FACIAL MODELLING AND ANIMATION TRENDS IN THE NEW MILLENNIUM: A SURVEY

by

MAURICIO RADOVAN

Submitted in fulfilment of the requirements for

the degree of

MASTER OF SCIENCE

in the subject

COMPUTER SCIENCE

at the

UNIVERSITY OF SOUTH AFRICA

SUPERVISOR: PROF L PRETORIUS

NOVEMBER 2008

Student number: **3068-775-6**

I declare that FACIAL MODELLING AND ANIMATION TRENDS IN THE NEW MILLENNIUM: A SURVEY is my own work and that all the sources that I have used or quoted have been indicated and acknowledged by means of complete references.

_____ _____

SIGNATURE DATE

(MR M RADOVAN)

Summary

Facial modelling and animation is considered one of the most challenging areas in the animation world. Since Parke and Waters's (1996) comprehensive book, no major work encompassing the entire field of facial animation has been published. This thesis covers Parke and Waters's work, while also providing a survey of the developments in the field since 1996. The thesis describes, analyses, and compares (where applicable) the existing techniques and practices used to produce the facial animation. Where applicable, the related techniques are grouped in the same chapter and described in a chronological fashion, outlining their differences, as well as their advantages and disadvantages. The thesis is concluded by exploratory work towards a talking head for Northern Sotho. Facial animation and lip synchronisation of a fragment of Northern Sotho is done by using software tools primarily designed for English.

Key terms:

Computer Graphics; Facial Animation; Lip Synchronisation; Computational Geometry; Object Modelling; Survey; Northern Sotho Talking Head

Acknowledgements

A single most important person I need to thank for the assistance and support during my masters studies is my supervisor, Professor Laurette Pretorius. First and foremost, I thank her for believing in me and allowing me to be her student. I also wish to thank Professor Albert Kotzé for his linguistic contribution to our experiment in Chapter 10. Lastly, I wish to thank Beverley Kempster for editing the thesis.

Table of contents

List of figures

PART I

Contextualisation

Chapter 1 Introduction

1.1 Background information

Facial modelling and animation refer to techniques of representing the face graphically on a computer system and animating the face in a manner consistent with real human movement. This is often considered one of the most challenging tasks undertaken in the field of animation, due to several factors. The first of these is that, because most of us experience so many natural human interactions every day, humans are skilled at identifying unnatural facial movements. Thus the slightest inconsistency in an animated face immediately alerts the viewer and the animation loses its realism. The human face is an incredibly complex system of a relatively large number of muscles that need to be perfectly coordinated in order to look realistic. Another factor that contributes to the difficulty of modelling and animation of the human face is its diversity. Different people have different facial features, caused by different bone structures and muscle sizes and proportions.

The ultimate goal of research in facial modelling and animation can be summarised as a system that creates realistic animation, operates in real time, is as automated as possible, and adapts easily to individual faces (Noh and Neumann, 1998).

Main drivers in the development of computer facial animation have been and still are the entertainment industry with its feature films, computer animated characters who have become household names and computer games; the broad field of visualisation in science, including medical science and forensic analysis; and the fast growing field of information and communication technology and human-computer interaction. Applications can be divided into interactive systems such as dialog-based interfaces, inhabited virtual worlds, animated pedagogical and/or conversational agent systems, and games that require real-time animation with limited realism; and off-line systems, including medical applications, forensic analysis and feature films, where realism and accuracy are paramount.

The field is inherently interdisciplinary, with computer science and 3D graphics acting as the ‘scientific glue’ that brings all the relevant fields together. In order to understand some of the facial animation concepts, one would require a substantial background in human anatomy. To model a human head, one needs to be familiar with the shapes of the bones that make up the human skull.

Another important discipline in this field is histology. To accurately simulate the skin's response to stress and deformation, we would need to know its anatomy and mechanical properties. The field of facial animation also penetrates into the studies of psychology. To create a believable facial animation, we need to know how other people perceive talking heads. Also, we would need to know how to synchronize all other facial expressions that normally manifest themselves along with speech, such as eye blinking, frowning, nodding, and others. One of the latest and most advanced trends in computer animation in general is simulation of the Newtonian physics that acts on animated objects. The programmer would place the objects in the 'world' and specify the constraints. Typical constraints would be masses of the objects and their interrelationships, such as springs or hinges. The world would then be in equilibrium, waiting for a force to be applied. Once the force is applied, the objects would react in the same way as if the same thing had happened in the real world under the same circumstances. This would continue until a new equilibrium is reached.

Applying these principles to facial animation, the head would wait for the synthetic muscle contraction force to form the desired expression. This requires extensive knowledge of physics, particularly kinematics and kinetics. Facial animation is tightly connected with speech synthesis and speech and expression recognition. This area penetrates into the field of artificial intelligence, as many of the automated lip synchronisation methods involve machine learning and neural networks. For the purpose of expression recognition and learning, most authors use machine vision principles.

Finally, but certainly not least importantly, facial animation and modelling have to take cognisance of human speech, and by implication, human language and linguistics. Moreover, in order to address the challenge of multiple cultures and multiple languages, both the visual and auditory aspects of the different languages and their importance for multilingual animation need to be investigated.

1.2 Problem statement

While the bleeding edge of these technologies with its associated intellectual property and copyright restrictions are not readily accessible in the research literature, there is an increasing interest in the techniques and technologies for facial modelling and animation in the broader scientific community. This was one of the main reasons why Noh and Neumann (1998) compiled their survey of 'theoretical approaches used in published work'.

However, vast strides have been made in this field in the ensuing decade, making affordable desktop processing and sophisticated animation toolkits more available to all. It is, therefore, the aim of this thesis to

- Provide an updated overview of the historically used technologies.
- Review state-of-the-art approaches in the field of facial animation on the basis of the published research.
- Venture into the animation of one of the Bantu languages, namely Northern Sotho.

Due to the vastness of the field, the scope of this thesis is limited to computer graphics as a sub-discipline of computer science, with a broad overview of the other disciplines. Facial animation techniques could be roughly divided into geometry deformation and image deformation approaches (Noh and Neumann, 1998). This thesis focuses on the geometry deformation techniques, while it briefly overviews the image deformation techniques. Within the geometry deformation techniques, more attention is paid to the currently used ones, representing the state of the art of the field. Where applicable, the advantages and disadvantages of each technique are outlined, and techniques compared with each other. Also, the future prospects of each particular technique are explained, along with the future research work as proposed in the scientific literature.

Northern Sotho, a Bantu language spoken in Southern Africa, is one of the eleven official languages of South Africa. In terms of language technology, it is considered a lesser-studied language. This thesis finally reports on a first and novel attempt at the animation of Northern Sotho by investigating the suitability of a commercial animation toolkit, primarily developed for English, for a language such as Northern Sotho.

1.3 Research objectives

Summarising, the purpose of this thesis is twofold. Firstly, it provides an updated overview of aspects of the field of facial modelling and animation. This is achieved by collecting, synthesising and comparing the available scholarly work in the field of facial animation. A brief summary of the associated topics from other fields of science are provided, where considered necessary.

Secondly, it investigates the application of a state-of-the-art commercial industry standard animation toolkit, designed for the English language, to the development of a Northern Sotho animation.

1.4 Methodological issues and challenges

Since this thesis has two quite different objectives and poses associated research questions, it is appropriate to briefly reflect on a suitable choice of research strategy or methodology in each case.

1.4.1 Research strategy for the overview

According to Mouton (2001), the research strategy for addressing the first research objective may be described as a study 'that provides an overview of scholarship in a certain discipline'. It is characterised as *non-empirical*, using *secondary* data. The selection and representativeness of the sources are therefore 'an important criterion of the final quality of the literature review' (Mouton, 2001:180). It is for this reason that we briefly address this issue.

The literature in terms of books is limited to three or four at most. The most helpful one was 'Computer Facial Animation' (Parke and Waters, 1996). The two authors made a monumental contribution to the field through their research. Fred Parke is regarded as the father of computer facial animation; the whole field seems to have evolved from his ideas back in 1972.

Their book not only details the results of their own research, but also outlines the most important achievements of other scientists in the field, and does this so comprehensively that there are very few significant contributions to facial animation that are not mentioned. It seemed only natural to explore the field using the pointers provided in the book and its references for the background of this thesis.

Advancements in the field between 1996 and 2007 are primarily obtained from the large body of research articles published in journals and conference proceedings.

There is a significant disparity between academic works and the entertainment industry. Companies such as Pixar, Dreamworks Studios or Alias-Wavefront have their own groups of animation researchers that work behind closed doors and do not publish their latest results. While this is understandable, since these results provide them with their competitive advantage, it also complicates any scientific assessment of the state of the art. Sir Isaac Newton once said 'If I have seen further, it is by standing on the shoulders of giants' (Isaac Newton, letter to Robert Hooke, 1676). In the field of animation, however, the giants do not allow the others stand on their shoulders.

Even academic publications explain only concepts; the source code is seldom released. The only significant pieces of code which are freely available are Parke's parametric model and Waters's linear muscle-based model. While Parke's model is obsolete by today's standards, Waters's model still

forms the basis of the majority of current facial animation research attempts and was well worth analysing. Unfortunately, there seems to be no source code of any of the late enhancements of this model available. This thesis therefore explains the concepts described in the publications, mostly without the benefit of access to the supporting code.

Most of the other references used in the thesis were obtained in an electronic format from the Internet. UNISA has access to two main sources of computer science journals, ACM (http://www.acm.org) and IEEE (http://www.computer.org/portal/site/csdl/index.jsp), which constitute authoritative, reliable and comprehensive sources in the field. I also found Google Scholar (http://scholar.google.com) superior to any other journal or citation search tool and invaluable to my research.

1.4.2 Research strategy for the artefact

The second objective and research question is addressed by means of the so-called design and creation research strategy (Oates, 2006:108). This strategy 'focuses on developing new IT products also called artefacts'. It is *empirical* in nature (Mouton, 2001:163) and makes use of both secondary (previous research and documentation) and primary data in the form of audio-visual recordings of a Northern Sotho speaker.

Main challenges in this research strategy are the explicit description, justification and use of a systematic and scientific development process, including the data collection; and the evaluation of the developed IT artefact – in this case the Northern Sotho talking head prototype.

Indeed, an important constraint of facial animation in general lies in the objectivity of evaluating the final results. There is often no suitable benchmark, as the final result borders on visual art. While one could calculate and compare the rendering speed, it is sometimes difficult to judge how realistic an animation appears to humans.

1.5 Thesis structure and chapter overview

The thesis is divided to four main parts, as shown in Figure 1. *Contextualisation* provides background information, problem statement, research objectives, methodology, research question and significance of the thesis. A broad overview of the field follows in Chapter 2 covering both geometry and image deformation. This includes a brief explanation of the taxonomy of the field, and a description of the historically and currently used technology, followed by speculation concerning future trends. Since the thesis itself concentrates solely on geometry-based approaches, there was a need to provide a brief overview of image-based technologies and convey their significance in the field.

In the section titled *Research Question 1*, geometry-based technologies are explained in a far greater detail, covering the period of time between 1970s to roughly 2007. To facilitate understanding of the matter, the necessary fundamentals of the related disciplines are provided in Chapters 3 to 9.

Research question 2 covers the detailed steps taken in creation of the first prototype of a Northern Sotho animated talking head, from choice of the sentence, through finding a speaker, recording and processing the media, to finally producing the animation in the format playable on a personal computer. This is discussed in Chapter 10.

The final part of the thesis is the conclusion (Chapter 11), which contains a discussion on the presented content and future work in the field.

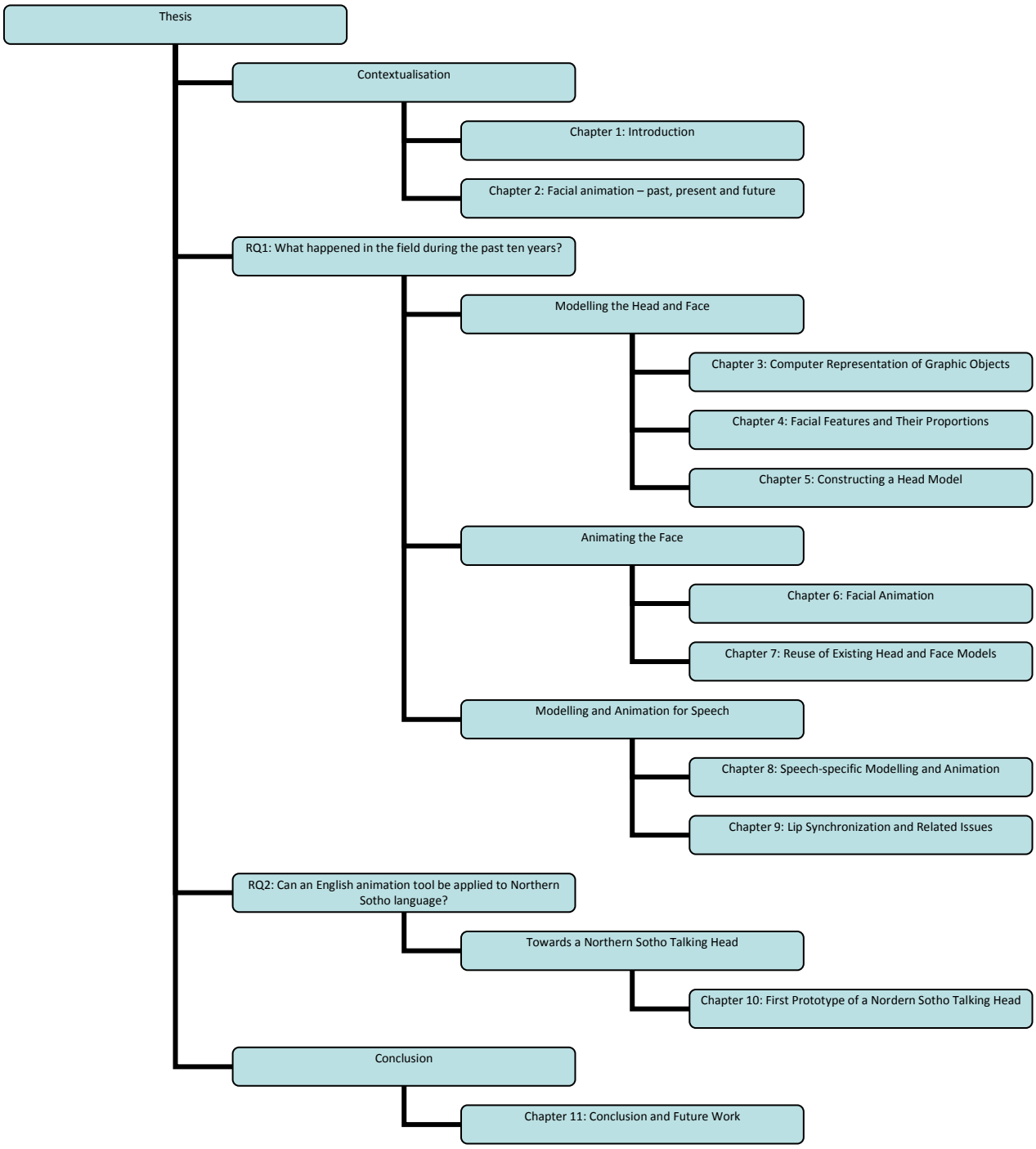


Figure 1: Chapter overview.

1.5.1 Addressing research question 1: what happened in the field for the past ten years?

1.5.1.1 Modelling the head and face

Intuitively, the first step would be to create a 3D computerised representation of a head; one that is photorealistic and indistinguishable from a real head. Let us start by erasing all our prior knowledge about computer graphics and try to think from the perspective of a scientist who is new to the field.

The first question that comes to mind is how to mathematically represent a 3D head. The basic concepts required to do this are described in Chapter 3. Having decided upon surface representation as a way forward, the natural progression would be to create a head. The intuitive method would be to position vertices in the shape of the head, connect the appropriate vertices with lines (edges) and fill the polygons bounded by the edges, forming so called faces. It soon becomes apparent that doing this manually, by hard-coding the coordinates in a programming language, would be an unrealistic task. Finding an easier and more productive method is thus required. Sculpting methods (see also Section 5.4) are manual and simple to grasp, but tend to be laborious and thus inadequate for today industry's animation needs. They require an artist to sculpt the head manually using a 3D software modelling tool. To perform such sculpting, it is helpful to know the proportions of various parts of the human head, described in Chapter 4. Although many artists still prefer these methods, scientists continued researching various ways of automating the modelling tasks. The results of this research are techniques such as contour reconstruction, 3D digitizers and photogrammetric techniques, also described in Chapter 5. Once complete, we have a static representation of the head.

1.5.1.2 Animating the face

The next step would be to animate such a head. The first question that needs to be answered before proceeding is whether the head needs to be animated in real-time, or whether the animator would have the luxury of unlimited rendering time. This is important, since the more time the animator has, the better the results. Next, various animation techniques to create the fundamental movements (Sections 6.2.1 to 6.2.5) need to be explored. One of the most advanced approaches in facial animation is physics-based (Section 6.2.5). It takes into consideration most aspects in human anatomy, in order to simulate the actual behaviour of joints and muscles involved. This approach requires a significant knowledge of human anatomy, including the histology and mechanical properties of the soft tissue (Section 6.2.4).

Having created the basic movements, the next challenge is to animate natural facial expressions. There are a fair number of muscles that contribute to every expression produced, just to guess how they work would not be acceptable. To assist the animator with this predicament, two fundamentally different methods are available. One captures the facial movements of a real person, writes them in the form of numbers in terms of spatial and temporal coordinates, and then reproduces them with a synthetic character. This method, known as performance-driven animation is covered in Section 6.1.1. The idea behind the second method is to identify which muscles contribute to a specific expression, to what extent and in what sequence.

There are extensive studies on how the muscles interact in producing facial expressions and it makes sense to use one of these in order to save time (Section 6.1.2). Some such research results provide the animator with muscle actions for every single phoneme in a language, which can then be used for lip synchronisation. Due to the inherent complexity of producing an animated talking head, it is clearly counterproductive to manufacture every head in an animated movie from first principles. That is why considerable research efforts were made to reuse existing head and face models, as described in Chapter 7.

1.5.1.3 Modelling and animation of speech

Because of its specific requirements over and above the animation of the rest of the face, speech animation is a field on its own. The anatomy of the speech related organs and body parts are briefly touched upon in Section 8.1. Section 8.2 provides a short introduction to phonemes and their visual representation. The modelling and animation of speech-specific body parts are discussed in Sections 8.3 to 8.5. Once the animator knows how to animate for speech, she soon realises the need for synchronisation of the visual and the audible representation. If the audible and visual speech are not in synch, the entire experience is unrealistic, due to human bimodal perception. Lip synchronisation can be done either manually or using one of the several automated methods, as described in Sections 9.1 and 9.2. Completion of the lip synchronisation does not conclude the efforts. The transition between phonemes does not look realistic. This is due to the mouth shape of a phoneme depending to a degree on preceding and succeeding phonemes. This phenomenon is called coarticulation (Section 9.3).

1.5.2 Addressing research question 2: can an English animation tool be applied to Northern Sotho language?

Chapter 10 is a novel attempt to create a speech synchronised animation in Northern Sotho language using 3D Studio Max. Linguistics information was provided by Northern Sotho linguistics expert, Prof AE Kotze, who outlined the envisaged challenges and selected a sentence used for the experiment. A native Northern Sotho speaker was then located and his performance recorded on a video clip. After the preparations, the animation was created in accordance with the recommendations by renowned industry professionals (Fleming and Dobbs, 1999).

1.6 Significance

This work has already resulted in *two publications*: Radovan and Pretorius (2006) and Radovan, Pretorius and Kotze (2007), and in a *software artefact* in the form of a Northern Sotho talking head. These three outcomes of the thesis can be found on the attached CD.

Radovan and Pretorius (2006) provided an overview of the historically-used technologies and attempts to predict the future trends in the field of facial animation as it permeates the entertainment industry with its feature films and computer games; the broad field of visualisation in science, including medical science and forensic analysis; and the fast growing field of information and communication technology and human-computer interaction. The content of this paper is presented in Chapter 2. We are of the opinion that surveys in this field are still needed, due to its size, scope, complexity and its multidisciplinary nature.

Radovan, Pretorius and Kotze (2007) presented an experiment in which a software system geared for production of speech animation in the English language, was used to produce animated speech in Northern Sotho. According to authors' records, this is the first time such an experiment has been performed and published in an academic publication. The content of this paper is presented in Chapter 10.

Finally, the thesis provides an *extended and up-to-date survey* of the field of geometry-based facial animation. The main shortcoming of survey papers is often their limited length and therefore limited coverage. A conference or a journal imposes a limit on the number of pages, while a survey by its very nature demands a more extensive coverage of the field. This thesis is not restricted in that respect, and it could therefore afford to provide a comprehensive coverage of the field, along with the necessary basics of the related disciplines.

Chapter 2 Facial animation: past, present and future[1]

2.1 Background

Facial animation approaches can be grouped into two groups, namely those based on geometry manipulation and those based on image manipulation (Noh and Neumann, 1998; Krinidis, Buciu and Pitas, 2003).

Geometry manipulation refers to manipulation of 3D models that consist of vertices in space forming polygons and thus representing the surface of the model. Image manipulation refers to 2D images or photos that are morphed from one to another in order to achieve the desired animation effect.

Beyond this classification it becomes more difficult to classify further, as there are often no well defined boundaries between the technologies (see also Deng and Noh, 2008: Chapter 1). It frequently happens that a newer technique uses methods introduced by the older technique, or they have perhaps both been used for two different parts of the face. It is, however, generally agreed that one could divide the geometry manipulation methods roughly into key-framing, parameterisation, pseudo-muscle methods, and physics-based methods (Tang, Liew and Yan, 2004); that image manipulation methods include morphing and blendshaping (Deng et al., 2006); and that performance-driven animation (also referred to as expression mapping) assumes a performer and makes appropriate use of both geometry-based and image-based techniques to do the animation (Zhang et al., 2006).

It should be noted that although automated facial expression analysis and synthesis/generation are often discussed together, the issues that need to be addressed, as well as the techniques and approaches, may vary (Krinidis, Buciu and Pitas, 2003; Ravyse, 2006). Indeed, facial modelling and animation largely corresponds to facial expression synthesis or generation, as is also confirmed by, for example, the brief taxonomy provided by Krinidis, Buciu and Pitas (2003) and by Terzopoulos et al. (2004). Computer vision with its focus on face analysis falls outside the scope of this overview.

It is our impression that facial animation is in many ways a superset of general body animation. In muscle-based modelling and animation in particular, general body animation and facial animation

---

[1] This chapter is based on an earlier work: Radovan, M. and Pretorius, L. (2006). Facial animation in a nutshell: past, present and future. *In: Proceedings of SAICSIT 2006*, Somerset West, South Africa, October 9-11 2006. ACM Press, pp 71 – 79.

have a number of similar research problems, including muscle and soft tissue deformation. The differences, however, include the following:

- The number of muscles participating in a facial expression is far greater;
- Soft tissue deformation needs to be much more realistic, due to its constant exposure to our scrutiny (see Section 6.2.4.1);
- The head contains one single joint, albeit more complex than any other joint in a human body, due to the number of degrees of freedom (see Section 8.3).

Both the finite-element method and the mass-spring approaches have their general body animation counterparts (see Sections 6.2.5.4 and 6.2.5.2), if not predecessors (Teran et al., 2005; Maurel, 1999).

2.1.1 Geometry-based approaches

2.1.1.1 Key-framing with interpolation

Key-framing methods with interpolation are by far the simplest and oldest approach. These methods entail taking a vertex in 3D space and moving it to another specific location. The computer computes the points in between and moves the vertex along the computed points as a function of time. If such a vertex forms part of a surface polygon, there would be an illusion of a deformation of the surface. Examples of such animation are described in the pioneering work of Parke (1972). He represents the facial surface with a relatively small number of polygons, ensuring that the edges and vertices are consistent with the animation requirements. While this sort of preparation of explicit data for each key-frame was satisfactory in 1972, today's demand for realism is much greater, resulting in the number of vertices growing by several orders of magnitude. For this reason, manufacturing each key-frame is no longer feasible.

2.1.1.2 Direct parameterisation

Parke continued working on the reduction of parameters and derived a method, which he called direct parameterisation (Parke and Waters, 1996:187-222). Using this method, the face is still represented by a polygonal surface, but the motion may be represented by a far smaller set of parameters. The underlying animation principles are still based on key-framing and interpolation. Although relatively advantageous when compared to plain key-framing and interpolation, direct parameterisation brings with it a new set of problems. A set of parameters is bound to a certain facial topology. In order to

create a different face, the set of parameters needs to be rewritten. Also, there is no mechanism to deform the skin under pressure. The most significant problem is the occasional conflict between parameters, which causes the expression to look unnatural.

2.1.1.3 Pseudo-muscle-based approach

As computer hardware evolved, new and more computationally intensive techniques became viable. The pseudo-muscle-based approach is a far more computationally intensive process than the two described previously. The muscle actions are simulated using geometric deformation methods, such as freeform deformations (Sederberg and Parry, 1986) and rational freeform deformations (Kalra et al., 1991a and 1992). An example of using rational freeform deformations for animation purposes is described by Chadwick, Haumann and Parent (1989). This technique produces better results than both key-framing and direct parameterisation, but disregards the subtle movements on the skin surface, such as wrinkles and bulges. The fundamental difference between this approach and its muscle-based successor in Section 2.1.1.4 is that pseudo-muscle merely simulates the effect of the muscle action on the skin surface while the muscle-based approach simulates the actual muscles, deforming the skin in accordance with their actions.

2.1.1.4 Muscle-based approach

The earliest attempt at a muscle-based model was that of Platt and Badler (1981). They constructed a mass-spring model of a muscle fibre as one of the abstraction layers in their system. Muscle-based approaches are characterised by simulating muscles and muscle actions consistent with the actual muscles in the human body. The next significant milestone in muscle-based approaches was by Waters (1987). He defines three different muscle types by the nature of their actions, namely linear, sheet and sphincter. Using his system, the muscles do not depend on the bone structure, enabling them to be ported to diverse facial topologies. Most physics-based models today are still built using Waters's basic principles.

2.1.2 Image-based techniques

According to Noh and Neumann (1998), image-based techniques could be suitably divided into morphing between photographic images, texture manipulation, image blending and vascular expressions. An early example of morphing between two images is seen in the work of Beier and Neely (1992). Pighin et al. (1998) combine 2D morphing techniques with 3D transformations to automatically produce highly realistic 3D facial expressions. Their image-manipulation work is an

example of image blending techniques. Texture manipulation is described in Oka et al. (1987) where they demonstrate a real-time texture mapping system of face photographs onto a mesh.

Vascular expression technique has not been widely adopted. It is not a facial animation technique in its own right, but is used to complement other techniques, enhancing the realism by modifying the skin colour according to emotions. Its most notable representative is the work by Kalra and Magnenat-Thalmann (1994).

Bregler, Covell and Slaney (1997), Ezzat, Geiger and Poggio (2002) and Cosatto and Graf (2000) are considered to be the most prominent researchers of their time in the field of image-based facial animation techniques (Ostermann and Weissenfeld 2004). Bregler, Covell and Slaney (1997) introduced a video-rewriting technique, where new mouth positions are synthesised and stitched onto the background image, creating the illusion of the speaking character. Ezzat, Geiger and Poggio (2002) followed a similar approach, with several improvements and reduced size of the expression database. Cosatto and Graf (2000) showed the best photorealistic results of the three techniques under review, in addition to being able to synthesise the animation in real-time.

2.1.3 Performance-driven methods

Performance-driven methods consist of capturing actual performers' movements and actions in order to use them to animate synthetic characters. These methods are therefore largely data-driven, capturing data by means of a multitude of facial markers. They are considered as a class on their own as they are independent of the implementation. That is, they could be implemented using both image and geometry manipulation. The earliest attempts at this method are traced back to the mid-eighties (Parke and Waters, 1996:111) for cartoon creation purposes.

Williams (1990a) was the first to synthesise expressions by changing the 2D texture coordinates using the differences between static images. Guenter et al. (1998) went a step further and derived their data from a video stream. Kouadio, Poulin and Lachapelle (1998) use pre-modelled 3D facial expressions and blending between them to produce a real-time animation. Pandzic et al. (1998) succeeded in doing away with the ubiquitous facial markers by using an edge extraction algorithm to acquire the performance data in real-time.

2.1.4 Expression coding systems

Generally speaking, expression coding systems may be considered as an alternative technology to performance-driven approaches. They provide a database of muscle actions for all possible facial expressions. The animator can then simply compile a script of desired expressions over time and allow the system to animate them. The most popular expression coding systems are Facial Action Coding System (Ekman, Friesen and Hager, 2002) and MPEG-4 models (Doenges et al., 1997 and Ostermann, 1998).

2.2 Present trends

We can safely conclude that purely manual key-framing and direct parameterisation are technologies of the past. They were adequate in their own time period, due to inferior computer hardware, when one could not afford to waste clock cycles and had to cut all possible corners to achieve the desirable effect. Even the most elementary techniques, such as smooth shading, had to be implemented manually through software. Such techniques are now significantly faster and implemented directly in the graphics hardware. They are also easily accessible and abstracted within graphics libraries, such as OpenGL and DirectX. Direct parameterisation has a relative advantage over pure key-framing because it dramatically reduces the amount of data required for the animation. However, frequent undesirable effects due to parameter clashing and the lack of skin deformation mechanisms cause scientists and developers to avoid this technique. The pseudo-muscle-based approach is viewed as little more than a bridge between earlier attempts and approaches and the muscle-based approach.

2.2.1 Muscle-based approach

While the development of the first pure muscle-based facial model by Waters (1987) constituted a significant milestone and was fairly advanced for its time, the model is relatively simple by modern criteria. It represents the skin as a geometric surface with no underlying structure. The deformations are implemented by means of a simple geometric distortion of the surface, which fails to reproduce subtle tissue deformation. Terzopoulos and Waters (1990) alleviate some of the mentioned problems by introducing anatomically-based arrangements of muscle models, along with a physically-based tissue model. This tissue model allows for more realistic surface deformations than the previous attempts. Zhang, Prakash and Sung (2001) model the skin using non-linear spring frames that are able to simulate the dynamics of real skin. The advantage of this approach is that the model does not need to be treated as a continuous surface since each mass point and each spring can be accessed individually.

Improved muscle action control was the goal of Pasquariello and Pelachaud (2001) and Bui (2004). This was achieved by dividing their respective models into a number of areas. For skin simulation purposes, both Pasquariello and Pelechaud (2001) and Bui (2004) diverted from the physics-based approach. Although Pasquariello and Pelechaud animated the skin in a realistic way, they did not use physical simulation of muscles and the visco-elastic behaviour of the skin. The two alternative techniques they used to simulate furrows, bulges and wrinkles were bump mapping and physical displacement of vertices.

Alternatively, Bui created wrinkles by displacing the affected vertices in the direction of the normal to the direction of muscle action. He also addressed the artefacts that occur on the skin surface under the influence of two or more of Waters's (1987) vector muscles. Waters's way of handling this was to add the displacements sequentially. Bui proposed simulating parallelism by calculating the resultant displacement internally, then applying it to the vertex.

Tang, Liew and Yan (2004) introduced a NURBS muscle-based system, defined by three to five control points. Using this system, muscle deformation is achieved by modifying the weighting of these control points. Internally, the weight modification forces the knots to move, which in turn moves the vertices of the model. To enhance realism, the authors attempted to simulate the fatty tissue reaction to deformation by adding control points between the two end control points. A promising anatomical model, offering unique versatility, was also described by Kahler, Haber and Seidel (2001) and Kahler et al. (2002). This model fitted the muscles and calculated the skull mesh based on the face geometry, thus greatly reducing manual intervention. A result of this technique can be seen in Figure 85 (pp 123), where a model of a boy has been automatically adapted to his different ages. The animation is based on a mass-spring system.

Sifakis et al. (2005) constructed an anatomically accurate facial muscle model, using the principles derived from the more general muscle construction principles of Teran et al. (2005). Their technique is based on finite element algorithms (for a detailed exposition of finite elements, the reader is referred to Fish and Belytschko, 2007). A significant feature of their model is that its muscle action can interact with the environment, that is, the muscle forces can be combined with external forces such as collision, producing the resultant effect shown in Figure 66 (pp 103).

Mass-spring and finite element algorithms seem to be the two dominating technologies for muscle-based animation today. The two schools of thought co-exist and superiority of one over the other has yet to be established.

2.2.2 Image-based techniques

Image-based techniques still remain the methods of choice in the movie industry, due to that industry's photorealistic requirements. Motion picture special effects are usually subject to post-production, so quality takes precedence over rendering speed. An example of sophisticated image-based modelling and animation techniques are those used in the movie Matrix Reloaded, as described in Borshukov and Lewis (2003) and Borshukov et al. (2003).

Another technique that may be classified as belonging to the image-based group of methods, and that has survived to this day, is so-called 'blendshape interpolation'. The basic principle consists of a series of static photos and the interpolation between them. Important recent work on blendshape modelling was carried out by Joshi et al. (2003), Lewis et al. (2005) and Deng et al. (2006).

Joshi et al. (2003) designed a method for automating the blendshape segmentation, greatly reducing the amount of manual work required. Due to the complexity of the human face, the blendshapes have to be segmented into smaller regions. Lewis et al. (2005) presented a new algorithm for solving the problem of blendshape interference. This undesirable effect appears when two interacting parameters are individually adjusted, and subsequently interfere with one another throughout the process.

Deng et al. (2006) describe a semi-automatic method of cross-mapping of facial data to pre-designed blendshape models. They also improve on the blendshape weight-solving algorithm.

2.2.3 Performance-driven methods

In performance-driven animation there are several distinguishable data acquisition research problems. Regarding implementation, the research problems are 'shared' between image- and geometry-based animation techniques. Here it should be mentioned that performance-driven techniques sometimes still use key-frames and interpolation. One of the main reasons for this is the fact that the large amount of key-frame data, which would be a major issue if constructed manually, can now be derived via an automatic acquisition method.

At the top of the list of the abovementioned performance-related problems is the issue of perfecting the method of capturing the performance data, that is reducing the human intervention to the minimum, to diminish or eliminate face markers and to reduce the need for custom acquisition hardware. Borshukov et al. (2003) use an optical flow and photogrammetric technique to record a live actor's performance. Optical flow refers to a technique of tracking each pixel in time using multiple cameras. The spatial position of each pixel can later be determined using triangulation. Blanz et al.

(2003) combine image- and geometry-based technologies to augment the performance by simulating motion that has not yet been performed.

Zhang et al. (2004) designed a system using several video cameras positioned around the subject (performer) at an angle. No facial markers were used, so that the footage is also suitable for texture and lighting purposes. Video cameras are relatively inexpensive and non-intrusive acquisition hardware. Once the videos have been produced, the computer derived the geometry of the subject using machine vision techniques. Zhang et al. (2006) also combined image- and geometry-based technologies, but for the purpose of simulating subtle facial details – such as wrinkles – that cannot be identified through performance.

Gutierrez-Osuna et al. (2005) created an interesting mixture of existing approaches in their performance-driven audio/visual synthetic system. The generic model contained a number of polygons with identified (standardised) MPEG-4 facial points (FPs). Facial expressions were achieved using muscle action, each of which conformed to MPEG-4 FPs. Although the model represented all of the 'anatomy' of a muscle-based system (mass-spring-based muscles, skull and jaw), the animation was not a free Newtonian physics system. The forces that acted on the muscles were compiled or defined in such a way that they conformed to MPEG-4 FPs.

There is an increasing trend towards the use of machine learning techniques in data-driven approaches to computer graphics, and in particular to facial animation (Hertzmann, 2003). Notable here is that of Steinke, Schölkopf and Blanz (2005), in which the use of support vector machine algorithms for 3D shape processing is presented. One of the case studies concerns the reconstruction of scans of human faces.

2.3 Future developments and conclusion

2.3.1 Muscle-based approach

The most advanced technology currently used for geometric deformation at present is the muscle-based model. The use of this technology is likely to persist among the majority of scientists who prefer geometric deformation as their most appropriate animation technique for some time to come.

- Improvements in this domain mainly concern the increased automation of manual tasks and the reduction of human intervention (Kahler et al., 2002), particularly regarding the modelling of the eyes, teeth and tongue (Ko et al., 2003).

- Another general trend is the appropriate combination of image- and geometry-based techniques. One example of this is the simulation of wrinkles and the skin colour. Advanced integration schemes are considered the way forward for advanced skin modelling and skin shading by taking into account self-shadowing, sub-surface scattering and anisotropic reflection (Kahler, 2003).
- Although the model described by Sifakis et al. (2005) is considered to be one of the latest and most progressive models, future improvements concern specialised details and features such as the inclusion of bone and flesh structure, increased realism of the facial expressions, an emphasis on the importance of the hair (Kahler, 2003) and deriving more accurate lip deformation data (Sifakis et al., 2005).

2.3.2 Image-based techniques

Regarding image-based deformation, the most popular trend is towards improving the blendshape modelling method. This approach was reportedly often used for special effects in some recent movies, such as Stuart Little, Star Wars and The Lord of the Rings (Lewis et al., 2005). One of the greatest problems with this approach lies in the amount of human work required to perform a successful animation (Joshi et al., 2003).

- Successful automation of the manual segmentation task and the improvement of the rendering algorithm have already been achieved. However, while the technique used, only takes into account the geometric information, the extension of the algorithm to texture information is considered as future work. Similarly, further research is required to automate the blendshape approach, which currently requires manual work in the setup stage (Deng et al., 2006).
- Most of the work in the academic community is constrained by the size of the datasets available for blendshaping, possibly due to financial constraints. Datasets produced for motion picture special effects are much larger, sometimes consisting of thousands of blendshapes (Lewis et al., 2005), as opposed to 46 used by Deng et al. (2006). Therefore, there is a strong need for testing the academic results on an industry-strength dataset, which has not happened, most likely due to the intellectual property issues.

2.3.3 Performance-driven methods

Performance-driven animation is undoubtedly also here to stay, as there is nothing more natural than the actual expressions created by real people. If such expressions are accurately captured and

reproduced, the results are quite astonishing. Clearly, the data acquisition methods of performance-driven animation warrant separate discussion as they have their own specific research problems (Chuang and Bregler, 2002; Chai, Xiao and Hodgins, 2003).

- Errors accumulate over time due to the use of optical flow principles (Borshukov et al., 2003), and thus need to be corrected, albeit manually. The improvement of the optical flow algorithm somewhat alleviated the manual correction (Reutersward et al., 2005).
- Extrapolation methods may result in unrealistic appearance to the viewer. In the case of occlusion it has been suggested that the occluded part could be deduced/interpolated from the neighbouring frames. However, Zhang et al. (2004) report that their algorithm does not work well for extrapolated faces.
- Specific aspects and issues regarding the application of machine learning techniques may arise, as listed by Hertzmann (2003), for example.

2.3.4 Conclusion

Both image- and geometry-based deformation techniques have their often specialised domains of application. It is not easy or even appropriate to speculate on which technology is better and will dominate in the future. The image-based technologies produce better results if we aim for believability and photorealism. On the downside, they are notoriously slow, inflexible and time-consuming. Since the emphasis is on photorealism in production of movies, the image-based techniques are still the methods of choice.

Conversely, the geometry-based techniques are far more flexible, faster and more manageable, but produce less realistic results. Because of these qualities, they are predominantly used for computer games and animated movies. Also, the anatomically-precise geometric models are used in medicine for cranio-facial surgery. Part II of this thesis is devoted to a closer look at geometry-based methods, for facial modelling and animation.

Performance-based methods do not conceptually compete with image- or geometry-based techniques. They are basically methods of acquiring data, and are thus required to power facial animation. Research is currently being conducted to improve the existing performance-based methods, and, in all likelihood this approach is also here to stay. The increased use of machine learning in data-driven approaches to facial modelling and animation seems inevitable and offers

challenging research opportunities. Applications involving facial animation will proliferate and become increasingly complex and multi-disciplinary.

PART II

Facial modelling and animation

Chapter 3 Computer representations of graphic objects

Computer representations of graphic objects can be broadly divided into those which represent *volume* and those which represent *surface*. Volume representation records data concerning the volume, that is, depending on the resolution, the area inside the object is defined. Surface representation defines the surface only. It has no record of the structure within the object. Both representations have their advantages and disadvantages, and these will be described in the following sections.

3.1 Volume representation

A 3D object can be described by one of three volume representation methods (Parke and Waters, 1996:57), namely constructive solid geometry (CSG), volume element (or voxel) arrays and aggregated volume elements (octrees).

3.1.1 Constructive solid geometry (CSG)

Constructive solid geometry (CSG) (Figure 2) is based on existing 3D primitives that form building blocks for a more complicated model. A real-world analogy of this method would be the use of Lego pieces and other building blocks, where complicated objects are composed from a number of predefined simple objects. Examples of such primitives are cubes, cones and spheres.

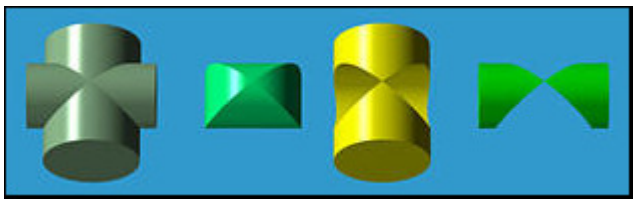


Figure 2: Constructive solid geometry (CSG) example.

CSG is more suitable for engineering or architecture models than for modelling faces, due to roughness of the building blocks. However, one could perhaps design a library of facial parts, such as multiple noses or ears, to cater for any possible nose and ear shape a human may have, similar to the

software used by the police to construct images of suspects in accordance with a witness' specification. Although this may seem feasible, there does not appear to be any current research to support it. This is probably due to a number of factors: firstly, the many diverse facial features to be covered, which makes the modelling task unrealistic. Secondly, grafting (see Section 8.4) such parts together into a whole may in itself turn out to be difficult. However, an obvious advantage of this approach would be the reuse of the building blocks, facilitating the construction of a head model.

3.1.2 Voxels

Voxels are the preferred way of describing structures in medical imaging. This technique consists of sampling cross-sectional images (Figure 3) of a three-dimensional object, then integrating these images into a data structure that accurately represents such an object. The samples are taken at regular intervals, the value of which is known to the computer. The computer then interpolates the volume in between the samples. The whole volume is then represented by pixels, each of which has a coordinate in 3D space and its own 'volume' (hence the name voxel, which stands for **VO**lume pi**XEL**).

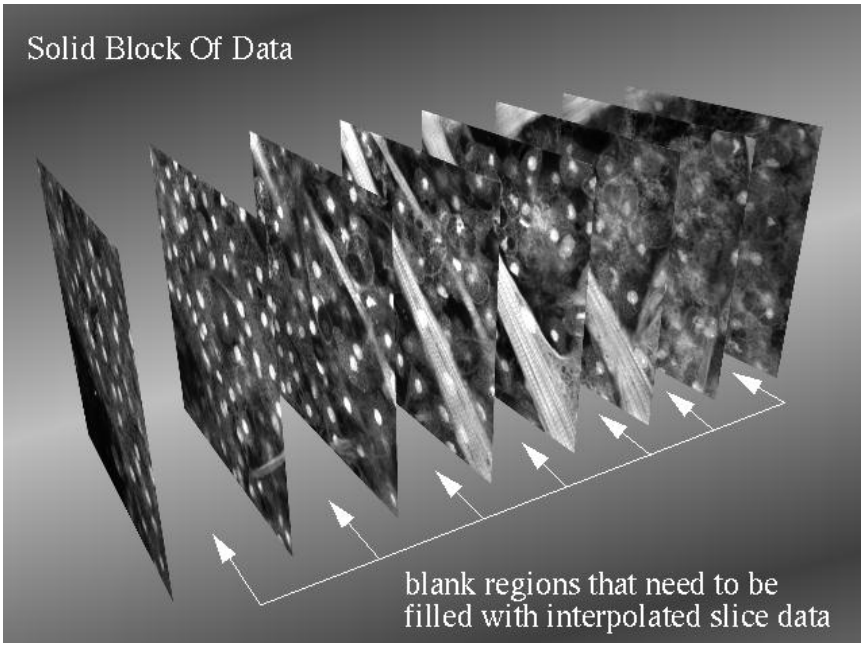


Figure 3: Cross-sectional images – source data for voxel processing (University of Iowa, 1996).

This technique is not used for modelling of faces due to the following two reasons:

Firstly, the detailed spatial representation may be considered ‘overkill’, since facial modelling finally requires only the surface appearance. Indeed, the required memory footprint for voxel images is often an order of magnitude greater than for surface representation. Consider an example of a 50x50x50 pixels solid cube. Voxel representation of such structure in a 24 bit colour would consume 375,000 bytes, with 3 bytes for each pixel. Representing the surface of the same solid cube would require 96 bytes for the geometry (8 vertices, 3 coordinates per vertex, 4 bytes per coordinate) and another 45,000 bytes to represent the surfaces (6 surfaces, 50x50 pixels on each surface, 3 colour bytes per pixel).

Secondly, animation of such models poses a problem, too. The structural shape change in facial animation is too complex for voxel models. Such models are better suited for rigid bodies, as the algorithms required to simulate their deformation are not trivial and would consume massive computing resources. There have been some recent voxel animation efforts, such as the one described by Chandru, Mahesh, Manivannan and Manohar (2000). They represented and animated a 3D world using a voxel model. The animation is done using the voxel-based keyframes, created through sculpting operations (more on keyframes in Section 6.2.1). The animated volume is crude and binary, hence there is no gray or colour information.

In conclusion, we mention a volume-to-surface conversion technique. Lorensen and Cline (1987) presented an algorithm which they call ‘Marching Cubes’ and it does exactly that – creates a polygonal representation of 3D data. Once the data is converted, the polygonal representation can be used for animation, using one of the surface methods described later. This technique has been used in a number of recent works in its pure or modified form (see Kobbelt, Botsch and Schwanecke, 2001 and Pollefeys et al., 2004).

3.1.3 Octrees

Octrees are yet another way of representing volumes. The technique consists of bounding an object with a cube, and then partitioning the cube recursively to eight equal parts at a time, marked from 0 to 7 (Figure 4).

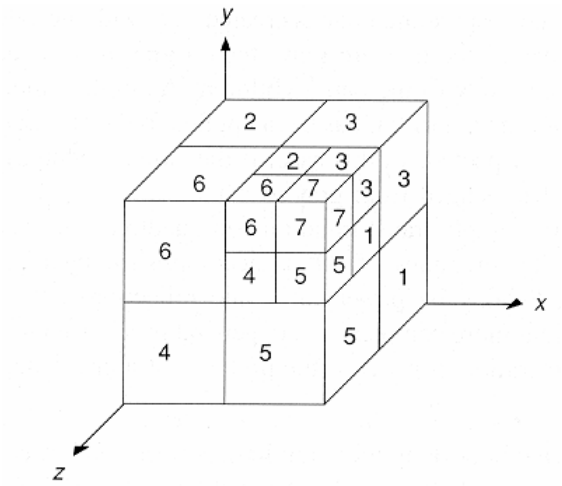


Figure 4: Octree subdivision.

Such a data structure is represented by a tree, with eight children for each internal node (hence the name *octree*). The subdivision is continued until the volume reaches the size of a pixel (voxels). While this technique is widely used in the organization and searching of the 3D space (such as collision detection), it is not apparent how it can be effectively used for the purpose of facial modelling and animation in practice.

3.2 Surface representation

In many instances, volume could be successfully emulated by surfaces. We could take a photo of each of the six sides of a brick, and then glue them together to form the shape of a brick. To a casual observer, such a creation from a distance would indeed look like a real brick. A similar concept has been used in computer graphics for many years. Being relatively computationally inexpensive in comparison to its volume counterparts, the surface representation has been a method of choice for both hardware and software designers. Surface representation techniques include *implicit surfaces* and *parametric surfaces*. On the implementation level, both surface types are subdivided and approximated by a number of flat primitives called *polygonal surfaces*. Graphics subsystems have been designed to natively handle and manipulate large quantities of polygonal surface information in 3D space. For a detailed exposition, the reader is referred to, for instance, Angel (2003:477).

3.2.1 Implicit surfaces

Implicit surfaces are familiar to us from analytical geometry. They are described by expressions that assign a scalar value to each point (x, y, z) in the three-dimensional Cartesian space with x-, y- and z-axis. An example of such an expression would be a sphere with centre at (0,0,0) and radius 1:

$$x^2 + y^2 + z^2 = 1 \tag{3-1}$$

In practice, implicit surfaces are not used for facial modelling. Their disadvantages include complexity in interfacing one implicit surface with another and the length of time required for interactive manipulation and display (Parke and Waters, 1996:59).

