on the single layer face model.

Thus, a new algorithm is proposed to prevent collapse of the face model in the absence of skull structure. This algorithm presents the idea of using 2D to 3D projections of muscle forces.

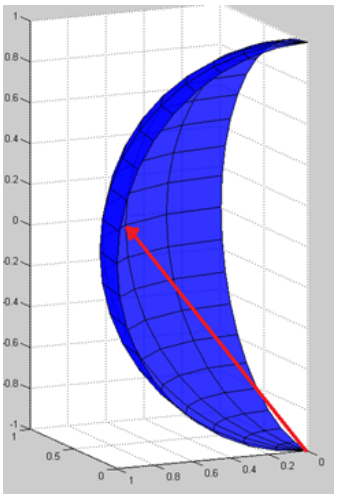

Figure 3.7: 3D muscle force direction in single layer face model.

3.2.2 Back and Forward Projections of Muscle Forces

When muscle forces are calculated in 3D space, the direction of the motion of the point masses affected by the muscle force is in 3D space as shown in 3.7. The spring forces and ER forces are all on the facial mesh surface. Thus, the skull penetration problem occurs due to the direction of the muscle forces. Since the calculation of muscle forces in 3D space presents the skull penetration problem, the calculations are proposed to be performed on 2D space as if the face model lies in 2D space. This ensures that the muscle forces are lying on the face mesh as spring forces and ER forces. This is achieved by projecting the facial mesh from 3D to 2D space with the following equations.

$$u = \arctan(\frac{x}{z}), \quad v = y \tag{3.21}$$

The muscle forces applied onto the point masses on the facial mesh is then calculated with equation 3.1 in 2D space. The calculated muscle forces for Medial Frontalis muscle can be observed in the following figure. The muscle force directions from the affected point mass to the attachment point while the forces are lying on the facial mesh.

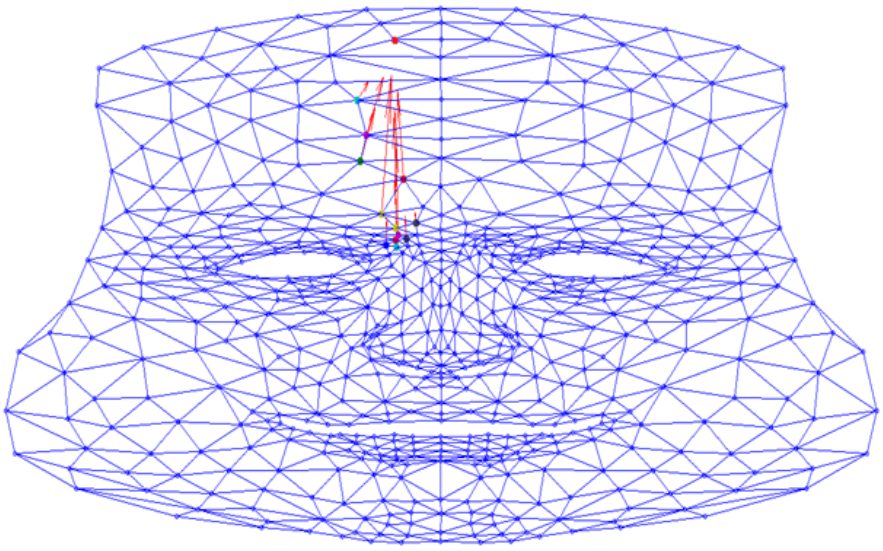

Figure 3.8: Muscle forces applied by Medial Frontalis to the point masses in its region of influence.

However, the muscle forces and the facial mesh shall reside in 3D space. Thus, the calculated 2D forces and the facial mesh are both projected back to 3D space. The projection of the facial mesh is easy since the z coordinates of the point masses are known. The back projection of the muscle forces shall be performed in a way that ensures the force is still lying on the surface of the facial mesh. This is achieved by introduction of a two-step projection algorithm.

The first step of the proposed projection algorithm is locating the triangular facial mesh element that 2D muscle force is lying on. This is achieved by calculations of dot and cross products of the created vector from affected mass point to its neighbors and the muscle force vector. These calculations help finding the triangle edges that lies on the left and right of the muscle force.

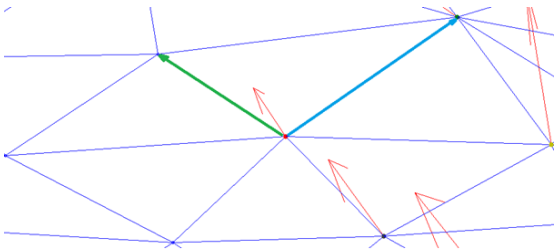

Figure 3.9: Locating the triangular facial mesh element that the muscle force lies on.

First the vectors emerging from the affected point mass to its neighbors are all created. Then the dot products and cross products of these vectors with the muscle force vector are calculated. The maximum valued dot product is chosen to be the closest vector to the muscle force. In 3.9, this vector is marked with green arrow. This vector's cross product is stored. Then the second maximum dot product is found. This is marked with blue arrow in 3.9. The cross product of this blue vector is compared with the green vector's cross product. If they both point in the opposite directions, then this means the blue vector is the second edge of the triangle, if not process of finding the next maximum dot product is repeated until all vectors are visited and a proper vector is found.