3.2.2 Parametric surfaces

Parametric surfaces are surfaces generated by functions with parametric variable(s). The parametric representation of the sphere in Section 3.2.1 is

$$\begin{aligned} x &= r\sin(\phi)\cos(\theta) \\ y &= r\sin(\phi)\sin(\theta) \\ z &= r\cos(\phi) \end{aligned} \quad , \tag{3-2}$$

where r is the radius, here r=1; and the parameters ϕ and θ range from 0 to 2π .

Bezier Surface Patch

An example of a parametric surface is provided in Figure 5. It depicts a Bezier surface patch. Cubic Bezier surface patches are bound by 16 control points, forming a 4x4 array. If the array is denoted as

$$P = [p_{i,j}], \tag{3-3}$$

then the corresponding cubic Bezier patch is formally defined as

$$P(u,v) = \sum_{i=0}^{3}\sum_{j=0}^{3} b_i(u) b_j(v) p_{i,j} \quad , \tag{3-4}$$

where $P(u,v)$ is a point on the patch, $b_i(u)$ is the i^{th} cubic blending Bernstein polynomial at value *u*, $b_j(v)$ is the j^{th} cubic blending Bernstein polynomial at value *v*, while $p_{i,j}$ is a control point.

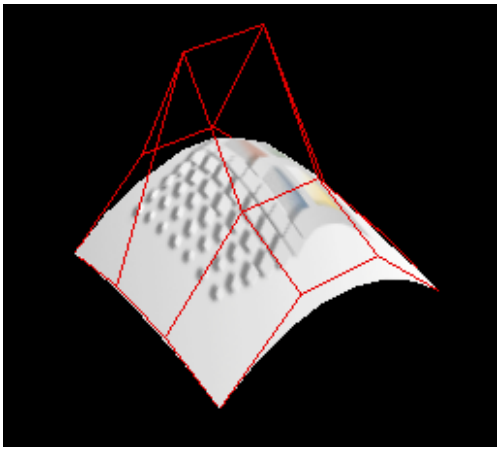


Figure 5: Example of a plotted parametric surface: Bezier surface patch.

The parameters u and v vary over a chosen rectangle, forming the surface patch. For a more detailed exposition of Bezier patches and Bernstein polynomials, the reader is referred to, for instance, Angel (2003:495). In Figure 5, the 16 control points are connected by red lines, forming the frame above the patch. The 16 points in question are

```
GLfloat p[][4][3] =
{
          {{-0.75,-0.75,-0.50}, {-0.25,-0.75, 0.00}, { 0.25,-0.75, 0.00}, { 0.75,-0.75,-0.50}},
          {{-0.75,-0.25,-0.40}, {-0.25,-0.25, 0.90}, { 0.25,-0.25, 0.90}, { 0.75,-0.25,-0.40}},
          {{-0.75, 0.25,-0.40}, {-0.25, 0.25, 0.20}, { 0.25, 0.25, 0.20}, { 0.75, 0.25,-0.40}},
          {{-0.75, 0.75,-0.50}, {-0.25, 0.75, 0.00}, { 0.25, 0.75, 0.00}, { 0.75, 0.75,-0.50}}
};
```

while the four cubic blending Bernstein polynomials are defined as

$$\begin{aligned} b_0(u) &= u^3 \\ b_1(u) &= 3u^2(1-u) \\ b_2(u) &= 3u(1-u)^2 \\ b_3(u) &= (1-u)^3, \end{aligned} \tag{3-5}$$

with u and v ranging between 0 and 1.

B-splines

Another example of parametric surfaces frequently used in computer graphics is that of *B-splines* (Angel, 2003:498). Several authors have described the use of *B-spline* surfaces for modelling of faces. These models produce faces with smooth curved surfaces that are defined using very few control points. Parke (1996:60) states the disadvantages of such models:

- Density of control points used does not support the detailed surface definition and surface control needed around the eyes and mouth;
- Creases in the face are difficult to implement since they often require defeating the natural surface continuity properties;
- Adding detail requires adding complete rows or columns to the control points array.

To overcome some of these disadvantages, further research has been conducted to increase the local detail in B-spline surfaces. Forsey and Bartels (1988) developed a B-spline subdivision algorithm that adds additional control points to a region of interest. This was done in an attempt to achieve more detailed control of the surface. Their study was later used by Wang (1993), who attempted to further improve on local surface details via hierarchical refinement. The B-spline facial models have not had further advancements, due to their disadvantages. Breton, Bouville and Pele (2001) cite difficulty to animate, as the control points are not on the surface and the seams between the patches are visible during the animation.

The B-spline surfaces are formally defined as (Rogers, 2001)

$$Q(u,v) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1`} B_{i,j} N_{i,k}(u) M_{j,l}(v) \qquad (3\text{-}6)$$

where $N_{i,k}(u)$ and $M_{j,l}(v)$ are the B-spline basis functions in the biparametric u and v directions. $B_{i,j}$ are the control points. The upper bounds m and n are one less than the number of control vertices in the u and v direction, respectively. The basis functions are defined recursively as

$$N_{i,1}(u) = 1 \text{ if } x_i \le u \le x_{i+1}, \text{ 0 otherwise}$$

$$N_{i,k}(u) = \frac{(u - x_i) N_{i,k-1}(u)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - u) N_{i+1,k-1}(u)}{x_{i+k} - x_{i+1}}$$

and (3-7)

$$M_{j,1}(v) = 1 \text{ if } y_j \le v \le y_{j+1}, \text{ 0 otherwise}$$

$$M_{j,l}(v) = \frac{(v - y_j)M_{j,l-1}(v)}{y_{i+k-1} - y_i} + \frac{(y_{j+l} - v)M_{j+1,l-1}(v)}{y_{j+l} - y_{j+1}}$$

where the x_i and y_j are elements of the knot vectors.

NURBS

Non-uniform rational B-splines, better known as NURBS, are the generalization of non-rational B-splines. NURBS are easily processed by a computer, are stable to floating-point errors, have little memory requirements and are able to represent a wide range of curves and surfaces. NURBS surfaces are defined as (Rogers, 2001)

$$Q(u,v) = \sum_{i=1}^{n+1}\sum_{j=1}^{m+1} B_{i,j} S_{i,k}(u,v) \tag{3-8}$$

where $B_{i,j}$ are the control points and $S_{i,k}(u,v)$ are the bivariate rational B-spline surface basis functions.

They are defined as

$$S_{ij}(u,v) = \frac{h_{i,j} N_{i,k}(u) M_{j,l}(v)}{\sum_{i_1=1}^{n+1}\sum_{j_1=1}^{m+1} h_{i_1,j_1} N_{i_1,k}(u) M_{j_1,l}(v)} \tag{3-9}$$

where $N_{i,k}(u)$ and $M_{j,l}(v)$ are the B-spline basis functions in the biparametric u and v directions, defined recursively in 3-7. $h_{i,j}$ is the weight at the point $B_{i,j}$.

A more detailed discussion of these surface approximations and their application may be found in Rogers (2001).

3.2.3 Polygonal surfaces

Polygonal surfaces (Figure 6) are formed from polygons; in this case, triangles. Polygons are bound by a number of explicitly defined vertices. The area bounded by the vertices is normally filled by a colour or a texture, giving the viewer the impression of a surface. Although the polygons are allowed to have a large number of vertices, they are usually reduced to triangles for practical reasons (hardware acceleration support, single-planar properties, guaranteed convexity, and so on). A large number of such triangles could be concatenated together, forming relatively complex surfaces. Most of the models and techniques referred to in the remainder of this thesis are formed using polygonal surfaces.

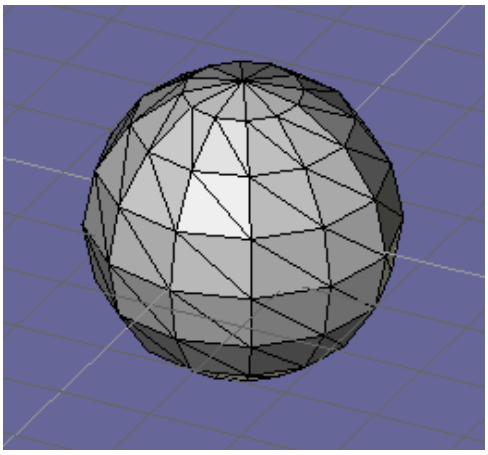


Figure 6: Example of a polygonal surface.

The chief advantage of using polygonal surfaces lies in their rendering speed. Polygons have been adopted as a rendering method of choice by most display card manufacturers, who embedded a vast amount of polygon rendering functions directly in the hardware. However, even the polygonal representation is far from ideal. It is often difficult to represent a curved surface using polygons (Figure 6). To create the illusion of smoothness, a large number of polygons needs to be used, along with some other rendering techniques, such as Gouraud shading or Phong shading.

3.2.4 Hybrid surfaces

It is often beneficial to combine several surface representation techniques. Bui (2004) combines polygonal and B-spline surfaces. He uses B-spline surfaces for small and smooth surfaces such as lips

and polygonal surfaces for the rest of the face, including the generation of wrinkles and bulges. The modelling of lips with B-spline surfaces was first described by King (2001). An issue that is specific to hybrid surfaces is so-called grafting, which refers to connecting diverse surfaces to form a homogenous unit. Grafting is described in Section 8.4.

3.3 Conclusion

To summarise, 3D objects on a computer could be represented either by volume or by surface. Volume representation is more intuitive and detailed, but requires large amounts of storage space and is more computationally intensive. Surface representation has opposite properties and has been widely accepted throughout the computer graphics community for the past 30 years.

The relative advantage of implicit and parametric surface representation over polygonal representations is their ability to describe curved surfaces.

Despite the multitude of surface representations techniques available, most of them are reduced to polygonal surfaces at a hardware display level. Lately, due to increases in processing speed, various volume representation techniques have resurfaced in the form of finite element methods. However, the latter technique is still in an experimental phase and it is expected that polygonal surfaces will continue to dominate hardware and software implementations for some time to come.

Chapter 4 Facial features and their proportions

The main object of study in this thesis is the human head, and in particular its modelling and animation. It is therefore necessary to discuss those aspects of human anatomy that pertain to the head, and its modelling.

Fleming and Dobbs (1999:3-52) in their book gave an overview of the most important aspects of manual head and face modelling. The remainder of this section is largely based on the relevant section of this book. Facial features for modelling purposes can be divided into the skull, brow ridge, eyes , nose, cheekbones, mouth, chin, lower jaw, and ear.

4.1 The skull

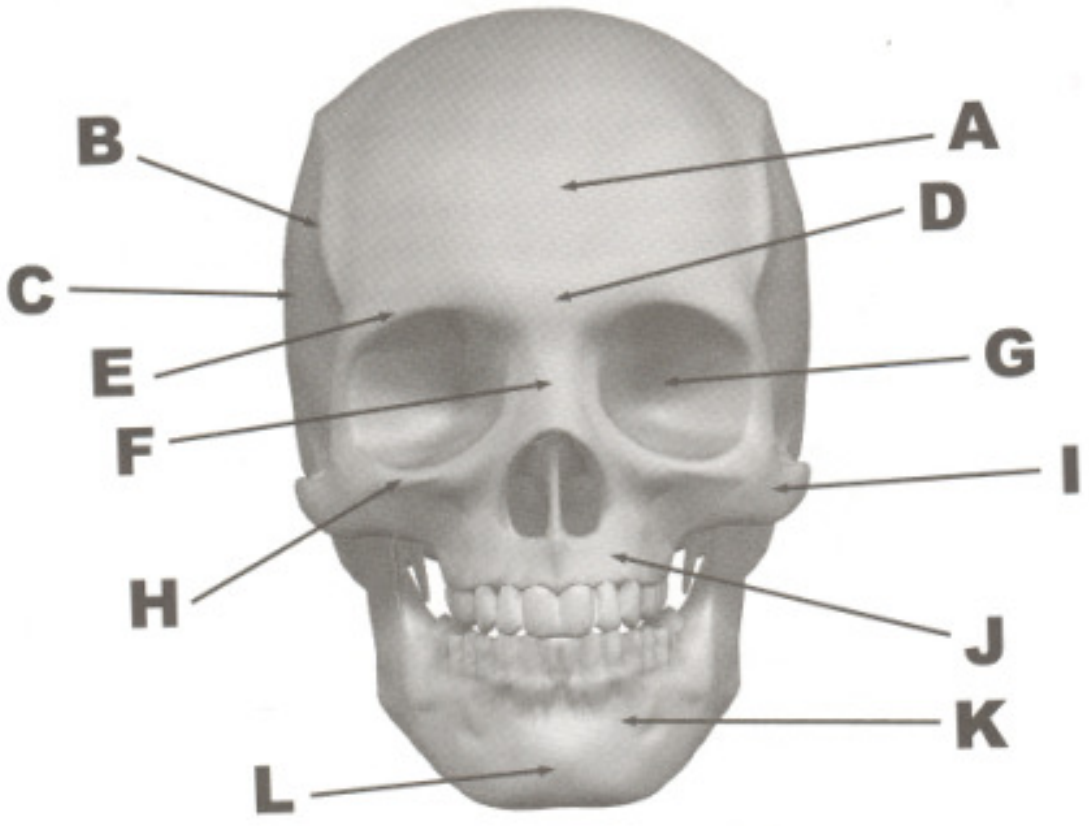


Figure 7: Skull parts (Fleming and Dobbs, 1999:3).

The skull's shape (Figure 7) has several distinct parts important for modelling purposes. These include:

- *Frontal bone (A)*, which forms the forehead structure.
- *Temporal ridge (B)*, which is not very pronounced, but is responsible for creating the square-shaped appearance of the upper skull.
- *Parietal bone (C)* on the side of the head.
- *Nasion (D)*, where the frontal bone meets the nasal bone.
- *Supraorbital margin (E)*, which is one of the most distinct bones on the face. It creates the ridge above the eyes.
- *Nasal bone (F)* is the structure on top of the nose, where it meets the nasion.
- *Orbital cavity (G)* is a large hole where the eyes are located.
- *Infraorbital margin (H)* is the lower portion of the orbital cavity
- *Zygomatic bone (I)* is the cheek bone that lies immediately under the infraorbital margin.
- *Maxilla (J)* is the upper jawbone, directly under the nose.
- *Mandible (K)* is the bulk of the lower jaw. It is the only movable bone on the skull.
- *Mental protuberance (L)* is the tip of the lower jawbone, or the chin.

In order to achieve a realistic model and subsequent animation, it is an imperative to preserve the relevant proportions of the facial features. The human skull is composed of two major parts, those being *cranial* and *facial*. The cranial part takes two thirds of skull mass, while the facial element occupies the remaining one third. Viewed as a profile, the human skull always fits into a square, that is, its height is the same value as its depth (Figure 8).

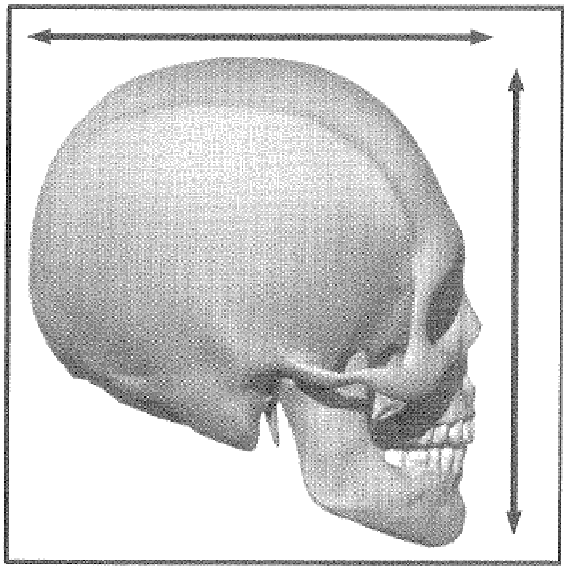


Figure 8: The skull proportions (Fleming and Dobbs, 1999:19).

When modelling a head from first principles, one should always do so inside the background of an actual square. This helps ensure the preservation of proportions. From the front, a line through the orbital cavity divides the skull into two halves.

There are several significant differences between a male and a female skull:

- Male cranial mass is more robust than that of a female. It tends to be blockier than a female, which is more rounded.
- The supraorbital margin of a female skull is sharper than that of a male, which tends to be round.
- Muscle attachments on the zygomatic bone are more pronounced in males.
- The mandible of a female is round, while that of the male one is squared.
- The superciliary arch (a ridge on the frontal bone above the eye socket) on a male is large and pronounced, overhanging the ocular cavity to provide better protection for the eyes.
- The canine teeth in the jaw of a male skull are significantly larger than their female counterparts.

The skull forms the base of the ‘future head’. It is always beneficial to have a skull template in the background, prior to embarking on design of other facial features. This makes it easier to maintain correct proportions.

4.2 The brow ridge

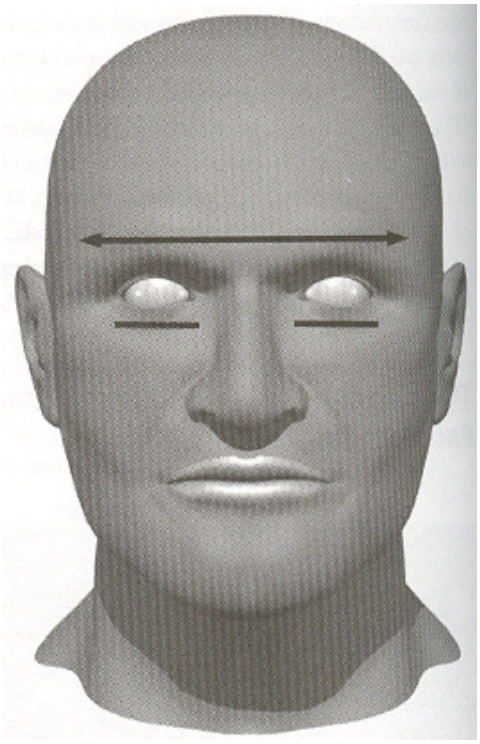


Figure 9: Proportions of the brow ridge (Fleming and Dobbs, 1999:30).

The brow ridge (Figure 9) is the midpoint of the face. The width of the head along the brow ridge is five times the width of the eye, while the brow itself is four times the width of the eye. It is an important detail to slightly indent the centre of the brow, just above the nasion.

4.3 The eyes

For the average face, the eye itself is about one and a quarter inch in diameter, and it is almost perfectly round, with the exception of the bump in front of the iris (conjunctiva and cornea).

Modelling eyes presents particular challenges. The exposed upper eyelid is extremely uncommon on a real head, as the supraorbital margin hangs over the upper eye. Normally, there is tissue under the supraorbital margin, covering a good portion of the upper eyelid. So-called 'floating eyelids' are another common mistake – it often happens that eyelids are modelled above the eyes and there is a visible shade falling over. In reality, eyelids are sliding over the eyes, and there is no space in between. Yet another detail to watch out for is the distance between the eyes, which is also commonly mismatched. Generally speaking, eyes should be set apart by the width of an eye. The shape of the eye opening is often made oval. This is close, but in reality the eye opening is not a symmetrical oval, but oblique. A proper placement of the iris is essential for facial expressions. The iris is partially covered by the upper eyelid, while hovering just above the lower eyelid. Subjectively, the more the iris is covered, the more depressed the character appears. The iris needs to be of appropriate size. If it is out of proportion, it will appear unnatural. The pupil is roughly half the width of the eye opening.

4.4 The nose

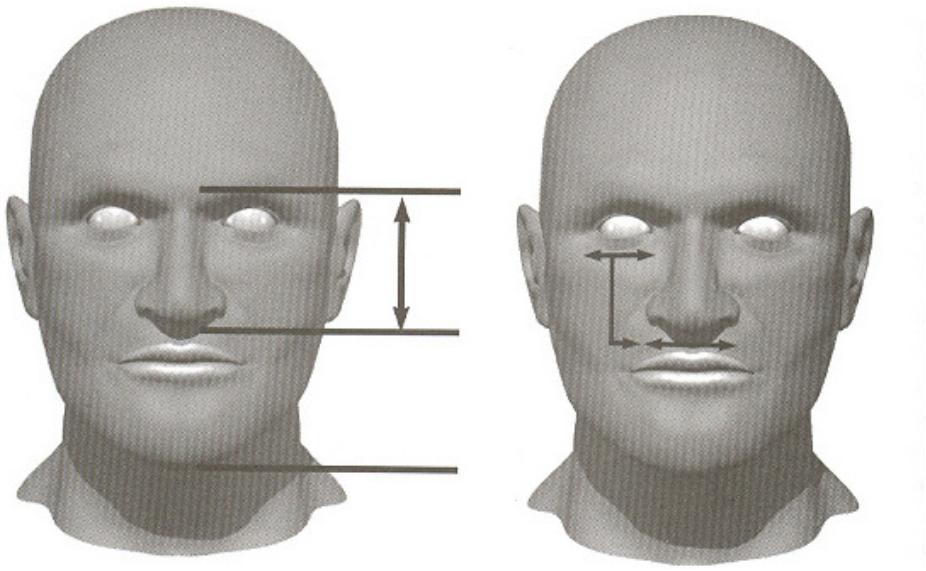


Figure 10: Proportions of the nose (Fleming and Dobbs, 1999:36).

The length of the nose from the nasion to the tip is the same as the distance from the tip to the bottom of the chin (Figure 10). The base of the nose is of the same width as the eye.

4.5 The cheekbones

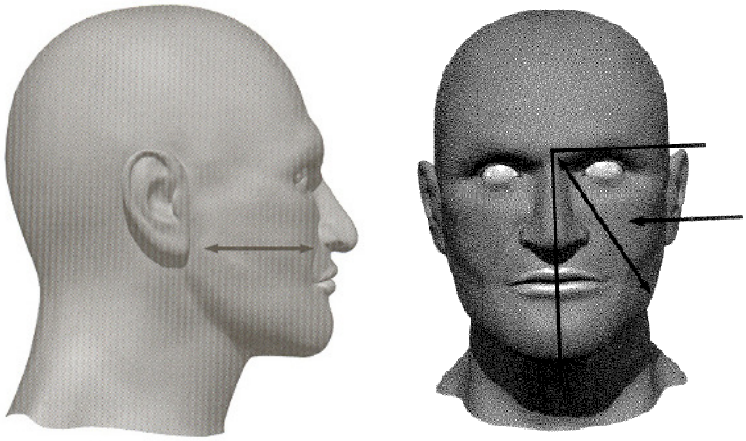


Figure 11: Cheekbone proportions (Fleming and Dobbs, 1999:38).

The cheekbone gives the head personality and character. The baseline of the cheekbones (Figure 11) is lined-up with base of the nose. It starts at the top of the nasal bone and runs down at a 30 degrees angle, measured from the nose line. Its depression is in the middle of this diagonal line.

4.6 The mouth

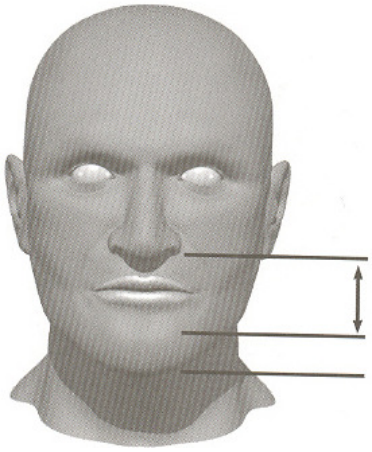


Figure 12: Mouth proportions (Fleming and Dobbs, 1999:40).

The mouth (Figure 12) is probably the most complicated facial feature to model. Because it is so mobile, opening and closing in speech or eating, it needs to be modelled from both inside and outside. Looking from the front, the mouth fits in a rectangle derived as follows:

- Drawing lines vertically from centres of the ocular cavities;
- Drawing a horizontal line just below the nose;
- Drawing a horizontal line at 2/3 way from nose to the chin.

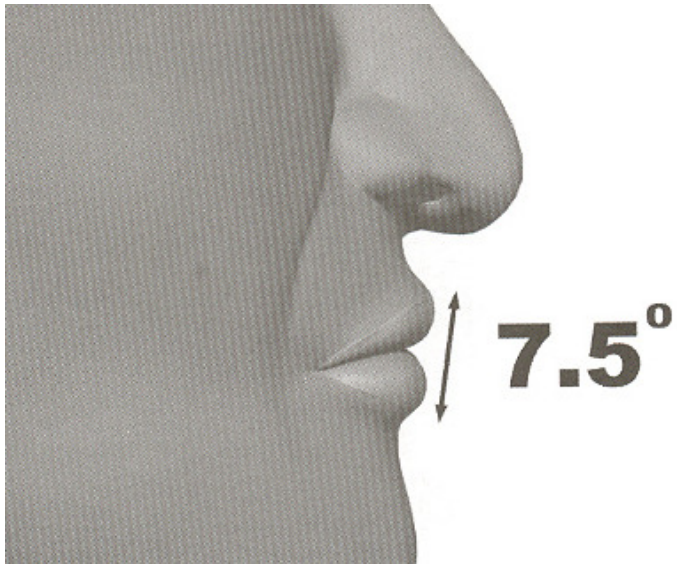


Figure 13: The lip angle (Fleming and Dobbs, 1999:41).

In profile, the mouth is in line with the corner of the jaw, and the lips are not flush, but slightly angled. The upper lip slightly overhangs the lower lip, so the tangent connecting both lips closes and angle of 7.5 degrees with the imaginary vertical line (Figure 13).

The interior of the mouth is far more complicated than the exterior. One of the most common modelling errors is to design cheek tissue away from the gums. On a real head, the cheek tissue is positioned tightly against the gums. The lips should contour around the form of the gums, pulling the sides of the mouth back into the head. Placement of the gums and teeth is crucial to the realism of the expression. The maxillary arch holds the upper teeth, while the mandibular arch holds the lower teeth.

The teeth meet in the same line where the lips meet, too. The width of the dental structure is equal to the distance between the centres of the ocular cavities (Figure 14).

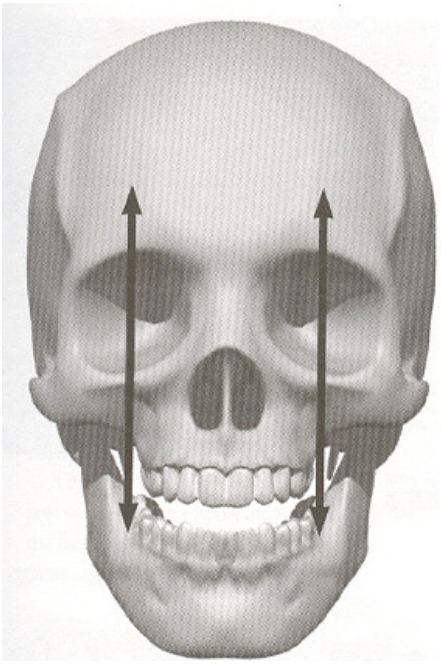


Figure 14: Width of the dental structure (Fleming and Dobbs, 1999:45).

Proportioning the teeth is somewhat more difficult, but the following guidelines may be used:

- The four front upper teeth together are the same width as the bottom of the nose.
- The gaps between the two front teeth of the upper and lower jaw are aligned.
- The lower canines line up with the outer incisors of the upper jaw.

Behind the teeth is the tongue, which requires some attention, too. The tongue is often modelled flat, while it is in fact really quite thick. It needs to be modelled as such, particularly when the tongue is resting on the lower palate (during yawning, for example). The tongue is flexible and its width varies depending on its position. However, the tongue assumes its maximum width when it is in its resting position, where it fills the lower gums.

4.7 The chin

The chin takes up the bottom third of the mass below the nose (since the mouth takes two thirds). There are no special considerations when modelling the chin.

4.8 The lower jaw

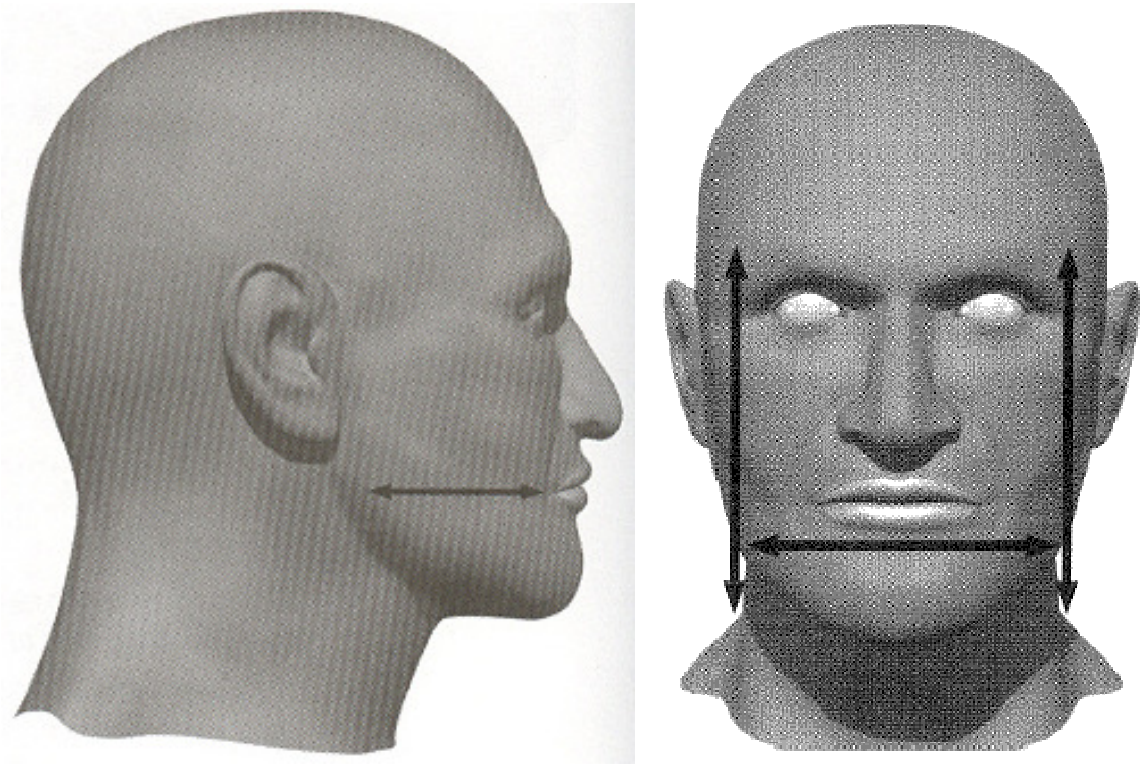


Figure 15: The angle of the lower jaw (Fleming and Dobbs, 1999:48).

The lower jaw defines the profile of the head. As mentioned before, the angle of the lower jaw aligns with the line where the lips meet (Figure 15). From a frontal perspective, the width of the lower jaw spans the imaginary vertical line drawn at the outer side of supraorbital margin.

4.9 The ear

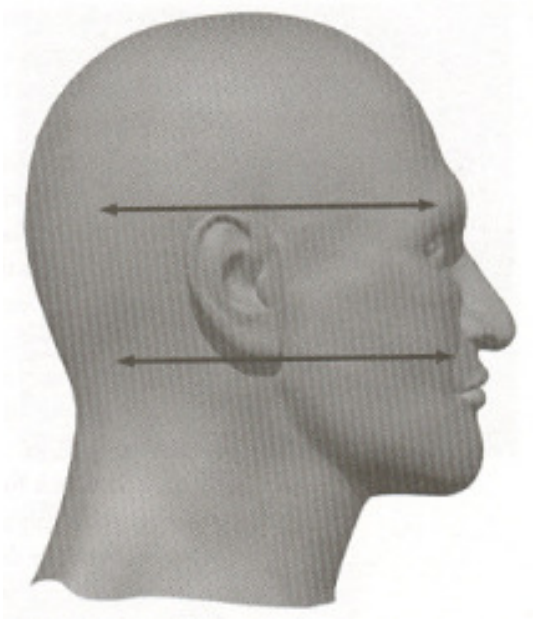


Figure 16: Ear placement (Fleming And Dobbs, 1999:49).

In profile, the vertical placement of ears is between the eyebrow and the base of the nose (Figure 16). Horizontally, they are placed approximately in the centre of the head. Ears are not placed along a vertical line, they are rotated by about 15 degrees, tilting backwards.

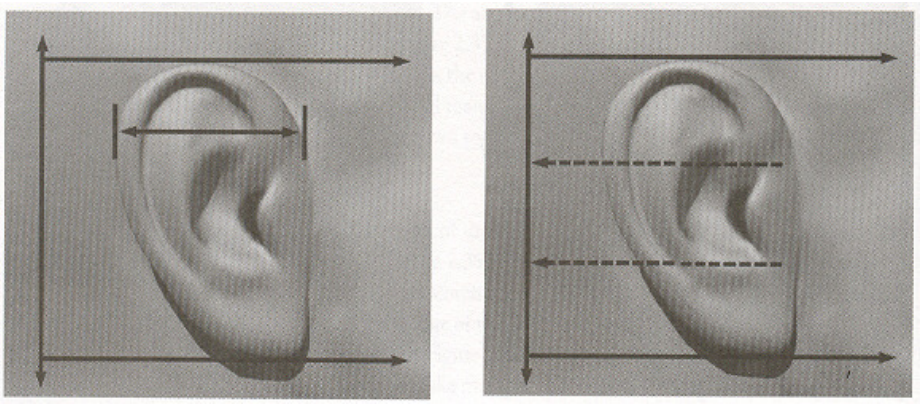


Figure 17: Ear proportions (Fleming and Dobbs, 1999:51).

The width of the ear is usually half of its height. The ear hole is one third of its length and is positioned centrally. The earlobe is also one third of the dimension of the ear. At its widest point, the earlobe is one half of the width of the ear (Figure 17).

4.10 Conclusion

The above-mentioned proportions and measurements are only approximate guidelines. Human heads and faces vary in shape and size. They largely depend on race, ethnicity and geographical region. A canonical head model could be created according to the mentioned proportions, then adapted to suit a particular character. On the part of the creator, taking care to ensure that the head is properly shaped will go a long way towards creating believable facial expressions and animation.

Chapter 5 Constructing a head model

As discussed in Chapter 3, there are a number of approaches which can be used when representing a head graphically. For the purpose of this thesis, we restrict ourselves to describing the most common and widely used method for geometry-manipulated animation: the surface representation method (see Section 3.2). Regardless of which surface representation is used at a higher level of abstraction (implicit, parametric or polygonal), any surface is finally approximated by a number of triangles at a graphics engine level. Each triangular surface is defined by three vertices in 3D space. The smaller and denser such triangular surfaces, the smoother and better defined is the resultant surface.

Due to its complexity and the degree of realism that needs to be achieved by current standards, it would be infeasible to model faces manually by positioning vertices and polygons in space, or perhaps by manually listing the vertex coordinates in a text file. Various automated methods and hardware devices have been invented to facilitate data collection. These methods are described in this chapter.

5.1 Contour reconstruction

Contour reconstruction is a technique of reconstructing 3D surfaces using multiple 2D data points and analysis of the positions of the object to be modelled. An algorithm for automatically creating such polygon skins was first described by Fuchs, Kedem and Uselton (1977). The data consisted of contours, taken at regular height intervals. The vertices on a contour are then connected to vertices on the neighbouring contours in such way that they form small triangular surfaces (Figure 18). The collection of contour data poses a challenge. It depends on the transparency of the surface – if the surface is transparent, the images could be taken at various focus levels of the camera or microscope. Opaque objects have to be cut into slices and recorded individually. The algorithm uses graph theory to construct a set of triangular surfaces over the contours, thus approximating the surface of the modelled object.

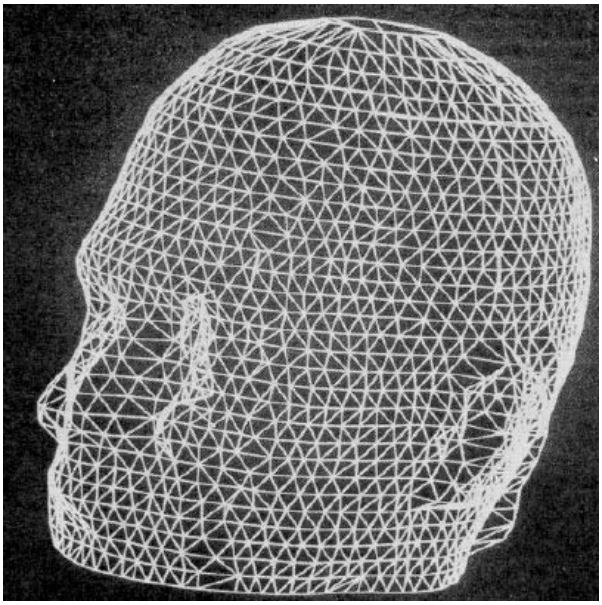


Figure 18: A head mesh created using contour reconstruction methods.

While contour reconstruction remains a popular modelling technique in general, it is not frequently used in facial modelling. The head and face are not transparent objects, and in most cases it would be impossible to slice them up for modelling purposes. Contour reconstruction techniques are better suited to modelling of the softer tissue, cross-sections of which are easier to obtain via *magnetic resonance imaging* (MRI) and *computer tomography* (CT) scans (Braude, 2005).

5.2 3D Digitizers

Older 3D digitizers were hardware devices that relied on mechanical, electromagnetic or acoustic measurements to locate positions in space. These digitizers required a probe at each surface that required measuring. Mechanical devices had a sensor at the surface to be measured. The position of the sensor was then converted to electrical impulses and translated into a point in 3D space. Their drawback lay in their not being able to reach all the points on complex surfaces. Acoustic digitizers relied on the time that is required for the sound to travel to the acoustic sensors. The position in space was determined by multiple sensors (minimum three).

Sometimes objects in the environment affected this reading, resulting in measurement errors. Because electromagnetic digitizers work by generating an orthogonal electromagnetic field, with

sensors providing signals that can be converted into coordinates in 3D space, metals may affect the electromagnetic field and generate reading errors (Parke and Waters, 1996:68).

Recently, non-contact laser based 3D digitizers were invented, such as the Konica-Minolta VIVID range of products (Konica Minolta, 2007). Figure 19 shows a human head captured using one of the Konica Minolta digitizers. The product comes with software that enables export to most of the major modelling file formats (Wavefront, 3D Studio Max, Maya, and others).

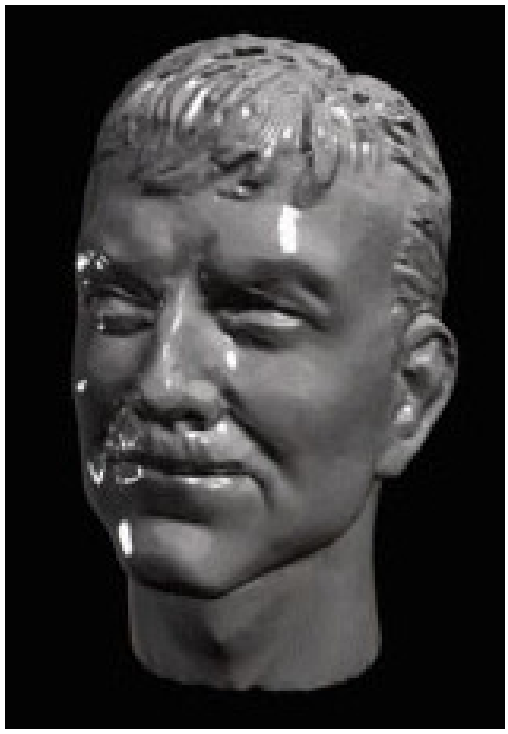


Figure 19: Digitized using a Konica Minolta device.

Another popular laser-based system is that of Cyberware (1999). Their Head & Face Colour 3D Scanner captures both surface information and colour simultaneously. The whole process is automated and controlled by the software and hardware supplied.

In Figure 20, an object is measured by moving the scanning device in a circular motion around it. The resultant data is given in a cylindrical coordinate system. Each mesh entry consists of a radius, azimuth and height triplet.

Figure 20: Result of a scanned surface colour data, using the Cyberware's hardware.

5.2.1 Issues surrounding the scanned data and their resolution

The main problem with laser-based scanning is that of missing points. Sometimes the laser beam gets obscured or dispersed – usually this happens under the chin, inside nostrils and areas under hair. These areas are depicted in the white areas in Figure 21. In an attempt to compensate for missing data, several methods have been designed. Two of these techniques are discussed below:

- The 'relaxation membrane interpolation' method was described by Terzopoulos (1988). In this method, relaxation interpolation approximates the missing data by indexing the nearest neighbour values, or 'stretching an elastic membrane' over the gaps.
- The other approach was described by Williams (1990a). He identified that processing of the surface data is similar to image processing. His method is also based on the neighbouring values, which gradually replace the missing data over a number of iterations.

Figure 21: Missing data – white patches – area covered by hair and under the chin (Lee, Terzopoulus & Waters, 1995).

Another problem with scanning is the noise which occurs on the scanned data, referred to by Williams (1990a) as 'salt and pepper' noise. This concept is taken from 2D scanned image processing, where the imperfection of the camera sensor introduces erroneous white ('salt') and black ('pepper') pixels (Chan, Ho and Nikolova, 2005). In order to eliminate this obstacle, Williams (1990a) computed a smooth estimate of the surface at a point, using a 3x3 blur kernel with unity gain and 0.0 as the centre sample coefficient. If the estimate differed from the centre sample by more than a predetermined threshold amount, the centre sample would be replaced by the estimate. Such a filter is also referred to as a Tukey filter or hysteresis filter.

Scanners produce massive amounts of data, creating small polygons on surfaces where it is not necessary. In such cases, multiple small polygons could be replaced by one large polygon on a surface with low curvature, to optimise storage and display. A number of algorithms exists for the purpose of automatically 'thinning' such meshes and eliminating all unnecessary polygons below the given curvature threshold (Heckbert and Garland, 1997).

5.2.2 Adaptation of the scanned data for the purposes of modelling and animation

Edges of the polygons of scanned data are not necessarily consistent with creases on a face, as desirable by the modelling requirements. Neither are they consistent with the requirements of division of polygons in such a way that surfaces are not deformed during animation. Several methods exist for matching scanned data and desired facial topology, namely *interactive fitting*, *topology adaptation* and *adaptive meshes*.

5.2.2.1 Interactive fitting

As suggested by its name, interactive fitting is a manual process and consists of matching each vertex of a polygon topology to a corresponding point on a scanned data mesh. One could approximately overlay the polygon topology on the scanned data mesh and manually slide the polygon vertices, until they match the data (Parke and Waters, 1996:81).

5.2.2.2 Adaptive meshes

Adaptive meshes (Terzopoulos and Vasilescu, 1991) automatically adapt to features of interest, increasing the polygon density for such areas. The idea behind adaptive meshes was to have a notion of adjustable ‘springs’ that connect the nodal masses to that of its neighbours. As a node is moved, these springs create forces which influence its neighbouring points, which in turn influence their neighbours, and so it continues. The entire mesh is repeatedly recalculated according to spring dynamics, until such time as it reaches a stable state.

5.2.2.3 Topology adaptation

Topology adaptation is described by Lee, Terzopoulos and Waters (1995), where they improve upon the existing adaptive meshes methods of Terzopoulos and Vasilescu (1991) and Vasilescu and Terzopoulos (1992). They do this using a generic face model (Figure 22) that is consistent with modelling requirements (efficient triangulation) and suitable for animation, then apply it to the data acquired by means of a scanner.

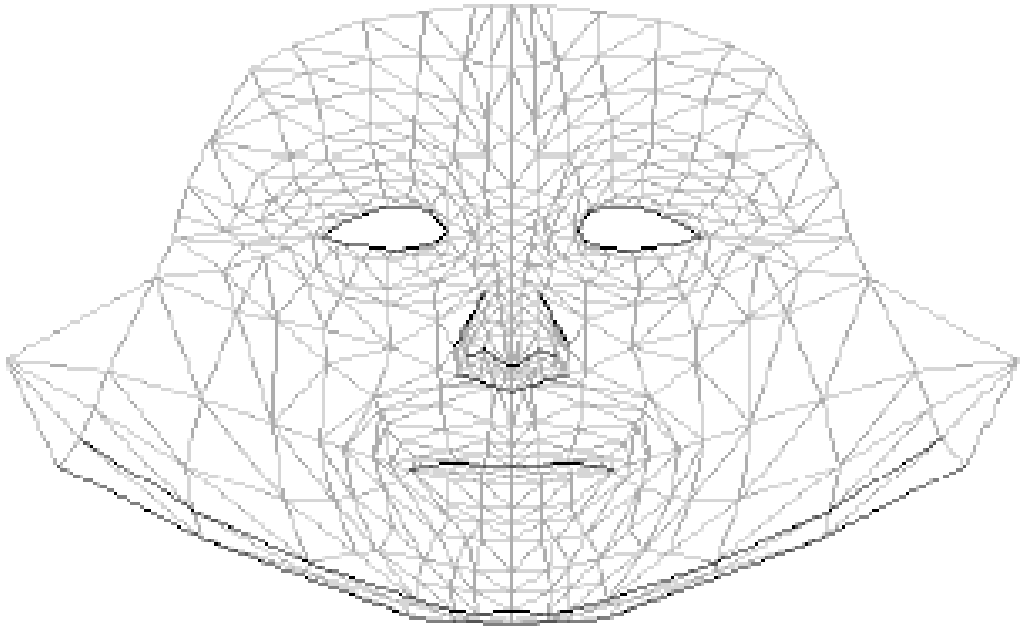


Figure 22: Facial portion of generic mesh in 2D cylindrical coordinates (Lee, Terzopoulos & Waters, 1995).

Conceptually, the steps in mesh adaptation start with location of the nose tip, chin tip, mouth contour, chin contour and ears. The vertices of the scanned face are then rescaled and mapped to the generic model in accordance with the previously mentioned landmarks. Vertices of the generic model are interconnected by modelled springs, in order to keep the vertices optimally distributed across the surface. Once the mapping of the vertices is completed, the spring 'forces' are activated, bringing the mesh into a localised equilibrium, within the boundaries of a set of landmarks (Lee, Terzopoulos and Waters, 1995).

5.3 Photogrammetric techniques

Photogrammetric techniques reconstruct a 3D shape from 2D photographs, each taken from a different viewpoint. PhotoModeler (2004) offers an example of photogrammetric based commercial software. It claims to be able to reconstruct a wide spectrum of 3D scenes and objects from photographs.

Traditionally, there were two main approaches for photogrammetric measurement: simple and complex. They differed significantly in terms of the quality of the final model. For the simple method, two photographs were required – one from a frontal perspective and one a profile (Figure 23).

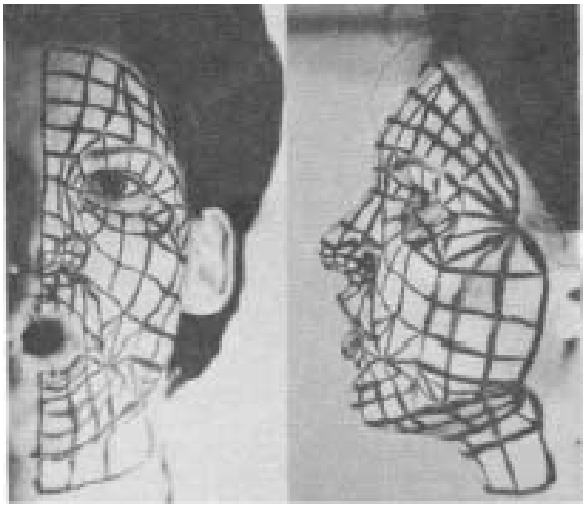


Figure 23: Image pair used for the simple photogrammetric method (Parke, 1972).

A coordinate system was established, with its origin at the centre of the head. The position of each point was then measured directly from the photos. Such measurement could have been done manually or using a 2D digitizer. The main shortcomings of this simple method were:

- The distortion of the images, since the photos are not true orthographic projections;
- The inability to represent points not visible in both views, such as underside of the chin.

The more complex method used two transformation matrices for projecting 3D points onto the 2D space shown on the two photographs. Two photos were taken in such a way that the angle between them was not too small and the point occlusion was minimal. This was described in greater detail by Parke and Waters (1996:74).

Synthesis of facial expressions from photographs has been attempted by Pighin et al. (1998). They departed from the more traditional photogrammetric technique of drawing a grid on a subject's face and replaced it with round markers (Figure 24). Since the small round facial markers were not as prominent as the grid (Figure 23), the photograph could have been used as the face texture.

Figure 24: Photograph with sample reference points and expression modelled by using 13 reference points (Pighin et al., 1998).

Gutierrez-Osuna et al. (2005) extended this technique to capture the necessary data for the purpose of MPEG-4 compliant animation (see Section 6.1.2.3 for a detailed discussion). Twenty-seven markers were placed on the face, corresponding to MPEG-4 facial points (FP – Figure 25). Head movements were differentiated from speech by tracking the four facial points that determine the head pose.

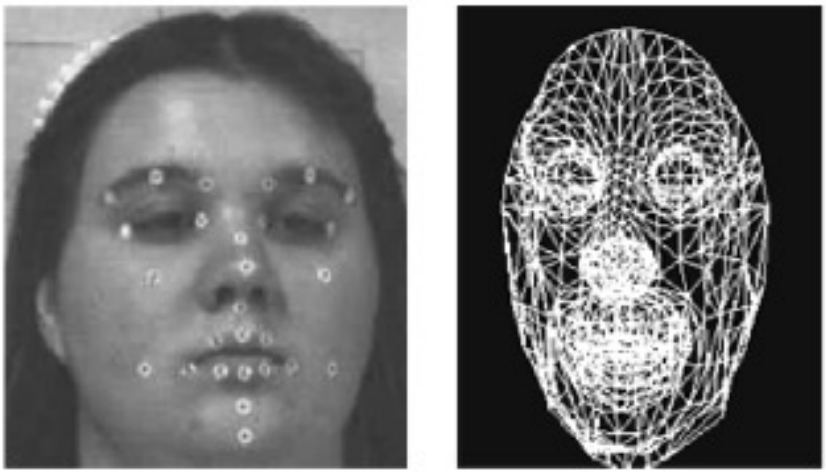


Figure 25: Neutral facial expression and its wireframe representation (Gutierrez-Osuna et al., 2005).

5.4 Sculpting methods

Head models can be created manually, using techniques similar to those used by sculptors. The various approaches to sculpting heads include *assembling faces from simple shapes*, *successive refinement from a simple object*, *head modelling using splines* and *intensity painting*. Each of these techniques is discussed in turn.

5.4.1 Assembling faces from simple shapes

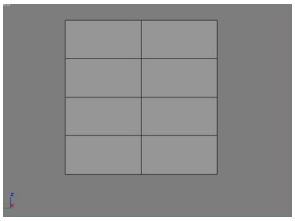
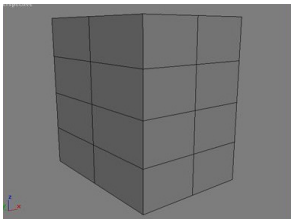
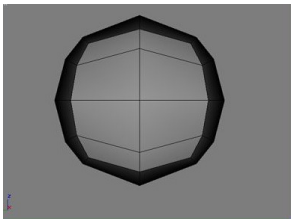
In some cases facial models do not need to be complex – they could be composed of simple objects. Such models are frequently easier to model and animate. They should, however, be able to express human emotions. A well known example of a character created by assembling simple shapes is Tinny, from Tin Toy (1988), a short animation by Pixar (Figure 26).

Figure 26: Tinny from Tin Toy (1988).

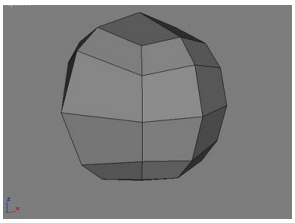
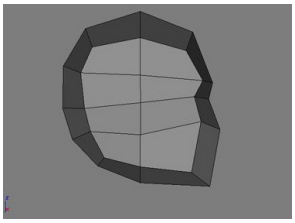
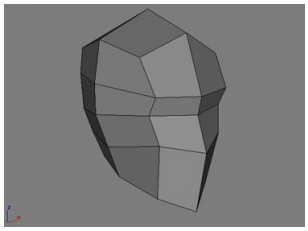
5.4.2 Successive refinement from a simple object

There are numerous tutorials on the Internet on the modelling of a human head, starting from a simple monolithic object. These tutorials are mostly dedicated to professional 3D modelling software, such as 3D Studio Max or Maya.

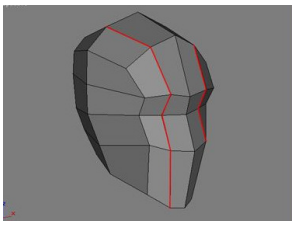
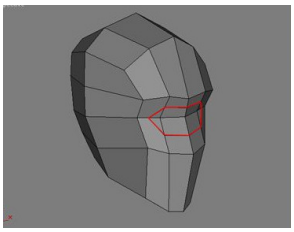
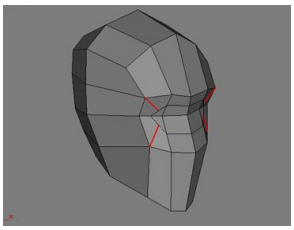
A tutorial by Second Reality (1988) starts with a single cube, which is first converted to a spherical shape by using a 3D Studio Max effect.

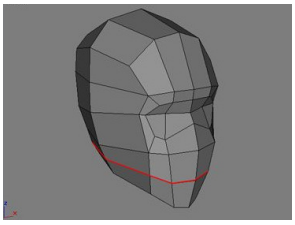
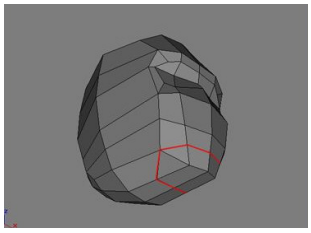
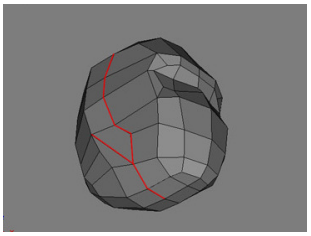
The reason for converting the cube to a sphere instead of starting off with a sphere is to avoid poles, as these are harder to manipulate than the more even quadratic surfaces of a cube.

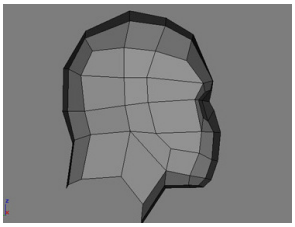
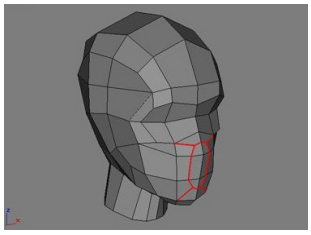
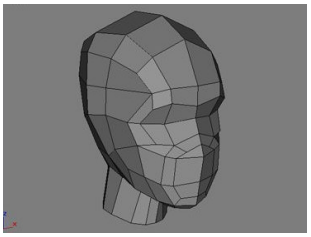
Once it has been created, one half of the sphere is deleted and the remaining half is mirrored using the 'instance-mirror' feature in 3D Studio Max. This ensures that all changes to one side automatically appear on the other side. The face's vertices are then manually moved to roughly resemble the shape of the face.

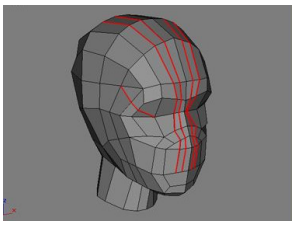
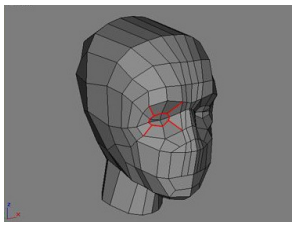
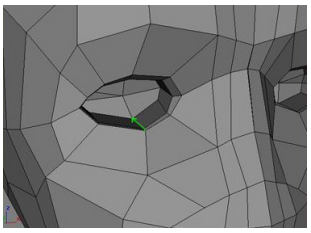
The mesh is further subdivided along the red lines to cater for eyes. A loop is added to outline eye sockets, then used as a border for further subdivision.

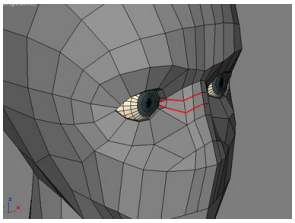
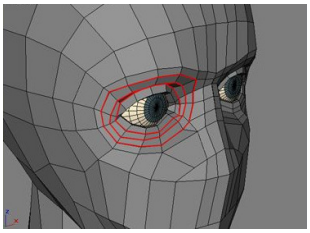
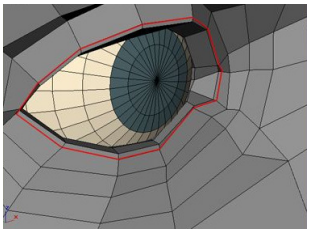
Another loop is added in the position of the mouth. Edges are added to adjust the loop, in order to improve the representation of the mouth. A further vertical loop is added, along with its connection to one vertex.

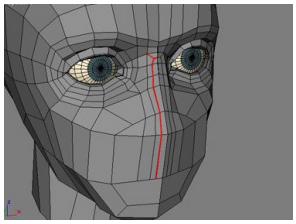
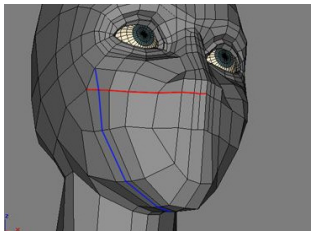
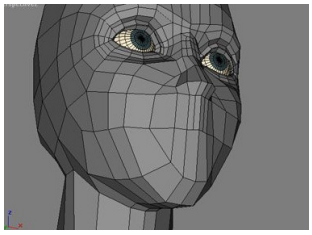
The neck is extruded and the faces under the neck deleted. The shaping of the nose begins, and several new edges are added. Vertices are moved to form the cheeks.

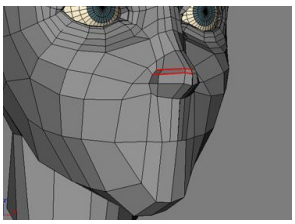
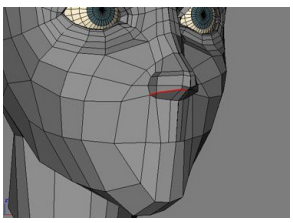
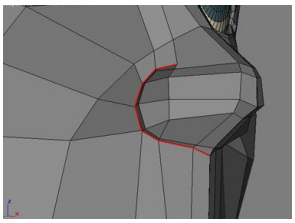
A series of new edges are added, to aid the shaping of the nose and eyes. A loop is added to create an eye socket. The polygons are then inwardly extruded.

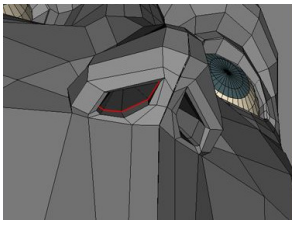
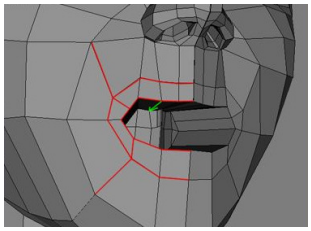
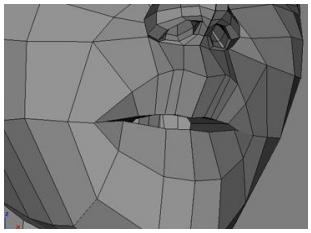
The eyeballs are added, along with ‘tear duct’ creases. Several more concentric loops are added, to help form the eyelids and the creases above and below the eye. A close row of edges helps to sharpen the edge around the eye when it is subdivided. This is to avoid so-called the ‘molten wax look’.

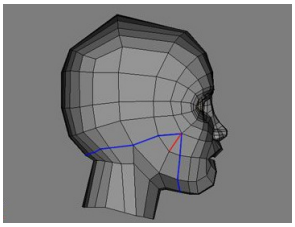
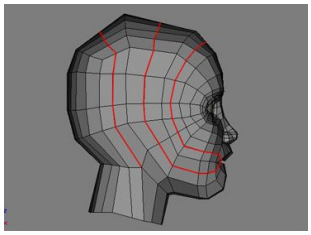
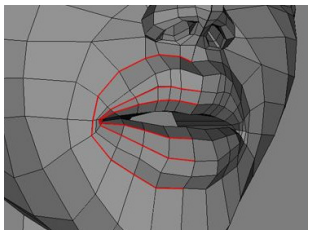
The eyes are further shaped by moving vertices around them. A few more edges are added for the nose. The base of the nose is added by altering an existing loop (to avoid adding unnecessary complexity to the model). The nose is shaped by moving vertices.

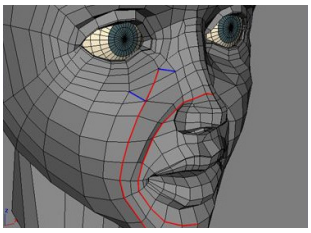
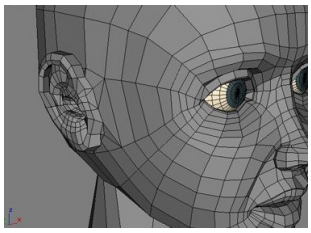
Further edges are added, for a more realistic nose.

Nostrils are shaped and faces removed. Edge-loops and inward extrusion creates an initial mouth. The outer edge-loop forms lips.

Several edges are removed, to avoid the ‘pinch’ effect on the cheek. A few more edge-loops are added, to facilitate shaping of the mouth, as well as yet more loops around the mouth, for more realistic-looking lips.


Additional edges are added, to smoothen the chin. A pre-modelled ear is added to the mesh. The final image is the smooth-shaded result seen above.

A variation of modelling by subdivision (tutorial by Comet, 2001) uses the actual photos as a background in 3D Studio Max, in order to generate the initial mesh (Figure 27).

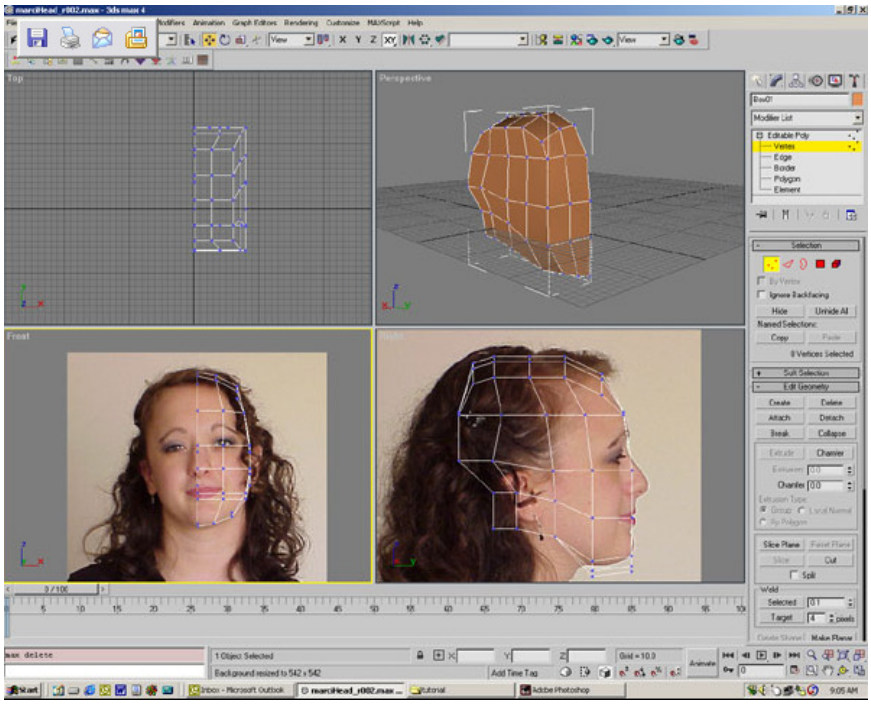


Figure 27: Using photographs to aid the subdivision (Comet, 2001).

An initial problem with the latter approach concerns the optimization of polygon modification. When the user moves a vertex, the software would need to iterate through the entire mesh, in order to identify all the polygons that are affected by moving that vertex. This may become quite intensive in a complex mesh and needs to be optimised. One such optimization is described by Baumgart (1975). In his model, each vertex is aware of its own coordinates and of all its neighbouring vertices, edges and polygons, which essentially eliminates the mentioned search.

5.4.3 Head modelling with splines

Modelling of the head using splines (Birn, 1996) consists of constructing an initial spline going from top of the head over the tip of the nose down to under the chin and neck (blue dots in Figure 28). The initial spline is then copied a number of times and each copy is rotated around the head by uniformly

incrementing the angle. Each copy is then modified to approximate the head and face shape at that longitude.

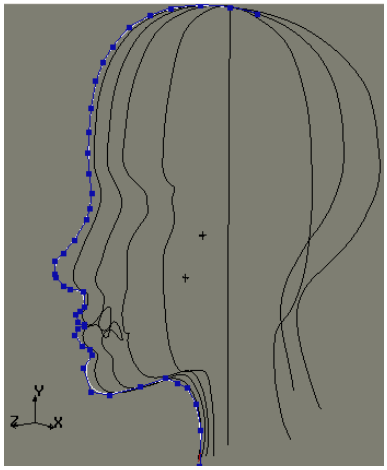


Figure 28: Initial spline and its copies around the head (Birn, 1996).

The corresponding latitude points are then connected and the whole creation becomes a polygon mesh. After a substantial amount of tweaking, the final result is shown in Figure 29.

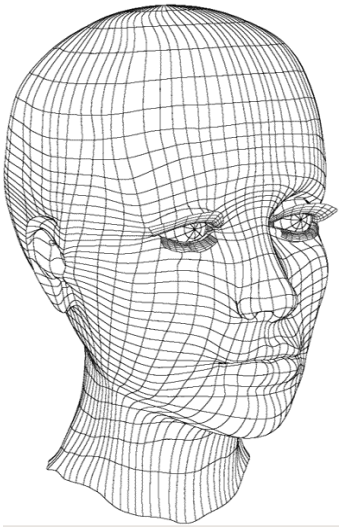


Figure 29: Finalized mesh (Birn, 1996).

5.4.4 Intensity painting

Williams (1990b) studies ways of extending the use of colours into the third dimension. The idea was to use 2D raster data to represent 3D surfaces. In this way, the whole arsenal of existing conventional 2D tools could be used for 3D modelling. He calls this technique 'displacement mapping', a technique which is similar to bump mapping. The difference is that bump mapping only gives the illusion of surface relief by the manipulation of normals, while displacement mapping actually changes the surface coordinates.

Using this technique, 2D image editing tools (PhotoShop™, PaintShop Pro™, and others) become instrumental in automated correction of incorrect and missing data due to imperfection of scanning. Williams demonstrates this on an example of damaged nose data. He claims that it took less than a minute to locate the tip of the nose and then created a smooth nose using the 'smear' filter of the image editing software called ImageStudio (Figure 30).

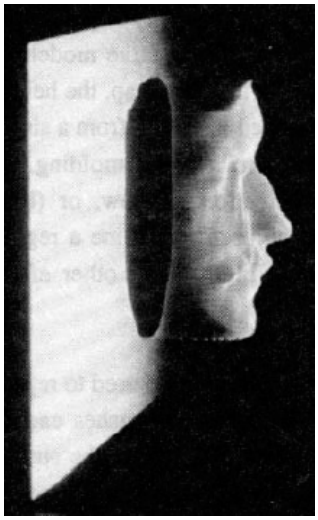


Figure 30: Williams (1990b) – one of the 3D paint images, rendered in 3D.

The major drawback of this technique is that a single image can represent only up to 180 degrees of a 3D object. Two or more images would be needed in order to represent the whole object.

5.5 Summary and conclusion

In this chapter we discussed various methods of creating models of the human head and face. Each method has its unique advantages and disadvantages. The methods evolved through the time and as the equipment advanced, some of the techniques have became more prominent, while others have faded into obsolescence. The most common ways to carry out acquisition today are 3D digitizers and the sculpting methods. The initial disadvantages of 3D digitizers are lately overcomed by scanning from the different angles to include the obscured areas. To overcome the large number of scanned vertices, various polygon reduction algorithms have been designed.

The properties of each acquisition method are outlined in Table 1:

	Contour Reconstruction	**3D Digitizers**	**Photogrammetric Techniques**	**Sculpting methods**
Based on	Multiple 2D images	Scanning	Multiple 2D images	3D software
Cost	Low	High	Low	Moderate to low
Need for human intervention	Moderate to low	Low	Low	High
Advantages	Inexpensive equipment	Relatively fast and accurate process.	Inexpensive equipment. Some heads are only available on photographs.	Human controlled process. Captures artist's imagination.
Applied in practice	No longer used	Used extensively	Practical application exists, but the extent of usage is unknown.	Used extensively
Disadvantages	Unsuitable for solid objects (requires slicing)	Inaccurate over obstructed areas (hair, chin). Acquire too many points, requires subsequent reduction.	Inaccurate when smaller object in front of a larger one.	Tedious and time consuming. Requires significant amount of training.

Table 1: Summary of different head and face acquisition methods.

Regardless of the acquisition method used, if the model is to be used for animation purposes, some manual adjustments will be necessary. The model ultimately needs to be suitable for animation. The density and position of vertices should be appropriate. Existing modelling and acquisition techniques often do not cater for the animation needs. For example, the edges sometimes cross over the moving parts of the face, such as across the upper and the lower lip. Automated data acquisition methods,

such as 3D scanners, acquire evenly spaced vertex coordinates, which result in an equal polygon density. For animation purposes, it is necessary to have higher density of polygons in the more expressive regions of the face. It is also beneficial if the polygonal structure of a particular region is conformable to the muscle fibres, to facilitate realistic movement. Pasquariello and Pelachaud (2001) identified these needs and refined their model in the most expressive areas: mouth and the eyes (as depicted in Figure 31).

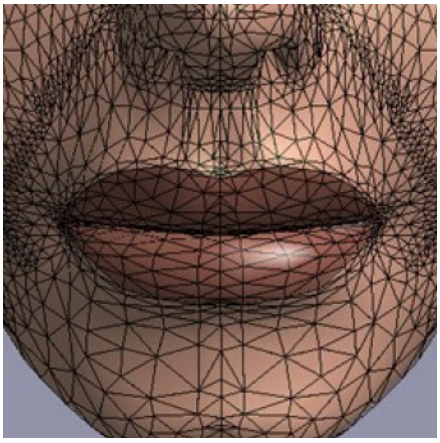


Figure 31: Polygon density and vertex distribution around the mouth (Pasquariello and Pelachaud, 2001).

Chapter 6 Facial animation

Computer facial animation could be loosely described as a set of techniques used to represent the face graphically on a computer system, and then to animate this face in a manner consistent with real human movement.

Roughly divided according to temporal requirements, facial animation can be split into two main categories: *real-time* and *non-real-time*. *Real-time* animation is produced in response to a non-predetermined requirement, such as a user's interaction in a computer game. The computer only learns the details of the animation shortly before the animation is required to begin. During this short lead time, the computer produces the first frame and sends it to the graphics subsystem to be displayed. While the first frame is being displayed, the computer produces the following frame, and so forth.

In the motion picture industry, it is believed that the human eye perceives rapid exchange of images as a smooth motion only if this movement occurs at a rate equal or greater than 24 images per second. This means that the computer is only permitted 1/24 seconds – or around 41.67 milliseconds – to prepare the next frame. Since there is a limit in the number of operations a computer can perform in a unit of time, the real-time animation models need to downgrade their requirements in accordance with the speed of the machine, usually by reducing the desired number of polygons and thus simplifying the rendering process. An example of a real-time animation system is Greta (Pasquariello and Pelechaud, 2001). Since the aim of the system is to perform real-time animation, the authors were forced to simplify the polygonal complexity. Instead of simulating wrinkling of the facial tissue, they had to use bump-mapping to approximate the effect, as it is much faster (bump-mapping is a technique of roughening the surface by changing the vertex normals – Angel, 2003:341-343).

Non-real-time animation, which is used for animated movies, has a predetermined script and thus rendering time is not a critical factor. The quality of the animation is paramount, so the machine is allowed sufficient time in order to achieve a perfect result. The produced frames are subsequently integrated into a video file and their display frequency is matched to the required video standard (PAL or NTSC). Examples of this are the facial models for craniofacial and maxillofacial surgery described by Koch et al. (1996) and Koch, Gross and Bosshard (1998), and special effects in movies such as Matrix Reloaded (Borshukov and Lewis, 2003).

It should be emphasised that strictly classifying facial animation techniques would be an unrealistic task, as the boundaries between techniques are often nebulous. In addition, many approaches integrate several unrelated methods in order to produce better results. Noh and Neumann (1998) made an attempt to classify the various facial modelling and animation approaches in their survey, according to which all facial animation approaches could be grouped into two groups, namely, those based on geometry manipulation and those based on image manipulation.

Geometry manipulation manipulates 3D models. These 3D models consist of vertices in space, forming polygons and giving an illusion of a surface and ultimately, a volume. Shifting these vertices through space in time deforms the perceived surfaces and gives an impression of movement. Image manipulation refers to morphing between 2D images, which provides an illusion of movement. Further taxonomy is rather blurred, as there are often no well defined boundaries between the technologies. It frequently happens that a newer technique uses methods introduced by the older technique, such as direct parameterisation using key-framing and interpolation, in its implementation. Sometimes two or more techniques are used for two different parts of the face.

The implementation of facial animation is often a layered approach. An example of abstraction-layered facial animation is described in Kalra et al. (1991b) in which a language is developed to synchronize speech, emotions and eye motions, in order to naturally specify animation sequences. They identified a need for the creation of abstract entities, to create a system that can be easily manipulated. The system they proposed is independent of the underlying animation system. To achieve the abstraction, they assumed a multilayered approach. Top layers concentrate on ‘what to do’, while the lower ones focus on ‘how to do it’. Intuitively, the facial animation could be logically and functionally divided into two abstraction layers. The higher layer, or muscle control methods, describes the movement required to represent a certain expression. The implementation of the movement is performed within the lower layer, using one or more of the techniques described in Sections 6.2.1 to 6.2.6.

6.1 Muscle control methods

Muscle control methods form the higher layer of the animation process. They provide the data that is required for the implementation layer. Based on functionality, they could be divided into *performance-driven* and *driven by control parameters*.

6.1.1 Performance-driven animation

Performance-driven animation captures the movement of real actors to drive the animation of models (Parke and Waters, 1996:111-113 and 287-307). This idea resulted from attempts to alleviate tedious manual animation, where dozens of parameters need to be perfectly synchronised in order to achieve a believable expression. From the temporal point of view, we could divide the performance-driven animation into two categories: *real-time* and *non-real-time*.

Real-time requires that the animation be displayed simultaneously with the actor's actions. *Non-real-time* records the actor's actions first, after which the computer produces an animation which is not restricted to the time of the actual motion. Obviously, the non-real-time method produces better quality animation, since there is no restriction on computing time and space in producing the animation. Performance-driven animation often uses hardware devices, such as body suits, data gloves and laser or video based motion-tracking systems.

Performance-driven animation techniques are considered to be in a class on their own as they are independent of the implementation. Early attempts to do this can be traced back to the mid eighties (Parke and Waters, 1996:111), for the purpose of creating cartoons. Williams (1990a) first synthesised expressions by changing the 2D texture coordinates using the differences between static images. Guenter et al. (1998) went a step further and derived their data from a video stream. Kouadio, Poulin and Lachapelle (1998) used pre-modelled 3D facial expressions and blending between them to produce a real-time animation. Pandzic et al. (1996) succeeded in doing away with the ubiquitous facial markers by using an edge extraction algorithm to acquire the performance data in real-time. Sifakis, Neverov and Fedkiw (2005) and Sifakis et. al. (2006) added an interesting twist to performance-driven animation by capturing and reproducing three types of facial data: speech, emotions and collision with external objects. This was made possible by employing an advanced physical model with animation based on the finite-element method (FEM), described in Section 6.2.5.4 (Sifakis, Neverov and Fedkiw, 2005).

In performance-driven animation there are several distinguishable data acquisition research problems. Regarding implementation, the research problems are 'shared' with the image-based and geometry-based animation techniques. Performance-driven techniques sometimes still use key-frames and interpolation. One of the main reasons for this is the fact that the large amount of the key-frame data, which would be a major issue if needed to be constructed manually, may now be derived via an automatic acquisition method.

At the top of the list of the above-mentioned performance related issues is the problem of perfecting the method of capturing the performance data, that is, reducing the human intervention to the minimum, in order to diminish or eliminate the face markers and to reduce the need for custom acquisition hardware. Borshukov et al. (2003) used optical flow and photogrammetric techniques to record a live actor's performance. Optical flow refers to a technique of tracking each pixel in time using multiple cameras. The spatial position of each pixel can later be determined using triangulation. Blanz et al. (2003) combined the image- and geometry-based technologies to augment the performance by simulating motion that has not yet been performed. Zhang et al. (2006) also combined the image- and geometry-based technologies, but for the purposes of simulating subtle facial details such as wrinkles that cannot be identified through performance. Zhang et al. (2004) designed a system using several video cameras positioned around the subject (performer) at various angles. No facial markers were used, so the footage was also suitable for texture and lighting purposes.

Video cameras are relatively inexpensive and non-intrusive acquisition hardware. Once the relevant videos have been produced, the computer derives the geometry of the subject using machine vision techniques. Gutierrez-Osuna et al. (2005) created an interesting mixture of existing approaches in their performance-driven audio/visual synthetic system. The generic model contained a number of polygons with identified (standardised) MPEG-4 facial points (FPs – see Section 6.1.2.3). Facial expressions were achieved using muscle action, each of which conforms to MPEG-4 FPs. Although the model represented all of the 'anatomy' of a muscle-based system (mass-spring-based muscles, skull and jaw), the animation was not a free Newtonian physics system. The forces that act on the muscles were compiled or defined in such a way that they conformed to MPEG-4 FPs.

Work on performance-based animation is far from complete. Some of the current research problems include correcting errors that have accumulated over time through the use of optical flow principles, as reported by Borshukov et. al. (2003). These errors have later been addressed by Reutersward et. al. (2005), who improved on the algorithm. Another problem with the performance-driven approach is that extrapolation methods may appear unrealistic to the viewer. In the case of occlusion it has been suggested that the occluded part could be deduced/interpolated from the neighbouring frames. However, Zhang et al. (2004) report that their algorithm does not work well for extrapolated faces.

Performance-driven animation is here to stay and continues to compete with the control parameters approach (Section 6.1.2). Nothing is more natural than the actual expressions expressed by real people. If such expressions are accurately captured and reproduced, the results are quite astonishing.

Advantages:

- The chief advantage of the performance-driven animation lies in the reduced need for human intervention.

Disadvantages:

- Making changes is tedious, as all the parameters are explicitly defined for each frame. This process is far simpler when using the key-frame approach, where the key-frames are set further apart.
- There is a possibility of occlusion of the markers, in which case the motion would be lost.

6.1.2 Control parameters methods

Once the basic animation principles are in place, albeit using simple key-frame or advanced muscle-based methods, there is still a long way to go to creating lifelike and convincing facial movements. It may be possible to, for example, control dozens of facial muscles, but which ones exactly should be contracted in order to create a smile? The number of combinations involved here is just too large for a trial and error approach. Significant research efforts have been dedicated to finding various coding standards that are independent of the basic animation principles and which result in various accurate facial expressions. These are higher level languages and/or scripts that describe which parts of the face need to move in order to produce the desired facial expression. This is usually provided in the form of list of muscles and percentages of the required contraction of these muscles in order to produce certain facial expressions. The actual contraction mechanism of these muscles is left entirely up to the fundamental modelling and animation techniques. Their contribution to animation is in the highest level of ‘what needs to be done’ in order to achieve certain facial expressions. A greater problem lies in ‘how to do it convincingly enough’, which continues to be the subject of extensive research.

6.1.2.1 FACS

The most widely used coding system for facial animation is the Facial Action Coding System (FACS) (Ekman, Friesen and Hagar, 2002). FACS was developed with the intention of describing every

possible facial movement. It can be learnt in a relatively short period of time and is therefore accessible to both the scientific and non-scientific community alike. It addresses only facial movement and ignores facial topology and surface. The expressions only include movement visible to the human eye; all other types of facial movement are ignored in favour of simplicity.

FACS has been used by many scientists researching facial animation as their high layer of abstraction, regardless of the underlying animation technique.

6.1.2.2 MIMIC

Fuchs, Haber and Seidel (2004) designed a specification language called MIMIC (Figure 32 and Figure 33) that can be used along with any other animation system that utilizes parameters varying over time to control animation, such as FACS. The main benefit of this system lies in the significantly reduced amount of time spent doing tedious and awkward work, as is required when specifying all the parameters manually.

```
defxpr smile "smile.xpr"
main {
[
  head_rot_y <in=5,hold=10,out=5,sin,value=-20>;
  eyelid_left <left=6,in=3,hold=2,out=1,sin,value=0.8>;
  head_rot_x <in=5,hold=10,out=5,exp,value=10>;
  jaw <min=0,max=0.09,freq=0.1,sin,duration=20>;
  lookat_x ( 0   12   20;
             0   100  0; )
]
smile <in=5,hold=10,out=5,value=1,sin>;
}
```

Figure 32: Sample of MIMIC code (Fuchs, Haber and Seidel, 2004).

Enhancements to this approach may include the introduction of dependencies between different animation parameters, in order to automate action, and speech synchronized animation which would extend the temporal alignment of actions to phonemes.

There has been very little mention of this technology since its initial publication and it is questionable whether it will be adopted by the scientific community in any significant way in the long term.

Figure 33: Resultant animation – executed MIMIC code snippet (Fuchs, Haber and Seidel, 2004).

6.1.2.3 MPEG-4

MPEG-4 is an ISO/IEC standard developed by Motion Picture Experts Group in 1999. Amongst other aspects, MPEG-4 defines specifications for the animation of face and body models (Doenges et al., 1997 and Ostermann, 1998). It proposes a set of Facial Animation Parameters (FAP) that deforms the face in a certain manner from its neutral position. The neutral state has its specific requirements, such as:

- Gaze in the direction of Z-axis;
- All muscles are relaxed;
- Eyelids are tangents to the iris;
- Lips are in contact;
- The mouth is closed;
- Upper teeth are touching the lower ones;
- The tongue is flat;
- Tip of the tongue is touching the boundary between the upper and lower teeth.

In order to be generic and thus cater for faces with different proportions, MPEG-4 stipulates the use of ratios, rather than absolute measurements. FAP values are therefore measured in so-called face animation parameter units (FAPU). FAPU are defined as fractions of distances between the key facial features.

Apart from FAP, MPEG-4 defines 84 facial feature points (Figure 34). The feature points define facial features and the shape of the face, and provide a reference for defining FAPs.

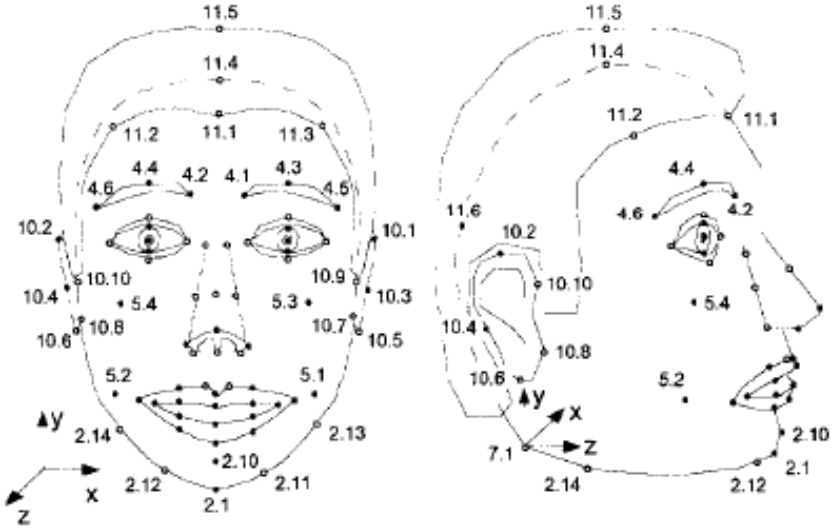


Figure 34: Examples of MPEG-4 facial feature points (Ostermann, 1998).

At a higher level, the FAP set contains visemes and expressions. Visemes correspond to the visual representation of phonemes. Only 14 static visemes are defined within the standard set (Figure 35).

#	Phonemes	Example	#	Phonemes	Example
1	p, b, m	put, bed, mill	8	n, l	lot, not
2	f, v	far, voice	9	r	red
3	T, D	think, that	10	A:	car
4	t, d	tip, doll	11	e	bed
5	k, g	call, gas	12	I	tip
6	tS, dZ, S	chair, join, she	13	Q	top
7	s,z	sir, zeal	14	U	book

Figure 35: Visemes in MPEG-4 (Ostermann, 1998).

Coarticulation and transition from one viseme to another is achieved by blending the two using a weighting factor. Expressions are non-vocal movements that express excitation of the expression. Here again, two expressions can be blended with a weighting factor (Ostermann, 1998). For a detailed exposition of coarticulation, the reader is referred to Section 9.3 and to Cohen and Massaro (1993) and Massaro (1998).

If a particular facial animation is coded according to the rules of the MPEG-4 standard, it can be read by an MPEG-4 decoder/terminal which would then perform the animation without the user having to resort to lower level programming. The implementation of a decoder is not stipulated by the MPEG-4 standard, and it may therefore utilise any of the implementation techniques described in Section 6.2. At a conceptual level, MPEG-4 defines three decoder profiles, classified by the extent of their external configurability (Ostermann, 1998):

- *Simple Facial Animation Object Profile* – the decoder has its own facial model that is animated by a FAP stream. It receives the FAP stream, decodes it, and then performs the animation internally in accordance with the received stream.

Figure 36: Greta – a simple facial animation object profile MPEG-4 decoder (Pasquariello and Pelachaud, 2001).

- *Calibration Facial Animation Object Profile* – a superset of the simple profile, the decoder can also receive the calibration data for facial features. This provides a limited external access to the modification of the facial features of the decoder's internal facial model.
- *Predictable Facial Animation Object Profile* – a superset of the calibration profile, the decoder can also receive the entire custom facial model and animate it.

An example of a Simple Facial Animation Object Profile decoder is Greta (Pasquariello and Pelachaud, 2001), visible in Figure 36. Greta contains a proprietary facial model that is directly animated by the FAPs. The model is composed of a polygonal surface and is powered by a pseudo-muscle-based animation system (see Section 6.2.3). The effect of facial actions using FAPs is illustrated in Figure 37.

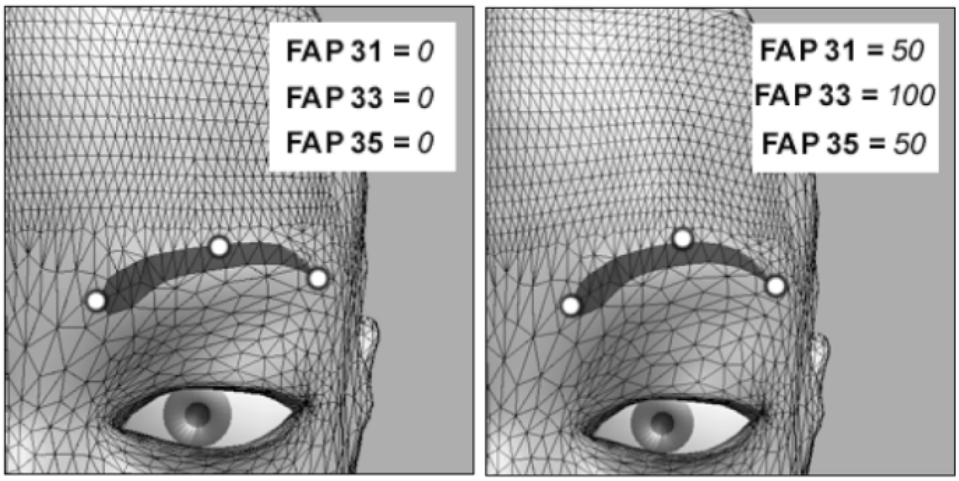


Figure 37: Change in FAP under the action of the frontalis muscle (Pasquariello and Pelachaud, 2001).

6.1.3 Summary and conclusion

This section discussed methods used to specify the control of muscles for animation purposes. Regardless of the implementation, data required to deform the face will need to be produced. That is, spatial and temporal coordinates and/or muscle parameters need to be fed to the implementation layer. The two fundamentally different methods of achieving this were described, namely performance-

driven and control parameters. Relative advantages and disadvantages of one method over the other are difficult to determine in an objective manner. They are mostly of qualitative nature, and depend on a viewer's perception of motion quality and believability. Essa et. al. (1996) outlined the differences between FACS and the performance-driven methods of the mid-nineties. They argued that control parameters do not simulate the complex coarticulation between the facial muscles to a sufficient degree. An example of this is a natural smile, where in addition to lip movement there is a subtle movement of the corner of the eyes. This motion may not normally be taken into consideration by control parameters systems, while it would be identified and reproduced by a good performance-driven method. However, intuitively the control parameters, by the very nature of their functionality, will always be behind the data derived through accurate capture of an actual subject's facial expressions. Control parameters are usually localised, while real facial motions often include multiple aspects of the face.

Another problem reported with FACS is that it does not specify the exact temporal parameters. The interpolation between two parameters, for example, would result in a constant rate of change between the two values, which would result in the animation appearing somewhat flat and mechanical. The natural facial expression rate of change is usually not linear, and varies through its stages.

As mentioned in Section 6.1.2.3, coarticulation and transition from one viseme to another in the MPEG-4 specification is achieved by blending the two with a weighting factor (described in detail by Cohen and Massaro, 1993 and Massaro, 1998). The objective success of this method is not reported. Intuitively, such temporal aspects would be far easier to capture using performance-based methods.

Both the performance-driven and the control parameter approach are still extensively used in the industry.

6.2 Geometry-based animation techniques

It is generally agreed that geometry manipulation methods can be roughly divided into key-framing, parameterisation, pseudo-muscle methods, and physics-based methods (Tang, Liew and Yan, 2004), while image manipulation methods include morphing and blendshaping (Deng et al., 2006). These techniques are discussed in the following several sections (image-based techniques are not covered, as this thesis focuses on geometry-based techniques). The emphasis is on muscle-based methods, as they seem to be the future in the field.

6.2.1 Key-frame based animation with interpolation

As far back as 1972, Parke represented a face using 250 polygons defined by 400 vertices (Parke, 1972). Animation was carried out using the key-frame and interpolation techniques. Interpolation was not linear, but based on cosines. The modelling of the acceleration and deceleration of facial movements benefitted from the smoothness and varying speed through time, resulting from cosine interpolation, in contrast with the discontinuities at the data points and constant tempo movement of linear interpolation.

In general, interpolation between values val_1 and val_2 can be described by the following expression:

$$val = (a)val_1 + (1-a)val_2, \text{ where } 0 < a < 1. \tag{6-1}$$

The above expression is of a single-dimensional nature, but it can easily be applied in 3D by interpolating over each dimension separately. A group of vertices would be subject to a common interpolation, to form a facial expression as a function of time. Cosine interpolation, as one of the simplest non-linear interpolation methods, is in common use in computer graphics. It may be formalised as follows:

$$\text{Let } a_1 = \frac{1-\cos(a\pi)}{2} \quad \text{with } a \in (0,1). \tag{6-2}$$

$$\text{Then } val = val_2 a_1 + val_1(1-a_1). \tag{6-3}$$

We note that $a_1 \in (0,1)$ too, and that it only alters the speed of change, which is visible on Figure 38 (the pink line represents the linear interpolation, while the blue one represents cosine).

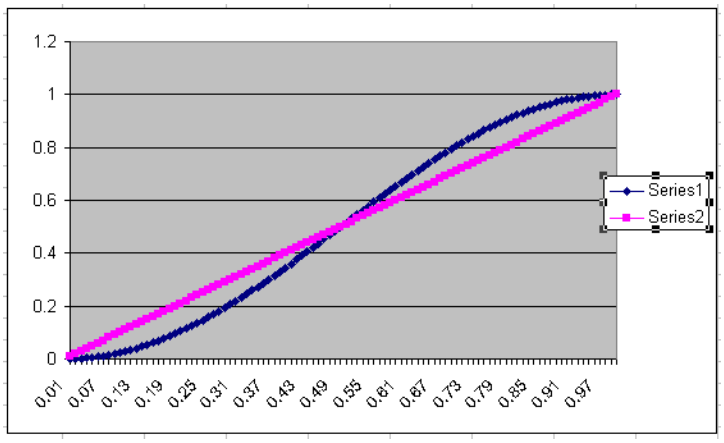


Figure 38: Linear and cosine interpolation.

Apart from the cosine-based nonlinear interpolation, there are several other functions that could be used for this purpose, along with other parametric curves, surfaces and blends of two of more of them (Brotman and Netravali, 1988 and Kochanek and Bartels, 1984).

Animation by interpolation assumes that the surface topology is fixed. Changing a face's expression involves moving each designated surface vertex by a small distance for each time frame. It should be noted that key-frame based animation techniques are, to a greater or lesser extent, used in almost all of the other more advanced techniques as a part of the animation process.

Advantages of this system include:

- Simplicity of understanding and implementation;
- It is not computationally intensive;
- Results are highly predictable.

Disadvantages include:

- Interpolation is restrictive, as there is a limited number of key-poses and any expression that does not conform cannot be achieved;
- Each key-pose requires explicit data-collection of vertex co-ordinates. For a large set of key-poses, this is a sizeable task;
- It is difficult to provide an interface to animators (orthogonal control parameters);

- The speed of animation may not be linear or it may not conform to any known function, in which case it would be difficult to calculate timing of the intermediate frames;
- Conversely, the path between two key frames may not be linear, nor does it necessarily conform to any existing function, making it difficult to calculate the position of intermediate frames.

6.2.2 Direct parameterisation

Parameterisation consists of creating what is usually a small set of parameters that characterize an object, or are considered sufficient to describe the object. For example, to describe a sphere, we may use the Cartesian coordinates of its centre and its radius. For simple objects, developing this set of parameters is relatively simple and intuitive.

The idea behind direct parameterisation is to create a model that is able to generate a wide range of faces and facial expressions based on a small set of input parameters. The ultimate goal would be to create a model which is able to assume every possible expression. However, most current models are still far from the ideal.

Developing a parameterised system involves two main steps:

- Development of a set of parameters;
- Development of a model that displays a face using the above set.

One could develop a set of parameters by simply observing the surface properties of the face in question, and then deduce the set of parameters from there. The second approach derives from studying the underlying anatomy and developing a set of parameters based on that. There is also a hybrid approach in which the study of anatomy is performed, but the final decision as to whether or not a parameter is required is based on observation of visibility of its presence.

Control parameters can be divided into those concerning either *expression* or concerning *conformation* (Parke and Waters, 1996:191). *Expression* control parameters usually deal with aspects such as opening of the eyelid, eyebrow arch, eyebrow separation, jaw rotation, width and expression of the mouth, upper lip position, mouth corner position and eye gaze. *Conformation* control parameters usually deal with width of the jaw, forehead shape, length and width of the nose, shape of the cheek and chin, shape of the neck, eye size and separation, face region proportions and overall face proportions.

The best known parameterised model is the one described by Parke and Waters (1996:187-222), which is essentially the same model presented by Parke for his PhD in 1974. In this model, image generation is based on polygonal surfaces. It contains several independent surfaces, such as facial mask, eyes and teeth. In this model, the topology always remains the same. Topology refers to the number of vertices and edges between them. The 3D positions of these vertices vary according to the parameters. As the vertex positions change, the surface of the face flexes and stretches, changing its shape.

Skin crease effects are achieved by modifying surface normals, causing discontinuity in the shading. The face is assumed to be symmetrical and only one side is modelled. The other side is derived by copying and mirroring the modelled half.

The transition between two expressions is done using interpolation. This method requires that the old and new position for each individual vertex is known and that the surface topology is constant, that is, each vertex maps to one and only one vertex of the new expression. Interpolation is applied locally to regions of the face, rather than globally to the whole face.

Parke and Waters distinguish between five distinct procedures that make use of the previously mentioned parameters to determine vertex positions:

- *Procedural Construction* is used to model eyes. It accepts parameters for eyeball, iris and pupil size, along with iris colour, eye position and eyeball orientation. It ultimately generates the polygon descriptors for the eyes.
- *Interpolation* is used for most of the facial regions that change shape, for instance forehead, cheekbones, neck, mouth, and others. Each area is first independently interpolated to their extreme positions, and then associated with a parameter value.
- *Rotation* is used to open the mouth by rotating the bottom jaw around its axis.
- *Scaling* controls size and placement of the facial features, for instance nose width, chin prominence, and others.
- *Position Offsets* control length of the nose, corners of the mouth and rising of the upper lip. They move the entire set of related vertices as a group.

Advantages:

- Relative advantage over key-framing (Section 6.2.1) – overcomes the problem of rigidity by grouping together vertices required to perform a certain task;

- Relative advantage over pseudo-muscle (Section 6.2.3) and muscle-based (Section 6.2.5) approaches – lower computational cost.

Disadvantages:

- The approach is not flexible, as it is bound to a particular topological mesh;
- It concentrates on skin only, mostly ignoring motivators of the dynamics of surface deformation;
- Conflicts between the parameters occur frequently, producing undesirable animation results.

6.2.3 Pseudo-muscle based approach

The formation of a facial expression is a complex process, involving the synchronised participation of a number of muscles. In addition, it causes visible skin deformations, in accordance with complex skin dynamics. Modelling this process anatomically requires considerable computing power. For this reason, scientists observed the results of the facial expressions in order to find a way to replicate these results without resorting to a complex anatomic modelling. A result of this attempt is the so-called pseudo-muscle based approach. It attempts to emulate the essential subset of facial muscles by directly deforming the facial mesh.