When both vectors lying on the right and left side of the affected point mass is located, all three points of the triangular facial mesh element is obtained. Since all point masses in the facial mesh is numbered in both 3D and 2D space, 3D coordinates of the triangle is automatically known. In other words, the surface that the muscle force in 3D space shall lie is located. Thus, this surface shall be defined. The 3D coordinates of this face is utilized to calculate the normal of the surface with following equation.

$$n = (P_2 - P_1) \times (P_3 - P_1) \tag{3.22}$$

where

$$P_1 = (x_1, y_1, z_1), \quad P_2 = (x_2, y_2, z_2), \quad P_3 = (x_3, y_3, z_3)$$

With the calculated surface normal, the surface equation can be obtained. The normal of a surface is formulated as;

$$n(r - r_0) = 0 \tag{3.23}$$

where

$$r = (x, y, z), \quad r_0 = (x_0, y_0, z_0)$$

Equation 3.24 is obtained by substitution of r and r_0 into equation 3.23.

$$n_x(x - x_0) + n_y(y - y_0) + n_z(z - z_0) = 0 \tag{3.24}$$

It is known that

$$u = \arctan(\frac{x}{z}) \quad \rightarrow \quad x = \tan(u)z,$$

$$v = y$$

If these are substituted into equation 3.24, equation 3.25 is obtained which gives the z component of the muscle vector. When equation 3.25 is solved, the x, y and z components of the muscle force can be calculated easily.

$$z = \frac{(n_x x_0 + n_z z_0 - n_y y_0)}{(n_x u + n_z)}, \quad x = \tan(u)z, \quad y = v \tag{3.25}$$

This muscle force is still lying on the surface of the located face while its direction is from the affected point mass to the attachment point. The resulting muscle force vectors can be observed in the following figures.

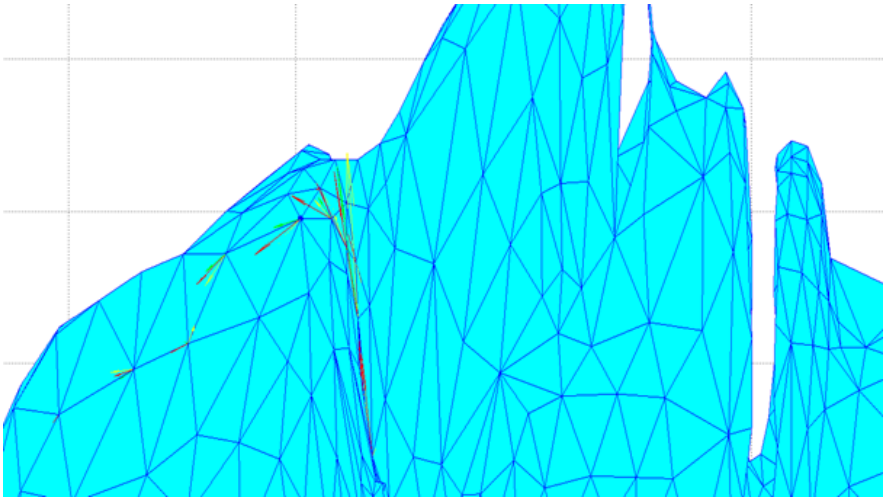

Figure 3.10: The muscle forces on 3D face model are shown with red arrows.

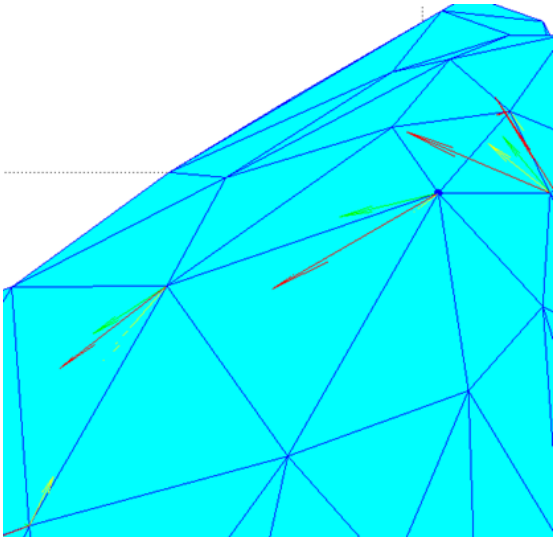

Figure 3.11: Close up of figure 3.10. Red arrows denote the muscle forces.

3.3 Stack Approach on Muscle Force Calculations

Physics based approach on facial animation is a daunting task due to the huge computational cost introduced by the complex facial structures. However, computational cost can be reduced by introduction of new algorithms. One of the

algorithms proposed in [52], suggests marking the point masses as dynamic, static or quasi-static to speed up the calculations of muscle force. This approach enables calculations to be performed only on some parts of the face model instead of each point mass that exist on the facial mesh. However, marking the point masses as dynamic or not depends on performing offline pre-computation of muscle forces. Also, selection of the dynamic point masses depends on the subject's facial topology and the activation levels of the muscles.

Here in this study, a new approach is proposed to eliminate the offline pre-computation to define the dynamic and static point masses by using stack. The muscle force calculations are started from the insertion point where the muscle force is at maximum value by pushing that point mass into the stack. The top element in the stack is then used for muscle force calculation according to equations given. If the calculated muscle force on this point mass is not negligible, then its direct neighbors are pushed into the stack. After addition of the neighbors to the stack is performed, the point mass on top of the stack is used for muscle force calculation again.

This approach enables only calculating the muscle forces for the point masses in the region of influence of the affected muscle. This is ensured by starting to push the direct neighbors of the insertion point in to the stack and checking the position of the pushed point mass to see whether it lies in the region of influence with equations (3.2) and (3.3). If the point mass does not lie in the region of influence of that muscle, then the muscle force is zero on that point mass, so the calculation continues by popping the next element in the stack. If the point mass lies in the region of influence of the muscle, neighbors of that point mass are pushed into the stack for muscle force calculation. This continues until there is not any element is left in the stack.