Advantages:

- It is simpler than muscle-based methods, as it does not replicate the anatomy;
- Less processor intensive than muscle-based methods, as the machine does not have to compute the muscle actions and tissue dynamics, but deform the mesh directly.

Disadvantages:

- Fails to display wrinkles and bulges, hence impairing the realism;
- Interaction between muscles is often neglected.

6.2.3.1 Abstract muscle action

The Magnenat-Thalmann, Primeau and Thalmann (1988) approach is based on muscle action abstraction, rather than actual muscles. They designed a set of procedures used to generate facial expressions, which is a more advanced method than the direct parameterised model. They called this set of procedures Abstract Muscle Action (AMA). Each procedure works on a specific region of the

human face, defined when the face is constructed. For complex muscles (such as lips) several procedures, each responsible for a simple motion, are allocated. By combining these procedures, a large number of expressions can be produced. Examples of AMA procedures are RIGHT_EYELID, MOVE_LEFT_EYE_VERTICAL and RIGHT_LIP_RAISER.

The authors divided facial expressions into two groups, namely phonemes and emotions. A phoneme is a facial expression which only uses the mouth motion and directly contributes to speech. An emotion is a facial expression which acts on many parts of the face, for example crying, smiling and laughing.

While faces may initially be created using any modelling technique, AMA procedures impose certain requirements, such as:

- The face is assumed to be symmetrical;
- The procedures may need to be scaled by a factor, along with the face model;
- The face needs to be divided into specific regions (such as skin, teeth and eyelids).

6.2.3.2 Freeform deformations

Freeform deformations (FFD) were first introduced by Sederberg and Parry (1986). They describe freeform deformations as a technique for deforming solid geometric models in a free-form manner. Kalra, Mangili, Magnenat-Thalmann and Thalmann (1991) provided the following compact formal definition of this technique:

FFD involves a mapping from R^3 to R^3 through a trivariate tensor product Bernstein polynomial (Angel, 2003:496). Physically, FFD corresponds to deformations applied to an imaginary parallelepiped of clear, flexible plastic in which are embedded the object(s) to be deformed. The objects are also considered to be flexible so that they are deformed along with the plastic that surrounds them. Mathematically, imposing a local coordinate system (S,T,U) on a parallelepiped region with origin at X_0, a point X has (s,t,u) coordinates in this system such that

$$X = X_0 + sS + tT + uU \tag{6-4}$$

A grid of control points P_{ijk} ($i = 0$ to l, $j = 0$ to m, $k = 0$ to n) is imposed on the parallelepiped.

The location of these points are defined as

$$P_{ijk} = X_0 + \frac{i}{l}S + \frac{j}{m}T + \frac{k}{n}U \tag{6-5}$$

The (s,t,u) coordinates of X can be found by the following equations:

$$s = \frac{TxU.(X - X_0)}{TxU.S}\ ,\ t = \frac{SxU.(X - X_0)}{SxU.T},\ u = \frac{SxT.(X - X_0)}{SxT.U} \tag{6-6}$$

For any point interior to the parallelepiped, $0 < s < 1$, $0 < t < 1$, $0 < u < 1$, the deformation is specified by moving the control point(s) from their undisplaced latticial position. The deformed position X' of a point X is computed from the following equation:

$$X' = \sum_{i=0}^{l}\sum_{j=0}^{m}\sum_{k=0}^{n} P_{ijk} B_i^l(s) B_j^m(t) B_k^n(u) \tag{6-7}$$

where $B_i^l(s)$, $B_j^m(t)$, $B_k^n(u)$ are the Bernstein polynomials defined as

$$B_i^l(s) = \binom{l}{i}(1-s)^{l-i}s^i \tag{6-8}$$

$$B_j^m(t) = \binom{m}{j}(1-t)^{m-j}t^j \tag{6-9}$$

$$B_k^n(u) = \binom{n}{k}(1-u)^{n-k}u^k \tag{6-10}$$

FFD can be described as multiple deformable objects placed into a cube-shaped mould and moulded together with a clear plastic compound (Figure 39).

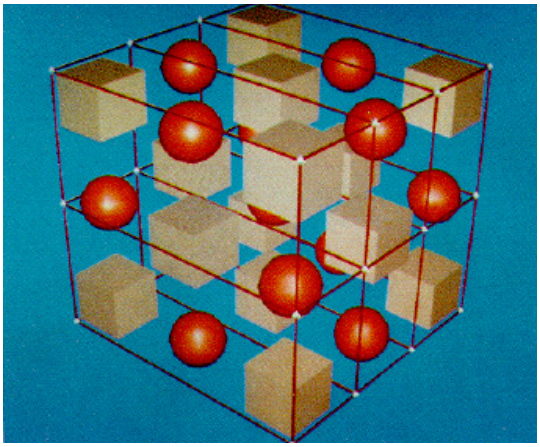


Figure 39: Initial imaginary cube with objects that are to be deformed (Sederberg and Parry, 1986).

Once the mixture has cooled down and is taken out if the mould, it looks like a clear plastic cube with various objects embedded inside. If the cube is then deformed, all the objects inside it would be deformed in a similar fashion (Figure 40).

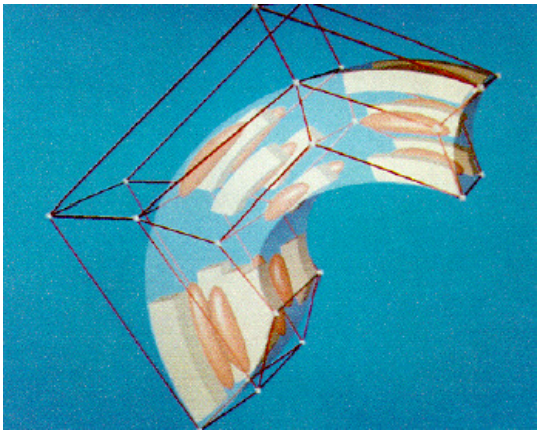


Figure 40: Deformed imaginary cube with control points (Sederberg and Parry, 1986).

FFD was used by Chadwick, Haumann and Parent (1989) to design a technique for controlling muscle actions based on deformation of skin surfaces. They divided the animation tasks into a *skeleton* and *muscle* with a *fatty tissue* layer. Using this technique, the skeleton is not deformable, but constrains deformation. The muscles map to the underlying skeleton and use the FFD constrained by

parameters to deform. On the basis of the type of constraints used, they differentiate between *kinematic*, *dynamic* and *sculpted* deformation.

Kinematic deformation (Figure 41) is consistent with the kinematic skeletal state. The skeleton is used as a base for muscle deformation (unlike in reality, where the situation is actually the opposite). Muscle action is determined by two parameters; elasticity and contractility (the ability of the muscle to shorten when activated by nervous stimuli). This method is more suitable for parts of the body with a greater number of bones and joints.

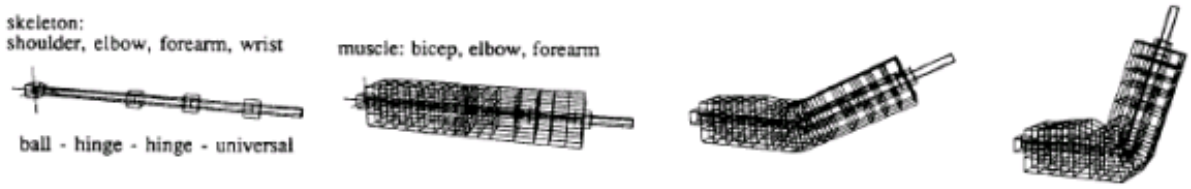


Figure 41: Kinematic deformation (Chadwick, Haumann and Parent, 1989).

Dynamic deformation is suitable for capturing the dynamic properties of soft body structures. The structure is a mass-spring model. Figure 42 represents an extract from the structure. It has one point mass at each of the eight corners, each of which is connected to the other seven with springs. Dynamics are simulated by calculating the spring forces and applying them to the point masses for each frame.

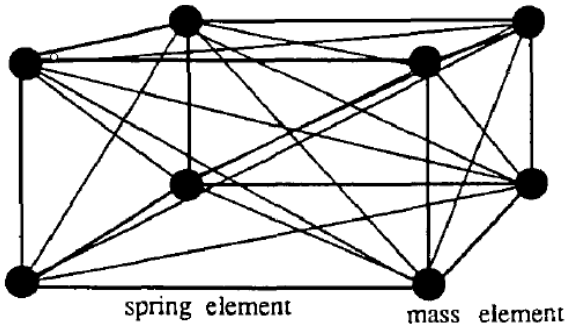


Figure 42: Dynamic deformation (Chadwick, Haumann and Parent, 1989).

Sculpted deformation is the most general, but at the same time the most labour intensive deformation method. Using this method, the animator defines the deformation by moving the individual control points. The benefit of this is that the animator is not constrained by the limiting nature of a set number of parameters.

The main advantage of FFD is its ability to produce a wide range of deformations without requiring multiple shapes. Despite being quite old and largely obsolete as a technique in its own right, this method is still in use as a component of some modern animation techniques for mesh warping purposes (Chang and Jenkins, 2008).

6.2.3.3 Rational free form deformations

Rational free form deformations (Kalra, Mangili, Magnenat-Thalmann and Thalmann, 1991 and 1992) are an extension of the free form deformations discussed above. They introduce another dimension and one more degree of freedom of manipulating the deformations by changing the weights at the control points. A basic form of RFFD is when all weights are equal, which then corresponds to FFD. They described the RFFD formally as follows, which is an extension to the expressions in Section 6.2.3.2.

The rational basis functions allow incorporation of weights defined for each of the control points (W_{ijk}) in the parallelepiped grid. With the new formulation the equation (6-7) changes as follows:

$$X' = \frac{\sum_{i=0}^{l}\sum_{j=0}^{m}\sum_{k=0}^{n} P_{ijk} B_i^l(s) B_j^m(t) B_k^n(u)}{\sum_{i=0}^{l}\sum_{j=0}^{m}\sum_{k=0}^{n} W_{ijk} B_i^l(s) B_j^m(t) B_k^n(u)}. \qquad (6\text{-}11)$$

The face is divided into smaller regions, using anatomical division in such a way that each region covers actions of the muscle(s) the deformation is meant to handle. A parallelepiped is then constructed around each region. Muscle actions are achieved by moving the lattice control points and changing their weights. This displacement is still much simpler than simulating the actual muscle, for fairly good results.

6.2.4 Biological background and tissue mechanics

This section provides an overview of the composition of facial tissues and their mechanical reaction to forces applied to them. This is particularly important for *muscle-based animation*, where simulation of the real muscle actions is attempted.

6.2.4.1 Facial Tissue

Human skin tissue is fairly complex (see Figure 43). It consists of two main layers: the epidermis on the surface and dermis below. The third layer, hypodermis, is not strictly part of the skin, but it is considered for the purpose of modelling and animation, as it influences skin mechanics. Because the study of the mechanics of facial tissue is in a way similar to the studies of material sciences, it assumes an engineering point of view. Soft tissue is viscoelastic in its response to stress, force, strain, deformation and stretch (Parke and Waters, 1996: 223-227).

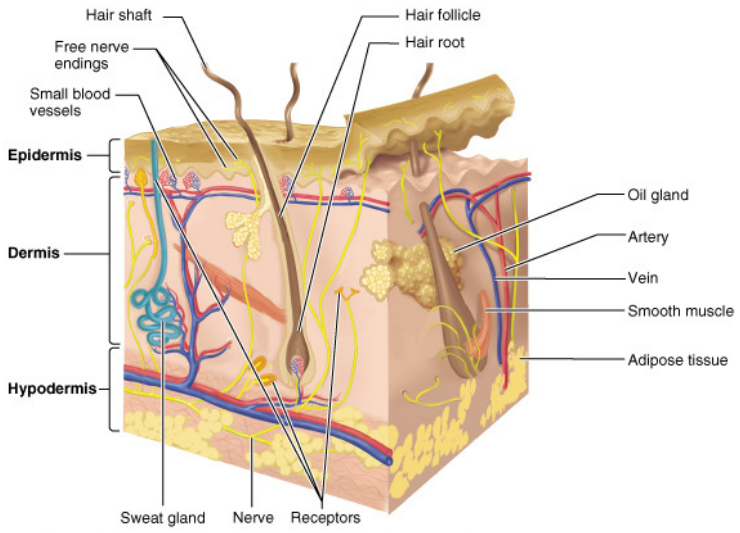


Figure 43: The skin model (Pennsylvania State University, 2001).

It has properties of both elastic solids and viscose fluids. Its elastic nature reflects in its storage of energy and also in its tendency to return to its original shape, once the force is removed. The relation between the force and the deformation is non-linear, as we can see in Figure 44. The viscous nature of a tissue is reflected in the behaviour of internal forces. They vary not only by the amount of deformation, but also by the rate of deformation. Under low stress, dermal tissue offers low resistance

to stretch as the collagen fibres begin to uncoil in the direction of the strain, but under greater stress the fully uncoiled collagen fibres resist stretch much more markedly (Zhang, Prakesh and Sung, 2001).

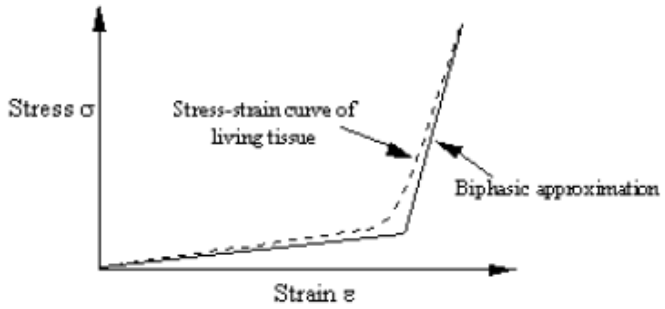


Figure 44: Stress/strain correlation of the facial tissue (Zhang, Prakesh and Sung, 2001).

In addition to pure elastic and pure viscous properties, the soft tissue exhibits the following additional properties (Parke and Waters, 1996:224-225):

- Hysteresis – change in response of the material under cyclic loading and unloading;
- Stress relaxation – reduction in the force opposing deformation through time;
- Creep – increase of the strain over time under a constant load;
- Preconditioning – repeated application of the same load results in different deformation responses.

Most of the skin's elasticity comes from its two ingredients, elastin and collagen. Elastin has rubber-like properties, with a linear stress/strain response. Collagen has stronger stress response and a limited range of deformation.

The viscous characteristic of skin comes from fat cells and the ground substance, which are composed mostly from water. When soft tissue is compressed, fat cells and the ground substance are forced outwards, perpendicular to the compression vector (Poisson effect).

The composition of the soft tissue of the face changes with age. Younger people have more collagen than elastin, and this ratio slowly reverses with age.

6.2.4.2 Muscles

A muscle consists of a group of muscle fibres. Each muscle fibre is composed of even smaller elements called myofibrils, between several hundreds and several thousands of them for each fibre. Along each myofibril, there is a pattern of filaments called sarcomeres. They contract, developing tension along their axis (Figure 45).

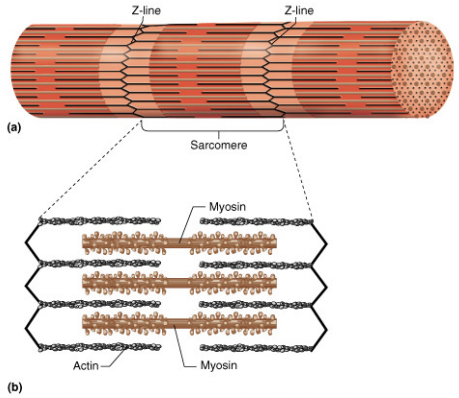


Figure 45: Image of a myofibril, showing sarcomeres along its axis (Pennsylvania State University, 2001).

At a basic level, the actual contraction is performed by the action of two proteins that form the filaments contained within a sarcomere: *myosin* and *actin* (Figure 46).

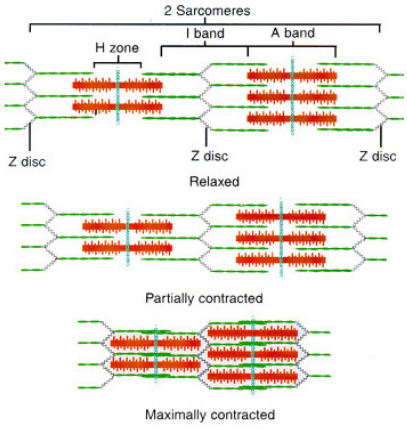


Figure 46: Muscle contraction and relaxation (Pennsylvania State University, 2001).

Facial muscles are voluntary muscles that generally arise from the bone or fascia of the head (Waters, 1987). At the highest abstraction, they could be divided into two groups: the muscles of the upper and lower face. In the lower face, there are five major groups:

- Uppers and downers – used to move the face upwards towards the brow and downwards towards the chin;
- Those that contract towards the ears and towards centre of the face;
- Oblique muscles that contract from the lips, upwards and outwards to the cheek bones;
- Orbitals – rounding the eyes and mouth;
- Sheet muscles which perform miscellaneous actions and the platysma muscles that move the mouth and jaw.

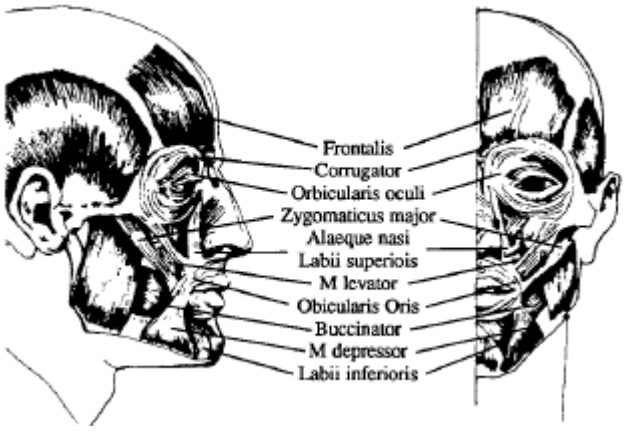


Figure 47: Major muscles of the face (Waters, 1987).

Waters (1987) listed the major facial muscles (Figure 47), but then grouped them into three major groups: linear (labii inferioris, labii superioris), sheet (frontalis) and sphincters (orbicularis oculi, orbiculars oris). Linear and sheet muscles have similar mechanical properties and are often analysed together. Most linear facial muscles are attached to a bone on one side, while the other side attaches to the soft tissue of the skin. The reason for this division is for modelling and animation purposes, as explained in Section 6.2.5, which discusses muscle-based animation.

6.2.5 Muscle-based approach

Parke and Waters (1996:55) claimed that there is no facial animation model based on the complete detailed anatomy of the human face. There is still no evidence of such model at the time of writing of this thesis. Although a muscle-based model seems to be the animation approach closest to the actual anatomy, it still does not replicate the complex assembly of bones, cartilage, muscles, vessels or skin. It focuses on the external manifestation of facial movement and concentrates on physical characteristics of the muscle and skin.

According to Parke and Waters (1996:131), the first worthy attempt at muscle-based animation was made by Platt and Badler (1981). Their approach involved FACS Action Units (Ekman, Friesen and Hagar, 2002 – see Section 6.1.2.1) as a source of parameters. The interesting part was the underlying architecture, which consisted of various layers of abstraction. At its lowest level, this method commenced with a point in 3D space. Arcs were used to represent the relationship between points (for example, elasticity information between two points of a skin surface). The simplest structure for force application was a muscle fibre (Figure 48). It consisted of a muscle point M, a bone point B and one or more skin points S.

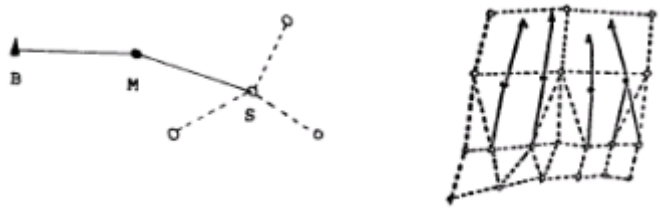


Figure 48: Muscle fibre (left) and muscle (right) – Platt and Badler (1981).

When force F is applied, vertex S is moved towards vertex M by offset S'

$$S' = \frac{F}{k} \tag{6-12}$$

where k represents the sum of spring constants at that point. This offset is calculated for all vertices to which the force propagates.

A higher abstraction from muscle fibres is the collections of fibres – the actual muscles, as depicted in the second diagram in Figure 48. When a muscle action is performed, all fibres in a particular muscle perform in parallel. There were several problems and shortcomings with this model. With muscles following the flow of a bone sheet (such as the area at the junction of an eye socket and its brow), a sudden change of direction would not be adequately modelled – the flesh may flow through the bone. The following aspects were not included in the model:

- Jaw actions;
- Cheek actions, such as puffing and sucking;
- Non-rigid objects, such as the tongue;
- Surface interactions, such as the tongue touching teeth.

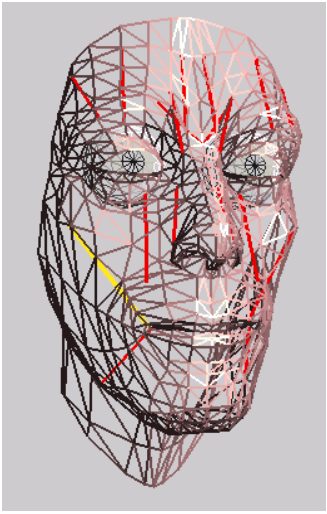


Figure 49: Simple muscle-based model for animation (Parke and Waters, 1996).

Even though Platt and Badler's (1981) was the first effort, nearly all muscle-based approaches to date refer to Waters (1987) as the pioneering contribution in muscle-based animation. Like Platt and Badler, he used FACS for the interface and the mass-spring concept to model the skin and muscles. He also improved on Platt and Badler's concept in the following aspects:

- Waters described three types of muscles – linear and sheet muscles that pull, and sphincter muscles that squeeze.

- His muscles had directional properties independent of bone structure, enabling them to be ported to diverse facial topologies.

Linear and sheet muscles are defined as linear muscle vectors, while the sphincter muscle is defined as an elliptical contraction. The main difference between linear and sheet muscles is that linear muscles contract towards a central point, while sheet muscles contract towards an entire area. Waters released the code for the above model, freely downloadable on the Internet. It is written in C, using OpenGL. Figure 49 is a screenshot from the running program, illustrating mesh and the muscles (depicted using red lines). The code uses three data files to build the face: a faceline file, an index file to construct the face mesh, and a muscle data file containing muscle descriptions.

Figure 50: Simple muscle-based model, smooth shaded and with skin texture (Parke and Waters, 1996).

In Figure 49, the red lines represent muscle vectors, while the yellow line represents the currently active muscle that can be contracted using the keyboard. The model has several preset expressions, such as sadness, fear and disgust. The transition between expressions is instantaneous, rather than interpolated. There is also little control over restricting the natural muscle contraction, which may yield unnatural expressions.

In Figure 50 the skin texture is applied to the simple muscle-based model mesh. The texture may have been acquired using a 3D scanning device such as the one from Cyberware in Figure 51.

Figure 51: Face texture used for the above model (Parke and Waters, 1996).

In the following section, we provide an overview of the theory behind Waters's model and its subsequent extensions by other authors.

6.2.5.1 Vector muscle system

a) Linear muscle modelling

In their simplest form, linear muscle vectors (Figure 52) deform the surface in a circular fashion, with cosine fall-off (Waters, 1987).

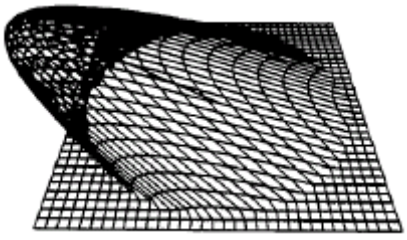


Figure 52: Linear muscle vector displacing a grid in a circular cosine fashion (Waters, 1987).

Figure 53 illustrates the muscle contraction in two dimensions. Attachment to the bone is in vertex V_1, while V_1V_2 represents the linear muscle vector. The aim is to calculate the displacement of P that moves to P' by contraction of the linear muscle with vector V_1V_2.

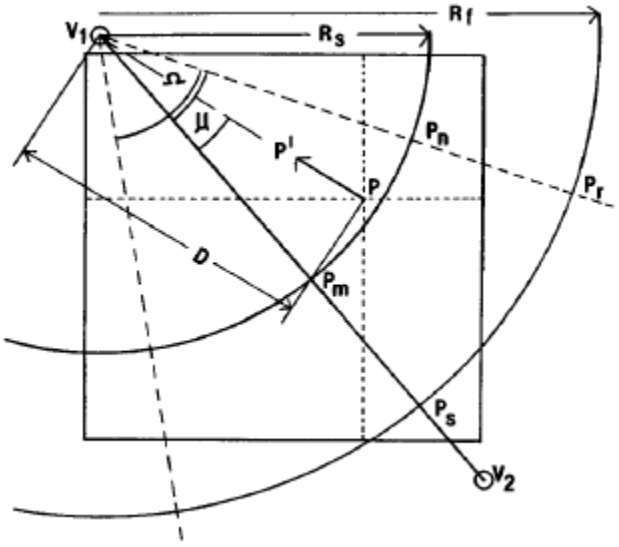


Figure 53: Linear muscle: zone of influence (Waters, 1987).

To calculate P', the following expression is used (Parke and Waters, 1996):

$$P' = P + akr\left(\frac{PV_1}{|PV_1|}\right). \tag{6-13}$$

a is the angular displacement parameter and is equal to $\cos(\mu)$. μ is the angle between V_1V_2 and V_1P. The distance D is between V_1 and P and r is the so-called radial displacement parameter and is defined as

$$r = \cos\left(\frac{1-D}{R_S}\right) \text{ for } P \text{ inside the region } V_1P_mP_n \tag{6-14}$$

and

$$r = \cos\left(\frac{D - R_S}{R_F - R_S}\right) \text{ for } P \text{ inside the region } P_n P_r P_s P_m \tag{6-15}$$

and k is a constant representing the elasticity of the skin.

Choe, Lee and Ko (2001) presented a slightly different linear muscle model (Figure 54).

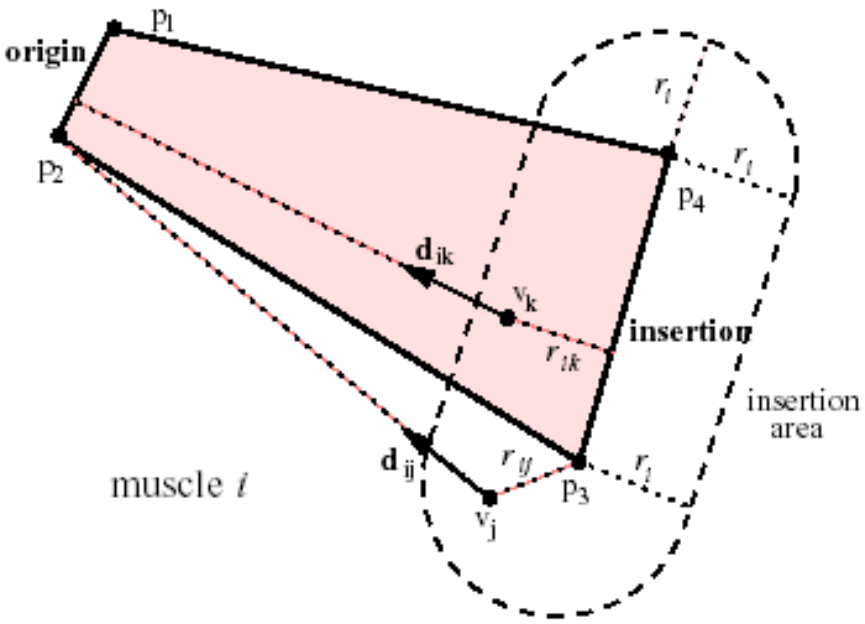


Figure 54: Parallel muscle model (Choe, Lee and Ko, 2001).

Their model is called the parallel muscle model, and it is represented by quadrilateral, p_1, p_2, p_3 and p_4. The origin is not a point, but a line between p_1 and p_2, which represents the attachment to the bone. r_i represents radius of the insertion area. r_{ij} is the distance from v_j to the closest point on p_3p_4. Vector d_{ij} connects v_j to the closest point on p_1p_2. The insertion area represents the area of influence on the skin surface. The force component at v_j is defined by the following expression:

$$f_{ij} = a_i f_i \left(1 - \frac{r_{ij}}{r_i}\right) d_{ij} \tag{6-16}$$

where a_i represents the muscle actuation parameter normalized on the interval [0,1] and f_i is the predefined unit muscle force.

b) Sphincter muscle modelling

The sphincter muscle (Figure 55) contracts around an imaginary central point and resembles a string bag.

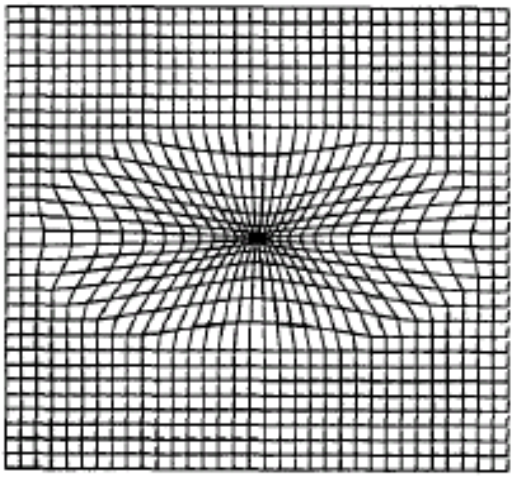


Figure 55: Result of contraction of a sphincter muscle in 2D (Waters, 1987).

Figure 56 illustrates a sphincter muscle contraction in two dimensions. It can be approximated to a parametric ellipsoid, with the semi-major axis lx, semi-minor axis ly, and with epicentre c, as depicted in Figure 56(a). The aim is to calculate the displacement of P that moves to P' by contraction of the sphincter muscle, using the following expression:

$$f = 1 - \frac{\sqrt{ly^2 p_x^2 + lx^2 p_y^2}}{l_x l_y}. \tag{6-17}$$

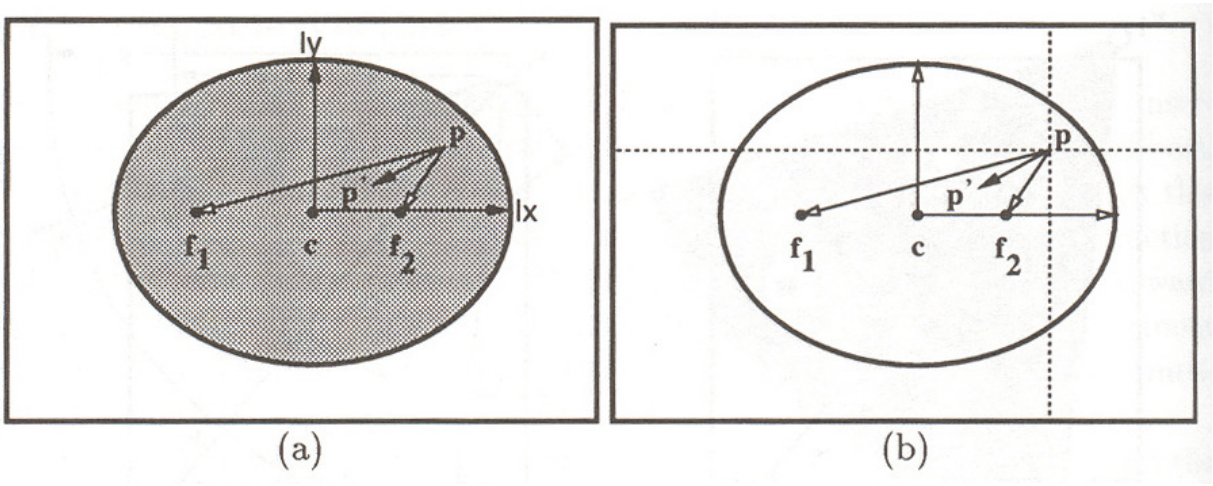


Figure 56: Sphincter muscle (Parke and Waters, 1996).

c) Sheet muscle modelling

An example of a sheet muscle is the frontalis major, which participates in the raising of the eyebrows. These muscles are composed of near parallel fibres, hence they do not have a radial component and they do not emanate from a point source.

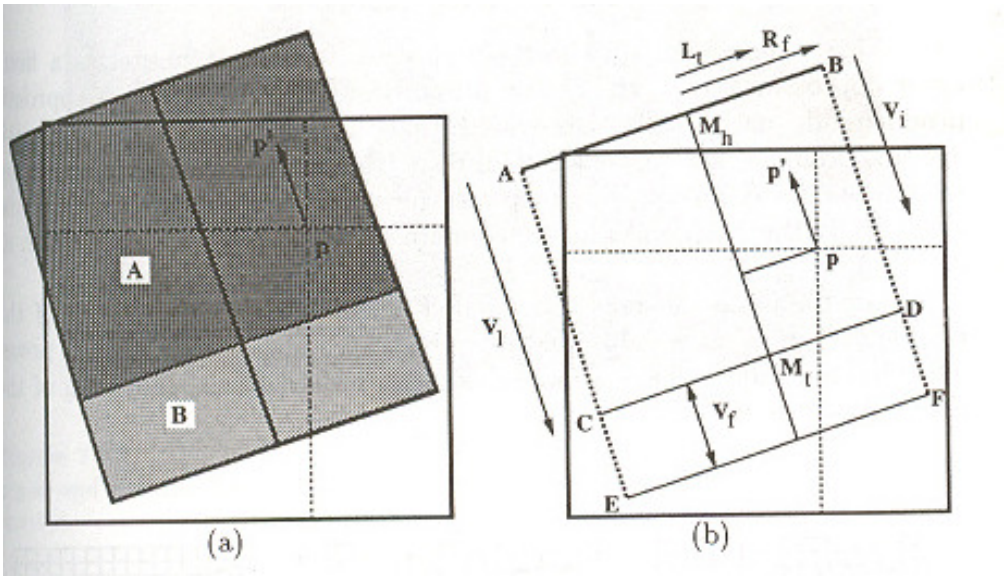


Figure 57: Sheet muscle (Parke and Waters, 1996).

As per diagram in Figure 57, the computation of node p can be defined as

$$d = \cos\left(1 - \frac{L_t}{R_f}\right), \qquad \text{for } p \text{ inside sector } ABDC \text{ and} \tag{6-18}$$

$$d = \cos\left(1 - \frac{L_t}{R_f}\left(\frac{v_i}{v_l} + v_f\right)\right), \qquad \text{for } p \text{ inside sector } CDFE\,. \tag{6-19}$$

The sheet muscle seems to be largely ignored by recent authors and I was unable to find details about its function. It was usually approximated by series of parallel linear muscles (Bui, Heylen and Nijholt, 2003).

Most – if not all – attempts at muscle-based animation have at least considered Waters's (1987) model as a base for their own approach. Many scientists worked on this model in isolation and have built in individual minor and major improvements to this model. The first that we consider concerns the subdivision of the face. Pasquariello and Pelachaud (2001) and Bui (2004) divided their models into a number of areas, to achieve better control of the muscle actions (Figure 58)

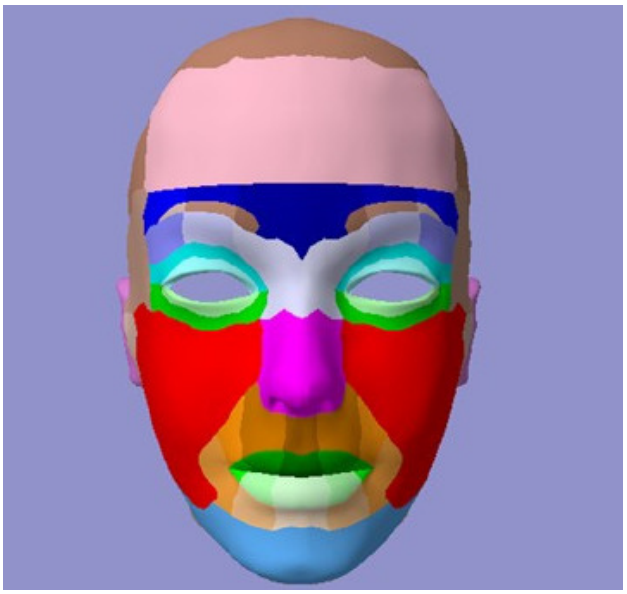


Figure 58: Subdivision of the face into areas of muscle actions Pasquariello and Pelachaud (2001).

The subdivision facilitates better control of displacement of vertices under the simultaneous action of multiple muscles. As discussed by Bui (2004), an optimal number of regions should be identified, in order for this technique to work. Too many smaller regions would complicate the finding of the region of influence of a muscle. Conversely, having fewer large regions would reduce the improvement on the speed of the muscle model's algorithm.

The original Waters's muscle model iterated through all the vertices in order to determine whether or not they are under a muscle's influence. While this approach was sufficient for the number of vertices of the original model, it is inefficient for today's models, which are composed of a significantly larger numbers of vertices. Subdivision of the face into regions introduces an important optimization concept to the algorithm, enabling it to query whether or not a vertex is inside the region upon which the muscle has the effect. This condition is far less computationally intensive than outright determination as to whether or not the muscle influences the vertex.

Secondly, modifications to the muscle model are briefly considered. Bui (2004) addresses the artefacts that occur on the skin surface under the influence of two or more Waters's (1987) vector muscles. Waters's way of handling this situation was to add the displacements one after the other, producing results similar to that in Figure 59.

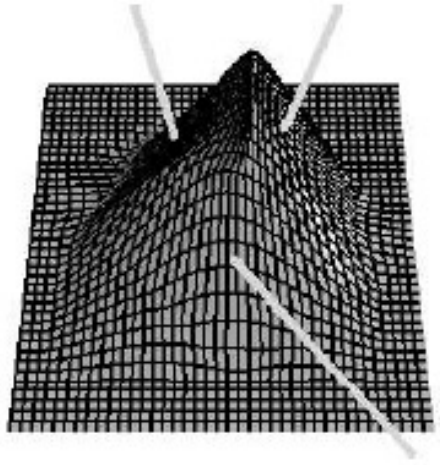


Figure 59: Multiple vector muscle actions – by adding displacements (Bui, 2004).

Bui (2004) proposed simulating parallelism by calculating the resultant displacement internally, then applying it to the vertex. This computation was performed in five steps (five being the number of

steps that Bui found to be a good trade-off between computational speed and a successful outcome) of muscle contraction. The results of this technique are visible in Figure 60.

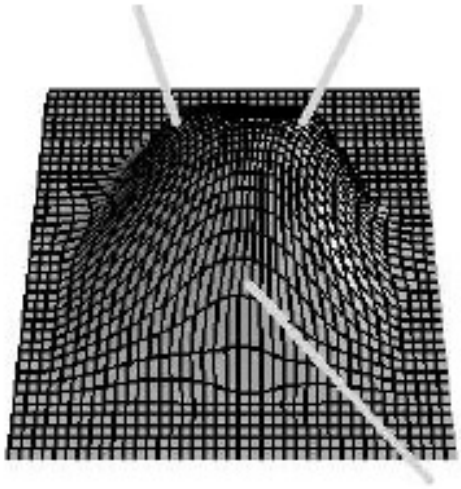


Figure 60: Multiple vector muscle actions – by simulating parallelism (Bui, 2004).

6.2.5.2 Mass-spring muscle system

Zhang, Prakesh and Sung's (2001) aim was to design an animated facial model that could be played in real-time, while also being sufficiently realistic. Their model is based on the human anatomy, featuring muscles and components of the skin. The skin is modelled using the non-linear spring frames that can realistically simulate the relevant dynamics. The advantage here is that the model does not need to be treated as a continuous surface, as each mass point and each spring can be accessed individually.

The facial model is a polygon mesh, and a mechanical law of particles is used to deform the facial skin tissue. Each vertex of the skin is represented as a particle with mass *m*. The particles are linked to each other with '*springs*'. A spring is the structure that conforms to the laws of elasticity and is modified to react in accordance with the stress/strain correlation of the skin. Figure 61 depicts the stress/strain correlation of the facial tissue. Under low stress, dermal tissue provides low resistance to stretch, due to the collagen fibres uncoil in the direction of stress. Once the collagen fibres are fully uncoiled under a greater stress, they resist stretch to a much greater extent.

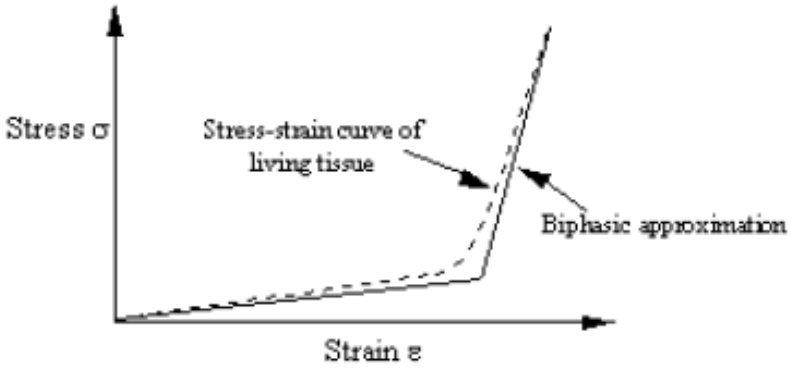


Figure 61: Stress/strain correlation of the facial tissue (Zhang, Prakesh and Sung, 2001).

The muscles are contracted in accordance with FACS (Ekman, Friesen and Hagar, 2002). Similarly to other muscle-based approaches, there are three distinct types of muscles, namely linear, sheet and sphincter muscles. Contraction of the muscles introduces an imbalance between the springs, which in turn propagates the force caused by the contraction throughout the system, until a new equilibrium is reached. Figure 62 illustrates the results of this attempt.

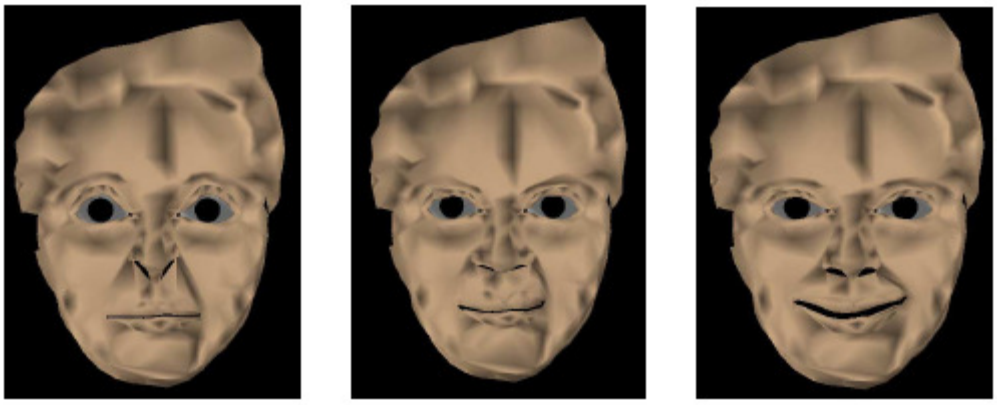


Figure 62: Facial expressions obtained using the mass-spring muscle system approach (Zhang, Prakesh and Sung, 2001).

Another anatomical model was described by Kahler et al. (2001 and 2002). The chief advantage of their model lies in its versatility. It fitted the muscles and calculated the skull mesh-based on the face geometry, thus greatly reducing the amount of manual intervention. A result of this technique is

visible in Figure 85, where a model of a boy has been automatically adapted to represent him at different ages.

6.2.5.3 NURBS vector muscle system

A NURBS (Section 3.2.2) muscle-based system was described by Tang, Liew and Yan (2004). A NURBS muscle is defined by three to five control points. The control points are classified into two groups, namely *reference control points* and *current control points*. Reference control points are used to relate the knot vectors and the nodes of the mesh inside the influence region, while the current control point is the one whose weight is currently being modified. Muscle deformation is achieved by modifying the weight of these control points. Internally, the weight modification forces the knots to move, which in turn moves the vertices of the model, as illustrated in Figure 63.

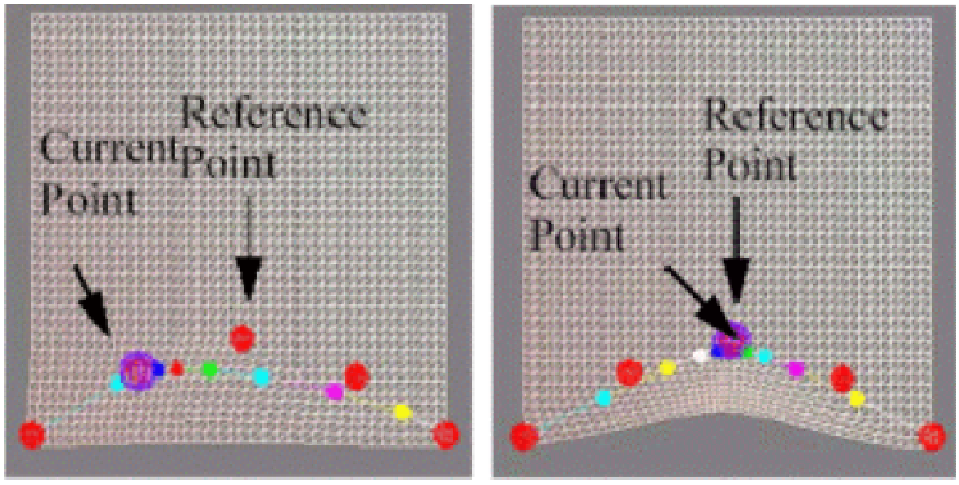


Figure 63: NURBS muscle model: the effect of weight modification at a control point (Tang, Liew and Yan, 2004).

As with other muscle-model implementations, the linear muscle is responsible for the majority of the facial expressions. The sphincter muscle model can be used for deformation of the mouth region. However, not all phonemes could be represented using a single sphincter muscle. This limitation was overcome by the addition of two more linear NURBS muscles. To contribute to realism, the authors tried to simulate the fatty tissue's reaction to deformation by adding control points between the two

end control points. The idea behind this was that the newly added points would drag the mesh slightly up, thus simulating the fatty tissue. The results are visible in Figure 64.

Figure 64: Some of the results achieved by using NURBS muscle model (Tang, Liew and Yan, 2004).

6.2.5.4 Finite-element method (FEM) based muscle system

Finite-element methods (FEM) are an alternative to the mass-spring system for muscle-based animation. FEM originates from the mid twentieth century and has primarily been used in mathematics for finding approximate solutions of partial differential equations. More recently, it has been found useful for solving certain problems in civil engineering and aeronautical engineering. FEM is a complex topic and its detailed description is beyond the scope of this thesis. Conceptually, FEM solves a complex problem by dividing it into a number of simpler problems. Applied to computer graphics, the complex geometry is divided into a relatively large number of simple shapes. The simple shapes are connected together using a function that approximates the inner forces of the material that the geometry represents. When acted upon by an external force, the inner and outer forces are calculated, determining the displacements of the simple shapes, until the equilibrium is reached.

Teran et. al. (2005) introduced a muscle-based animation system based on FEM. They derived the muscle, tendon and skeleton geometry for their model from the US National Library of Medicine (1994), as depicted in Figure 65. In this system, the geometry volume is represented by a number of tetrahedrons, along with the appropriate internal forces binding the tetrahedrons together.

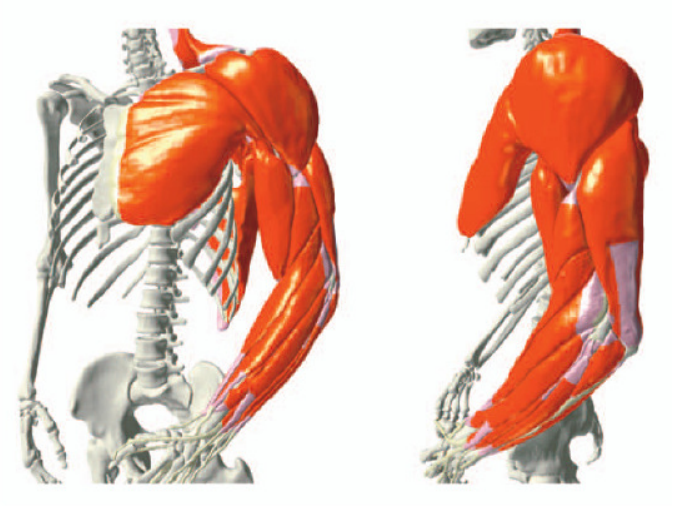


Figure 65: Muscle/skeleton model by Teran et. al. (2005): muscles are depicted in red, while tendons are pink.

Sifakis, Neverov and Fedkiw (2005) constructed an anatomically accurate facial muscle model, using the principles derived from the more general muscle construction principles of Teran et al. (2005). Their technique is also based on finite element algorithms, but it differs from other facial models because its muscle action can interact with the environment. That is, the muscle forces can be combined with external forces such as collision, producing the resultant effect shown in Figure 66.