3.4 Simulation Algorithm of the System

The proposed physics based facial expression animation system runs the following algorithm to generate the frames of the resulting animation. The generated frames are stored in a special matrix and plotted on the screen at run time.

Algorithm 3.4.1: ANIMATE MESH($facemodel, timesteps$)

for $i \leftarrow 1$ **to** $timesteps$

do
- **for** $vertex \leftarrow 1$ **to** $totalNumberOfVertices$
  - **do** *Estimate the positions with equation (3.14) at timestep $i+1$ by using v_i^n*
- **for** $muscle \leftarrow 1$ **to** $totalNumberOfMuscles$
  - **do** **if** *vertex $\exists$ in region of influence*
    - **then** *Calculate F_{Mus} with equation (3.1)*
- **for** $affectedVertex \leftarrow 1$ **to** $totalNumberOfAffectedVertices$
  - **do** *Calculate F_{Vis} between the vertices with equation (3.17)*
- **for** $vertex \leftarrow 1$ **to** $totalNumberOfVertices$
  - **do** **if** *$vertexF_{Mus}$ is non-zero*
    - **then** *Prevent collapse of the faces with UV conversions*
- **for** $vertex \leftarrow 1$ **to** $totalNumberOfVertices$
  - **do** *Calculate v_i^{n+1} with equation (3.20)*
    - *Determine the new coordinates of the vertices with equation (3.14) by using v_i^{n+1}*
    - *Update v_i^n with v_i^{n+1} for timestep $i+1$*
- *Display generated frame*

return

3.5 Simulation Results

3.5.1 Generated Facial Expressions

The proposed physics based facial expression animation system is tested with generation of eight facial expressions. These eight facial expressions are divided into two groups. First group is the basic facial expressions. A total of six basic facial expressions generated. These facial expressions are *Anger, Disgust, Fear, Happy, Sad* and *Surprise*. Second group is the composite facial expressions. Two composite facial expressions are generated with the system. These facial expressions are *Disgusted Anger* and *Happy Surprise*. The results of the animation can be observed in the figures 3.12 and 3.13.

The first subfigure in figure 3.12 is the *Neutral* state of the facial mesh when there is no muscle activation.

The second subfigure shows the facial expession *Anger*. The muscles *Procerus*, *Depressor Anguli Oris* on both sides, *Levator Labii Alaeque Nasi* on both sides and *Risorius* on both sides of the face are activated. Contraction of *Procerus* moves the vertices downwards and creates frown while the contraction of *Depressor Anguli Oris* and *Risorius* on both sides curls the sides of the mouth downwards.

The third subfigure shows the facial expression *Disgust*. The muscles *Depressor Anguli Oris* on one side, *Levator Labii Superioris* on one side, *Corrugator* on one side and *Levator Labii Alaeque Nasi* on both sides of the face are activated. Contraction of *Levator Labii Superioris* provides the upturn of the one side of the mouth while nose wrinkling is generated by contraction of *Levator Labii Alaeque Nasi* on both sides. *Contraction* of Corrugator generates the frown while *Depressor Anguli Oris* on the other side of the face move the corner of the mouth downwards.