Figure 66: Impact of a colliding object on the face (Sifakis, Neverov and Fedkiw, 2005).

6.2.5.5 Representation and animation of the facial tissue

Pasquariello and Pelachaud (2001) provided for the creation of wrinkles and furrows in their animation. The animation itself is muscle-based, so the wrinkles and furrows would be created by muscle actions. Two areas received special attention: the forehead and the naso-labial area (Figure 67). The forehead was intended to produce the horizontal wrinkles which occur when the eyebrows are raised, while the naso-labial area produced the dimples which result from smiling.

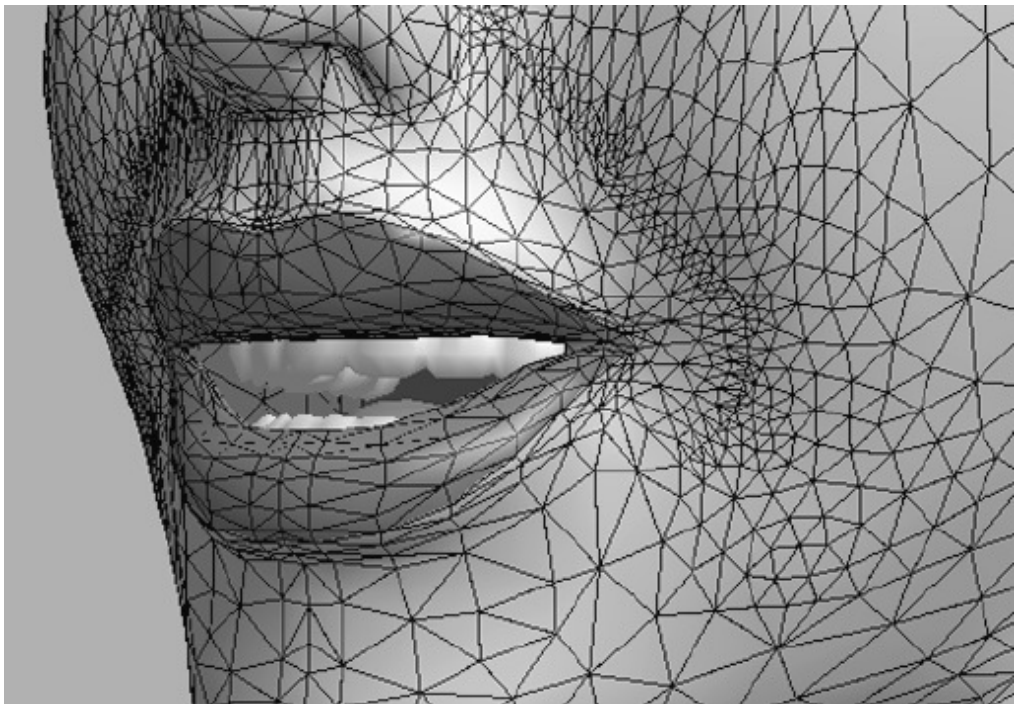


Figure 67: Increase in polygon density in the naso-labial furrow area (Pasquariello and Pelachaud, 2001).

Although Pasquariello and Pelachaud animated the skin in a realistic way, they did not use physical simulation of muscles and the subsequent viscoelastic reaction of the skin. They used two techniques to simulate furrows, bulges and wrinkles: bump mapping and physical displacement of vertices.

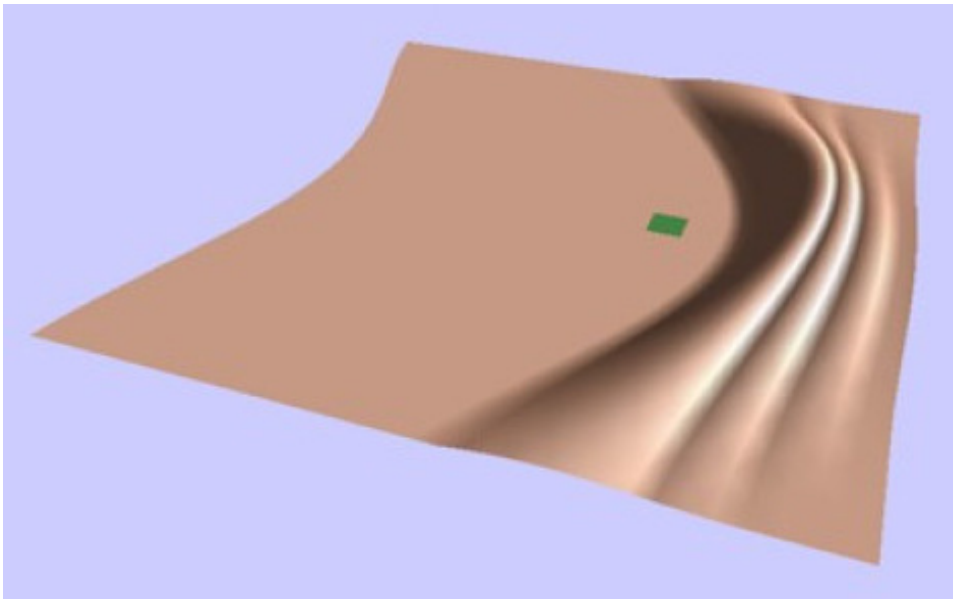


Figure 68: Wrinkles achieved through bump mapping combined with vertex displacement (Pasquariello and Pelachaud, 2001).

Furthermore, when modelling a wrinkle as in Figure 68, the displacements in the X and Y directions need to be computed. There is no displacement in the direction Z, as also suggested by Figure 69. Physical displacement of vertices in the XZ plane is limited to an ellipsoid area and fades towards the edges.

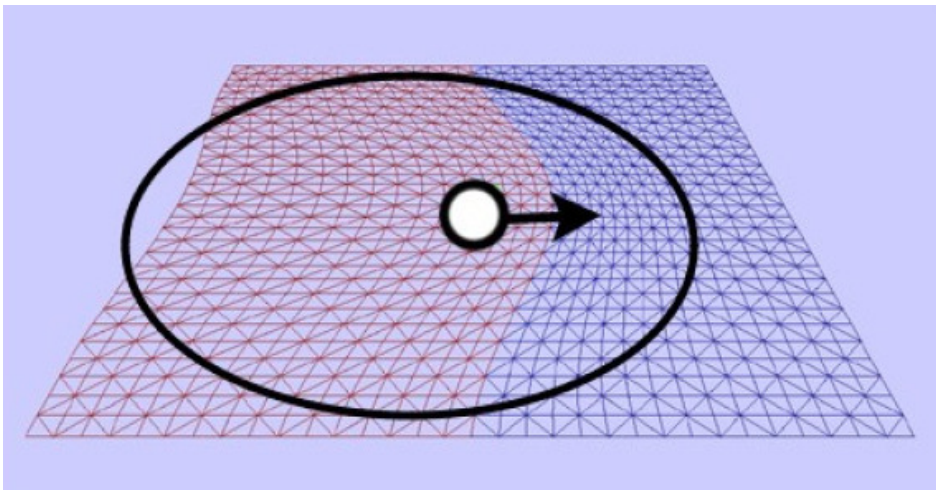


Figure 69: Vertex displacement on the XZ plane (Pasquariello and Pelachaud, 2001).

The displacement is calculated using the following expression:

$$\Delta x_i = (W_j)(FAP_x), \tag{6-20}$$

where Δx_i is the value of displacement, FAP_x is the direction of the FAP action (for definition of FAP, see section 6.1.2.3), in this instance in the direction of the X-axis. W_j is the weight assigned to a vertex, which decreases from 1 to 0 with the distance of the point of action. The decrease is non-linear and is depicted in Figure 70.

The displacement in the direction of the Y-axis is calculated using the following expression:

$$\Delta y_i = \Delta x_i K_1 (0.5(1+\cos(d_i)))(1-\exp(-d_i'/K_2)) \tag{6-21}$$

Δy_i is the value of displacement in the direction of the Y-axis, Δx_i is the value of displacement in the direction of the X-axis, d_i is the distance of the vertex *i* from the point of muscle action, d'_i is the distance between the vertex *i* from the Z-axis, while K_1 and K_2 are constants.

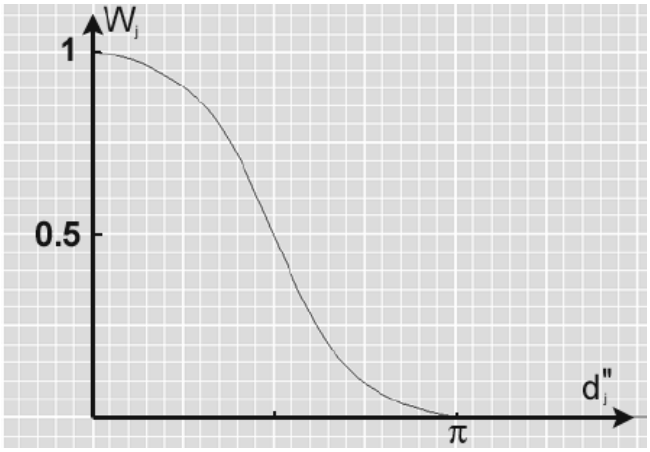


Figure 70: Reduction of weight with increment of distance of a vertex from the point of influence (Pasquariello and Pelachaud, 2001).

Bui (2004) created wrinkles by displacing the affected vertices in the direction of the normal to the direction of muscle action. If we consider Waters's linear (vector) muscle model,

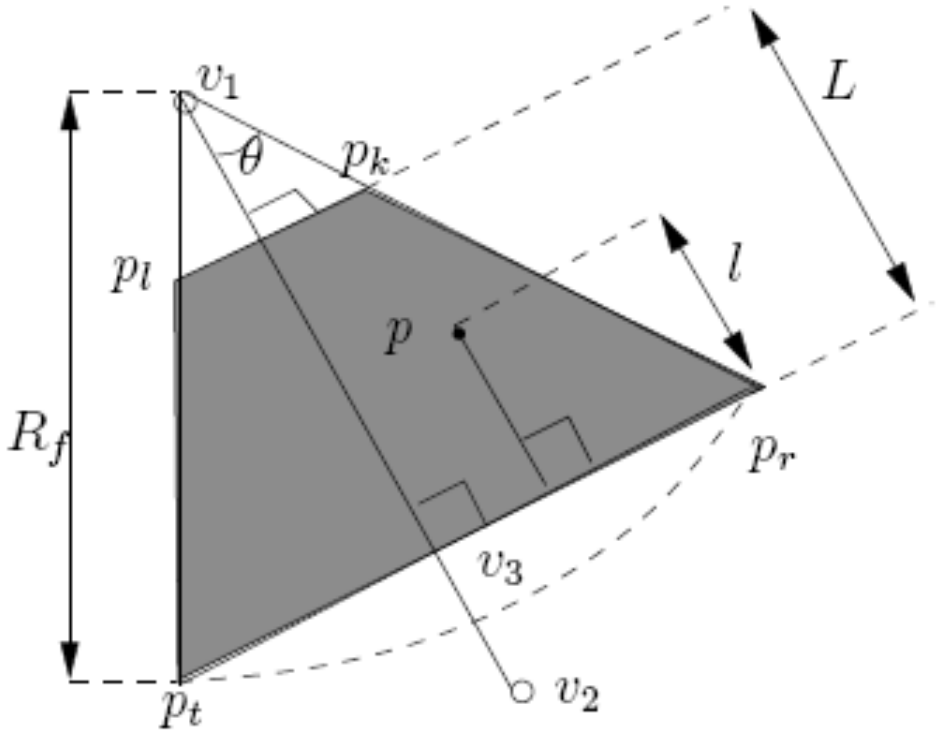


Figure 71: Linear (vector) muscle model (Waters, 1987 and Bui, 2004).

the zone bounded by p_l, p_k, p_r and p_t would contain wrinkles (Figure 71). The displacement of the vertices in order to simulate wrinkles would be in the form of multiple parabolas, as depicted in Figure 72. The wrinkle function $f(l)$ is defined as

$$f(l) = a(1 - \frac{(u(l) - b)^2}{b^2} \tag{6-22}$$

where

$$u(l) = l - \left\lfloor \frac{l}{b} \right\rfloor b \tag{6-23}$$

and $2b$ is the 'period' of the series of parabolas.

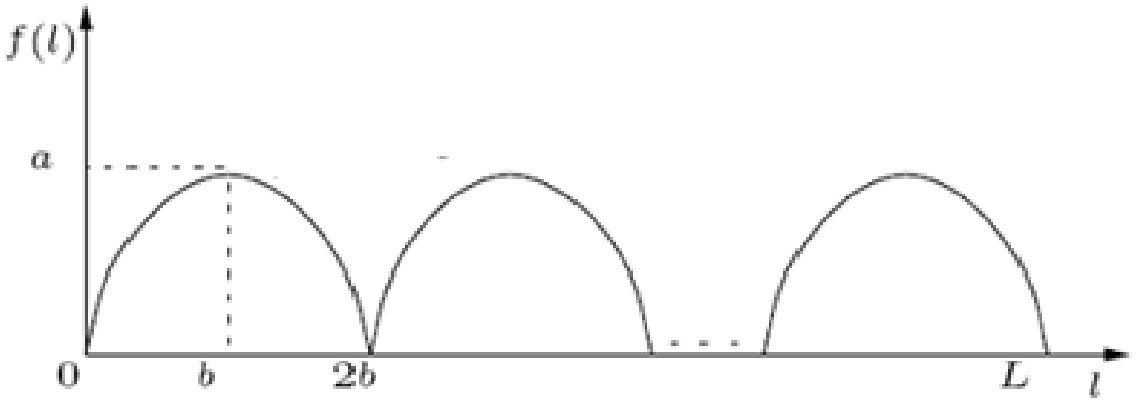


Figure 72: The wrinkle function $f(l)$ (Bui, 2004).

The reason for choosing a parabola is that it is computationally inexpensive, but it still adequately represents wrinkling.

The main disadvantage of this approach is a result of phenomena called 'unrepresentative vertex normal'. We can see from Figure 73 that the normal to the curve for *l-values 2b, 4b…* is undefined, which upsets the smooth shading algorithms. To alleviate this problem, the system uses the normals of triangular polygons that contain the offending vertices.

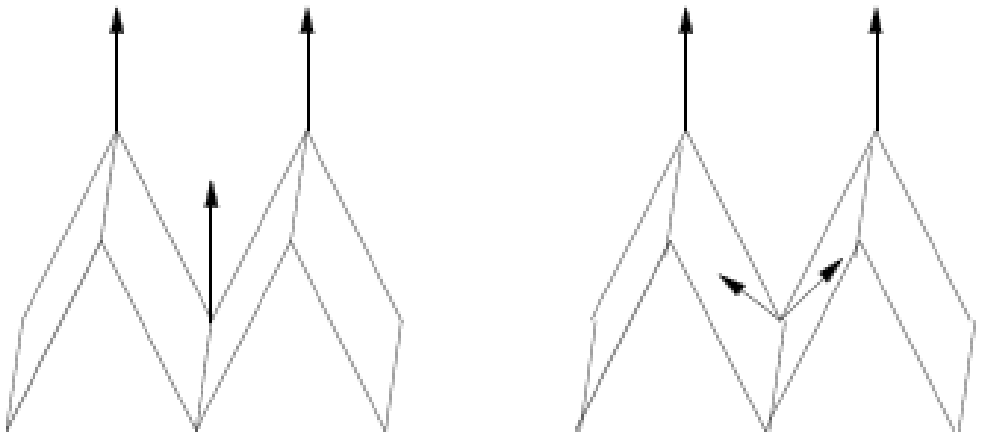


Figure 73: Unrepresentative vertex normal and its solution (Bui, 2004).

The final result of Bui's (2004) approach for creation of wrinkles is shown in Figure 74.

Figure 74: Wrinkles due to muscle contraction (Bui, 2004).

6.2.6 Hybrid approach

More recent facial animation solutions are made up of combinations of several of the techniques discussed earlier in previous sections. Bui, Heylen and Nijholt (2003) extended Waters's (1987) model to provide wrinkles and bulges and to extend muscle interactions. They also attempted to optimise the model to increase the animation speed. Like Waters, they used a triangular polygon mesh to represent the face, while the lips were modelled using a simple B-spline surface.

An eye tracking algorithm and an algorithm for opening and closing eyelids were adopted from the direct parameterisation method of Parke and Waters's (1996:187-222). Squeezing of the eye was implemented by using Waters's (1987) sphincter muscle.

Waters's (1987) linear muscle method performed muscle action in isolation. In reality, facial muscle actions influence each other, and the resultant force deforms the skin tissue. Overcoming this problem has been attempted many times before, but with limited success. Bui, Heylen and Nijholt introduced 'parallelism', a novel approach which is illustrated in Figure 75.

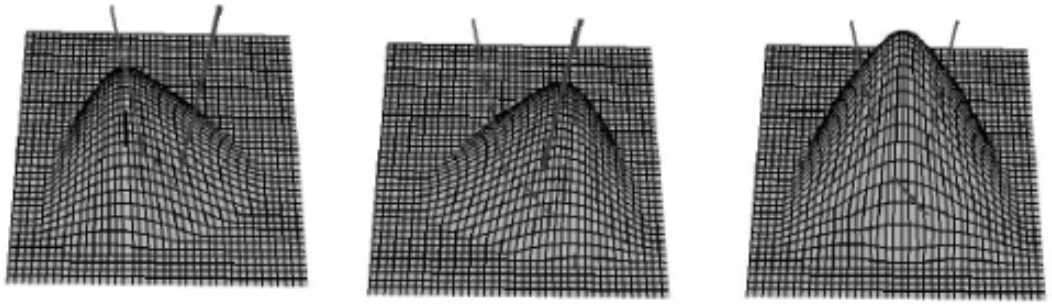


Figure 75: Effect of single muscles on the mesh (left and centre) and their resultant action, modelled using parallelism (right) (Bui, Heylen and Nijholt, 2003).

The sheet muscle model for the frontalis has been replaced by a set of linear muscles, since the frontalis is not completely flat. Wrinkles were simulated by changing the height of the skin affected by muscle actions in regular intervals. The results can be seen in Figure 76.

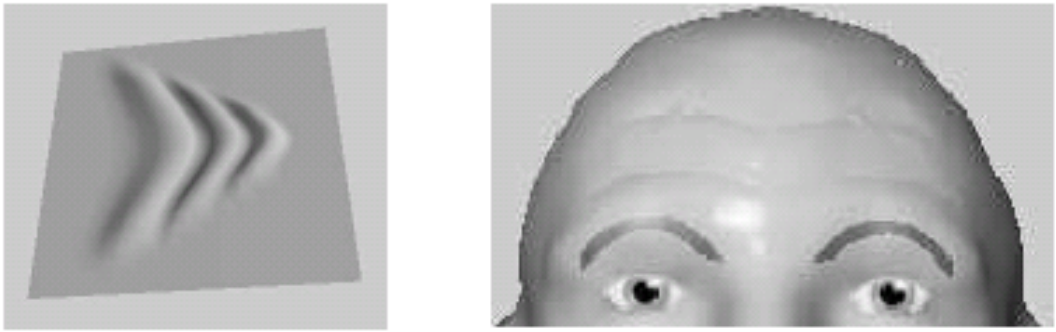


Figure 76: Wrinkles as a result of muscle actions (Bui, Heylen and Nijholt, 2003).

This approach still requires a considerable amount of human intervention for the initial mesh reduction. The regional division is also done manually and there is no scanned face texture applied to the model, which would introduce computational overhead.

Gutierrez-Osuna et al. (2005) created an interesting mixture of existing approaches in their performance-driven audio/visual synthetic system. The generic model contained a number of polygons with identified MPEG-4 FPs. Facial expressions (each of which conformed to an MPEG-4 FP**)** were achieved using muscle action. Although the model had all of the 'anatomy' of a muscle-

based system (mass-spring based muscles, skull and jaw), the animation was not a free Newtonian physics system. The forces that acted on the muscles were combined in such a way that they conformed to MPEG-4 FPs (for MPEG-4, see Section 6.1.2.3).

6.2.7 Facial animation from an artistic perspective

Because it is usually scientists who design algorithms and try to improve approaches to facial animation, the focus is often on proof of concept rather than artistic realism. For this reason, the models created by their methods are often not sculpted to perfection. Their algorithms are frequently integrated into various high-end modelling and animation software such as 3D Studio Max, Maya or Render Man. Artists, on the other hand, do not usually know the nuts and bolts of their modelling software, but are experts in mastering and exploiting the available functionality. They also specialise in creating and animating faces to be both visually appealing and highly convincing. It is therefore beneficial to take cognisance of the facial animation problems and best practices from an artistic perspective. Fleming and Dobbs (1999:8-11) mention common caveats when attempting facial animation. The remainder of this section is based on their observations.

The supraorbital margin (Figure 77) is the bone that lies directly below the eyebrows. When humans change their facial expressions, the skin moves over the supraorbital margin. A common mistake in facial animation is to actually move the supraorbital margin on the model, which tends to make the effect unrealistic. One should rather aim to move the physical tissue on the upper portion of the supraorbital margin, keeping the lower portion in place.

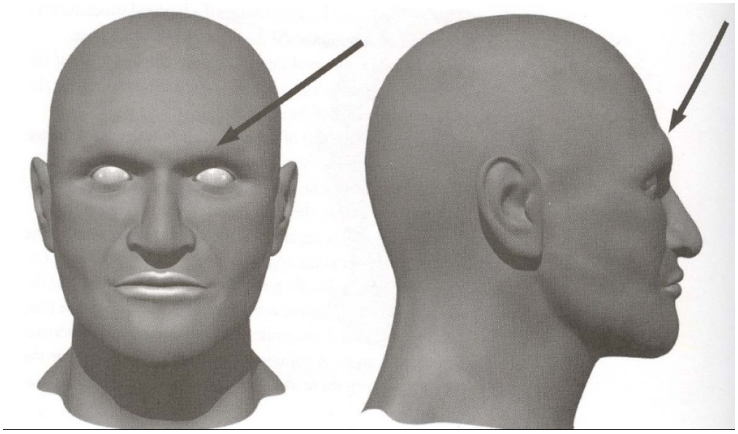


Figure 77: Supraorbital margin (Fleming and Dobbs, 1999).

Another common mistake is to move the tip of the nose. It does not happen in reality as there are no muscles connected to the cartilage – it is too flexible (Figure 78).

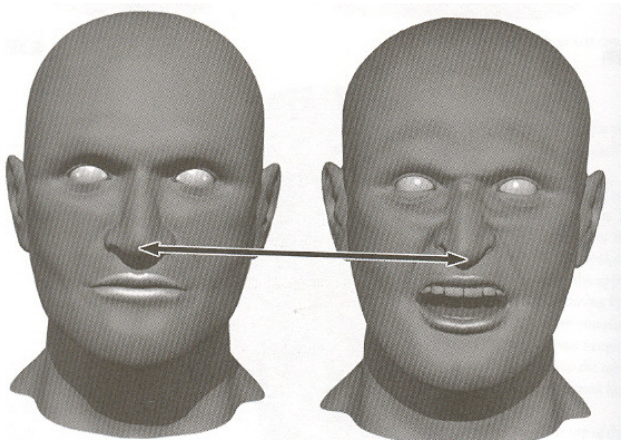


Figure 78: Locking the tip of the nose (Fleming and Dobbs, 1999).

An infraorbital margin (Figure 79) is formed by the lower portion of the orbital cavity and the upper portion of the cheekbone. Moving the infraorbital margin is another common mistake in facial animation. When the cheeks are raised, the tissue goes up and over the infraorbital margin, forcing the lower eyelid to puff up. The muscle tissue cannot move over the infraorbital margin, so it collects under it and creates ‘puffy cheeks’.

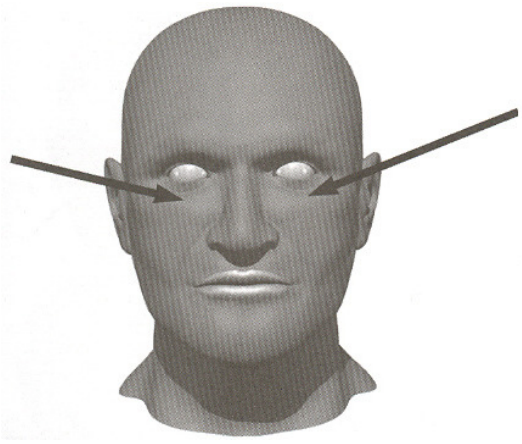


Figure 79: Infraorbital margin (Fleming and Dobbs, 1999).

An important aspect of facial animation is the mandible, which is the lower jawbone and the only movable bone on the skull. The axis of rotation of the mandible should always be correctly placed, that is, in line with the lower part of the ear. The mandible also moves sideways, but that motion is relatively limited and, as a rule of thumb, it can only move laterally by the width of an incisor.

6.2.8 Summary and conclusion

In the preceding sections we discussed five main facial animation techniques. The simplest amongst them is the key-frame based technique. On its own, this technique has little merit. Despite its simplicity of implementation, it is fairly difficult to achieve realistic facial motions and expressions. However, it is extensively used as an auxiliary technique, where another superior algorithm is used to calculate key frames.

Performance-based animation is distinct from most other techniques, as it captures expressions and movements of an actual person, thus animating an analogous character. This means that there is no algorithm or coding system which computes the expressions. This approach is still successfully used in creation of animation. Direct parameterisation was not extensively covered, as it has become obsolete and is no longer used. The pseudo-muscle based system simulates muscle actions in a simplified manner. Its merit lies in being relatively computationally inexpensive in comparison with the musce-based model, on the account of realistic impression. This approach has lately been increasingly replaced by the muscle model, due to advances in hardware design and hence reduced need for preservation of the processing power.

Finally, the muscle-based model is the current state-of-the-art technology in which the muscle actions are closely simulated in accordance with the human anatomy. Hardware restrictions are the main reason that this technique has not yet been fully perfected, and some shortcuts and simplifying assumptions are still used. Muscle-based models continue to be a subject of extensive research. A comparison of different animation techniques is summarised in Table 2.

	Interpolation and key-frame	Direct Parameterization	Pseudo-muscle based	Muscle based
Complexity	Simple	Intermediate	Intermediate	Complex
Need for human intervention	High	Intermediate	Intermediate	Intermediate
Applied in practice	Only used as a subset of other methods	No longer used	Limited current use	Used extensively
Advantages	Simplicity.	Overcomes rigidity by grouping associated parameters together	Simpler than muscle-based, less processor intensive	Accurately simulates muscle mechanics
Disadvantages	Restrictive technique. Requires explicit data. Tedious process.	Restricted to a certain topological mesh. Ignores surface deformation. Conflict between parameters. Noticeable motion boundaries.	Fails to display wrinkles and bulges. Interaction between muscles difficult to simulate.	Could be extremely processor intensive

Table 2 : Comparison of different facial animation techniques.

In particular, mass-spring and finite element algorithms seem to be the two dominating technologies for muscle-based animation today. Currently, the two schools of thought co-exist and the superiority of one over the other has not yet been established.

Relative to FEM, the mass-spring method is simpler and produces reasonably good results. It is computationally less expensive, and hence better suited to real-time applications. However, recent mass-spring models are increasingly flexible, allowing for automated adaptation of the skull and muscles as a function of the facial geometry (Kahler et al., 2001 and 2002).

Conversely, FEM is regarded as being more stable and precise. However, it is far more computationally demanding. While mass-spring animation is used extensively for real-time requirements, FEM algorithms often take in the order of minutes to complete. In addition, the initial FEM volumetric mesh is relatively complex and time consuming to build, requiring considerable human intervention. The FEM mesh is inflexible, requiring rebuilding for any change in geometry.

We could indicate that as computer hardware becomes faster, the relative computational speed advantage of mass-spring algorithms over FEM diminishes, possibly giving way to FEM as a method of choice. Table 3 outlines the comparison between these two techniques.

	Mass-Spring Model	Finite-Element Method
Complexity	Simpler	Complex
Adaptability	Flexible	Inflexible
Computation	Fast	Slow, expensive
Quailty	Reasonable	Very Good
Need for human intervention	Lower	Higher

Table 3: Relative advantages and disadvantages of mass-spring over FEM.

Chapter 7 Reuse of existing head and face models

Creating new facial and head models from first principles is a time consuming and tedious task. Streamlining this task has been so extensively researched that it deserves a chapter of its own. Various notable methods to reuse existing head and face models are as follows:

- Interpolation between existing faces (Parke, 1996:94-96);
- Local deformation of existing faces (Magnenat-Thalmann et al., 1989);
- Freeform deformations (Sederberg and Parry, 1986);
- Stochastic noise deformations (Lewis, 1989);
- Parameterised conformation models (Parke, 1996:104);
- Adaptation from a canonical model (Zhang, Sim and Tan, 2004);
- Expression cloning (Noh and Neumann, 2001);
- Cloning of MPEG-4 face models (Mani and Ostermann, 2001);
- Transfer of multi-layered anatomical structures (Kahler et al., 2002);
- Hybrid approach (Paouri, Magnenat-Thalmann and Thalmann, 1991).

7.1 Interpolation between existing faces

Interpolation of existing faces was first described by Parke (1996:94). Intermediate forms of the surface are achieved by interpolating each vertex between its extreme positions. The source and resultant face could be of either the same or of diverse topology. If they are of the same topology, it means that the number of vertices and their interconnections must be identical. If that is the case, creating a new face simply involves an interpolation of the vertex positions.

If the faces have different topologies, one would need to resort to different methods. According to Parke (1996:95), a fairly obvious approach is to add or delete polygons and vertices from the faces, until their topology is the same. However, algorithms for adding and deleting polygons on an arbitrary topology are not obvious.

Another approach, described by Magnenat-Thalmann et al. (1989), is to first convert the topology of each face into a rectangular grid topology ($m \times n$ matrix) and then to interpolate between these

grids. This method involves two steps: profile determination and grid generation. Profiles are determined by finding intersections between the object and the series of horizontal and vertical planes. Grid generation consists of finding the length of the longest profile first, subdividing this length into (m-1) intervals, and then finding the grid points.

The grids for two faces with diverse topology are likely to be different. Corresponding points in the smaller grid are determined for each point in the larger grid. The polygon network is transformed to a regular grid by the resampling of the polygon surface. Profiles are selected in such way that there is a correspondence between similar physical regions. That is, even in a diverse topology case, both faces still have a nose, mouth and eyes that are more or less similar. The algorithm expects borders of the profiles to be at extremities of the eyes and mouth. Finally, an in-between of the two faces is again obtained using linear interpolation.

7.2 Local deformation of existing faces

Local deformation is a transformation to a portion of the affected object only. It involves the selection of the region and the choice of transformation to be used. Magnenat-Thalmann et al. (1989) list five methods for selecting a region, namely by indicating vertex numbers, by selecting vertices within a defined box (defined by six vectors), by selecting part of a cylinder (starting angle and percentage of the circle), by colour (all the vertices assigned a specific colour) and by set-theory operations between two previously selected regions.

In addition, the authors describe four methods of transformation:

- *Percentage to a vertex* – each vertex is moved towards a reference vertex (according to the percentage of the distance between two vertices);
- *Guided translation* – translation of movement is first calculated, then applied to all the vertices within the selected region;
- *Scale according to a plane* – scale with an amplitude proportional to the distance between the vertex and the plane is applied to the region;
- *Variable translation* – applied with consideration to a variation degree and possibility of a change in acceleration factor. This factor determines the degree of influence and is most intense in the centre of the region, fading towards the edges. The implementation technique is based on Allan, Wyvill and Witten's (1989) work on the so called decay function.

7.3 Freeform deformations

Freeform deformations (see Section 6.2.3.2) are described by Sederberg and Parry (1986). FFD can be described as multiple deformable objects placed into a cube-shaped mould and blended together with a clear plastic compound. Once the mixture cools down and is taken out if the mould, it would look like a clear plastic cube with various objects embedded inside it. If we deform the cube, all the objects inside it would be deformed accordingly. Figure 80 shows how FFD may be used to deform a particular section of an object.

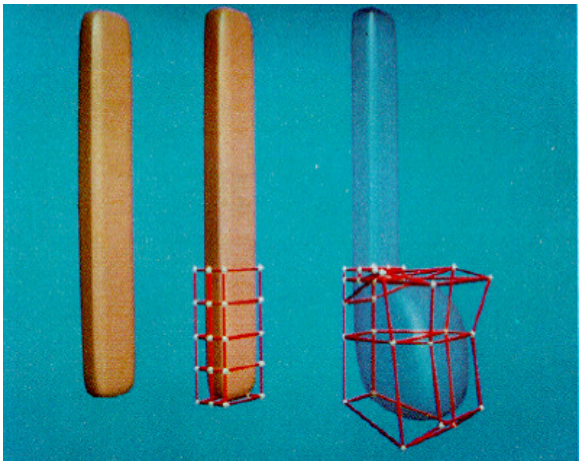


Figure 80: Example of FFD in action (Sederberg and Parry, 1986).

It can be applied to most geometric primitives, locally or globally. It is an interesting and useful fact that there is a family of FDDs with the property of preserving the volume after deformation. These techniques can be used within modelling software, to smoothly bend, extrude and intrude existing models.

Coquillart (1990) identified limitations of FFD and extended it to be more versatile. She called her extension EFFD (extended freeform deformation). The extension was achieved by using non-parallelepipedical lattices. This would, for example, be useful for defining a circular bump on a surface.

7.4 Stochastic noise deformations

Solid noise is a random-valued function from R^3 to R with some known statistical properties. The use of solid noise to create new models from existing ones was described by Lewis (1989). He described

two algorithms for the synthesis of high-quality solid noise with control of the power spectrum and distribution function. The solid noise vector is applied to vertices of the face surface, which results in a deformed face. The noise can be adjusted, to produce a range of desirable effects.

7.5 Parameterised conformation models

Another approach, described by Parke (1996:104), is the use of parameterised facial models. The idea consisted of the creation of a face or facial expression based on a number of controlling parameter values. The number of different faces and/or expressions depended on the number and type of control parameters.

7.6 Adaptation from a canonical model

Zhang, Sim and Tan (2004) described the latest approach to creating realistic facial models from a low resolution generic mesh, a canonical model of which is shown in Figure 81. The main drawback of the previous attempts (Lee, Terzopoulos and Waters, 1995; Pighin et al., 1998; Kahler et al., 2002) lies in the amount of human intervention and/or artistic sense required in order to achieve satisfactory results. Zhang, Sim and Tan (2004) not only succeeded in automating the procedure, but produced models suitable for animation purposes. The canonical model already contains data required for expressions and animation.

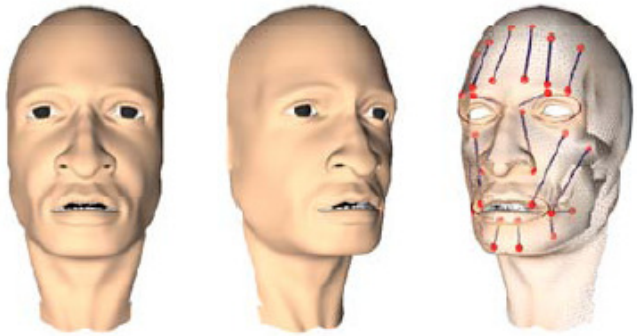


Figure 81: Canonical model used to adapt to scan (Zhang, Sim and Tan, 2004).

Texture and surface of the new face are acquired using a laser-based 3D scan. The scanned data are adapted to the canonical model in four steps. Firstly, landmark locations are automatically recovered on both the canonical model and the acquired data using the novel projection-mapping approach.

Afterwards, global adaptation is performed. It automatically adapts the size of the canonical model and aligns it with the geometry of the scanned face, based on the landmark positions. Upon completion of the second phase, local adaptation is performed – the positions of the vertices of the canonical model automatically adapt to the scanned data. Finally, the muscle layer is adapted – all muscles defined under the skin of the canonical model are transferred to the new skin geometry.

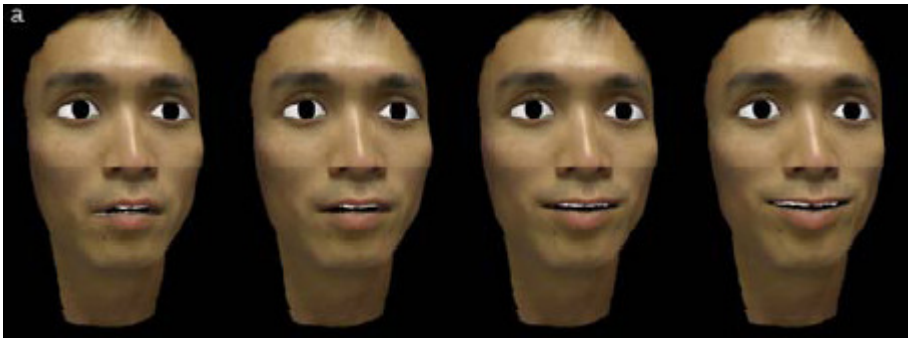


Figure 82: Resultant model, following the adaptation (Zhang, Sim and Tan, 2004).

The authors achieved the results shown in Figure 82 in only 30 seconds (on a P4 2.4GHz CPU). Further eye and teeth positioning would take several more minutes, which is indeed a remarkable improvement on historical methods. The only manual task in the processes constitutes selection of the initial landmark and the adaptation of the ears and teeth. In future, the manual selection of landmarks could be eliminated by using a face recognition approach to automatically estimate facial features from an image. Automatic adaptation of separate components (ears and teeth) could also be achieved.

Gutierrez-Osuna et al. (2005) designed their canonical model in accordance with Parke and Waters's (1996) approach. They also assigned 64 MPEG-4 FPs to the model for the purposes of animation. To modify the canonical model to suit the current speaker, 3D coordinates of FPs are obtained from the video (see photogrammatic techniques in Section 5.3). Besides these coordinates, the authors used MPEG-4 distance fractions and other anthropometric measurements to enhance the adaptation.

7.7 Expression cloning

Noh and Neumann (2001) mapped expression from an existing model directly to the surface of the target model. Their system requires that a number of reference points be determined initially, viz. those vertices in the target model that correspond to the vertices on the source. Depending on the

diversity of the two models, the system may require manual specification of the initial few points (never more than ten). The computer then calculates the other reference points (normally fifteen to thirty five), after which it proceeds transferring the motion vectors from the source to target vertices. The radial basis function (RBF) is used to approximate the alignment of the features of the source to the target model. Cylindrical projection is then used to ensure that all the vertices of the source model lie on the target model surface. The results are depicted in Figure 83.

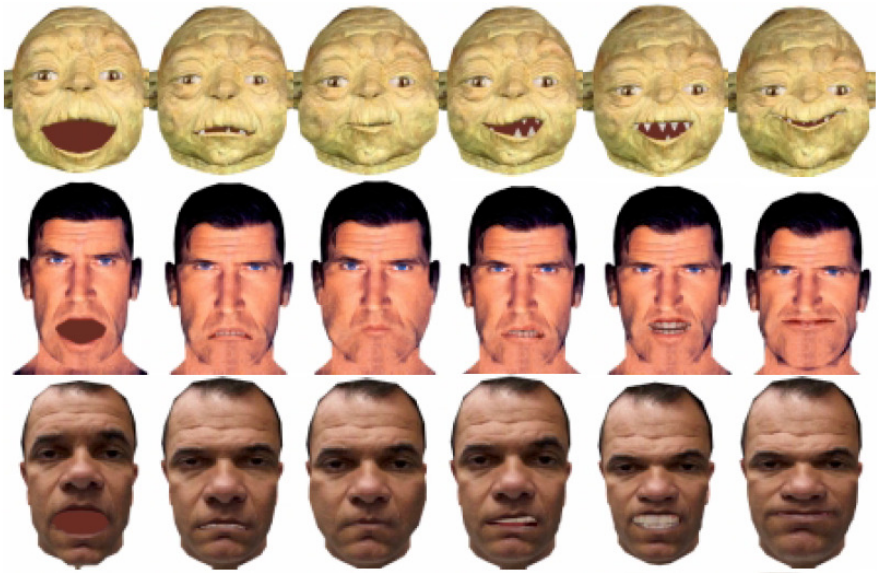


Figure 83: Results of the expression cloning (Noh and Neumann, 2001).

Some of the drawbacks of this approach are the necessary human involvement and the requirement that the expression is created on the original model first. Also, due to its nature, it cannot be run in real-time. The tongue and teeth had not been addressed at the time of publication of this thesis.

7.8 Cloning of MPEG-4 face models

Cloning of MPEG-4 face models was described by Mani and Ostermann (2001). In order to clone the faces, they manipulated B-splines, using their weighted sum as a mapping function. The user needs to manually select the corresponding landmarks on both the source and the target face. Once that is done, the facial animation tables (FAT) of the source model are applied to the target model, resulting

in cloned expressions. To enhance correctness, the target model's B-spline weights could be fine-tuned manually. The results are seen in Figure 84.

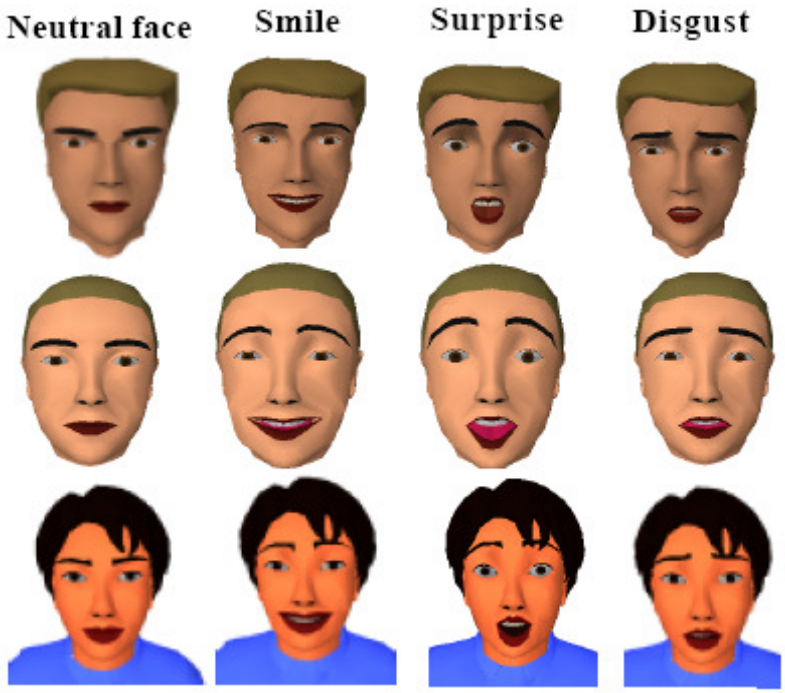


Figure 84: Results of MPEG-4 face cloning (Mani and Ostermann, 2001).

The main drawback of this approach is the considerable amount of human intervention required. The quality of the animation directly depends on the number of landmarks selected and the tuning of the weights.

7.9 Transfer of multi-layered anatomical structures

Kahler et al. (2002) described a method of deforming an anatomical face model, consisting of skull, muscles and skin, the result of which is shown in Figure 85. Extending Noh and Neumann's (2001) work, they used radial basis functions to deform their source model. Again, the user first chooses a set of landmarks. These landmarks are then used to define the so-called warp function. The main difference between this approach and the expression cloning defined by Noh and Neumann lies in the way in which the warp function is applied to different parts of the model. The skin mesh is deformed by direct application of the function on the skin vertices. Since the skull mesh is dependent on the skin mesh, it is recalculated from the newly derived skin vertices coordinates. Likewise, muscles are

computed in such way that they fill the gap between the skin and the skull. The eyes, teeth and tongue are considered to be independent rigid components and are automatically scaled and repositioned.

Figure 85: Deformation of anatomical structures (Kahler et al., 2002).

Once again, the main disadvantage of this approach is the unavoidable human involvement. Kahler et al. (2002) are currently looking into statistical data and anthropometric modelling, to assist them in the automation of the manual tasks.

7.10 Hybrid approach

Paouri, Magnenat-Thalmann and Thalmann (1991) used several different techniques to produce new faces from existing ones. They used interpolation (Magnenat-Thalmann et al., 1989) to create a face which is in between two other faces. They also used concepts of local deformations (Sederberg and Parry, 1986) and polygonal mesh modification (Allan, Wyvill and Witten, 1989), and extended these techniques as follows:

- Definition of range of influence around a vertex;
- Decay function over a range of influence;
- Binding, anchoring, stretch, grow and randomize.

Local deformations are done manually, using the sculpting methods. The user interface consists of a Spaceball (LeBlanc et al., 1991) and a mouse. Spaceball is an interactive input device that enables three-dimensional manipulation of the object on the screen.

7.11 Conclusion

Considering the difficulty and tediousness of creating new models from first principles, it is worthwhile researching methods that may assist in automating this task. This chapter described a number of methods used to derive new models from existing ones. Different methods have been shown to be suitable for different stages of the evolution of facial animation. While the interpolation described in Section 7.1 was sufficient in the early stages, it would be unsuitable for a complex muscle-based system with thousands of vertices, as described in Section 6.2.5. To be entirely fair, most of the more advanced methods do eventually move the vertices in order to create a new face, but this step is always preceded by a higher degree of automation. The state of the art in this discipline seems to be the transfer of multi-layered anatomical structures, as described in Section 7.9. All of the existing approaches are still plagued by a high degree of the human intervention, with the way forward consisting mostly of attempts to minimise such intervention.

Chapter 8 Speech-specific modelling and animation

This chapter concentrates on the modelling and animation of the elements of the human body involved in the production of speech. The focus is on the physical model and muscle-based facial animation, as it seems to be the most advanced and actively researched approach. Bones that participate in the production of visual speech are explained first, followed by specific parts of the face, mouth and tongue.

8.1 Speech anatomy and mechanics

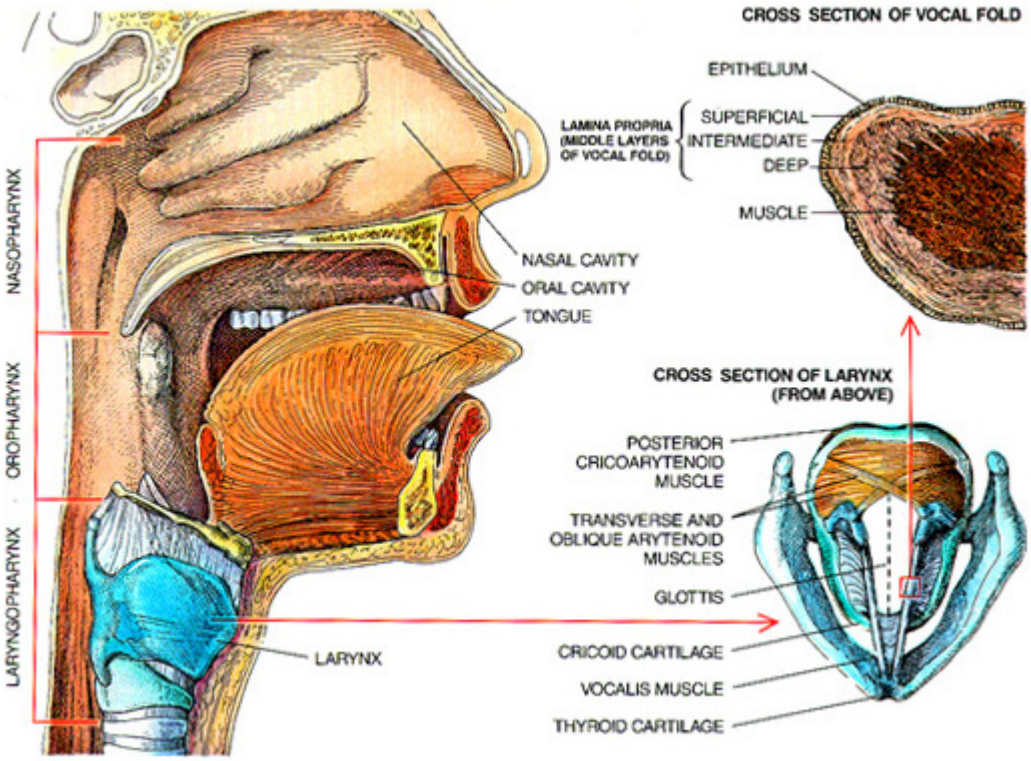


Figure 86: Organs of speech (University of Pittsburgh Voice Center, 1996).

The main elements used for the production of speech are the *vocal folds* (*glottis*), *velum (soft palate)*, *nasal cavity*, *oral cavity*, *jaw* and *lips* (Parent, King and Fujimura, 2002). *Vocal folds* are the fleshy

surfaces that can control flow of air in between them. The *velum* is the flap at the back of the oral cavity. The *palate* is the upper bound of the oral cavity.

The *tongue* is one of the most important elements in the modification of sound. It is controlled by four external muscles, as well as four internal muscles that control its shape. The lips are controlled by about twenty linear muscles, in addition to the orbicularis oris – the sphincter muscle that controls the opening of the mouth. The organs involved in the production of speech are illustrated in Figure 86.

Sound production occurs due to the vibration of the mucosa at the inner edge of each vocal cord, as illustrated in Figure 87. The vocal cord is closed prior to production of the sound (A). Air pressure develops below the vocal cords due to exhalation (B). The vocal cords then briefly separate, releasing the air (C). After the release, the vocal cords assume a particular shape, in order to produce the sound (D), after which they close up again (E). Most human sounds are created by movement of the air in and out of the lungs (Hall, 1992). This air stream is modified by parts of the body through which it travels, thus modifying the sound frequency and producing different sounds. The lungs are not the only part of the body able to produce the air flow – to a lesser degree, the pharynx and the mouth are also capable of this action.

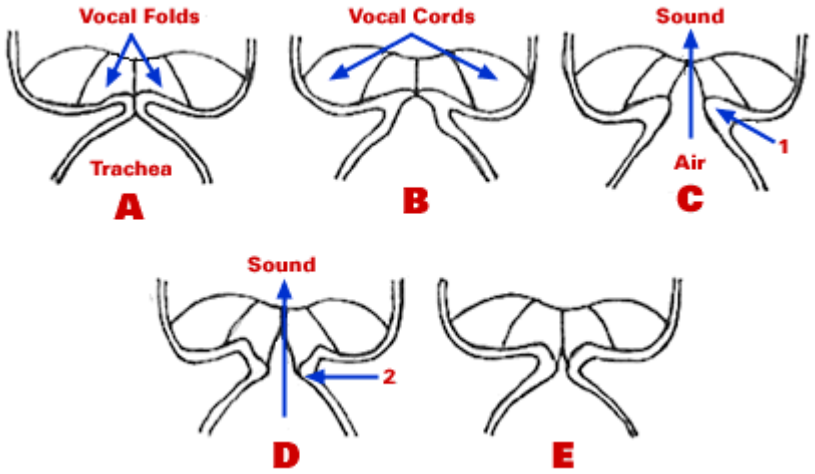


Figure 87: Sound production (University of Pittsburgh Voice Center, 1996).

The larynx area is solely responsible for voicing and the pitch of the air flow. When the vocal cords in the larynx narrow while the air flow is passing through it, they cause the speech to be voiced by vibrating. Conversely, if the voice cords are relaxed, the vibration does not occur and the speech is

thus not voiced. Once the air has passed through the larynx area, it passes through one or more of the three cavities, namely pharyngeal, oral and nasal cavities. The human body can independently close or open any of the three cavities, producing unique sound combinations. The tongue is another important aspect of the production of sound. The sound varies as different parts of the tongue touch different parts of the mouth.

8.2 Phonemes and visemes

The world authority when it comes to phonetics is the International Phonetic Association (IPA, 2005). On their web site several charts are published, depicting all possible phonemes a human mouth can articulate. According to Fleming and Dobbs (1999:107), there are 40 phonemes in English. In the task of assigning visemes to these 40 phonemes, two main divisions can be made in order to simplify the task – facial expressions and tongue positions. Although there are only ten different facial expressions that represent all 40 phonemes, they are combined with several tongue positions, forming 16 unique combinations for English. Figure 88 depicts the distinct facial expressions and tongue positions used to represent all 40 phonemes used in the English language.

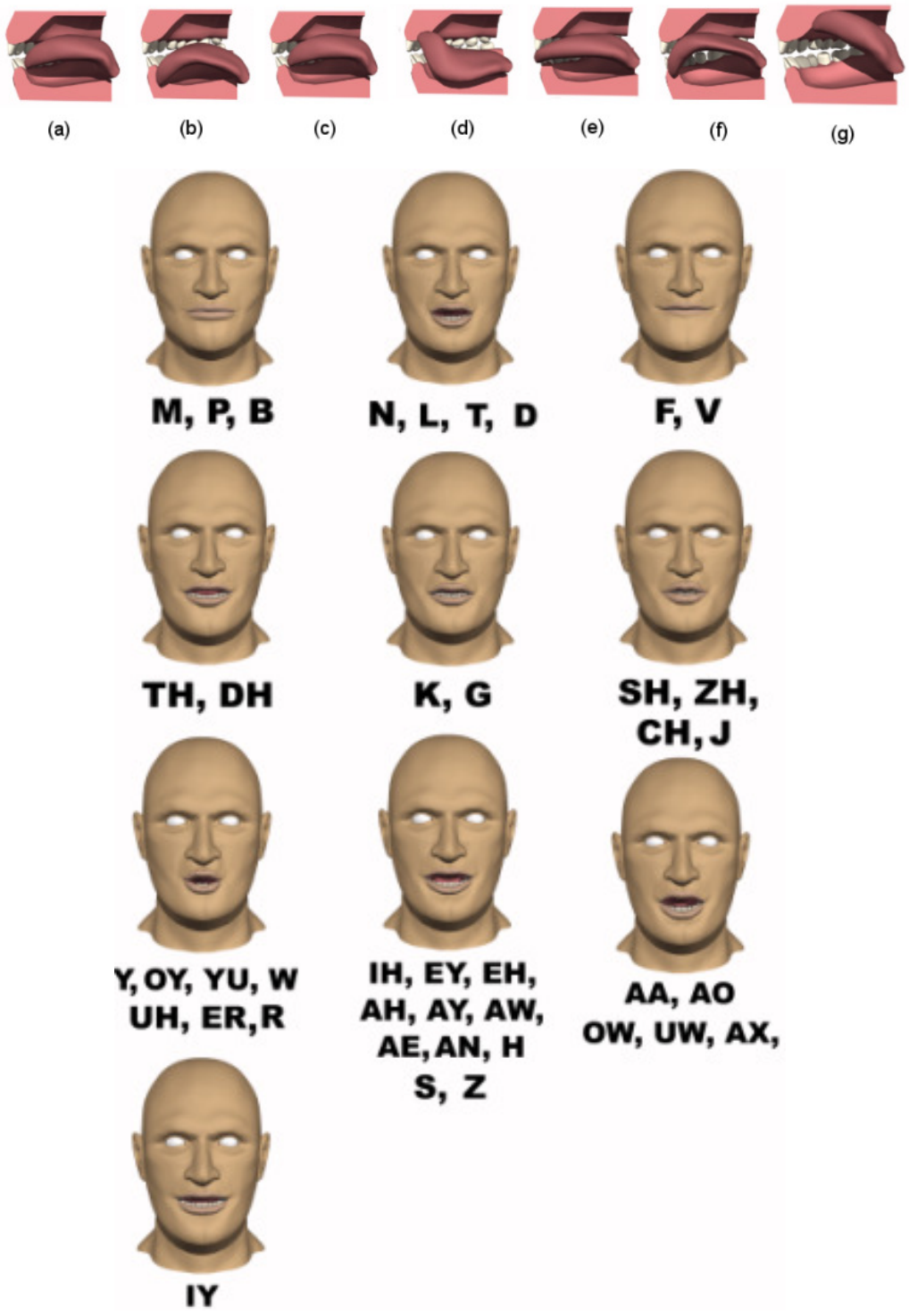


Figure 88: Distinctive tongue positions and facial expressions for speech synthesis (Fleming and Dobbs, 1999:112).

A simplified speech animation approach ignores tongue movement and bases speech synthesis purely on the ten facial expressions. While this may be adequate when seen from a distance, it looks unnatural from a close range, as viewers are accustomed to seeing tongue movements.

Phonemes could be divided into groups by using different criteria:

- By *point of articulation* – physical point in the mouth where the air stream is obstructed:
    - *Lips* (labial);
    - *Teeth* (dental);
    - *Hard palate* (palatal);
    - *Soft palate* (velar);
    - *Back of throat* (uvula/glottis).
- By *manner of articulation* – type of obstruction of the air stream:
    - *Vowels* – with non-obstructed air stream.
    - *Fricative* – formed by forcing air through a narrow gap, creating a hissing sound. They last longer than any other type and are awarded a higher frame count in an animation. Fricative consonants are **F, V, TH, DH, S, Z, SH, ZH** and **H**.
    - *Plosives* – sound is produced at the point of articulation. Since they are formed by a burst of air, their lifespan is short. Plosive consonants are **P, B, T, D, K** and **G**. They often last shorter than a frame, hence are usually dropped from the visual representation.
    - *Affricative* – a special case of a plosive consonant immediately followed by a fricative in the same place of articulation, forming a new unique phoneme. An example of this is **D** followed by **Z**, forming **J** as in ‘jump’. Affricatives are **CH** and **J**.
    - *Nasal* – consonants which require air to go through the nose. The oral cavity is blocked, while the soft palate is lowered to allow air to pass. Nasal consonants are **M**, **N** and **AN**.

- *Voicing* – phonemes are either voiced or voiceless. A phoneme is voiceless when vocal cords do not vibrate during the articulation.

Vowels are formed by a change in the position of tongue, lips and palate. Vowels are divided into

- *Unitary* – in which there is no change in articulation position. These include **IY, EY, EH, AE, AA** and **AO**.
- *Diphthong* – those composed of two vowels gliding from one to another (between two different articulation positions). They are usually quite long, with the first vowel of the two dominating. Diphthongs are **AY, OY, AW** and **YU**.
- *Glides* – are a subclass of diphthongs and are even slower. Glides in English language are **Y** and **W**.
- *Liquids* – subclass of diphthongs that produce a rolling sound. These are **L** and **R**.

8.3 Jaw

The human jaw or *mandible* (Figure 90) is the only mobile bone of the entire skull. It is attached to the *temporal bone* via a joint, suitably named *temporomandibular* joint or TMJ (Figure 89).

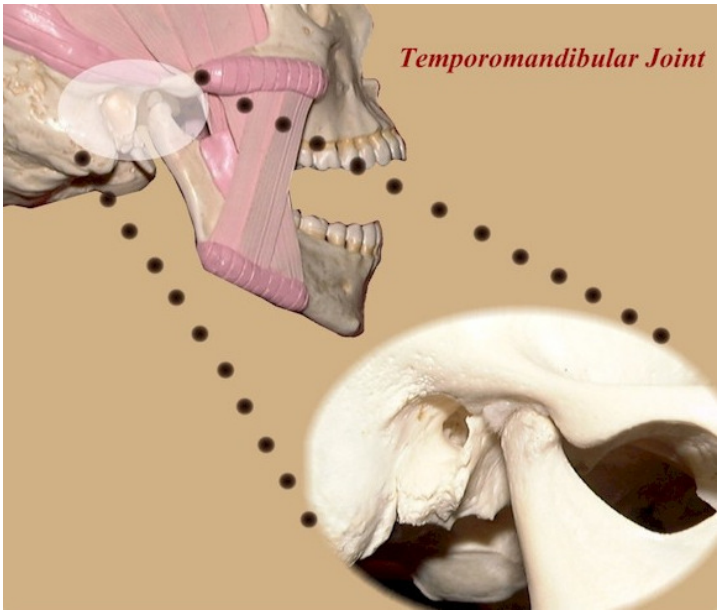


Figure 89: Temporomandibular joint (Daeman College, 2004).

TMJ is the most complex joint in the human body, as it allows for both rotating and sliding movements. Fortunately for animators, the jaw mostly rotates (as opposed to sliding) during the articulation of speech, and hence its animation can be simplified without significantly impairing the realism. However, if one attempts simulation of mastication or yawning, sliding of the TMJ should be considered.

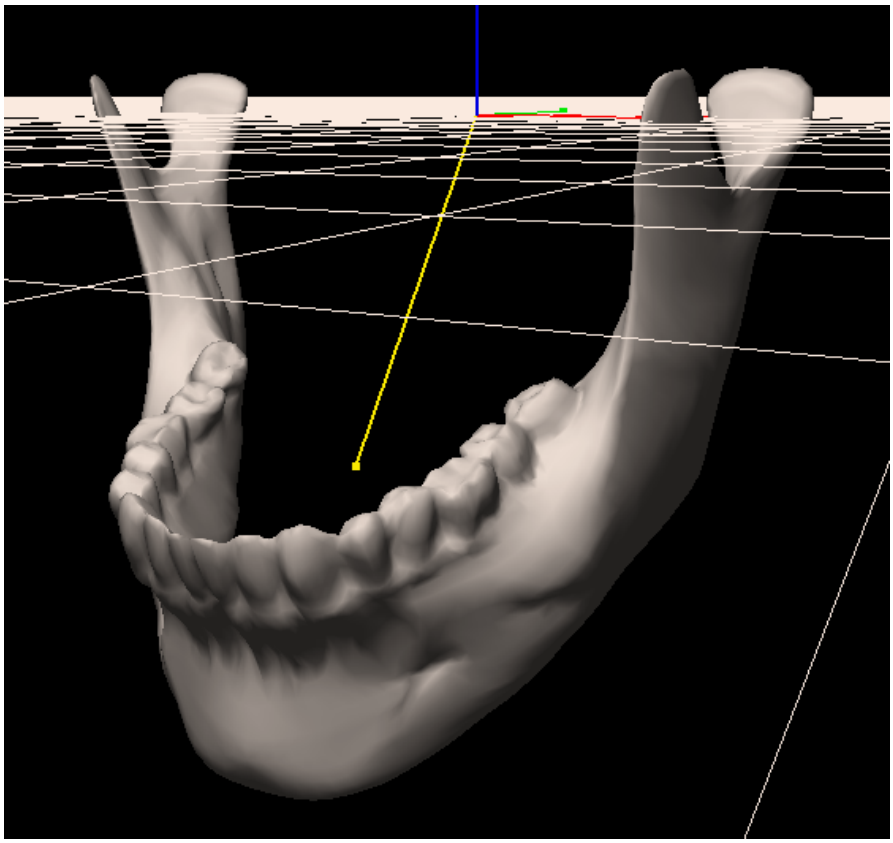


Figure 90: Human female mandible model (Fleming and Dobbs, 1999).

The mandible itself is not often modelled, as it is not visible externally. Instead, an approximation of its motion is simulated. Parke and Waters (1996:207) in Parke's direct parameterised model first establish the jaw pivot axis. To simulate the mandible motion, all the vertices of the lower part of the face are rotated about such an axis. The rotation is not even, but tapered, and it stretches the skin. This deformation is not constant. Intuitively, there is less resistance to movement in the middle of the lower lip, than it is at the edges of lips. The upper lip is not affected by the rotation, but raised or

lowered by using an extra parameter. With minor modifications, this approach has been used by most of the attempts to date.

Choe, Lee and Ko (2001) described the mandible mechanics in greater detail. They defined the rotation axis, and then a point on the chin v_j. Vector e_j connects v_j to the closest point in the rotation axis. A predefined unit of force f_i acts perpendicularly to both e_j and the rotation axis, in the direction of the vector d_{ij}. This is depicted in Figure 91.

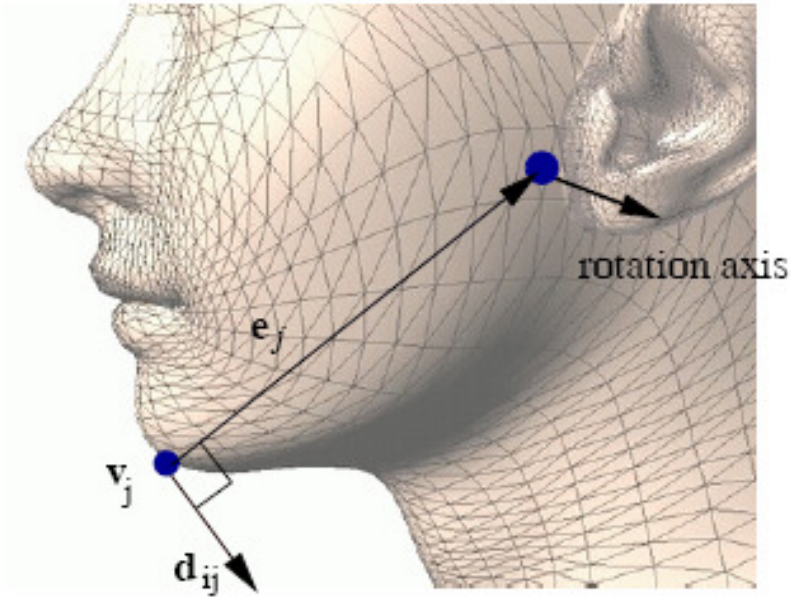


Figure 91: Jaw rotation (Choe, Lee and Ko, 2001).

The force expression is then

$$f_{ij} = a_i f_i \left| e_j \right| d_{ij} \tag{8-1}$$

where a_i is the rotation parameter for the jaw, normalized between 0 and 1, and $\left|e_j\right|$ is the magnitude of e_j. Although this method shows satisfactory results for the cheek and chin, it fails to realistically deform the lips. To alleviate this, heuristic correction forces are applied.

A concept similar to Parke's has been used by Bui (2004). He did not model the jaw bone, but simulated the effect of its opening on the skin surface. He also rotated vertices of the bottom part of the face around an imaginary mandible rotation axis and used the tapered rotation of the jaw vertices. Bui did deform the upper lip too, as it attempted to provide resistance to stretching. Again, the deformation was more pronounced towards the edges of the mouth.

8.4 Lips

In recent years, lip modelling and animation has established itself as a research area in its own right. It poses some unique additional challenges in comparison to the rest of the face. The lips are largely controlled by a muscle called orbicularis oris, which is a sphincter muscle situated just below the lip surface. Sphincter muscles contract around an imaginary central point and resemble a string bag. This requires a unique skin surface which is capable of being deformed in a way that is consistent with the muscle. As we know from personal experience, lips are a highly elastic and deformable part of the face – they can assume countless different expressions. The colour and surface of the lips are also different from the rest of the face.

Over the last three decades of research in the field of facial modelling and animation, the mouth and lips have largely been treated as an equal part of the face, not receiving their deserved attention. Parke and Waters (1996:207) described Parke's parametric method of lip control. He used ten parameters to control the lips, teeth and jaw. This was one of the earliest computerized efforts and was included in Parke's PhD thesis in 1974. It permitted only limited lip motion and little consideration for the shape of the lips. Lips were represented as a polygonal sub-mesh, together with the rest of the face. Almost two decades later, Waters and Levergood (1993, 1994) used a simplistic polygon-based model of the face, animated using the mass-spring method. The lips were still an integral part of the face mesh, while mouth positions for the phonemes were formed from the predefined dataset (illustrated in Figure 92), created by observing the real lips.

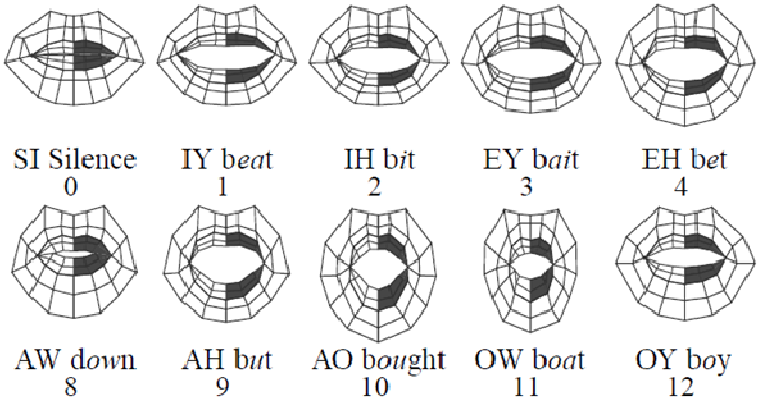


Figure 92: Sample of mouth shapes (Waters and Levergood, 1993 and 1994).

One year later, Beskow (1995) updated Parke's original model (Parke and Waters, 1996) by introducing additional parameters to control the lips (Figure 93). The most important of these were *lip rounding*, *bilabial occlusion* and *labiodental occlusion*. *Lip rounding* facilitates the synthesis of the round vowels. Lip vertices are moved from their initial position towards the centre of the lip opening by a percentage specified by an appropriate parameter.

Figure 93: Change of lip rounding parameter (Beskow, 1995).

Bilabial occlusion pulls the lips towards each other. This would have been possible even without introducing an additional parameter, but it would involve changing up to three original parameters on Parke's model. Finally, *labiodental occlusion* pulls the lower lip towards the edge of the upper front teeth. This is required to achieve the visual difference in visemes /f/ and /v/.

A novel approach, which involved porting the polygonal lip model to an implicit surface model, was presented by Guiard-Marigny et al. (1996). The lip shape of their model is controlled by only five

parameters and limited between ten extreme shapes (two per parameter). All the shapes are achievable by interpolating between the ten extreme shapes. This model was initially used by King, Parent and Olsafsky (2000) and King (2001), but has later been abandoned in favour of a muscle-based approach. The reasons for this departure were the inability to express emotions and unrealistic rounded lip positions for phonemes such as /o/. They represented the lips as a B-spline surface with a 16x9 control grid, as shown in Figure 94. Their lip model contains the vermillion zone (red area), along with the invisible part on the inner side of the mouth. The positions of the control points are influenced by direct muscle actions and the movement of the jaw. This approach appears to be the currently most advanced lip modelling and animation technique within the field of geometric deformations, as explained in greater detail in King (2001) and King, Parent and Olsafsky (2000), as follows:

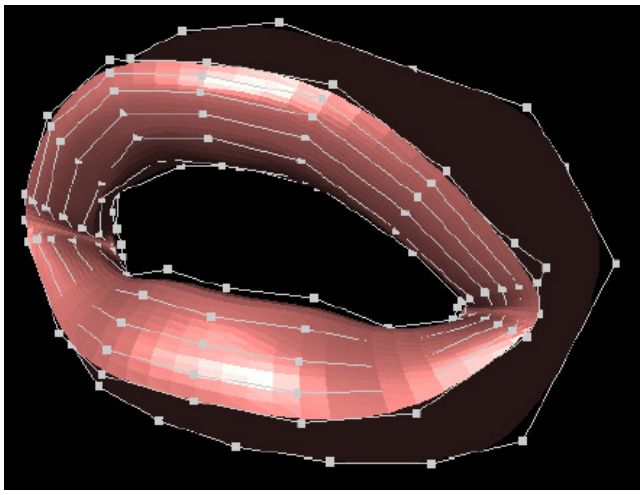


Figure 94: Lip model showing the control points (King, Parent and Olsafsky, 2000 and King, 2001).

All of the muscles except the orbicularis oris are treated as a vector displacement acting upon its insertion points. The orbicularis oris constricts the shape of the lips into an oval while also extruding them. The parameters for the jaw articulate a virtual mandible based on the three jaw parameters, J_{open}, J_{in} and J_{side}, and its resulting transform, J_i, is used to move the lower lips. The lower lip is rotated outward with the opening of the jaw as well. For each control point, its position is calculated based on the parameters by the following:

$$p_i' = p_i'' + L_i + J_i \tag{8-2}$$

where p_i'' is the starting value for control point i, L_i is the sum of the displacements from the linear muscles and J_i is the contribution of the jaw:

$$L_i = \sum_{j=0}^{m} \rho_j M_j \delta_{ij} \tag{8-3}$$

where m is the number of linear muscles, $\delta_{ij} = 1$ if muscle j inserts into p_i and is 0 otherwise, M_j is a vector representing the maximum displacement caused by muscle j, and ρ_j is the parameter value for muscle j. The value J_i represents movement of the lips due to the articulation of the mandible and it is calculated as:

$$J_i = J_{open} + J_{in} + J_{side} + LD\alpha_i + Open\gamma_i \tag{8-4}$$

where J_{open} is the effect of opening the mouth using the mandible, and is a rotation about the axis running through the condyles. J_{in} is the movement of the mandible in or out and J_{side} is the lateral movement of the jaw. LD is the motion vector for the lower lip and α_i is a constant which represents how much pulling the lower lip down will affect the upper lip point. $Open$ is how much the mouth is being open (which tightens the lips) and γ_i is how much opening the mouth will pull control point p_i towards the centre of the mouth. $\alpha_i = \gamma_i = 0$ for the lower lip points.

The orbicularis oris pulls the mouth shut like a draw string. The result of its contraction is dependent on the mandible and contraction of the other muscles.

Combining the displacements of the other muscles with those of the orbicularis oris are difficult, so instead the effect of the other muscles is calculated first, which is then used to calculate the effect of the orbicularis oris. The function O_i calculates the motion resulting from contraction of the orbicularis oris on control point i. The final location of control point p_i is

$$p_i = O_i(p_i') \tag{8-5}$$

where

$$O_i(p) = o(\theta_i + e_i(p) + \mathrm{X}_i) \tag{8-6}$$

and o is the value of the orbicularis oris parameter, θ is the maximum rotation due to the puckering of the lips, and X_i is the maximum extrusion from contraction of the orbicularis oris. $e_i(p)$ calculates an ellipse shape for the mouth and returns a motion vector for moving the control point p to a point on the ellipse created when contracting the orbicularis oris.

An independent approach to modelling and animation of lips using a NURBS muscle model (Figure 95) is described by Tang, Liew and Yan (2004).

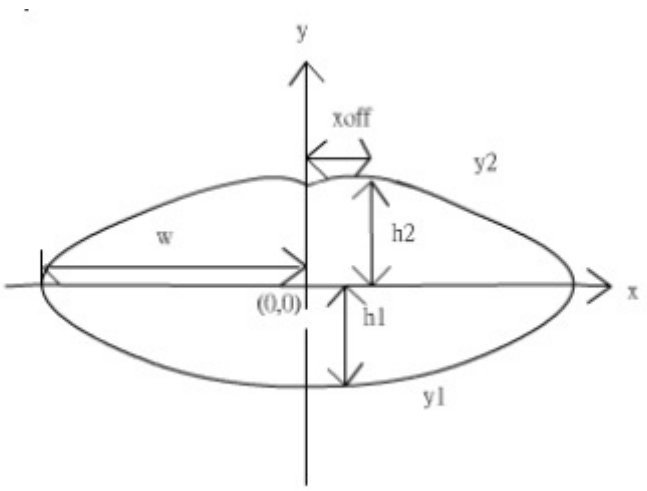


Figure 95: Geometric lip model (Tang, Liew and Yan, 2004).

They developed a method of reconstruction of the lip shapes from video, and then used this information to animate their model. The model is constrained by the two curves, 8-7 and 8-8, illustrated in Figure 95:

$$y_1 = h_1\left(\left(\frac{x - sy_1}{w}\right)^2\right)^{1+\delta^2} - h_1 \tag{8-7}$$

$$y_2 = \frac{-h_2}{\left(w - x_{off}\right)^2}\left(\left|x - sy_2\right| - x_{off}\right)^2 + h_2 \tag{8-8}$$

where s describes the skewness of the lip shape, δ represents the deviation of the curve from a quadratic, while the rest of the dimensions are visible in Figure 95. For more details about NURBS and the NURBS muscle model, the reader may refer to Sections 3.2.2 and 6.2.5.3. According to the above lip model, five parameters are used to control the lip contour:

$$p = \left\{w, h_1, h_2, x_{off}, \delta\right\}. \tag{8-9}$$

The parameters influence the weights of certain muscle control points in order to achieve the desired lip shape. Animation is achieved by appropriately varying these parameters.

Another challenge is that the lip opening is often asymmetrical during speech production (Cave et al., 1996). Massaro et al. (2005) addressed this problem in their model, and found that asymmetry of the lip opening causes a rip along the facial midline. To correct this, algorithms compromise different sides of the face by adjusting Y and Z coordinates of each point within a threshold distance from the midline. Implementation of the asymmetry is achieved by implementing steeper dominance functions to the dominant side (see Section 9.3). Alternatively, one could simply have two segment definitions, one for each side.

When the lips are modelled and animated using different techniques from the rest of the face, the animator may encounter the problem of *grafting*. Grafting refers to the synchronisation of a face subset with the rest of the face**.** Due to their unique properties and requirements imposed on them, the lips are often considered in isolation from the rest of the face. This introduces the problem of 'fitting' the lips into the face model. King (2001) described his method of synchronisation of the lips with the rest of the face model, and this process is illustrated in Figure 96. He did it interactively, by first fitting the lip model over the existing facial model (a, b). Overlapping triangles of the facial model are then removed (c). Edges are added, to connect the lip model with vertices of the facial model where the removed triangles were, thus triangulating the space between the lips and the rest of the face (d, e). Finally, the lip model is added to the rest of the geometry (f).

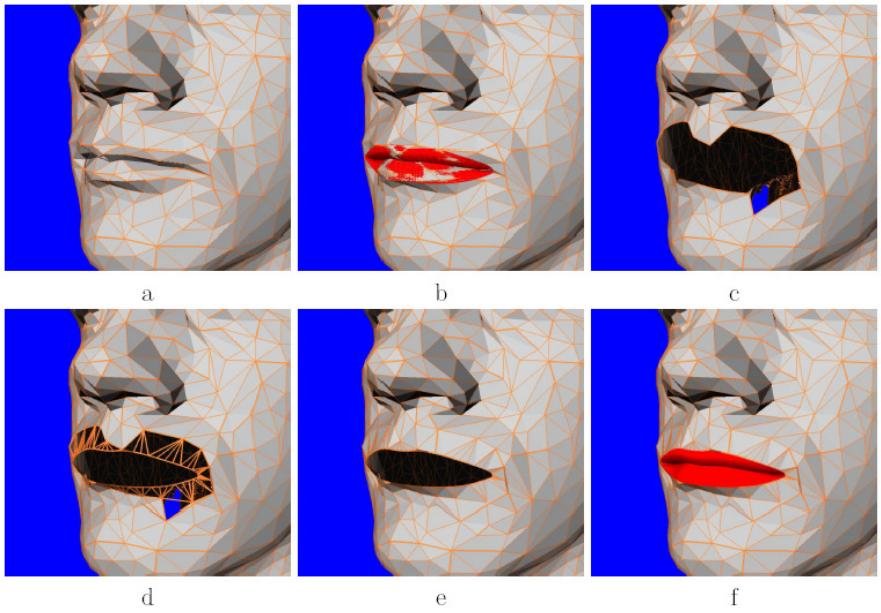


Figure 96: Grafting of lips onto the facial model (King, 2001).

8.5 Tongue

Tongue modelling has for a long time been considered unnecessary due to its limited visibility during speech. During the early days when facial modelling and animation were far from realistic, the absence of an appropriate tongue model was not all that important. In recent years, facial modelling and animation have achieved such a degree of realism that the tongue has become an important contributor to the realistic perception of a human face. The difference in perception of a face with and without a tongue is illustrated in Figure 97.

Figure 97: Supporting viseme for the phoneme /n/, with tongue (left) and without tongue (right) (King, 2001).

Parke's original parametric model (Parke and Waters, 1996) did not cater for the tongue. Cohen and Massaro (1993) modified Parke's model, adding the tongue and its parameters. They did not concentrate on realistic representation, but focused on correct linguistic and psychological aspects for the purpose of their coarticulation research. They added four tongue parameters, namely length, angle, width and thickness. A more realistic tongue implementation was attempted by Pelachaud, van Overveld and Seah (1994).

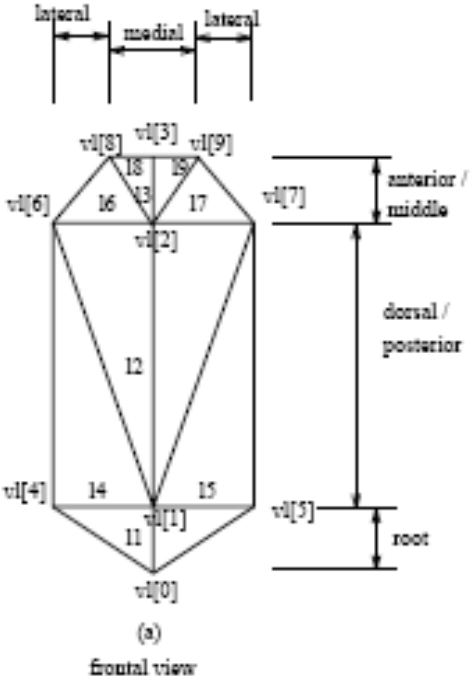


Figure 98: Tongue frame used by Pelechaud, van Overveld and Seah (1994).

They used a frame composed of triangles to represent the tongue base (Figure 98). The vertices of the base are used in animation, while the actual tongue is built at a relative offset from the frame. Positions for the vertices are defined for each viseme, and animation is effected via interpolation between two viseme frames. For the purpose of penetration avoidance, the mouth cavity is approximated to simple geometric shapes, such as spheres and planes. During the animation, each

vertex is tested for penetration. If the penetration occurs, the vertex is projected onto the surface in question.

Another tongue implementation originating from Parke's model (Parke and Waters, 1996) was that of Beskow (1995). He used a modified Parke's model for his text-driven speech synthesis, approximating the tongue movement by its tip, and disregarding the tongue body for simplicity. The model was composed of 64 polygons and controlled by four parameters, namely length, width, thickness and apex (Figure 99).

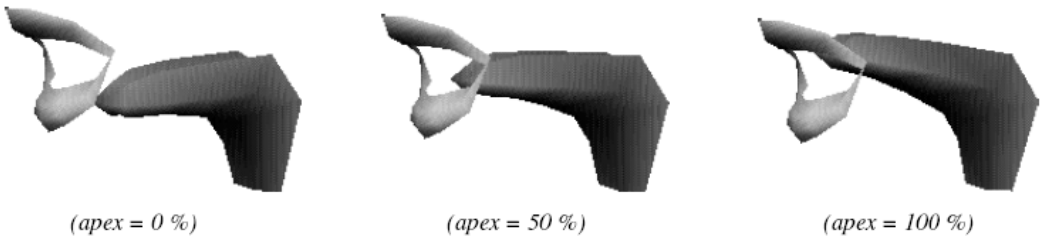


Figure 99: Beskow (1995) tongue model.

A recent B-spline surface tongue model was described by King and Parent (2001) and King (2001). Their tongue model was fast to render and deform, realistic in shape, capable of any tongue shape needed for speech, capable of other tongue shapes for general facial animation, able to perform collision detection between itself and rest of the oral cavity and preserved volume during movement and collision. This method consisted of an 8x13 grid of bi-cubic patches over 60 control points. The control points are arranged in a 6x10 grid (Figure 100). Since this approach appears to be the currently most advanced tongue animation technique within the field of geometric deformations, it is explained in greater detail. The following is an extract from King (2001):

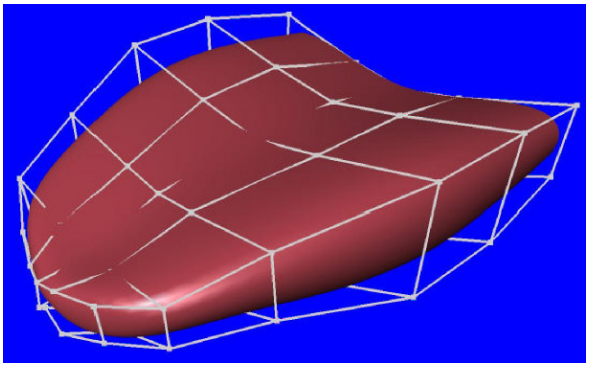


Figure 100: Tongue model with its control grid (King, 2001).

The new control points $P'_{i,j}$ are calculated as

$$P'_{i,j} = P_{i,j} + \sum_{k=1}^{6} \omega_{k,j} p_k \alpha_{k,i} \qquad (8\text{-}10)$$

where $\omega_{k,j}$ is the weight for row j of parameter k, and $\alpha_{k,i}$ is the weight for column i of parameter k (number of bi-cubic spline basis functions), $1 \le i \le 10$ and $1 \le j \le 6$. There are separate weights for the x, y and z directions. For example, the following row weights are used in the x direction:

$$\omega_x = \begin{bmatrix} 1 & .9 & .6 & 3 & 0 & 0 \\ 1 & .5 & .1 & 0 & 0 & 0 \\ 1 & .9 & .9 & .2 & .2 & 0 \\ 0 & .1 & .3 & .9 & 1 & .2 \\ .5 & .9 & 1 & 1 & 1 & 1 \\ .1 & .7 & .9 & 1 & 1 & 1 \end{bmatrix}$$

Reducing the tongue width when the tip is extended, and expanding the tongue during retraction simulates volume preservation. This crude approximation gives reasonable results with no performance penalty.

8.6 Body language

Another important aspect of communication is body language. Research has shown that the total impact of a communication is about seven percent verbal (words only), 38 percent vocal (tone of voice, inflection and the related sounds) and 55 percent nonverbal (Mehrabian, 1972). Humans are not usually conscious that they are perceiving, and thus are influenced by, body language. However, whenever the body language and the spoken words do not 'agree', or seem discordant, they are aware of the inconsistency (Pease, 1981). An example of an attempt to integrate body gestures with facial animation is described by Masaro et al. (2005). They augmented the existing model (Massaro, 1998) by adding the body, arms, hands, shoulders and legs. The primary purpose of this addition was to extend the communication through gesture. In addition, it allowed the character to become more of a general-purpose avatar. The body was animated at a high-level using scripts, and forward kinematics was used for low-level animation. Their body model was not composed of multiple rigid bodies connected by joints, which is the most common approach in computer games. It consisted of a single elastic skin surface, with body gestures being assigned certain meanings, as shown in table in Figure 101.

Visual Gesture	**Meaning**
Head nod	Yes
Thumb up	I approve
Greeting wave	Hello
Shoulder shrug	I don't know
Head shake	No
Thumb down	I reject
Farewell wave	Good-bye
Wink	I'm kidding
Auditory Gesture	**Meaning**
Clap	I approve
Wolf whistle	How beautiful
Raspberry	I dishonor you
Hiss	I disapprove
Rise-fall whistle	How surprising
Tongue-click	Shame on you

Figure 101: Some body gestures and their meanings (Massaro et al., 2005).

8.7 Conclusion

This chapter described various techniques which have been developed for modelling and animating the parts of the face that participate in speech. Apart from featuring all the elements of general animation, facial animation poses its own specific challenges. The face features several sphincter muscles, the most significant of which, regarding speech, is the one that controls the mouth. Due to the peculiarity of its actions and its importance for communication, scientists have realised that it requires special attention. We discussed several approaches to lip modelling and animation, and described in detail one which is considered the most advanced technique based on geometric deformation. The tongue is another peculiar muscle, which is significant for a realistic image of visual communication. It does not conform to a muscle stereotype as its purpose is not to support another form of skeleton structure. It can expand and contract, widen and narrow, having six degrees of freedom. For the purposes of facial animation, it is not necessary to cover all possible tongue movements, as this muscle is largely obscured by the mouth. However, realistic simulation of the visible parts is an important ingredient of successful animation. This is now acknowledged by many scientists in the field. Several tongue modelling approaches were covered, providing details of the one that we consider the most advanced within the realm of geometric deformation techniques.

Chapter 9 Lip synchronisation and related issues

Lip synchronisation refers to the matching of phonemes uttered to the relevant and appropriate lip and tongue positions. The aim is to create a realistic illusion of the animated character speaking. This is difficult to achieve, as human audiences are highly sensitive to any inconsistency between visual and auditive stimuli. Lip synchronisation can be divided into manual and automated synchronisation. Manual synchronisation occurs when the animator listens over and over again to the relevant audio track until he or she can allocate a viseme to each phoneme that is heard. Interpolation between the visemes is usually executed by an animation software system.

Alternatively, automated lip synchronisation reads and animates a prepared sequence of visemes without the need for human intervention. By the method used to prepare such sequences of visemes, automated lip synchronisation can be divided into three main areas: text-, speech- and image-driven approaches.

The text-driven approach is the simplest of the three. The sequence of visemes is given in the form of a text file. The speech-driven approach analyses an audio track of the speech that is to be animated and then prepares the sequence of visemes from this analysis. Conversely, the image driven approach analyses video footage, then uses this information to prepare the sequence of visemes for the animation.

9.1 Manual lip synchronisation

Fleming and Dobbs (1999:122-131) described a practical approach to lip synchronisation. In their method, the main prerequisite for lip synchronisation is a 3D model of the character to be animated. It is important that this model is suitable for animation, having its vertices and edges consistent with the required deformation caused by expressions and speech. The next step consists of modifying the 3D model and creating each individual viseme that is needed. Once this preparatory work is completed, the animator would proceed with a breakdown of the speech pattern. There are many accents and speech patterns, even within the same language, for the same sample of a printed text. For example, some people slur their speech (skipping consonants), which sounds very different from ordinary speech. Likewise, the visual representation of the two would also differ. The animator would need to listen to the speech and write down its phonetic representation, disregarding the actual spelling. Fleming and Dobbs (1999:124) demonstrated this using a cartoon character named Knuckles. He

would pronounce the sentence '*You shouldn't ought to talk to me like that*' as '*Ya shudnada tak tuh me like dat*'.

The next step is breaking down the newly derived speech into phonemes. An audio editing tool may be used here, preferably one with a time scale in both frames and seconds. Another useful feature is the 'scrub' tool, which allows the animator to move forwards and backwards through the sound at will. The following phonemes represent the mentioned example:

'Y AH SH UH D N AA D AH T AA K T AH M IY L AY K D AE T'

Once the content is determined, the animator would locate the range of frames over which the phoneme lasts. The scrubbing tool may facilitate this task, as will the visual representation of the speech (Figure 102).

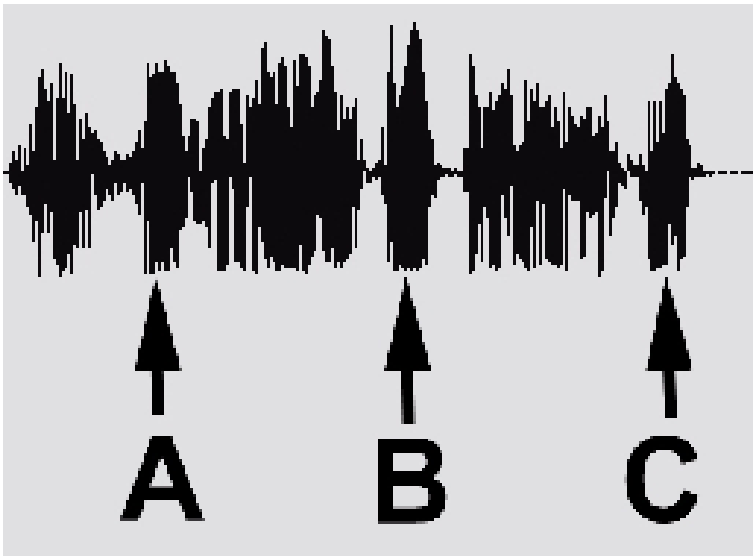


Figure 102: Visual representation of a speech sample (Fleming and Dobbs, 1999:126).

The visible bursts marked as **A, B** and **C** represent the words *should*, *talk* and *that*, respectively. In addition to that, we could visually identify the plosive consonants, as they form peaks on the graphic representation of the sound file (Figure 103).

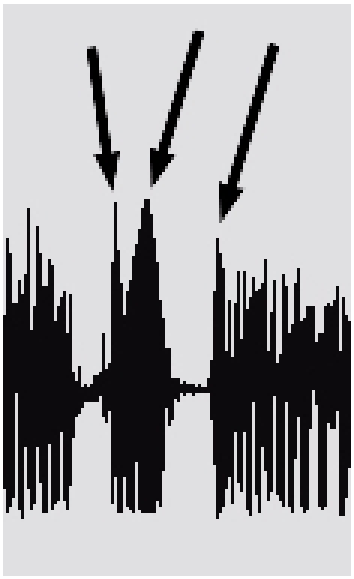


Figure 103: Visual identification of plosive consonants (Fleming and Dobbs, 1999:127).

Other ‘landmarks’ in our wave files are vowels (Figure 104). They are voiced and the airflow is not obstructed during their articulation, which is why they tend to be louder.

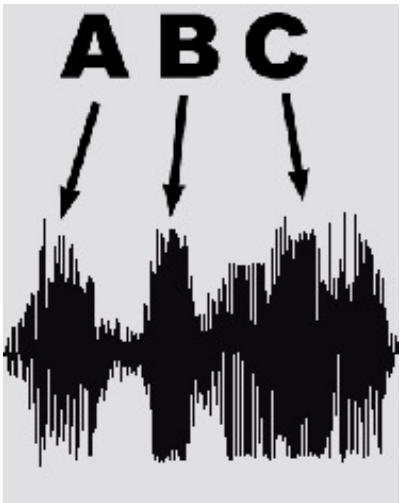


Figure 104: Example of vowels in a sound file (Fleming and Dobbs, 1999:128).

The sound file is played backwards and forwards, until the precise frame position of each phoneme is established. The results are entered in a table, as in Figure 105.

PHONEME	Target	Frame		PHONEME	Target	Frame
Y	7	2		L	2	53
AH	8	6		AY	8	57
				K	5	61
SH	6	10				
UH	7	15		D	2	62
D	2	19		AE	8	63
N	2	20		T	2	66
AA	9	25				
D	2	30				
AH	8	32				
T	2	37				
AA	9	38				
K	5	42				
T	2	45				
AH	8	46				
M	1	48				
IY	10	50				

Figure 105: Individual phonemes and their position in the sound file, measured in frames (Fleming and Dobbs, 1999:129).

The frame position of each phoneme is a key-frame, which serves as a reference to the animation system for interpolation (Fleming and Dobbs refer to interpolation as ‘morphing’). Visemes produced for the individual phonemes are mapped to the appropriate frame according to the table.

9.1.1 Phoneme reduction techniques

Traditional animators of cartoons advocate simplicity. They maintain that attempting accurate lip motion with animated characters usually looks unnatural. Rizvic and Avdagic (2004) performed an experiment to demonstrate this claim. The software tools used for this experiment were 3D Studio Max, Morpher modifier and MaxScript. They progressed through several phoneme reduction phases, producing a video clip for each phase. The initial phase contained all phonemes in a chosen sentence. While the lip movement appeared accurate, it was not sufficiently smooth. The authors attributed this to visemes that last less than 1 frame (1/25 sec in PAL standard), which are too short for a regular PAL or NTSC motion picture standard. Subsequent phoneme reduction phases yielded better smoothness, without noticeable loss in accuracy. The last phase, containing only four visemes, was oversimplified and unconvincing. The limitation of the experiment was that the tongue was not modelled, and it did not take coarticulation into consideration.

Fleming and Dobbs (1999:131-134) also described some practical guidelines for phoneme reduction:

- A phoneme at the beginning of a word should never be dropped.
- Consonants at the end of words often may be dropped without a noticeable effect.
- Commonly dropped phonemes are plosives, as they are fast and not often registered visually.
- Nasal phonemes are frequently dropped in order to smoothen the transition. The greatest problem is 'm', for which the mouth briefly closes and opens again. This movement is usually less than a frame long and if performed, it makes the animation look unnatural.

Albrecht, Haber and Seidel (2002) identified closure (when the lips are held closed for a period of time) as an important visual cue, regardless of how short it is. They assigned high importance to the dominance function of a closure phoneme (see Section 9.3.3 on coarticulation), ensuring that the lips are sufficiently close at closure. Short phonemes, other than closures and releases, were distributed in such a way that the animation frame matched the peak dominance function of the phoneme. In this way, even short phonemes made their appearance in the animation. Figure 106 represents phonemes and frames as a function of time. Phonemes are depicted as alternately painted flat boxes indexed s_1 to s_5. The frames are denoted by green arrows and indexed f_1 to f_8. If the frames were distributed evenly through time, the phoneme s_3 would not be visible in the animation. In order to include the phoneme in the animation, the frame f_5 had to be moved back in time.

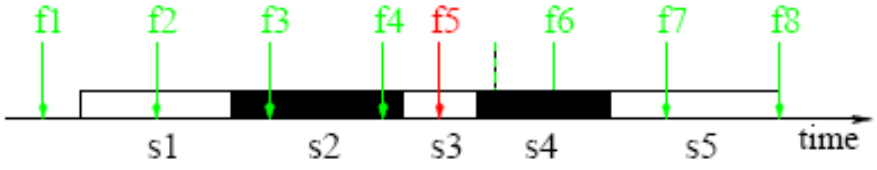


Figure 106: Manipulation of animation frames, so that they match the peak dominance function of a phoneme (frame f5 was repositioned in order to cater for phoneme s3: Albrecht, Haber and Seidel, 2002).

9.2 Automated synchronisation

With the aim of avoiding the tedious manual lip synchronisation discussed in the previous section, a significant part of lip synchronization efforts has been devoted to the automation of the process. These efforts can be divided into three major functional groups, namely those driven by text, speech and image. Text-driven lip synchronisation attempts to produce visual and audible phonemes from a printed text. Speech-driven analyses a pre-recorded speech file, extracting the required phonemes from it. These phonemes are then used to produce the visual aspect of synthesized speech. The image-driven approach uses video frames in an attempt to identify the uttered phoneme. Another approach that is worth mentioning is the hybrid approach, which combines text-driven and speech-driven approaches in an attempt to further improve the automation of visual speech.