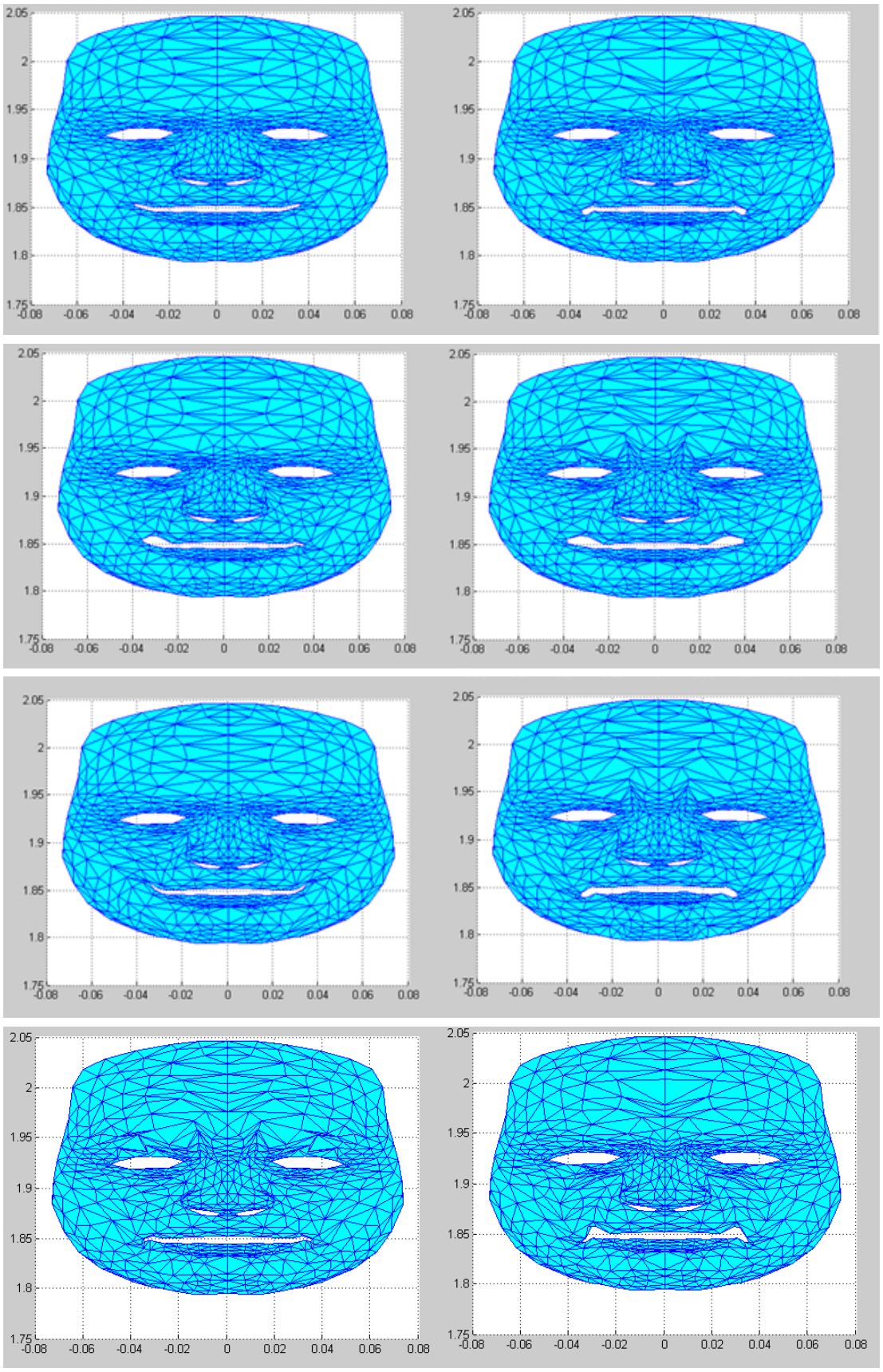

Figure 3.12: Facial expressions Neutral, Anger, Disgust, Fear, Happy, Sad, Surprise and Disgusted Anger are presented respectively.

The fourth subfigure shows the facial expression *Fear*. The muscles *Medial* and *Lateral Frontalis*, *Levator Labii Superioris*, *Risorius* and *Depressor Anguli Oris* on both sides of the face are activated. The contraction of *Medial* and *Lateral Frontalis* muscles on both sides provides the movement of the vertices in the forehead region to move upwards while contraction of *Levator Labii Superioris*, *Risorius* and *Depressor Anguli Oris* on both sides generates the movement of the mouth.

The fifth subfigure shows the facial expression *Happy*. The muscles *Zygomatic Major*, *Levator Anguli Oris* and *Levator Labii Alaeque Nasi* on both sides of the face are activated. The contraction of *Zygomatic Major* and *Levator Anguli Oris* provides the upward curling of sides of the mouth while the contraction of *Levator Labii Alaeque Nasi* realize the enlargement of the nose.

The sixth subfigure shows the facial expression *Sad*. The muscles *Medial Frontalis*, *Depressor Anguli Oris*, *Risorius*, *Levator Labii Alaeque Nasi*, *Depressor Labii Inferioris* on both sides of the face are activated. The contraction of *Medial Frontalis* moves the inner eyebrow region upwards. The contraction of *Depressor Anguli Oris*, *Depressor Labii Inferioris* and *Risorius* curls the mouth downwards while pouting the lower lip.

The seventh subfigure shows the facial expression *Surprise*. The muscles *Medial*, *Lateral* and *Outer Frontalis* and *Depressor Anguli Oris* on both sides of the face are activated. All of the *Frontalis* muscles are contracted to move the vertices in the forehead region upwards while *Depressor Anguli Oris* on both sides of the face provided the slight downward curl of the sides of the mouth.

The eight subfigure shows the facial expression *Disgusted Anger*. The muscles *Procerus*, *Depressor Anguli Oris*, *Levator Labii Alaeque Nasi*, *Risorius* and *Levator Labii Superioris* on both sides of the face are activated. The contraction of *Procerus* gives the frown to the facial expression while *Levator Labii Alaeque Nasi* provides the nose wrinkling. The contraction of *Depressor Anguli Oris* and

Risorius generates the downward curl of the mouth while *Levator Labii Superioris* pulls the corners of the mouth upwards.

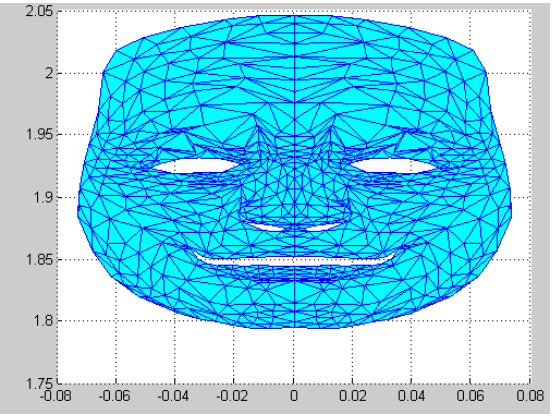

Figure 3.13: Facial expression Happy Surprise.

The facial expression *Happy Surprise* is generated by activating the muscles *Zygomatic Major*, *Levator Anguli Oris*, *Levator Labii Alaeque Nasi*, *Medial,Lateral* and *Outer Frontalis*. The upwards movement of the vertices in the forehead region is provided by the contraction of *Medial,Lateral* and *Outer Frontalis* muscles on both sides of the face. The expansion of the nose is generated with the contraction of *Levator Labii Alaeque Nasi* on both sides of the face while the upwards curl of the mouth is provided by *Zygomatic Major* and *Levator Anguli Oris* muscles.

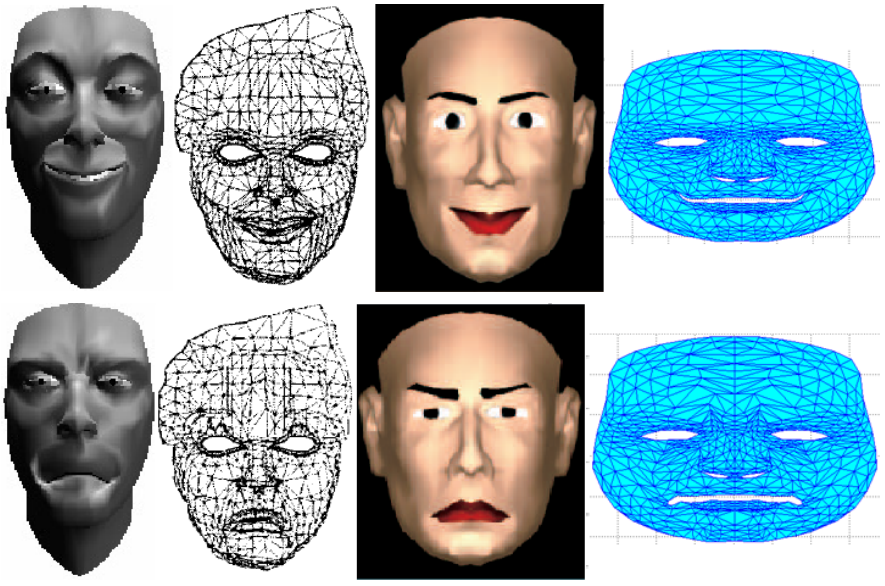

Figure 3.14: Results of different works compared to single layer approach presented here.

Obtained results of different facial expression generation techniques and the results of the animation system proposed here is displayed together in figure 3.14. Figure 3.14 depicts *Happy* and *Sad* facial expressions in first and second rows respectively. The results are taken from the works [9], [50] and [21] from left to right. These three works all employs three layered approach while the proposed animation system here utilizes a single layered approach. Even though the face model is a single layered one, the generated facial expressions are accurate compared to the results of the previous works.

3.5.2 Runtime Performance

The runtime performance of the proposed system is slow compared to the other physics based approaches. This is mainly due to introduction of the model collapse avoidance algorithm and removal of offline pre-computations. The UV conversion algorithm employed in this study ensures the prevention of model collapse. However, the back and forward projections bring a computational cost. This cost is due to the cross and dot products computed to locate the triangular face that the force shall lie on 3D space. The stack based approach for muscle force calculations provided a generic system. The system can deform facial models with different topologies without the need of offline pre-computations. However, since the pre-computations are not performed to mark the dynamic point masses, proposed system needs to run the stack based algorithm to find the dynamic point masses at each timestep. This re-calculations results in slower animation generation compared to the approaches that employ offline pre-computations.

Expressions	**Initial** N_T	**Final** N_T	**Computational time (s)**				**Frame-rate (fps)**
			T_d	T_s	T_a	T_r	
Happiness	1394	2832	5.76	0.05	0.42	3.42	21.6
Anger	1394	2216	4.35	0.04	0.25	2.75	27.4
Surprise	1394	2962	6.17	0.05	0.46	3.53	20.5
Sadness	1394	2527	5.22	0.05	0.34	3.10	24.8
Disgust	1394	3241	6.54	0.06	0.54	3.82	18.6

Figure 3.15: Performance values of animation system presented in [21].

Table 3.1: Elapsed Times for basic and composite facial expressions for an animation with 35 frames.

Facial Expression	Number of Activated Muscles	Elapsed Time (sec.)	FPS
Happy	6	8.078	4.33
Sad	10	9.753	3.59
Angry	7	6.823	5.13
Feared	12	9.987	3.50
Disgusted	5	3.873	9.03
Surprise	10	5.615	6.23
Disgusted Anger	9	8.298	4.22
Happy Surprise	12	11.691	2.99

Elapsed time for generation of basic and composite facial expression can be observed in table 3.1. Each generated facial expression animation consists 35 frames. Δt is set to 0.01 seconds and non-linearity factor α is set to 5. Table 3.2 displays the minimum, maximum and average elapsed times for ten consecutive runs of the system to generate six basic facial expressions.

Table 3.2: Elapsed Time statistics in 10 consecutive runs.

Facial Expression	Min	Max	Average	Average Deviation	Standard Deviation
Happy	7.772	8.134	7.982	0.118	0.134
Sad	9.705	9.915	9.799	0.063	0.073
Angry	6.689	6.851	6.789	0.043	0.053
Fear	9.847	10.252	10.100	0.100	0.127
Disgust	3.791	3.956	3.884	0.038	0.050
Surprise	5.615	5.734	5.692	0.026	0.034

When the generated Frames Per Second (FPS) values presented in figure 3.15 and the table 3.1 are compared, it is obvious that the previous work performs faster than the single layer face model animation system presented in this work. However, since both systems generates facial expression on face models that has different number of layers and deals with the challenges presented by the choice

of face model, simply comparing the resulting FPS values is not enough for evaluating the performance of the proposed animation system.

As table 3.1 suggests, the elapsed time of the facial expressions are dependent on different factors. The factors determining the elapsed time could be one or a group of following; number of activated muscles, region of influence, configured Δt value.

Table 3.2 shows the statistics of the elapsed times for 10 consecutive runs for generation of basic facial expressions with 35 frames. Fastest facial expression generation of the system is *Disgust* with an average runtime of 3.884 seconds while the slowest expression generation is *Fear* with and average runtime of 10.100 seconds. The smallest standard deviation calculated for these elapsed times is 0.034 for *Surprise* while the biggest deviation for elapsed times is observed in generation of expression *Fear* with a value of 0.127.