9.2.1 Text-driven approach

In a text-driven approach, visible and audible speech are synthetically generated from a text. There are two main directions in the text-driven approach. One consists of mapping text to visemes by means of vector quantization, initiated by Waters and Levergood (1993 and 1994). The other utilizes a rule-based system, as described by Pelachaud, Badler and Steedman (1996) and Beskow (1995).

Waters and Levergood (1993 and 1994) relied on speech synthesizers (DECtalk) to interpret the text and provide their system with phonemes. Their system follows the synthesized pronunciation with the corresponding expressions. Time is a critical aspect of this approach. In order to be realistic, the visual and the audible systems need to have a common time base reference. Since the audible system operates at a far higher frequency, the visual system calculates its interpolation in terms of the audible system's clock. Pelachaud, Badler and Steedman (1996) based their approach on the Facial Action Coding System (FACS) (see, for example, Ekman, Friesen and Hager, 2002) , using a text file as their speech source. In addition, the source text is required to be in its phonetic representation beforehand. Apart from the phonemes, the text would contain affect and its intensity, phoneme timing and intonation structure. The algorithm reads the file and computes the lip shapes. Action Units (AU) (Ekman, Friesen and Hager, 2002) are used for the basic lip shapes. These basic shapes are then modified by taking into account the influence of emotions, coarticulation and other contributing factors. As the animation engine, the authors used Jack – animation software developed at the University of Pennsylvania (Badler, Phillips, and Webber, 1993). Beskow (1995) used a parameterised model based on the one originally designed by Parke (Parke and Waters, 1996). The main differences between Parke's and Beskow's model are in the introduction of lip movement

parameters and the tongue. Beskow used two existing systems, namely RULESYS, that converts the plain text into phonemes and other speech data, and GLOVE, to synthesize audible speech. The role of synchronisation between the visual and audible speech is performed by the RULESYS system. Ezzat and Poggio (1998) presented a 2D audio/visual text-to-speech system. The system mapped phonemes to pre-recorded images representing visemes. The images in between two phonemes were achieved by morphing. In the interests of simplicity, coarticulation was ignored.

A pure text-driven approach often forms part of a speech-driven approach. In a speech-driven approach, the speech is often analysed, then converted into a sequence of phonemes, using a text-driven approach thereafter. Speech-driven approaches are explained in the following section.

9.2.2 Speech-driven approach

In a speech-driven approach, the computer first analyses the pre-recorded speech, and then generates visual facial representations. Pioneering the first notable speech-driven automated lip-synchronisation attempt, Lewis (1991) first outlined some of the historical methods. The early naïve approach to lip-synchronisation consisted of opening the mouth in proportion to the loudness of the sound. It is obvious that this approach did not have good results, as pronouncing ‘m’ can be loud, while the mouth hardly opens at all. A more advanced approach involved passing the sound data through a series of filters and producing the animation based on spectra outputs. Although acceptable, this approach, used in the early 1980s, was still unable to produce the fully realistic lip motion. The main drawback was in the spectrum containing the vocal tract and pitch mixed together, while speech depends exclusively on vocal tract data. Lewis’s approach was based on a linear prediction model, and involved the sound being broken down to sound source and vocal tract filtering components. The filtering components were the parts used for lip-synchronisation. Using the linear prediction algorithms, they were converted into a phonetic script. References phonemes were extracted from the script first. Reference phonemes are English language vowels (Figure 107) and the consonants m, s and f.

Figure 107: Mouth positions for the vowels in the English language (Lewis, 1991). The top row are the vowels in hat, hot and the f/v sound. The bottom row are the vowels in head, hit and hoot.

While identifying the vowels using a linear prediction identification approach was not difficult, correctly identifying the consonants posed a significantly greater challenge. For example, when the consonant 't' is at end of a word, the mouth may have to remain open in anticipation of the next word. The model used for the animation was based on Parke's (1996) parameterised approach. The tongue was not included in the animation. An interesting contribution to the speech-driven lip synchronisation school of thought was made by Brand (1999). He used artificial intelligence principles to train his system on face reaction in response to speech. Using this system, face trajectories were mapped directly to the voice patterns, bypassing the process of decomposition of the sound signal to phonemes. This was followed by animation of the phonemes, taking coarticulation into account. Some authors tried to bypass the intermediate step of analysing the audio information to a sequence of phonemes first. Bondy et al. (2001) described a method of concurrent analysis of audio and video. During the training phase, an audio signal is mapped to lip positions deduced from the video. Once the training is completed, the system is able to analyse new audio speech and retrieve the facial expression mappings accordingly. One of the latest additions to speech-driven animation is the capture and reproduction of other audio-conveyable information, such as intonation (Gutierrez-Osuna et al., 2005).

9.2.3 Image-driven approach

Apart from audio, speech can also be deduced from a video source. A computer can employ machine vision techniques to perform lip-reading from video footage. One of the first notable works in this area was by Ezzat, Geiger and Poggio (2002). Their system was capable of analysing video footage, then synthesising visual speech from the analysed data. Once the initial fifteen minute footage is recorded, the system would spend several days analysing the footage and learning (using gradient descent learning) how to synthesise the speech. During the learning period, no user intervention was required.

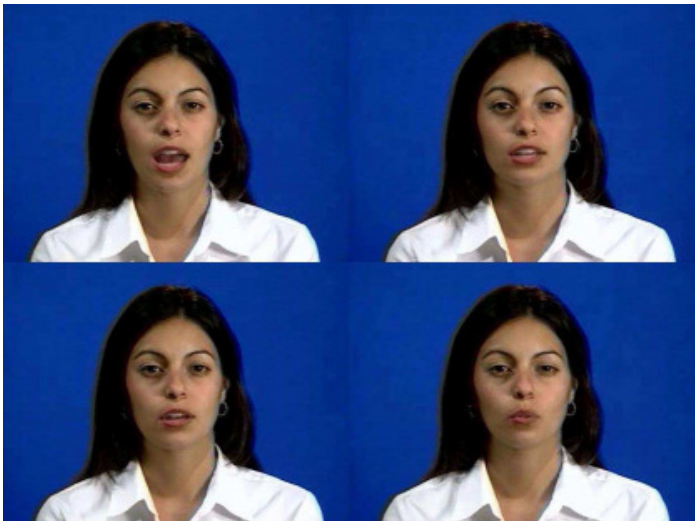


Figure 108: Some results of Ezzat, Geiger and Poggio's (2002) visual speech synthesis approach.

The new video was synthesized using 2D image processing techniques. A multidimensional morphable model (MMM) was used for the generation of expressions, while a trajectory synthesis technique based on regularization was used to generate trajectories within the MMM space. The main disadvantage of this approach was in the difficulty of blending the newly created images into the background: catering for substantial movement of the head, changes in lighting conditions and changes in the viewpoint. Some sample frames produced using this approach are shown in Figure 108. Basu, Oliver and Pentland (2003) designed a system capable of capturing 2D lip motion from a video file, and then animating a 3D lip model in accordance with the analysed data. Finding an accurate way to capture lip movement from video is challenging, partly because there is not much

visual information with which to work. Also, the head seldom remains still. It rotates constantly, leaving a different angle 2D projection of the lips for analysis every time. Also, the contour of the lips is often obscured by lighting, facial hair and other disturbances. The primary way of differentiating the lips from the rest of the face is using their colour content, which is often noisy. Due to all the abovementioned obstacles, it would be extremely hard, if not impossible, to capture accurate lip movement information using only machine vision principles.

As an auxiliary method, Basu, Oliver and Pentland (2003) used statistics of expectancy of posterior behaviour, taking into consideration visible 2D posture. To aid machine vision, sixteen ink marks were plotted on the actor's lips prior to filming, along with one on the nose as a reference. This initial filming was used to train the system (the actual footage did not have any markings). The details of the computer vision principles utilized are omitted, as they are beyond the scope of this thesis.

9.3 Speech animation issues

Due to its nature, lip synchronisation poses several unique challenges, which the animator would need to consider in order to produce a successful speech synthesis.

9.3.1 Bimodal communication

Humans perceive speech both visually and audibly (Massaro, 1998:3-34). Massaro conducted numerous experiments where he subjected the participants to combinations of visual and audible representations of a particular phoneme (or group of phonemes), and then recorded the participants' answers as to which phoneme they had heard. The visual and audible representations were not always consistent, a visual representation of one phoneme was supported by an audible representation of a similar but different phoneme. The experiments demonstrated that the highest percentage of correct matches occurred when the visual and audible representation were consistent. Conversely, the least correct matches occurred when the representations were inconsistent, which indicates that visual experiences indeed affect our perception of speech. The experiments also demonstrated that audible speech has far more influence over perception than visual speech does.

Noting the results from experiments conducted by Massaro (1998:231-240) and de Gelder and Vroomen (2000), we could extend the meaning of bimodal communication to emotions. The authors demonstrated human bimodal perception of emotions by alternating visual and audible cues. They found that, generally, when visual and audible cues were consistent (such as a sad voice with a sad face), the frequency of correct guessing of the emotion by participants was the greatest. The

frequency was lower for unimodal cues (only audible or only visual), and it dropped considerably for confusing inconsistent bimodal cues (such as a sad voice with a happy face).

9.3.2 Intonation

Intonation refers to the involvement of emotions into the utterance, changing the pitch and melody of the words. This feature has been largely neglected in synthetic speech, which is one of the reasons why synthetic speech is usually easily recognizable. Although intonation affects mostly audible speech, visible speech needs to be consistent with audible in order to look realistic. One area where intonation affects visual speech is the rate at which it happens. For example, angry speech is fast, while sad speech is slow. This also has a cumulative effect on coarticulation, as mentioned in the section on coarticulation below.

When discussing the topic of intonation, it is difficult to separate visual and audible sources of information. They complement each other and if either of the two is inaccurate, realistic perception is impaired. Since the field of synthetic speech seems to be far from successful implementation of intonation (Murray and Arnott, 1996), pre-recorded voices of live actors are still likely to be used for some time. If the existing speech is analysed for the purpose of automated animation on a phonetic level only, even with perfect lip synchronisation, the face would remain emotionless. Since the viewer can sense the intonation from the speech which is not matched to an appropriate facial expression, the entire perception would appear unnatural. The question remains: how to extract data related to emotions from the speech?

9.3.3 Coarticulation

Coarticulation can be loosely described as the effect of adjacent phonemes on each other's articulation (Massaro, 1998:372-386). The influence of the preceding segment is known as perseverative coarticulation, while the influence of the upcoming segment is called anticipatory coarticulation. As an example, one could consider the difference in articulation of '**t**' in *boot* and in *beet*. It is obvious that the mouth position is different at the point of articulation of '**t**'. There is also a more subtle difference in the sound of the phoneme.

The mouth cannot assume the perfect posture for every phoneme in a word because of the speed at which words are uttered. While forming the posture for a phoneme, the following phoneme posture is

anticipated. In this process, the posture for the current phoneme is compromised to a certain degree, in order to be able to assume the posture for the following phoneme in time.

This has a significant impact on visual speech synthesis and needs to be considered if one wishes to achieve a realistic facial animation. We discuss two approaches to coarticulation: a three-step algorithm by Pelachaud (1991) and a dominance/blending algorithm by Massaro (1998:376-386).

Pelachaud's (1991) algorithm was based on deformability of the lips (deformability refers to the degree of influence that neighbouring phonemes are allowed to have over the phoneme under review). Different phonemes have different deformability. 'F' is an example of the least deformable, while 'm' is one of the most deformable phonemes. In this coarticulation algorithm, the deformability also depends on the speed of speech. It is evident that the slower the speech, the more time is available for the lips to shape. Hence, the effect of coarticulation diminishes.

The basic idea of this approach is identifying and examining a highly visible vowel before and after the phoneme under review. The viseme shape of the phoneme is adjusted to conform to the shapes of the two neighbouring vowels.

In addition, the lapse of time between two phonemes is considered. Suppose that we observe two consecutive phonemes **a** and **b**. Each phoneme has its articulation time. This time can be functionally divided into three parts:

- Time required for the facial muscles to contract into the correct shape for articulation;
- The actual articulation time in which the viseme does not change and;
- Relaxation time during which the lips restore their neutral form.

If the relaxation time of phoneme **a,** when added to contraction time of phoneme **b** is longer than the time required for articulation during the common speech, phoneme **a** would visually influence the phoneme **b** in such a way that the lips would have to start contracting to articulate **b** somewhere on the relaxation path of phoneme **a**. Where exactly this occurs depends on various factors, such as position in the word in question, and the accent and language of the speaker.

Elements of this approach have been put into practice by Pelachaud, Badler and Steedman (1996). Their system was discussed earlier in Section 9.2.1. The authors admitted that it is often not enough to analyse the segments immediately before and after the current one, as the current position can depend on up to five segments before or after (Figure 109).

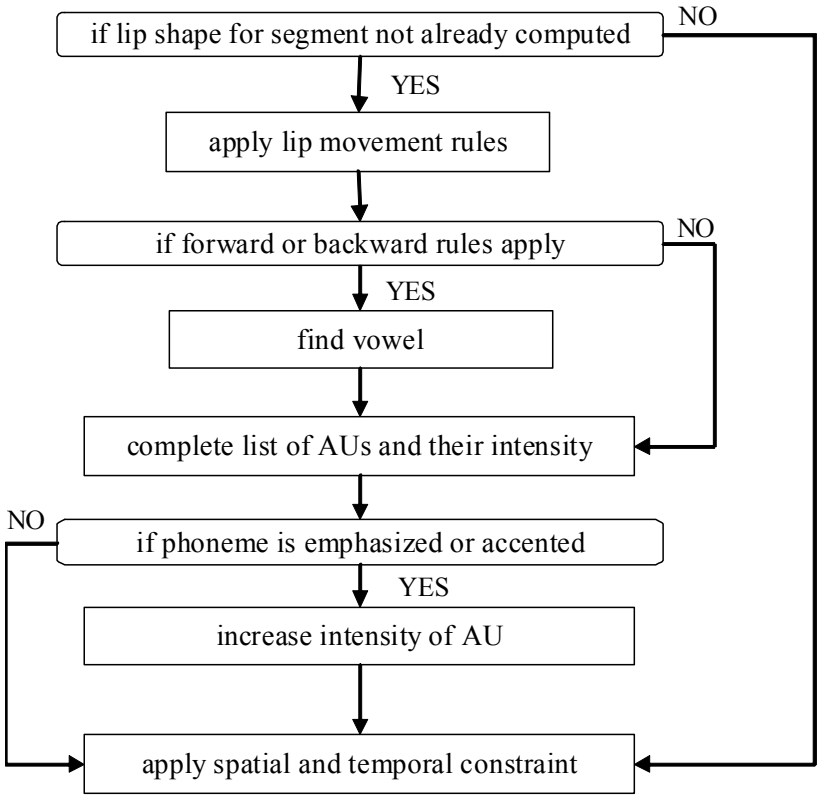


Figure 109: Lip shape computation and coarticulation algorithm (Pelachaud, Badler and Steedman, 1996).

Massaro's (1998:376-386) approach used the dominance and blending functions. He took a phoneme and its timing information to produce key-frames at specific intervals. Each speech segment has a varying degree of dominance over articulators, which is calculated by a function for each articulator-phoneme combination. Using this function, it is possible to accurately determine the position of each articulator at any given point in time. The dominance falls accordingly with the time distance from the centre. The weighted average of all dominances acting within a given time frame determines the lip and tongue position.

This method is still considered superior to the other existing ones (Albrecht, Haber and Seidel, 2002). Albrecht, Haber and Seidel listed the advantages of Massaro's algorithm: low memory usage, no neural network training, convincing results and fast animation. They themselves used Massaro's

algorithm and improved on it by adopting the muscle-based facial animation model described by Kahler, Haber and Seidel (2001). They also restricted the influence of a segment to only seven preceding or following segments, in order to reduce computational overhead.

Ezzat, Geiger and Poggio (2002) included the solution to the coarticulation problem based on artificial intelligence principles (using the gradient descending learning procedure). While the actual technique is similar to Cohen and Massaro's (1998:376-386), its main benefit is in that it does not need human intervention.

9.4 Conclusion

It is certain that speech communication is at the very least bi-modal, as humans perceive talking faces both visually and audibly. If either of the two aspects lacks in realism, the animation attempt will fail. Synthetic speech is out of the scope of this text, as the emphasis is primarily on the facial animation.

Lip synchronisation can be either manual or automated. Manual lip synchronisation is simple but laborious, and the success of the result depends to a great extent on the artistic skills of the animator. Automated lip synchronisation is far more complicated, but less dependent on human intervention and ability. According to the data acquisition method, lip synchronisation can be divided into three areas: text-, speech- and image-driven synchronisation. The speech- and image-driven approaches both eventually rely on a text-driven method. Both these approaches have their loyal followers and it is unclear which approach produces better results. Intuitively, using both approaches independently would enhance the recognition by having two points of reference.

PART III

Towards a Northern Sotho talking head

Chapter 10 First prototype of a Northern Sotho talking head[2]

10.1 Introduction

Northern Sotho, one of the eleven official languages of South Africa, belongs to the so-called Bantu language family and has approximately 4.1 million mother-tongue speakers. It is considered a resource-scarce (in terms of language resources, technological infra-structure and funding), lesser-studied language of the world. In particular, the authors are not aware of any published research on speech animation/visible speech synthesis for Northern Sotho or any of the other indigenous official languages of South Africa.

As a first step towards the development of a Northern Sotho talking head, the following general question is posed: Are specific facial animation tools – those that have been designed and used for the English language – appropriate for Northern Sotho speech animation? More specifically, what can be achieved with commercially available animation products for English? What lessons are to be learnt from this experiment in terms of the development of a Northern Sotho talking head in the most parsimonious of ways?

To address this general question we proceeded as follows: As our source, we recorded a video of a Northern Sotho male mother-tongue speaker, uttering one sentence in his native language. To make the experiment meaningful and challenging, the sentence contained a selection of non-English phonemes. The sound system of Northern Sotho is characterised by a variety of complex consonants. These include a number of homorganic and heterorganic affricates as well as labio-palatal fricatives. The Northern Sotho sentence that is featured in this thesis also includes a lateral plosive. Articulations such as these pose specific animation challenges in terms of facial expressions and the appropriate synchronization of lip, tongue and other movements.

We used lip synchronization techniques described by established artists (Fleming and Dobbs, 1999), first approximating the lip synchronization in accordance with rules for English, and then applying a variety of modifications to suit Northern Sotho. As our principal modelling and animation tool, we used 3D Studio Max (Autodesk, 2007), a software system of choice among animators in the

---

[2] This chapter is based on an earlier work: Radovan, M., Pretorius, L. and Kotze, A.E. (2007). Towards a Northern Sotho talking head. *In: Proceedings of AFRIGRAPH 2007*, Grahamstown, South Africa, October 29-31, 2007. ACM Press, pp 125-133.

industry. As an additional aid, we used an evaluation copy of Facial Studio (Di-o-Matic, 2000), a plug-in dedicated to rapid development of facial animation.

This chapter reports on the process followed, the first results obtained, and insights – both linguistically and computationally – acquired. In our opinion, it has been successfully demonstrated that a variety of non-English (Northern Sotho) phonemes can indeed be modelled by tools developed for English by manually combining multiple different English phonemes and using some direct access to facial muscles and their actions. The chapter is concluded with a brief discussion of possible future work.

10.2 Animation context

Building a talking head basically requires a facial model with appropriate animation capabilities such as a moving jaw and deformable mouth, including the tongue and teeth, the speech that the talking head should produce, the synchronization of the visual (visemes, visyllables) and the auditory (phonemes, syllables) aspects of speech, and the so-called coarticulation model that approximates the ways that neighbouring phonemes/visemes influence one another.

Since the focus is on the speech animation of Northern Sotho, we briefly outline the different directions in speech production and in modelling speech coarticulation. In terms of speech production a distinction is made between real recorded human speech and synthetic speech produced by speech synthesis systems such as Festival, Lucent TTS, Euler/Mbrola, and others (King and Parent, 2005; Massaro et al., 2005; Ouni, Cohen and Massaro, 2005). Concerning coarticulation, a distinction can be made between physics-based approaches where mouth movement is governed by facial muscle simulation; rule-based approaches that attempt to generate speech animation using rules or so-called dominance functions; data-driven methods based on pre-recorded real human motion and audio data that may be searched and from which new audiovisual (phoneme-viseme) sequences may be constructed; and compact statistical models obtained from large data sets by means of machine learning techniques (Deng et al., 2006; Yong et al., 2005).

Other tools of relevance for automated lip synchronization systems in the industry are the widely known product Voice-O-Matic (Di-o-Matic, 2000), a plug-in to 3D Studio Max. The user first assigns the phonemes to visemes and selects an appropriate audio track, after which the software analyses the chosen track and creates an animation. The software manufacturer claims that it will work with almost any language using audio only as a source. It also offers an optional text input, to enhance

analysis (this feature can only be used with the English language). Another similar product is Ventriloquist, originally developed by LIPSinc and currently marketed by Digimation (1999). It utilises LIPSinc's voice analysis system, voiceDSP, which uses digital signal processing technology to analyse the voice and output the corresponding 3D facial coordinates. Its support for languages other than English could not be established from available documentation. A third technology in the form of Magpie Pro (1997) concludes our brief discussion of automated lip synchronization systems. A less-automated system than its sister products, Magpie Pro's speech recognition module requires training of the user as it insists upon the mapping of sample waveforms to their correlating phonemes.

The main tool used to produce our animation is 3D Studio Max. It is a complex, sophisticated software program with a multitude of features, and it is used by many 3D graphics and animation professionals. Its highly flexible and extensible architecture permits third-party applications to easily interface with the core system, further enriching its already extensive set of features. One such plug-in is Facial Studio, which we utilised to produce our head. It builds on 3D Studio Max's modelling engine by producing generic polygonal models of various faces. Another important feature is the animation subsystem. 3D Studio Max requires the setting of key-frames, while automatically taking care of interpolation between the key-frames, synchronizing both animation and the speech. For our model, this could be done in real-time, which is a great help when previewing work and correcting errors. Finally, 3D Studio Max allows for the exporting of the complete animation to a video file, which is then playable on any computer with no extra hardware requirements.

Our speech, one sentence of Northern Sotho, is video-recorded. This recording forms the basis of the manual construction of our phoneme-to-viseme mapping using Magpie. Coarticulation (key frames with linear interpolation) is taken care of by the 3D Studio Max toolkit, as indicated above.

10.3 Linguistic context

The linguistic issues have been prepared by Prof AE Kotzé. This paragraph is the summary of the original chapter published in Radovan, Pretorius and Kotze (2007).

10.3.1 The test sentence

The formulation of the test sentence for the Northern Sotho talking head centred on the challenge to test the performance of 3D Studio Max in the animation of non-English phonemes, where a phoneme is thought of as the smallest contrastive unit in the sound system of a language (SIL International, 2004). To this end the following sentence, which contains as many non-English phonemes as

possible, but which is not unnaturally long, was formulated and subsequently recorded while being spoken by a Northern Sotho adult male:

Diphahla tše mokgalabje o di utswitšego, o di beile ka tlase ga tafola
'The goods that the old man had stolen, he put underneath the table'

The phonetic transcription of the sentence is

[ʆipʰała t͡ʃɛ mok͡xʰalaβ͡ʒɛ o ʆi ut͡switʃ͡ɛɣo o ʆi βɛilɛ ka t͡ɬase ɣa tafola].

Textbook descriptions of Northern Sotho suggest that voiceless stops of the standard variety are ejected (produced on a flow of air by an upward movement of the larynx), but in practice it is found that many speakers do not eject. This was also the case with our informant.

10.3.2 The animation challenge

Speech animation mainly concerns articulatory actions that are observable, such as jaw, mouth and lip movements and/or adjustments. Tongue positioning, adjustment and transition play a major role during speech production but, for the most part, these articulatory movements of the tongue or specific parts of the tongue are not visible during speech production. The actions of the tongue may be partially visible from specific angles when the lips are apart and, then, primarily if the articulation is anterior. Velar, pharyngeal and glottal articulations and adjustments, which are regularly executed in the majority of languages, cannot be observed at all. For this reason convincing speech animation relies heavily on the correct animation of facial movements which involve the actions of the lips and the jaw.

The animation cannot succeed if it does not depict the facial articulatory movements convincingly and it also requires that the visual part of speech is accurate and appropriately synchronised with its auditory correlate. Animation software typically provides the animator with a set of visemes. A viseme is a facial expression that best depicts any particular phoneme and hence they are 'the visual manifestation of phonemes' (Ezzat and Poggio, 1999).

When speech-synchronised animation software such as Facial Studio is tested in respect of its capabilities towards a language that falls outside its intended scope, the following questions come to mind:

- Will the visemes available to 3D Studio Max via Facial Studio account for Northern Sotho phonemes that do not occur in English?
- How adaptable is Facial Studio in terms of dealing with non-English phonemes?

A Northern Sotho language specialist would entertain the following curiosities about the task at hand from an articulatory phonetics point of view.

10.3.2.1 Timing issues

Different consonant types typically have different durations in English (Kent and Read, 2002). For instance, stops have the shortest duration, followed by affricatives and then fricatives. Their data also show that the presence or absence of voicing has durational implications in stops and fricatives. They furthermore include duration values in respect of English sonorants.

The sound system of Northern Sotho includes a number of affricatives and fricatives that are characterised by what Ladefoged and Maddieson (1996) refer to as 'multiple articulatory gestures' and Van Wyk (1977) as 'double articulations' ('dubbelartikulasies'). These consonants all involve the lips for the articulation of the first segment of the phoneme and either an alveolar or post-alveolar articulation for the second. Examples are

[p͡ʃ] Voiceless labio-palatal affricative as in *-hlapša* 'is washed';

[f͡s] Voiceless labio-alveolar fricative as in *lefsifsi* 'darkness';

[β͡ʒ] Voiced labio-palatal fricative as in *bjalwa* 'beer'.

Although one would expect complex sounds to have a longer duration than sounds that are simple, for instance a voiceless homorganic affricative versus a voiceless fricative with just one source of frication, this is not the case (Kent and Read, 2002). It would therefore be incorrect to assume that articulatory complexity necessarily has a durational implication. The Northern Sotho consonants with double articulations referred to above are, however, exceptions in this regard. Ladefoged and Maddieson (1996) argue that these sounds constitute phonetic sequences as opposed to simultaneous articulations. According to them these sounds may be seen as single entities phonologically but not phonetically. Their spectrograms of *fs* and *fš* and *pš* (Ladefoged and Maddieson 1996) clearly reveal that the total articulatory duration of Northern Sotho double consonants can be expected to exceed the accepted or known durations of consonants of which there is better evidence that the temporal centre of its two segments is aligned. The question arises as to whether the (viseme) coarticulation

capabilities of the Facial Studio would be able to provide a convincing animation of the word *mokgalabje* which contains the complex fricative *bj* [β͡ʒ]. Furthermore, given the fact that *bj* is a phonetic sequence of [β] followed by [ʒ], rather than simultaneous bilabial and palatal segments, will 3D Studio Max and Facial Studio be able to animate a smooth transition from [β] to [ʒ]?

10.3.2.2 Consonants in the data

An animation would fail if a phoneme requires a viseme which is not provided by the software that is used. Therefore, the animation of the following Northern Sotho consonants that do not occur in English justifies special attention:

- *d* [ɹ], a voiced retroflexive flap: When the tongue is curled backwards prior to it rapidly uncurling and the apex striking the alveolar ridge, the lower jaw is possibly drawn closer to the upper jaw than in the case of an alveolar plosive /d/. This may have implications for a suitable facial expression to represent the Northern Sotho *d* in the recorded sentence.
- *kg* [k͡xʰ], an aspirated velar affricative: One could expect the viseme for velar consonants to accommodate *kg*.
- *bj* [β͡ʒ], a voiced labio-palatal fricative: As has been pointed out previously, the duration of this sound is exceptional. The segment *b* is a fricative and not a plosive like English /*b*/ but this is not expected to complicate successful animation as much as, possibly, the durational factor could. With reference to Northern Sotho *pš* Ladefoged and Maddieson (1996) argue that the articulation of its segments can be equated to the sequence /*pt*/ in English 'caption' because of the fact that there is little evidence of segment overlap in Northern Sotho *pš*. Since the sequence labial followed by palatal does exist in English, Facial Studio should facilitate the successful animation of the labial+palatal sequence in Northern Sotho *bj*.
- *b* [β], a voiced bilabial fricative: As mentioned before, it is not expected that the feature plosive versus fricative at the same place of articulation would be problematic. Although Northern Sotho fricatives are most probably also much longer in duration than stops (this has not been established) *b* is voiced and therefore probably has a duration which falls within the established duration of at least voiceless stops.

- *hl* [ɬ], a voiceless alveolar-lateral fricative: This sound does not occur in English, except in an adopted place name like *Llanelli*. The laterality being an intra-oral articulation, it is to be expected that the viseme for alveolar consonants would be able to account for *hl*.
- g [ɣ], a voiced velar fricative: It is expected that the 3D Studio Max viseme for English velar phonemes will be adequate to animate Northern Sotho *g*.
- *tl* [t͡ɬ], a voiceless alveolar-lateral plosive: It is expected that a viseme designed to account for alveolar phonemes would serve the animation requirements associated with *tl* as the lateral release occurs intra-orally.

10.3.2.3 Labialization of consonant phonemes in Northern Sotho

The widespread effect of assimilation (in the form of labialization) in the environment preceding back vowels and the semi vowel /*w*/ is also a challenge. In as early as 1929 Tucker noticed that the back vowels of, amongst others, Northern Sotho are characterised by rather prominent lip-rounding. 'This phenomenon is so strong that the native invariably rounds his lips ... during the articulation of any foregoing consonants in the same syllable' (Tucker 1929). Roux (1979) shows that the extent of the lip-rounding in Southern Sotho, another member of the Sotho language group and with a similar vowel system to that of Northern Sotho, is actually more in respect of the semi vowel /*w*/ than for the back vowel [ɔ], for instance. This kind of labialization has particular implications for the consonants *ts* and *g* in the chosen Northern Sotho sentence. If indeed the recording of the Northern Sotho confirms that these phonemes are produced with lip-rounding, would Facial Studio be able to supply in the need for visemes with an unusual lip configuration: rounded lips instead of perhaps neutral lips?

The abovementioned 'anticipatory coarticulation' (Ladefoged, 1975) is not brought about by back vowels and /*w*/ alone, but also by the front vowels of Northern Sotho. Scholars often overlook the fact that, in anticipation of the production of front vowels, the lips are spread during the articulation of a consonant preceding a front vowel or the semi vowel /*y*/ in the Bantu languages. Hence, it is to be expected that consonants followed by front vowels may also present the animator with visemes that have unsuitable lip configurations.

10.3.2.4 Vowels in the data

The major challenge in respect of the animation of Northern Sotho sounds was not foreseen to include any vowels, as all the vowel phonemes of Northern Sotho have English counterparts that are, to a

considerable degree, perceptually similar. However, it had been noted by Tucker (1929), for instance, that lip-rounding in the case of the back vowels of Northern Sotho is ‘very pronounced’. Six of the seven vowel phonemes of Northern Sotho have been included in the selected sentence. These vowels can be said to resemble the following English vowels:

[i] as in English *’heed’;

[e] as in English *’hid’;

[ɛ] as in English *’head’;

[a] as in English ‘car’;

[o] as in English *’hood’;

[u] as in English ‘who’d’ (Examples with asterisks taken from Ladefoged (1975)).

10.4 Preparation

The principal environment for the exercise was 3D Studio Max (www.autodesk.com). Additional tools used were an evaluation copy of Di-O-Matic’s plug-in Facial Studio (Di-o-Matic, 2000) and a shareware utility Magpie (Magpie Pro, 1997). The first challenge was conversion of an analog video tape to a digital format. This was achieved using an old Leadtek VIVO graphic card and its Winfast PVR utility (www.leadtek.com.tw). The clip was encoded using an MPEG2 codec, which was a good balance between the quality and available disk space.

Magpie requires the audio file in WAV format. To extract audio from the digitized clip, I used a freeware utility AoA Audio Extractor (AoA Media, 2008; Figure 110).

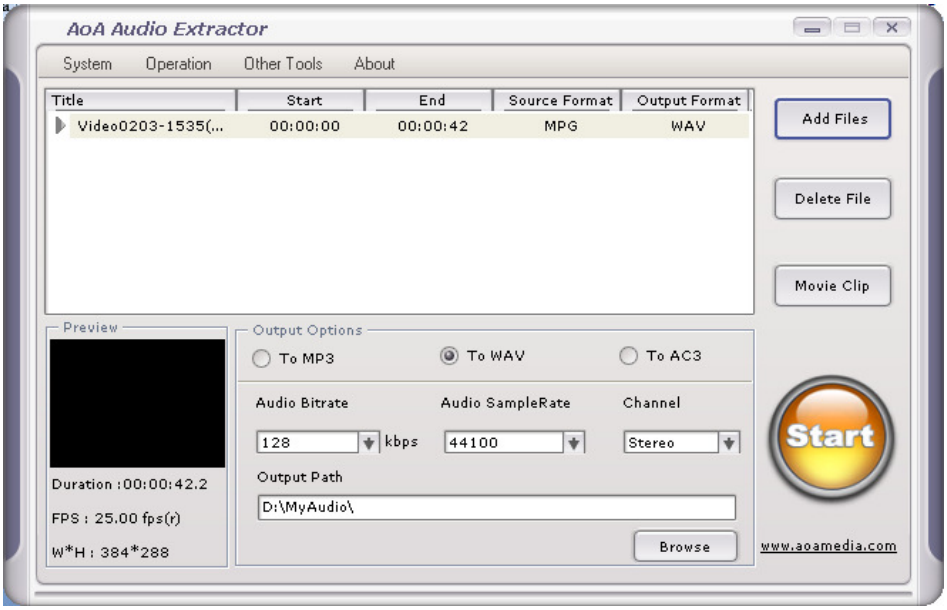


Figure 110: AoA Audio Extractor: user interface (AoA Media, 2008).

The original video recording repeated the sentence four times, and was fairly quiet and with lots of crackling background noise. I used a freeware utility WavePad (NCH Software, 2008; Figure 111) to extract a single sentence, normalize the volume and reduce some background noise.

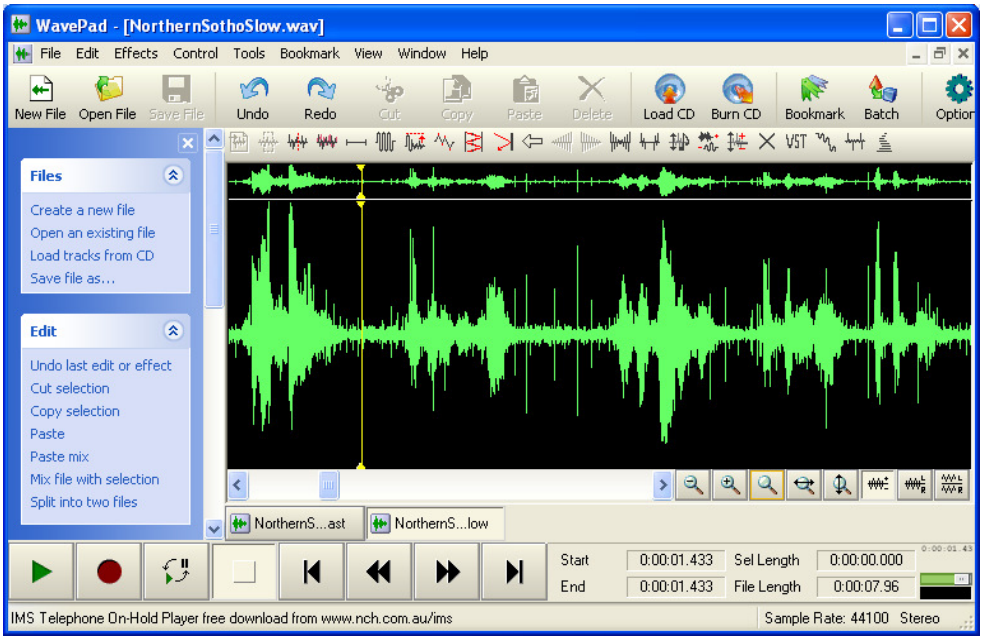


Figure 111: WavePad: user interface (NCH Software, 2008).

The language of graphical voice representation is universal, regardless of the language being English or African: it can be represented in a waveform and its graphic representation is depicted in Figure 112.

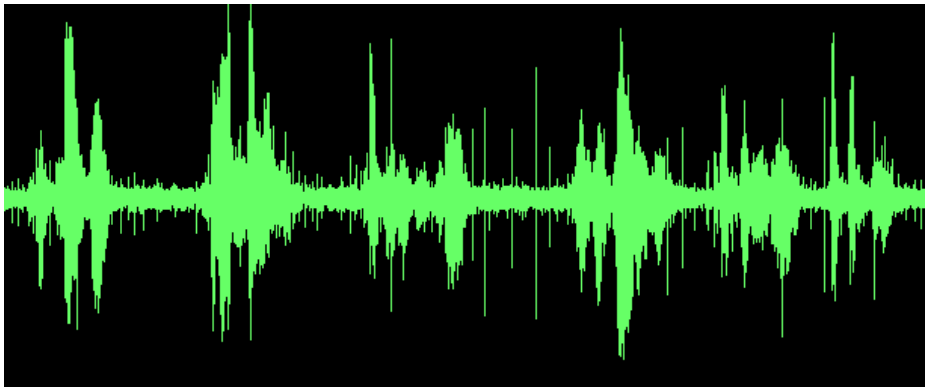


Figure 112: Graphic representation of the Northern Sotho sentence.

10.5 Modelling and lip-synchronisation

The first phase in lip synchronisation consists of breaking down the speech pattern. Although we have the text of the spoken sentence, different people speak with different accents and speeds. Our Northern Sotho sentence is

Diphahla tšê mokgalabje a di utswitšwêgo, o di bêilê ka tlase ga tafola.

***(*The goods which the old man had stolen, he put underneath the table.)**

I loaded the produced file into Magpie (1997), to determine the phonetic representation over time, expressed in frames per second. Magpie GUI is presented in Figure 113. In the top section is the graphic representation of the speech file, divided into frames by a set of vertical lines. In the left frame is a choice of visemes that could be used. It contains 22 distinct visemes by default, which is more than enough for the English language. Each viseme is linked to a bitmap image of a cartoon mouth position, which gives the user a visual cue when playing the sound files and verifying the phoneme allocation. This animation is visible in the top-right corner of the window. Fleming and Dobbs provide a fairly detailed study on distinct visemes in English language. At the end, they settled

on ten distinct visemes if the tongue is ignored and sixteen if the tongue is taken taking into the consideration. The sixteen visemes are shown in Figure 114.

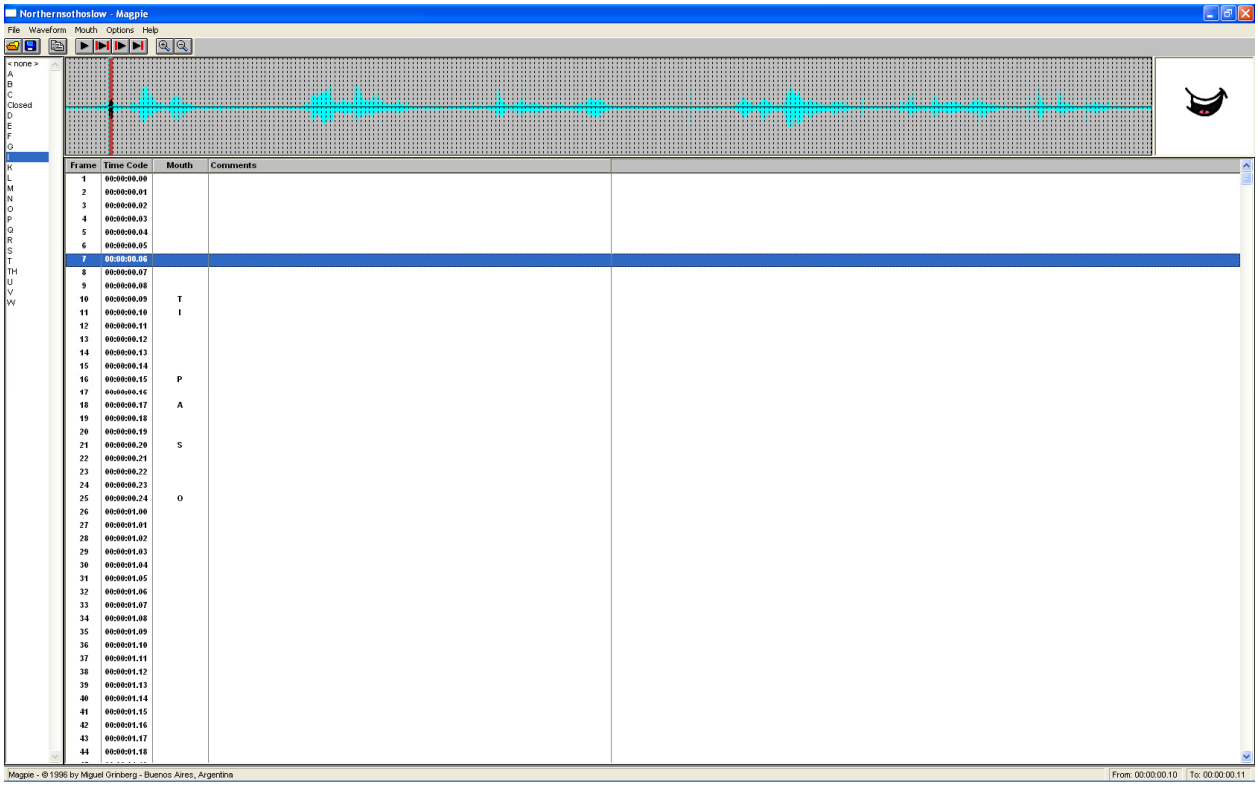


Figure 113: Magpie (1997): user interface.

Furthermore, American English has 41 distinct phonemes, which map to the mentioned 16 visemes. The whole mapping procedure consisted of observing the lip movements of the subject, listening to the speech and attempting to map each phoneme to its nearest counterpart in English. The whole process was more of an art than science.

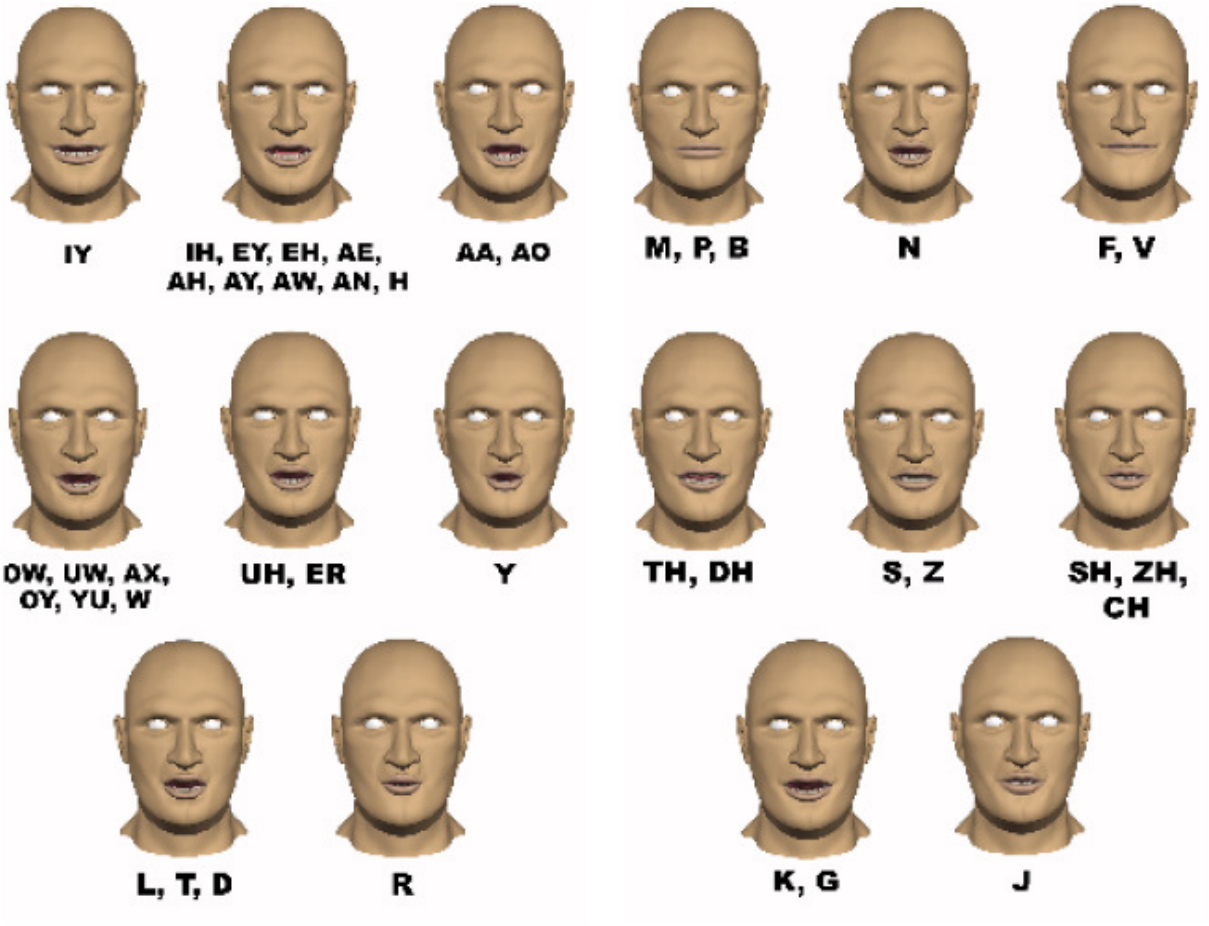


Figure 114: Sixteen distinct visemes in English language.

The next step was to apply the viseme distribution to a 3D model in 3D Studio Max. First, we constructed an African male head. Facial Studio ships with various presets for different races and nationalities, and their characteristic features (Figure 115). For the purpose of this experiment, I have simply chosen an African male using a 100% for each parameter. Facial Studio includes a ready-made generic 3D head model, thus enabling me to produce quite spectacular results with little effort (Figure 116).

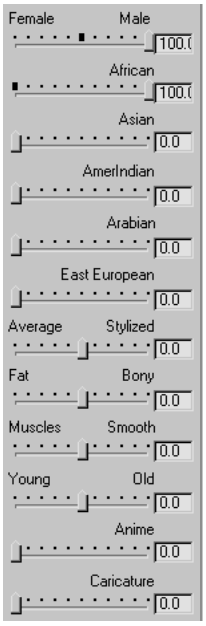


Figure 115: Facial Studio high-level facial features presets.

Next, we loaded the sound and adjusted the frame rate to PAL system (25 frames per second). The animation technique used was a simple linear key-framing with interpolation. Even though more advanced and better animation methods do exist, I had to limit myself to simpler techniques at this stage.

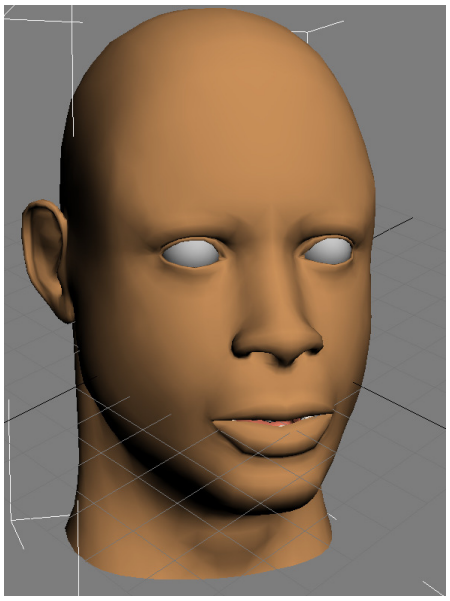


Figure 116: The resultant African male head model.

This was due to my limited knowledge of the modelling software. 3D Studio Max is a complex set of tools, taking graphics professionals years to master.

The key-framing concept consists of moving the timeline thumb to a desired position in time, then morphing the head in such way that it assumes the viseme consistent with the currently uttered phoneme. The system is capable of performing the linear interpolation automatically and in real-time. Due to the repetitive nature of this experiment, I will cover only the first section in detail. Our previous work with Magpie will aid us in selection of key-frames. The first of five parts is the word *Diphahla*. Playing the sound repeatedly, I decided upon D-IY-P-AH-S-AA to be the closest English phonetic representation. Allocated to frames in Magpie, the distribution is shown in Figure 117.