In general, when the number of activated muscles is increased, the animation speed slows down due to the increase in computations. However, in some cases, the elapsed time for a facial animation is faster than another facial animation even though the number of muscles activated in the first facial animation is greater than the second facial animation. This is due to the characteristics of the activated muscle. The length and the angle of influence of the muscle define the region of influence. When this region of influence is small, the number of affected vertices will be small; and if the region of influence is big, then a large set of vertices is affected. Therefore, if the regions of influence of the muscles that are activated in a facial expression cover a large set of vertices, then the generation of this facial expression is slower. An example of this can be observed in Table 3.1. The elapsed times for *Happy* and *Surprised* facial expressions are 8.078 and 5.615 seconds respectively. Even though *Surprised* facial expression contains more activated muscles than *Happy* facial expression, the generation of the *Surprised* animation is faster than the *Happy* animation.

The value of Δt is another factor affecting the elapsed time of facial expression generation. Increase in the value of Δt results in achieving the desired facial expression in less timesteps. Since one frame is generated at each timestep,less timesteps mean generation of less frames. Thus, by increasing the value of Δt, total number of calculations are decreased. Therefore, the animation generation completed in less time. However, the less the generated frames are the less smooth the animation becomes. On the other hand, it is possible to desire generation of more frames. One can decrease the value of Δt to achieve generation of more frames for obtaining a smoother animation. However, the increase in the number of generated frames results in slower animation generation due to the computation increase with the increase in the number of frames. Table 3.3 displays the elapsed times for *Happy* facial expression for varying values of Δt and total generated frame numbers.

Table 3.3: Elapsed Times for varying Δt and frame numbers.

Number of Frames Generated	Δt	Elapsed Time (sec.)
140	0.0025	31.399
35	0.01	8.078
17	0.015	3.754

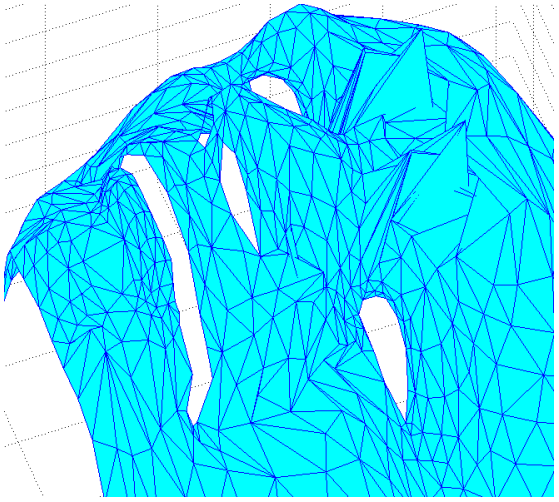

Figure 3.16: Element collapse in facial mesh.

Element collapse of the model is prevented with Edge Repulsion approach. In the absence of edge repulsion method, face model deformation occurs fast. This quick deformation continues until the individual elements of the model starts collapsing. An example of element collapse can be observed in figure 3.16. This example is observed when the computations for F_{ER} is omitted from the system. The deformation occurs quickly, in the 12^{th} timestep, due to the absence of a repulsion force.

Model collapse due to the absence of skull is prevented with a new algorithm presented in this study. The resulting facial expression animations proved the prevention of model collapse. However, introduction of back and forward projections increased the computational load of the system at runtime. Matlab Profiler points out 20% of total runtime of *Happy* facial expression generation is consumed by this UV conversion process.

Chapter 4

Conclusion

The goal of our research is to provide the most realistic looking facial expression animations. Vast range of approaches ranging from key framing and geometric approximations to physical systems that are based on facial anatomy were suggested by researchers in the literature. In this study, anatomy based approach is employed due to the potential of providing the best results. A new facial expression animation system along with two new algorithms for prevention of facial mesh collapse and locating the dynamic point masses are proposed.

In multi layer face models, skull penetration occurs when the facial mesh collapses into the skull layer due to the muscle forces applied on the point masses. This problem is generally addressed to with repulsory forces from skull layer to the facial mesh layer. In single layer face models, this approach cannot be employed due to the absence of a skull model. Thus, the first algorithm we propose is for prevention of facial mesh collapse on facial animation systems that deploy a single layer face model.

The second algorithm is proposed to locate the dynamic point masses in the region of influence of the muscles at runtime. In previous works, the dynamic point masses on the facial mesh were located with offline pre-computations and stored to be used at runtime. Thus, the process was dependent on the face model. The new algorithm allows the proposed facial expression animation system to animate any single layer facial mesh topology due to the removal of offline pre-computations.

However, due to the constant factor on the computational complexity introduced by the new algorithms, the proposed animation system does not meet the realtime requirement of animation systems.

In this study, creation of a generic animation system, prevention of model collapse in single layer face models, and efficient traversal of vertices in the influence region of muscles are explored for computer aided animation of facial expressions. Further research can be started from here to meet the realtime requirements of animation systems.

References

[1] A. Mehrabian. "Communication without words." *Psychology Today,* 2(9), pp.52-55, 1968.

[2] A. Mehrabian. *Silent Messages: Implicit Communication of Emotions and Attitudes (2nd ed.).* Belmont, CA: Wadsworth. ISBN 0-534-00910-7.

[3] "Speaktoit Asistant." *Google Play.* Internet:https://play.google.com/store/apps/details?id=com.speaktoit.assistant&feature=search_result#?t=W251bGwsMSwxLDEsImNvbS5zcGVha3RvaXQuYXNzaXN0YW50Il0, Dec. 27, 1012 [Jan. 1, 2013]

[4] J.Y. Noh, U. Neumann."A Survey of Facial Modeling and Animation Techniques." Technical Report 99-705, University of Southern California, 1998.

[5] J. Königsson. "Facial Animation." M.A. thesis, Umeå University, Sweden, 2005.

[6] P. Ekman, W. Friesen. *Manual for the the Facial Action Coding System.* Consulting Psychologist Press, Palo Alto California, 1977.

[7] S. Platt, N. Badler. "Animating Facial Expressions." *Proc. SIGGRAPH 81,* vol. 15, pp. 245-252, 1981

[8] K. Waters. "A Muscle Model for Animating Three-Dimensional Facial Expression." *Computer Graphics,* vol. 22(4), pp. 1724, 1987

[9] D. Terzopoulus, K. Waters. "Physically-Based Facial Modeling, Analysis and Animation." *J. Visualization and Computer Animation,* vol. 1, pp. 73-80, 1990

[10] Y. Lee, D. Terzopoulos, K. Waters. "Realistic Modeling for Facial Animation." *Proc. SIGGRAPH 95,* vol. 29, pp. 55-62, Aug. 1995

[11] P. Kalra, A. Mangili, N. Magnenat-Thalmann, D. Thalmann. "Simulation of Facial Muscle Actions Based on Rational Free Form Deformations." *Proc. EUROGRAPHICS 92,* vol. 11, pp. 59-69, Aug. 1992

[12] E. Goldfinger. *Human anatomy for artists: The elements of form.* Oxford University Press, 1991

[13] F.I. Parke. "Computer Generated Animation of Faces." *Proceedings ACM annual conference,* August 1972.

[14] M. Rydfalk. "CANDIDE, a parameterized face." Technical Report LiTH-ISY-I-866, Dept. of Electrical Engineering, Linkping University, Sweden, 1987

[15] B. Welsh. "Model-Based Coding of Images." PhD dissertation, British Telecom Research Lab, Jan. 1991.

[16] J. Ahlberg. "CANDIDE-3 An Updated Parameterised Face." Technical Report LiTH-ISY-R-2326 Department of Electrical Engineering, Linköping University, Sweden, 2001.

[17] MPEG WorkingGroup on Visual, International Standard on Coding of Audio-Visual Objects, Part 2 (Visual), ISO-14496-2, 1999.

[18] J. Huang, Z. Su, R. Wang. "3D Face Reconstruction based on Improved CANDIDE-3 model." *Fourth International Conference on Digital Home,* 2012.

[19] S. Pasquariello, C. Pelachaud. "Greta: A Simple Facial Animation Engine." *Proc. of the 6th Online World Conference on Soft Computing in Industrial Applications,* Sept. 2001.

[20] A. Shin, S. Lee, H. Bülthoff, C. Wallraven. "A Morphable 3D-Model of Korean Faces." *IEEE International Conference on Systems, Man, and Cybernetics,* October 14-17, 2012.

[21] Y. Zhang, E.C. Prakash, E. Sung. "A New Physical Model with Multilayer Architecture for Facial Expression Animation Using Dynamic Adaptive Mesh." *IEEE Transactions On Visualization And Computer Graphics,* 2004.

[22] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, D.H. Salesin. "Synthesizing Realistic Facial Expressions from Photographs." *Computer Graphics Proceedings SIGGRAPH98,* pp. 7584, 1998.

[23] E. Mendi and C. Bayrak. "Facial Animation Framework for Web and Mobile Platforms." *IEEE 13th International Conference on e-Health Networking. Applications and Services,* 2011.

[24] W. Zhou, N. Xiang, X. Zhou. "Towards 3D Communications: Real Time Emotion Driven 3D Virtual Facial Animation." *Workshop on Digital Media and Digital Content Management,* 2011.

[25] T. Beier, S. Neely. "Feature-Based Image Metamorphosis." *Computer Graphics Siggraph Proceedings,* vol. 26, pp. 35-42, 1992.

[26] F. Pighin, J. Auslander, D. Lischinski, D. H. Salesin, R. Szeliski. "RealisticFacialAnimation Using Image-Based 3D Morphing." Technical report UW-CSE-97-01-03, 1997.

[27] J. Noh, U. Neumann. "Expression Cloning." *Proceedings of ACM SIGGRAPH,* pp. 277-288,2001.

[28] M. Zamith, A. Montenegro, E. Passos, E. Clua, A. Conci, R. Leal-Toledo, P.T. Mouro. "Real time feature-based parallel morphing in GPU applied to texture-based animation." *IEEE Proceedings of the 16th International Workshop on Systems, Signals and Image Processing,* vol. 16, pp. 145-150, 2009.

[29] J. Areeyapinan, P. Kanongchaiyos. "Face Morphing Using Critical Point Filters." *In Computer Science and Software Engineering (JCSSE), 2012 International Joint Conference on,* pp. 283-288, IEEE, 2012.

[30] F.I. Parke. "Parameterized Models for Facial Animation." *IEEE computer graphics and applications,* pp. 61-68, 1982.

[31] M. Hoch, G. Fleischmann, B Girod. "Modeling and animation of facial expressions based on B-splines." *The Visual Computer,* vol. 11.2, pp. 87-95, 1994.

[32] E. Akagündüz, I. Ulusoy, N. Bozkurt, U. Halc. "A Physically-based Facial Skin Model to Simulate Facial Expressions on Digitally Scanned 3D Models." *In Computer and Information Sciences, 2007. ISCIS 2007. 22nd International Symposium on,* pp. 1-5, 2007.

[33] T.W. Sederberg, S. R. Parry. "Free-form deformation of solid geometric models." *ACM Siggraph Computer Graphics,* vol. 20.4, pp. 151-160, August 1986.

[34] W.S. Lee, N. Magnenat-Thalmann. "Fast Head Modeling for Animation." *Image and Vision Computing,* vol. 18.4, pp. 355-364, 2000.

[35] M. Nahas, H. Huitric, M. Rioux, J. Domey. "Facial image synthesis using skin texture recording." *The Visual Computer,* vol. 6.6, pp. 337-343, 1990.

[36] M. Nahas, H. Huitric, M. Saintourens. "Animation of a B-Spline Figure." *The Visual Computer,* vol. 3.5, pp. 272-276, 1988.

[37] C.L.Y. Wang, D.R. Forsey. "Langwidere: A new facial animation system." *In Computer Animation'94., Proceedings of,* pp. 59-68, IEEE, 1994.

[38] K. Singh, E. Fiume. "Wires: a geometric deformation technique." *Proceedings of the 25th annual conference on Computer graphics and interactive techniques,* pp. 405-414, ACM, 1998.

[39] T.D. Bui. "Creating Emotions and Facial Expressions for Embodied Agents." PhD dissertation, University of Twente, Netherlands, 2004.

[40] I.S. Pandzic, R. Forchheimer, A. Pakstas. "MPEG-4 Facial Animation: The Standard, Implementation and Applications.", 2002.

[41] T. Goto, M. Escher, C. Zanardi, N. Magnenat-Thalmann. "MPEG-4 Based Animation with Face Feature Tracking." *Proceedings Eurographics Workshop Computer Animation and Simulation 99,* pp. 89-98, 1999.

[42] P. Dai, G. Xu, T. Riegel, E. Hundt. "Automatic face modeling and synthesis based on image pairs." *Proceedings of the Fourth IEEE International Conference on Computer Vision Systems,* pp. 18-24, 2006.

[43] N. Patel, M. Zaveri. "3D Facial Model Construction and Expressions Synthesis From a Single Frontal Face Image." *International Conference on Computer and Communication Technology (ICCCT),* pp. 652-657,IEEE, 2010.

[44] J. Sheu, Y. Wu, Y. Chuang,Y. Hsiao, C. Chen. "Production of Facial Expressions Using Facial Feature Positioning and Deformation." *International Symposium of Industrial Electronics (ISIE),* pp. 1158-1163, IEEE, 2012.

[45] B. Yamina, M.H. Farida. "MPEG4 Parameterization for Facial Deformation." *International Conference on Multimedia Computing and Systems (ICMCS),* pp. 414-419, IEEE, May 2012.

[46] I.A. Essa. "Coding, analysis, interpretation, and recognition of facial expressions." *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 19.7, pp. 757-763, July 1997.

[47] Y. Wu, N.M. Thalmann, D. Thalmann. "A Dynamic Wrinkle Model in Facial Animation and Skin Aging." *Journal of Visualization and Computer Animation,* vol. 6, pp. 195-202, 1998.

[48] Y.C. Fung. Biomechanics: Mechanical Properties of Living Tissues. New York:Springer-Verlag, 1993.

[49] Y. Zhang, E. Sung, E. Prakash. "Animation of Facial Expressions by Physical Modeling." *Proceedings of EUROGRAPHICS 2001, short presentations,* pp. 335-345, 2001.

[50] Y. Zhang, E. Sung, E. Prakash. "A physically-based model for real-time facial expression animation." *Proceedings of Third International Conference on 3-D Digital Imaging and Modeling,* pp. 399- 406, IEEE, 2001.

[51] Y. Zhang, E. Sung, E. Prakash. "Hierarchical Face Modeling and Fast 3D Facial Expression Synthesis." *Proceedings of the XV Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI02),* pp. 357364, October 7-10, 2002.

[52] Y. Zhang, E. Sung, E. Prakash. "Efficient Modeling of an Anatomy-based Face and Fast 3D Facial Expression Synthesis." *Computer Graphics Forum,* vol.22.2, pp. 159170, 2003.

[53] T.H. Bui, D. Heylen, A. Nijholt. "Improvements on a Simple Muscle-Based 3D Face for Realistic Facial Expressions." *Proceedings of the 16th International Conference on Computer Animation and Social Agents (CASA 2003),* pp.33, May 08-09, 2003.

[54] M.T. Eskil, M. Özgüz, D. Aydoğan, K.S. Benli. "Semi-Automatic Adaptation of High-Polygon Wireframe Face Models Through Inverse Perspective Projection." *Proc. of International Symposium on Computer and Information Sciences (ISCIS 2011),* London, September 2011.

[55] L. Cooper, S. Maddock. "Preventing Collapse within Mass-Spring-Damper Models of Deformable Objects." *The Fifth International Conference in Central Europe on Computer Graphics and Visualization,* 1997.

[56] N. Molino, R. Bridson, J. Teran, and R. Fedkiw. "Adaptive Physics Based Tetrahedral Mesh Generation Using Level Sets." *Eng. with Computers,* vol.21, pp. 2-18, 2005.

Curriculum Vitae

Tuğba Erkoç was born in 14 June 1987, in Istanbul. She received her B.S. degree in Computer Engineering in 2010 from Işık University. She has been working as an R&D Engineer at Netaş Telecommunications since 2010. Her research interests are image processing, computer graphics, computer based animation.