7	00:00:00.06	Closed
8	00:00:00.07	D
9	00:00:00.08	D
10	00:00:00.09	D
11	00:00:00.10	IY
12	00:00:00.11	IY
13	00:00:00.12	IY
14	00:00:00.13	IY
15	00:00:00.14	P
16	00:00:00.15	P
17	00:00:00.16	P
18	00:00:00.17	AH
19	00:00:00.18	AH
20	00:00:00.19	AH
21	00:00:00.20	AH
22	00:00:00.21	S
23	00:00:00.22	S
24	00:00:00.23	S
25	00:00:00.24	AA
26	00:00:01.00	AA
27	00:00:01.01	AA
28	00:00:01.02	AA
29	00:00:01.03	AA
30	00:00:01.04	AA
31	00:00:01.05	Closed

Figure 117: Allocation of phonemes to frames in Magpie.

In Facial Studio, we found a group of preset visemes that did not fully match Fleming and Dobbs' classification (Figure 118 and Figure 114).

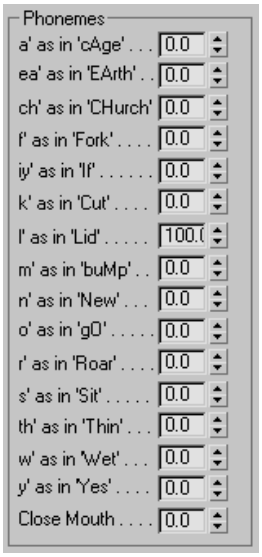


Figure 118: Facial Studio: preset visemes and pronunciation samples.

Since we know that the viseme for D is similar to that of L, we tried to use 100% L at frame number 8 (Figure 119).

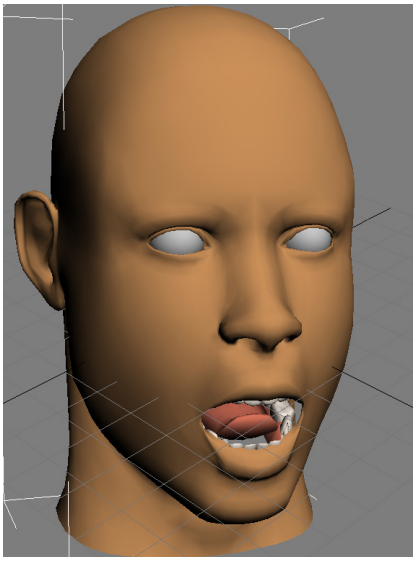


Figure 119: Animated head pronouncing the phoneme ‘L’.

Looking at frame number 8 of the actual video (Figure 120), the facial expression did not match my attempt. This was yet another proof that the process is not automatic, requires a lot of practice, consideration, and reviewing of the source material, including both audio and video.

Figure 120: Frame #8 of the source video: the speaker pronouncing the phoneme ‘D’.

It is becoming clear that our first phoneme is not a clear **D** after all. The subject has tightened his teeth, instead of opening the mouth wide and touching the hard palate with the tongue, as expected from someone who utters **D**. This phoneme is more something in between **D** (**d**i**d**) and J (**j**u**dg**e). Facial Studio does not have **J**, but **CH** (**ch**ur**ch**) was close enough. Combined with a degree of closed mouth parameter, we achieved relatively satisfactory result (Figure 121).

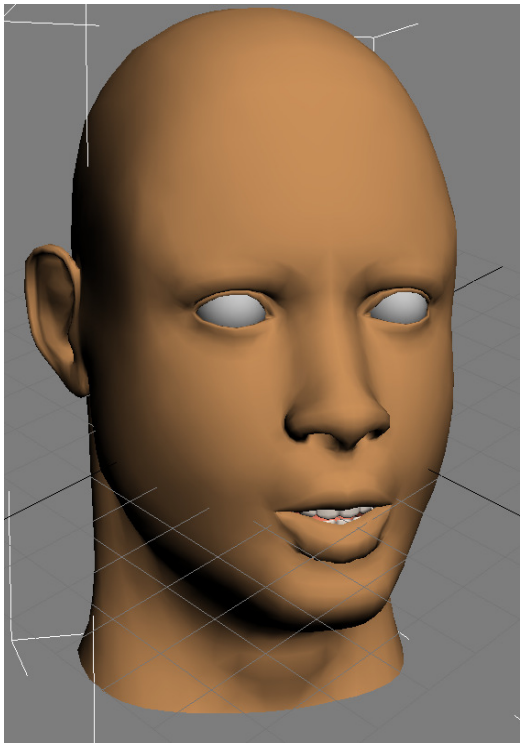


Figure 121: Frame #8 of the animation: the model ‘pronouncing’ the phoneme ‘D’.

IY (b**ea**t) follows at frame number 11. Again, a 100% IY looks unnatural (Figure 122), as it is far too wide.

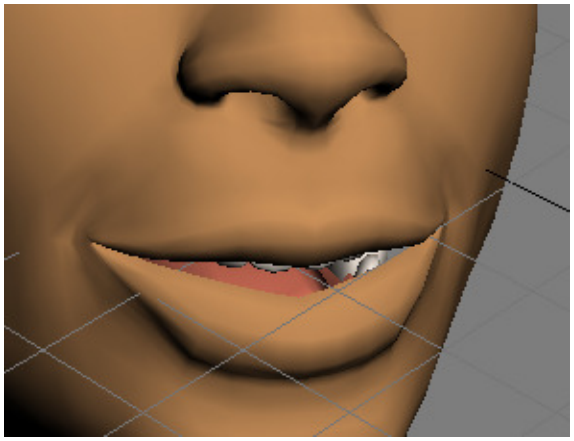


Figure 122: Frame #11 of the animation: first attempt at the viseme for IY (beat).

Watching the video several more times, I noticed there was hardly any movement resulting from the transition **D** to **IY**. The subject has pronounced **D** with full anticipation of **IY** that follows. The two visemes are nearly identical; the difference in sound is caused purely by the tongue, moving away from the hard palate. Since all this happens behind the closed teeth, we do not need to change the expression.

The next phoneme was **P** (**p**o**p**) at frame 15. There was no **P** in Facial Studio, but we can use **M** (**m**ai**m**) instead, as their visemes are interchangeable. The subject at frame 15 is shown in Figure 123.

Figure 123: Frame #15 of the source video: the speaker pronouncing the phoneme 'P'.

100% M was not satisfactory, as the mouth was too wide (Figure 124). One could assume that different people have different speech styles. We could possibly succeed in reproducing the visual speech in a realistic fashion, even though the lip positions do not exactly match the subject on the video. However, it was my aim to produce graphics as close to the video as possible.

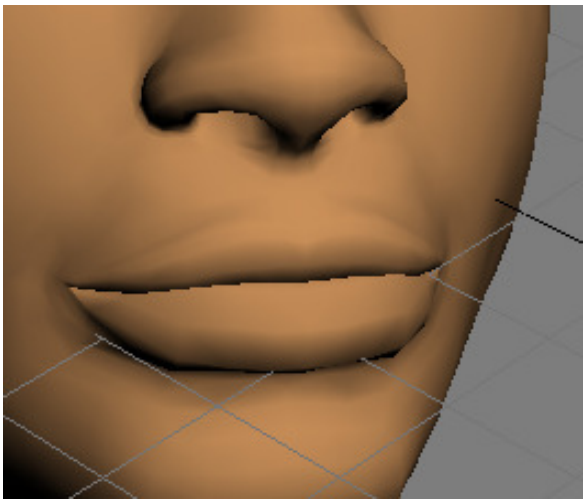


Figure 124: Frame #11 of the animation: first attempt at the viseme for *P* (*pop*).

Experimenting for a while with other visemes, it appeared that adding 70% of TH (**th**in) produced satisfactory result, as the expression was much more similar to the subject's (Figure 125).

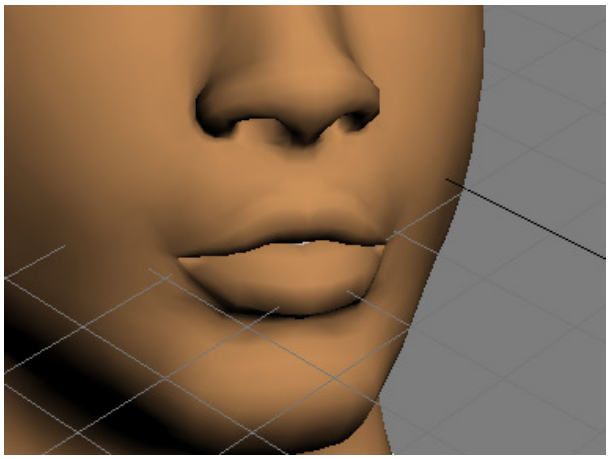


Figure 125: Frame #11 of the animation: a corrected *P* (*pop*).

AH (b**u**t) was fairly easy, as 70% of A in Facial Studio looks similar to the subject's expression. This is shown in Figure 126.

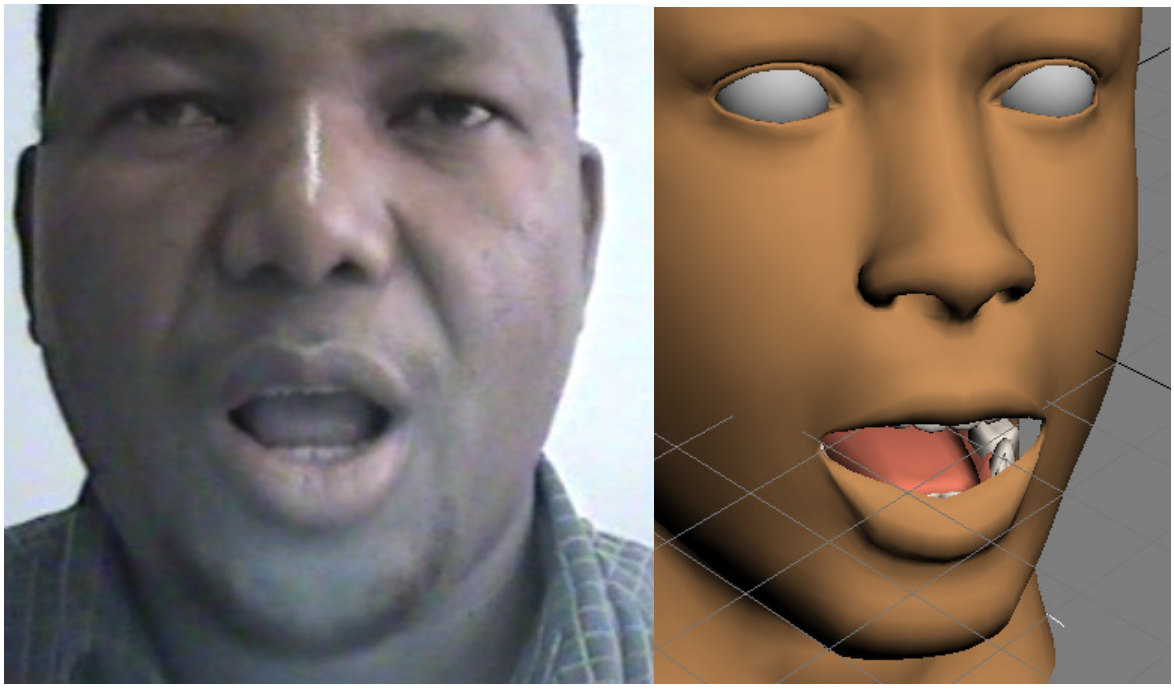


Figure 126: Speaker and the animation pronouncing the phoneme *AH* (b*u*t).

S (sass) was almost silent; the mouth made a direct transition to **AA** (h**o**t), which is equivalent to **O** (g**o**) in Facial Studio. The only part that moved was his tongue, which rolled up and backwards. This is unusual for **S**, as we would squeeze the tongue between the hard and the soft palate, then forward, in order to produce the hissing sound of **S**. This movement does not seem to exist in English visemes, so I tried to simulate it directly with the tongue (-20.0 down and 10.0 tip). **O** also needed some attention. The subject did not utter an open mouth **O**, but rather tightened the lips almost into a kiss. Apart from 100% **O**, I simulated the expression by adding 100% Closed, 100% **TH** and 100% **W** (**w**et). The final result is displayed in Figure 127.

This concludes the detailed first part. The other four parts were synchronised in a similar fashion:

- read the phoneme from Magpie;
- locate the corresponding viseme in Facial Studio;
- apply the viseme;
- find the frame in the video;
- compare to the viseme and correct if necessary;

- repeat the above until end of the sentence.

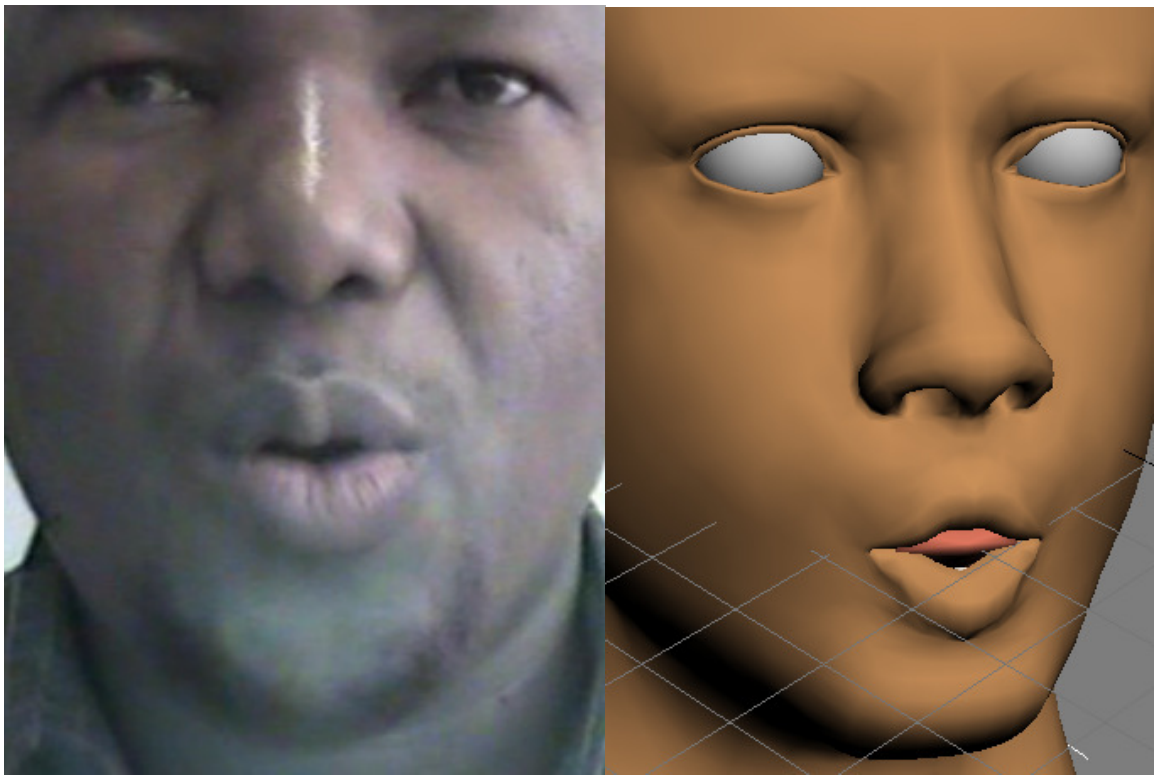


Figure 127: Speaker and the animation pronouncing the phoneme *S*.

10.6 Discussion

From the experiment it is clear that the phoneme-viseme mapping for Northern Sotho differs significantly from that for English.

Specific observations are:

1. The use of the viseme for English S to animate Northern Sotho hl in Diphahla succeeds because the tongue movement for English s is sufficiently similar to that of hl, notwithstanding the mentioned difference. The differences in the actual actions of the tongue in the case of the one sound versus the other are not obvious and therefore not a requirement for the animation to be effective.

2. The unsuitability of the S-viseme in animating Northern Sotho hl is ascribed to the laterality of hl. In order to effect a lateral release as required by hl the sides of the tongue have to be placed against the sides of the upper teeth ridges. The impression may be created that the front of the tongue moves backwards because, immediately prior, the tongue body was low when Northern Sotho a was articulated. Obviously an upwards movement of the tongue from this low position will also seem to involve a degree of retraction. Nonetheless, tongue options had to be manipulated, as described earlier, in order to achieve a production of hl that is convincing.

3. The viseme for L was not suitable as a stand-in for Northern Sotho d, probably because the latter has a retroflexive articulation. No published comparative phonetic data on the position of the lower jaw in retroflexive flapped d versus alveolar plosive d was traced and hence the assumption that the jaw position of the one would be different as compared of the other is just an assumption.

4. Reference was made to the fact that the transition from D to IY in Diphahla was not accompanied by any movement because the subject pronounced D while already anticipating IY is confirmation for the nature of anticipatory coarticulation that was referred to earlier. This obviously has implications for all Northern Sotho consonants. Their animation may require special attention when compared to English same-viseme counterparts.

5. The lip-rounding of the Northern Sotho phoneme /a/ as pronounced by the Northern Sotho speaker was so severe that the viseme for O had to be customised. From a Northern Sotho perspective this is not regarded as rounding but rather an incidental idiosyncrasy. Only back vowels and /w/ are produced with lip-rounding in Northern Sotho.

6. With respect to the modelling of rapid speech versus slow, deliberate speech the most important difference may be found in the phoneme reduction techniques. In fast speech, some of the phonemes are lost due to the limitation of the video to 25 frames per second. Also, the coarticulation effect seems more pronounced, as the speaker does not have enough time to adjust the lips perfectly for every phoneme.

Issues that require further investigation are the animation of vowels, *bj*, the labialized consonants *ts* and *g* in *utswitšego*, and timing.

General observations are:

We explored one of the many ways to produce a first prototype of a Northern Sotho talking head and demonstrated that it could be done with 3D Studio Max. In order to streamline the process using this tool, we could investigate the exact muscle movements for Northern Sotho phonemes and create a custom Northern Sotho set of phonemes-visemes, which would not require the blending of existing English oriented visemes. The software provides the facility to achieve this. There is also a wide variety of other tools and toolkits available that need to be investigated if and when necessary.

In this first experiment we used video-recorded real speech. The decision as to whether real or synthetic speech should eventually be used often, among others, depends on the application. In cases where a prescribed text is provided, real speech may be preferred. For autonomous conversational agents where the content is not predetermined, synthesised speech may be more appropriate. An investigation into the use of and trade-offs between using real and/or synthetic speech also forms part of future work.

Phoneme-viseme correspondence and coarticulation remains a challenge in speech-synthesized facial animation (Deng et al. 2006; Leszczynski and Skarbek 2005; Massaro et al. 2006; Sifakis et al. 2006), particularly for languages that are significantly different from English (King and Parent 2005; Pei and Zha 2006). In our experiment we associated various phonemes of Northern Sotho that do not also occur in English with blends of visemes for English – the process was not scientifically sophisticated, but rather subjective, manual, time-consuming and trial-and-error driven. We consider the insights and understanding acquired both linguistically and computational, most valuable and are now in a position to investigate the development of a phoneme-viseme correspondence set specifically for Northern Sotho. Towards this end the use of muscle or physics-base approaches, rule-based methods including dominance functions, as well as data-driven techniques and machine learning techniques will be systematically explored.

Chapter 11 Conclusion and future work

This thesis covers the field of facial animation from its inception to date (2008). Many techniques have been mentioned for reference only, as they are no longer used. However, each of the main approaches, namely geometry manipulation, image manipulation and performance-driven animation have their current representative state-of-the-art techniques and loyal support in the scientific community.

The surviving representative of geometry manipulation is the muscle-based approach. It is the method of choice for most of the interactive applications, requiring a real-time rendering solution. The main streams competing for supremacy of the muscle-based approach are mass-spring and finite-element methods. The superiority of one over the other has not yet been established. Improvements and future work in this domain consist mainly of increased automation of the traditionally manual tasks and general reduction of human intervention (Kahler et al., 2002), and modelling of the eyes, teeth and tongue, in particular (Ko et al., 2003). Proposed future improvements of the most advanced finite-element method in facial animation (Sifakis et al., 2005) consist of inclusion of bone and flesh structure, advancement of realism of the facial expressions and deriving more accurate lip deformation data.

The state-of-the-art of image-based manipulation is currently the blendshape modelling method. This approach was reportedly often used for special effects in recent movies, such as Stuart Little, Star Wars and The Lord of the Rings (Lewis et al., 2005). As with geometry methods, the greatest issue lies in the amount of human intervention required during the process (Joshi et al., 2003). Further research is required to automate the blendshape approach, as it currently requires extensive work in the setup stage (Deng at al., 2006). Most of the work in the academic community is constrained by the size of the blendshape datasets, probably due to financial constraints. Datasets produced for the motion picture special effects are much larger, sometimes consisting of thousands of blendshapes (Lewis et al., 2005), as opposed to 46 used by Deng et al. (2006). Therefore, there is a strong need for testing of the academic results on an industry strength dataset, which has not happened, most likely due to the intellectual property issues.

Performance-driven animation is also here to stay – nothing is more natural than actual expressions created by real people. If such expressions are accurately captured and reproduced, the results are quite astonishing. Clearly, the data acquisition methods of performance-driven animation warrant

separate discussion as they have their own specific research problems (Chuang and Bregler, 2002; Chai, Xiao and Hodgins, 2003). Errors accumulate over time due to the use of optical flow principles (Borshukov et al., 2003) and need to be corrected, albeit manually. The improvement of the optical flow algorithm somewhat alleviated the manual correction (Reutersward et al., 2005). Extrapolation methods may result in unrealistic appearance to the viewer. In the case of occlusion it has been suggested that the occluded part could be deduced/interpolated from the neighbouring frames. However, Zhang et al. (2004) report that their algorithm does not work well for extrapolated faces. Another area for future work in the improvement of performance-driven techniques is machine learning, as listed by, for example, Hertzmann (2003).

As the software and hardware technology for facial animation and modelling mature and become more accessible, diverse applications of growing complexity may be explored. While the entertainment industry and medical scientific visualisation will continue to generate increasingly sophisticated applications, we expect significant developments in the broad field of human-computer interaction, web-based (Kshirsagar et al., 2003) and otherwise. For example, active and emerging fields there relate to affective computing and embodied conversational agents that address, among others, the problem of ‘attempting to enhance interactions between humans and technology through the development of artificial systems responsive to the moods and emotions of a human user’ (Anderson and McOwan, 2006). These interactions increasingly take place between humans and humanoid animated conversational agents. It has been shown conclusively that such interaction is enhanced by animated agents that exhibit socio-emotive abilities and/or socio-cultural associations, and are lifelike (for example, Krenn et al., 2004). In achieving these goals the modelling and animation of the face, facial expressions, voice, visual style and resulting personality of such an agent play a vital role (Gulz and Haake, 2006) and offers ample opportunity for multi-disciplinary research (Ochs et al., 2005; Pelachaud, 2005), also within the Southern African multi-lingual multi-cultural context.

We conducted an experiment where we combined principles of the facial animation and linguistic aspects of the Northern Sotho language, an African language talking head. Finally, the exploratory work towards a talking head for Northern Sotho opened up various avenues for further research, as discussed in the concluding section of Chapter 10.

References

[1] Albrecht, I., Haber, J. and Seidel, H.P. (2002). Speech synchronisation for physics-based facial animation. *In: Proceedings of WSCG 2002*, Plzen-Bory, Czech Republic, February 4-8 2002. University of West Bohemia, pp 9-16.

[2] Allan, J. B., Wyvill, B. and Witten, I. H. (1989). A methodology for direct manipulation of polygon meshes. *In: Proceedings of CG International '89*, Leeds, UK, June 27-30 1989. Springer, pp 451-469.

[3] Anderson, K. and McOwan, P.W. (2006). A real-time automated system for the recognition of human facial expressions. *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, **36**(1), pp 96-105.

[4] Angel, E. (2003). *Interactive computer graphics a top-down approach using OpenGL*. 3rd ed. Addison Wesley.

[5] AoA Media. (2008) AoA Audio Extractor. [online] Available From: http://www.aoamedia.com/audioextractor.htm. [Accessed 22 Nov 2008].

[6] Autodesk/Discreet. (2007). *Products*. [online]. Available from: http://usa.autodesk.com/adsk/servlet/index?id=331041&siteID=123112 [Accessed 25 Oct 2008].

[7] Badler, N.I., Phillips, C. and Webber, B. (1993). *Simulating humans: computer graphics animation and control*. Oxford University Press.

[8] Basu, S., Oliver, N. and Pentland, A. (2003) 3D modelling and tracking of human lip motions. *In: Proceedings of the Sixth International Conference on Computer Vision (ICCV'03)*, Nice, France, October 14-17 2003. IEEE Computer Society, pp 337-343.

[9] Baumgart, B. G. (1975). *Winged-edge polyhedron representation*. [online]. National Computer Conference 1975. Stanford University. Available from: http://www.baumgart.org/winged-edge/winged-edge.html. [Accessed 25 October 2008].

[10] Beier, T. and Neely, S. (1992). Feature-based image metamorphosis. *In*: *Proceedings of SIGGRAPH 1992*, Chicago, USA, July 27-31 1992. ACM Press, pp 35-42.

[11] Beskow, J. (1995). Rule-based visual speech synthesis. *In: Proceedings of Eurospeech '95*, Madrid, Spain, 1995. European Speech Communication Association, pp 299-302.

[12] Birn, J. (1996). *TUTORIAL: NURBS head modelling*. [online] Available from: http://www.3drender.com/jbirn/ea/HeadModel.html. [Accessed 25 Oct 2008].

[13] Blanz, V., Basso, C., Poggio, T. and Vetter, T. (2003). Reanimating faces in images and video. *Computer Graphics Forum*, **22**(1), pp 641-650.

[14] Bondy, M.D., Petriu, E. M., Cordea, M. D., Georganas, N. D., Petriu, D. C. and Whalen, T. E. (2001). Model-based face and lip animation for interactive virtual reality Applications. *In: Proceedings of ACM Multimedia 2001*, Ottawa, Canada, September 30-October 5, 2001. ACM Press, pp 559-563.

[15] Borshukov, G. and Lewis, J.P. (2003). Realistic human face rendering for 'The Matrix Reloaded'. *In: SIGGRAPH 2003 conference on Sketches & Applications*, San Diego, USA, July 27-31 2003. ACM Press.

[16] Borshukov, G., Piponi, D., Larsen, O., Lewis, J.P. and Tempelaar-Lietz, C. (2003). Universal capture – image-based facial animation for 'The Matrix Reloaded'. *In: SIGGRAPH 2003 conference on Sketches & Applications*, San Diego, USA, July 27-31 2003. ACM Press.

[17] Brand, M., (1999). Voice puppetry. *In*: *Proceedings of SIGGRAPH 1999*, Los Angeles, USA, August 8-13 1999. ACM Press, pp. 21-28.

[18] Braude I. (2005). *Smooth 3D surface reconstruction from contours of biological data with MPU implicits*. MSc thesis. Drexel University.

[19] Bregler, C., Covell, M. and Slaney, M. (1997). Video rewrite: driving visual speech with audio. *In: Proceedings of SIGGRAPH 1997*, Los Angeles, USA, August 3-8 1997. ACM Press, pp 353-360.

[20] Breton, G., Bouville, C. and Pele, D. (2001). A 3d facial animation engine for real time applications. *In: Proceedings of WEB3d*, Paderbon, Germany, February 19-22 2001. ACM Press, pp 15–22.

[21] Brotman, J. S., Netravali, A. N. (1988). Motion interpolation by optimal control. *In: Proceedings of SIGGRAPH 1988*, Atlanta, USA, August 1-5 1988. ACM Press, pp 309-315.

[22] Bui, T. D., Heylen, D. and Nijholt, A. (2003). Improvements on a simple muscle-based 3D face for realistic facial expressions. *In: Proceedings of 16th International Conference on Computer Animation and Social Agents (CASA 2003)*, Los Alamos, USA, 7-9 May 2003. IEEE Computer Society, pp 33-40.

[23] Bui, T. D. (2004). *Creating emotions and facial expressions for embodied agents*. PhD Thesis. University of Twente, Netherlands.

[24] Cave, C., Guaitella, I., Bertrand, R., Santi, S., Harlay, F. and Espesser, R. (1996). About the relationship between eyebrow movements and f0 variations. *In: Proceedings of ICSLP '96*, Philadelphia, USA, October 3-6 1996. IEEE Computer Society, pp 2175-2178.

[25] Chadwick, J., Haumann, D. and Parent, R. (1989). Layered construction for deformable animated characters, *Computer Graphics*, **23**(3), pp 234-243.

[26] Chai, J., Xiao, J. and Hodgins, J. (2003). Vision-based control of 3D facial animation. *In: Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, San Diego, USA, July 26 - 27 2003. ACM Press, pp 193-206.

[27] Chan, R.H., Ho, C. W. and Nikolova, M. (2005). Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization, *IEEE Transactions on Image Processing*, **14**(10), pp 1479–1485.

[28] Chandru, V., Mahesh, N., Manivannan, M. and Manohar, S. (2000). Volume sculpting and keyframe animation system. *In: Proceedings of Computer Animation 2000*, Philadelphia, USA, May 3-5 2000, University of Pennsylvania, pp 134.

[29] Chang, E. and Jenkins, O. C. (2008) Sketching articulation and pose for facial animation. *In:* Deng, Z. and Neumann, U., eds. *Data-Driven 3D Facial Animation*. Springer, pp 145-161.

[30] Choe, B., Lee, H. and Ko, H. S. (2001). Performance-driven muscle-based facial animation. *The Journal of Visualization and Computer Animation*, **12**(2), pp 67-79.

[31] Chuang, E. and Bregler, C. (2002). *Performance driven facial animation using blendshape interpolation*. Stanford University. Computer Science Technical Report CS-TR-2002-02.

[32] Chuang, E. and Bregler, C. (2005). Mood swings: expressive speech animation. *ACM Transactions on Graphics*, **24**(2), pp 331-347.

[33] Cohen, M. M. and Massaro, D. W. (1993). Modelling coarticulation in synthetic visual speech. *In*: Magnenat Thalmann, N. and Thalmann, D., eds. *Models and Techniques in Computer Animation*. Springer, pp 139-156.

[34] Comet, M. B. (2001). *Modelling a head with polys in 3D Studio Max.* [online]. Available from: http://www.comet-cartoons.com/3ddocs/headpolymodel/. [Accessed 8 November 2008].

[35] Coquillart, S. (1990). Extended free-form deformation: a sculpturing tool for 3D geometric modelling. *In: Proceedings of SIGGRAPH 1990,* Dallas, USA, August 6-10 1990. ACM Press, pp 187 – 196.

[36] Cosatto, E. and Graf, H.P. (2000). Photo-realistic talking heads from image samples. *IEEE Transactions on Multimedia*, **2**(3), pp 152-163.

[37] Cyberware (1999) *Head & Face color 3D scanner*. [online]. Available from: http://www.cyberware.com/products/psInfo.html. [Accessed 8 November 2008].

[38] Daeman College. (2004). *TMJ research at Daemen*. [online]. Available from: http://www.daemen.edu/offices/grants/Funded%20Projects/TMJ.HTM. [Accessed 8 November 2008].

[39] De Gelder, B. and Vroomen, J. (2000). The perception of emotion by ear and by eye. *Cognition & Emotion*, **14**(3), pp 289-311.

[40] Deng, Z., Chiang, P.Y., Fox, P. and Neumann, U. (2006). Animating blendshape faces by cross-mapping motion capture data. *In: Proceedings of SIGGRAPH 2006 Symposium on*

Interactive 3D Graphics and Games (SI3D), Redwood City, USA, March 14-17 2006. ACM Press, pp 43-48.

[41] Deng, Z., Neumann, U., Lewis, J. P., Kim, T-Y., Bulut, M. and Narayanan, S. (2006). Expressive facial animation synthesis by learning speech coarticulation and expression spaces. *IEEE Transactions on Visualization and Computer Graphics*, **12**(6), pp 1523–1534.

[42] Di-O-Matic. (2000). *Facial Studio*. [online]. Available from: http://www.di-o-matic.com/products/Plugins/FacialStudio/. [Accessed 8 November 2008].

[43] Digimation. (1999). *Ventriloquist.* [online] Available from: http://www.digimation.com/home/?Content=ProductDetail&ProductCode=SD389. [Accessed 8 November 2008].

[44] Doenges P., Lavagetto F., Ostermann J., Pandzic I. S. and Petajan E. (1997). MPEG-4: audio/video and synthetic graphics/audio for mixed media. *Image Communications Journal*, **5**(4), pp 433-464.

[45] Ekman, P., Friesen, W. V. and Hager, J. C. (2002), *Facial action coding system: the manual*. [online] Available from: http://face-and-emotion.com/dataface/facs/manual/TitlePage.html. [Accessed 8 November 2008].

[46] Essa, I. A., Basu, S., Darrell, T. and Pentland, A., (1996). Modeling, tracking and interactive animation of faces and heads using input from video. *In: Proceedings of Computer Animation Conference*, Geneva, Switzerland, June 3-4 1996. IEEE Computer Society, pp 68-79.

[47] Ezzat, T. and Poggio, T. (1998). MikeTalk: A talking facial display based on morphing visemes. *In: Proceedings of Computer Animation Conference*, Philadelphia, USA, June 8-10 1998. IEEE Computer Society, pp 96-102.

[48] Ezzat, T. and Poggio, T. (1999). Visual speech synthesis by morphing visemes. *International Journal of Computer Vision*, **38**(1), pp 45-57.

[49] Ezzat, T., Geiger, G. and Poggio. T. (2002). Trainable videorealistic speech animation. *In: Proceedings of SIGGRAPH 2002*, San Antonio, USA, July 21-26 2002. ACM Press, pp 388-398.

[50] Fish, J. and Belytschko, T. (2007). *A First Course in Finite Elements*. John Wiley & Sons.

[51] Fleming, B. and Dobbs, D. (1999). *Animating facial features and expressions*. Charles River Media.

[52] Forsey, D. R. and Bartels, R. H. (1988). Hierarchical B-spline refinement. *ACM SIGGRAPH Computer Graphics*, **22**(4), pp 205-212.

[53] Fuchs, T., Haber, J. and Seidel, H.P. (2004). MIMIC — a language for specifying facial animations, *In: Proceedings WSCG SHORT Communication*, WSCG'2004, February 2–6, 2004, Plzen, Czech Republic. University of West Bohemia, pp 71-78.

[54] Fuchs, H., Kedem, Z. M. and Uselton, S. P. (1977). Optimal surface reconstruction from planar contours. *ACM SIGGRAPH Computer Graphics*, **11**(2), pp 236-236.

[55] Guenter, B., Grimm, C., Wood, D., Malvar, H. and Pighin, F. (1998). Making faces. *In: Proceedings of SIGGRAPH 1998*, Orlando, USA, July 19-24 1998. ACM Press, pp 55–66.

[56] Guiard-Marigny, T., Tsingos, N., Adjoudani, A., Benoit, C. and Gascuel, M-P. (1996). 3D models of the lips for realistic speech animation. *In: Proceedings of the Computer Animation*, Geneva, Switzerland, June 3-4 1996. IEEE Computer Society, pp 80-89.

[57] Gulz, A. and Haake, M. (2006). Design of animated pedagogical agents – A look at their look. *International Journal of Human-Computer Studies* **64**(4), pp 322-339.

[58] Gutierrez-Osuna*, R.,* Kakumanu, P.K., Esposito, A., Garcia*, O.N.,* Bojorquez, A., Castillo, J.L. and Rudomin, I. (2005). Speech-driven facial animation with realistic dynamics. *IEEE Transactions on Multimedia*, **7**(1), pp 33-42.

[59] Hall, V. (1992). *Speech driven facial animation*. BSc Thesis. Curtin University of Technology.

[60] Heckbert, P. S., and Garland, M. (1997). *Survey of polygonal surface simplification algorithms*. Carnegie Mellon University. CMU-CS-95-194.

[61] Hertzmann, A. (2003). Machine learning for computer graphics: a manifesto and tutorial. *In: Proceedings of the 11th Pacific Conference on Computer Graphics and Applications (PG 2003)*, Canmore, Canada, October 8-10 2003. IEEE Computer Society, pp 22-26.

[62] IPA. (2005). *Alphabet*. [online]. Available from: http://www.arts.gla.ac.uk/IPA/ipachart.html. [Accessed 8 November 2008].

[63] Joshi, P., Tien, W. C., Desbrun, M. and Pighin, F. (2003). Learning controls for blend shape based realistic facial animation. *In: Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, San Diego, USA, July 26 – 27 2003. ACM Press, pp 187-192.

[64] Kahler, K., Haber, J. and Seidel, H.P. (2001). Geometry-based muscle modelling for facial animation. *In: Proceedings of Graphic Interface*, Ottawa, Canada, June 7-9 2001. A K Peters, pp 37-46.

[65] Kahler, K., Haber, J., Yamauchi, H., and Seidel, H. P. (2002). Head shop: generating animated head models with anatomical structure. *In:* Spencer, S. N., editor, *Proceedings of the 2002 ACM SIGGRAPH Symposium on Computer Animation (SCA-02)*, San Antonio, USA, July 21-22 2002. ACM Press, pp 55-64.

[66] Kahler, K. (2003). *A head model with anatomical structure for facial modelling and animation*. PhD Thesis. Universität des Saarlandes.

[67] Kalra, P., Mangili, A., Magnenat Thalmann, N. and Thalmann, D. (1991a). 3D interactive free form deformations for facial expressions, *In: Proceedings of Compugraphics '91*, Sesimbra, Portugal, September 16-20 1991. Tech. Univ., pp 129-141.

[68] Kalra, P., Mangili, A., Magnenat Thalmann, N and Thalmann, D. (1991b). SMILE: a multilayered facial animation system, *In:* Kunii, T. L., editor, *Proceedings of IFIP WG 5.10*, Tokyo, Japan, April 8-12 1991. Springer, pp 189-198.

[69] Kalra, P., Mangili, A., Magnenat Thalmann, N and Thalmann, D. (1992). Simulation of facial muscle action based on rational free form deformations. *Computer Graphics Forum*, **11**(3), pp 59-69.

[70] Kalra, P. and Magnenat - Thalmann, N. (1994). Modelling of vascular expressions in facial animation. *In: Proceedings of Computer Animation*, Geneva, Switzerland, May 25-28 1994. IEEE Computer Society, pp 50 -58.

[71] Kent, R. D. and Read, C. (2002). *Acoustic analysis of speech*. 2nd ed. Thomson Learning.

[72] King, S. A. and Parent, E. (2001). A 3D parametric tongue model for animated speech. *Journal of Visualization and Computer Animation*, **12**(3), pp 107–115.

[73] King, S. A., Parent, E. and Olsafsky, B. L. (2000). An anatomically-based 3D parametric lip model to support facial animation and synchronized speech. *In: Proceedings of Deform*, Geneva, Switzerland, November 29-30 2000. Kluwer Academic Publishers, pp 7-19.

[74] King, S. A. and Parent, R. E. (2005). Creating speech-synchronized animation. *IEEE Transactions on Visualization and Computer Graphics*, **11**(3), pp 341–352.

[75] King, S. A. (2001). *A facial model and animation techniques for animated speech*. PhD Thesis. Ohio State University.

[76] Ko, H.S., Choi, K.J., Choi, M. G., Tak, S., Choe, B. and Song, O.Y. (2003). Research problems for creating digital actors. *In: EUROGRAPHICS 2003 STAR, state of the art reports 24th annual conference of the European Association for Computer Graphics*, Grenada, September 1-6 2003. Universidad de Granada.

[77] Kobbelt, L. P., Botsch, M., Schwanecke, U., and Seidel, H.-P. (2001). Feature sensitive surface extraction from volume data. *In: Proceedings of SIGGRAPH 2001*, Los Angeles, USA, August 12-17 2001. ACM Press, pp 57–66.

[78] Koch, R. M., Gross, M. H., Carls, F. R., Buren, D. F., Fankhauser, G. and Parosh, Y. I. H. (1996). Simulating facial surgery using finite element models. *In: Proceedings of SIGGRAPH 1996*, New Orleans, USA, August 4-9 1996. ACM Press, pp 421-428.

[79] Koch, R.M., Gross, M. H. and Bosshard, A. A. (1998). Emotion editing using finite elements. *Computer Graphics Forum*, **17**(3), pp 295-302.

[80] Kochanek, D. and Bartels, R. (1984). Interpolating splines with local tension, continuity and bias tension. *In: Proceedings of SIGGRAPH 1984*, Minneapolis, USA, July 23-27 1984. ACM Press, pp 33-41.

[81] Konica Minolta. (2007). *For 3D measurement*. [online]. Available from: http://www.konicaminolta.com/sensingusa/products/3d. [Accessed 11 November 2008].

[82] Kouadio, C., Poulin, P. and Lachapelle, P. (1998). Real-time facial animation based upon a bank of 3D facial expressions. *In: Proceedings of Computer Animation*, Philadelphia, USA, June 8-10 1998. IEEE Computer Society, pp 128-136.

[83] Krenn, B, Neumayr, B, Gstrein, M. and Grice, M. (2004). Lifelike agents for the Internet: a cross-cultural case study. *In:* Payr, S. and Trappi, R., eds. *Agent Culture: Human-Agent Interaction in a Multicultural World*. Lawrence Erlbaum Associates, pp 197-229.

[84] Krinidis, S., Buciu, I. and Pitas, I. (2003). Facial expression analysis and synthesis: a survey. *In: Proceedings of the 10th international Conference on Human-Computer Interaction (HCI 2003)*, Heraklion, Greece, June 22-27 2003. Lawrence Erlbaum Associates, pp 1432-1436.

[85] Kshirsagar, S., Garchery, S., Sannier, G. and Magnent-Thalmann, N. (2003). Synthetic faces: analysis and applications. *International Journal of Imaging Systems and Technology* **13**(1), pp 65-73.

[86] Ladefoged, P. (1975). *A course in phonetics*. Harcourt Brace Jovanovich, Inc.

[87] Ladefoged, P. and Maddieson, I. (1996). *The sounds of the world's languages*. Blackwell Publishers.

[88] LeBlanc, A., Kalra, P., Magnenat-Thalmann, N. and Thalmann, D. (1991). Sculpting with the 'ball & mouse' metaphor. *In: Proceedings of Graphic Interface 1991*. Calgary, Canada, June 3-7 1991. AK Peters, pp 152-159.

[89] Lee, S.P., Badler, J.B. and Badler, N.I. (2002). Eyes alive. *In: Proceedings of SIGGRAPH 2002*, San Antonio, USA, July 21-26 2002. ACM Press, pp 637-644.

[90] Li, Q. and Deng, Z. (2007). *Facial motion capture editing by automated orthogonal blendshape construction and weight propagation*. University of Houston, Department of Computer Science, Technical Report Number UH-CS-07-12.

[91] Lee, Y., Terzopoulos and Waters, K. (1995). Realistic modelling for facial animation. *In: Proceedings of SIGGRAPH 1995*, Los Angeles, USA, August 6-11 1995. ACM Press, pp 55-62.

[92] Leszczynski, M. and Skarbek, W. (2005). Viseme classification for talking head application. *In: Proceedings of CAIP 2005*, Versailles, France, September 5-8 2005. Springer, pp 773–780.

[93] Lewis, J.P. (1989). Algorithms for solid noise synthesis. *In: Proceedings of SIGGRAPH 1989*, Boston, USA, July 31 - August 4 1989. ACM Press, pp 263-270.

[94] Lewis, J.P. (1991) Automated lip-sync: Background and techniques. *The Journal of Visualization and Computer Animation*, **2**(4), pp 118–122.

[95] Lewis, J. P., Mooser, J., Deng, Z. and Neumann, U. (2005). Reducing blendshape interference by selected motion attenuation. *In: Proceedings of SIGGRAPH 2005 Symposium on Interactive 3D Graphics and Games (I3D)*, Washington, USA, April 3-6 2005. ACM Press, pp 25–29.

[96] Lorensen, W. E. and Cline, H. E. (1987). Marching cubes: a high resolution 3D surface construction algorithm. *In: Proceedings of SIGGRAPH 1987*, Anaheim, USA, July 27-31 1987. ACM Press, pp 163-169.

[97] Lundeberg, M. and Beskow, J. (1999). Developing a 3D-agent for the AUGUST dialogue system. *In: Proceedings of Audio-Visual Speech Processing (AVSP) 1999*, Santa Cruz, USA, August 7-10 1999. pp 151-154.

[98] Magnenat-Thalmann, N., Primeau, E. and Thalmann, D. (1988). Abstract muscle action procedures for human face animation, *The Visual Computer*, **3**(5), pp 290-297.

[99] Magnenat-Thalmann, N., Minh, H.T., de Angelis, M. and Thalmann, D. (1989). Design, transformation and animation of human faces. *The Visual Computer*, **5**(1&2), pp 32-39.

[100] Magpie (1997). *Magpie Pro*. [online]. Available from: http://www.thirdwishsoftware.com/magpiepro.html, 1999-2007. [Accessed 14 November 2008].

[101] Mani, M. V. and Ostermann, J. (2001). Cloning of MPEG-4 face models. *In: Proceedings of International Workshop on Very Low Bit rate Video Coding (VLBV01).* Athens, Greece, 11-12 October 2001. Springer, pp 206-211.

[102] Massaro, D. W. (1998). *Perceiving talking faces: from speech perception to behavioral principle*. MIT Press.

[103] Massaro, D. W., Ouni, S., Cohen, M.M. and Clark, R. (2005) A multilingual embodied conversational agent. *In: Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, January 3-6 2005, Los Alimitos, USA. IEEE Computer Society, pp 296-303.

[104] Massaro, D.W., Liu, Y., Chen, T.H. and Perfetti, C.A. A. (2006). Multilingual embodied conversational agent for tutoring speech and language learning. *In Proceedings of the 9th International Conference on Spoken Language Processing (Interspeech 2006)*, Pittsburgh, USA, September 17-21 2006. International Speech Communication Association, pp 825-828.

[105] Maurel, W. (1999). *3D Modeling of the human upper limb including the biomechanics of joints, muscles and soft tissues*. PhD thesis. Ecole Polytechnique Federale De Lausanne.

[106] Mehrabian, A. (1972). *Nonverbal communication*. Chicago: Aldine-Atherton.

[107] Mouton, J. (2001). *How to succeed in your master's and doctoral studies: a South African guide and resource book*. Van Schaik Publishers.

[108] Murray, I. R. and Arnott, J. L. (1996). Synthesizing emotions in speech: is it time to get excited?. *In: Proceedings of The Fourth International Conference on Spoken Language Processing (ICSLP) 1996*, Philadelphia, USA, October 3-6 1996. IEEE Computer Society, pp. 1816-1819.

[109] NCH Software. (2008). WavePad sound editor. [online]. Available From: http://www.nch.com.au/wavepad/. [Accessed 22 November 2008].

[110] Noh, J.Y. and Neumann, U. (1998). *A survey of facial modelling and animation techniques*. University of Southern California, Technical Report 99-705.

[111] Noh, J. and Neumann, U. (2001). Expression cloning. *In: Proceedings of SIGGRAPH 2001*, Los Angeles, USA. August 12-17 2001. ACM Press, pp 277-288.

[112] Oates, B. (2006). *Researching information systems and computing*, London: Sage Publications Ltd.

[113] Ochs, M., Niewiadomski, R., Pelachaud, C. and Sadek, D. (2005). Intelligent expressions of emotions. *In: Proceedings of The First International Conference on Affective Computing & Intelligent Interaction (ACII) 2005*, Beijing, China, October 22-24 2005. Springer, pp 707-714.

[114] Oka, M., Tsutsui, K., Ohba, A., Kurauchi, Y. and Tago, T. (1987). Real-time manipulation of texture-mapped surfaces. *In: Proceedings of SIGGRAPH 1987*, Anaheim, USA, July 27-31 1987. ACM Press, pp 181-188.

[115] Ostermann, J. (1998). Animation of synthetic faces in MPEG-4. *In: Proceedings of Computer Animation '98*, Philadelphia, USA, June 8-10 1998. IEEE Computer Society, pp 49-51.

[116] Ostermann, J. and Weissenfeld, A. (2004). Talking faces – technologies and applications. *In: Proceedings of the 17th International Conference on Pattern Recognition (ICPR 2004)*, Cambridge, UK, August 23-26 2004. IEEE Computer Society, pp 826- 833.

[117] Ouni, S., Cohen, M.M. and Massaro, D.W. (2005). Training Baldi to be multilingual: A case study for an Arabic Badr. *Speech Communication*, **45**(2), pp 115-137.

[118] Paeschke, A., Kienast, M. and Sendlmeier, W. (1999) F0-contours in emotional speech. *In: Proceedings of 14th International Congress of Phonetic Sciences (ICPhP 99)*, San Francisco, USA, August 1-7 1999. University of California, pp 929–931.

[119] Pandzic, I.S., Capin, T.K., Magnenat-Thalmann, N. and Thalmann, D. (1996). Towards natural communication in networked collaborative virtual environments. *In: Proceedings of FIVE '96*, Pisa, Italy, December 19-20 1996. pp 37-47.

[120] Paouri, A., Magnenat-Thalmann, N. and Thalmann, D. (1991). Creating realistic three-dimensional human shape characters for computer generated films. *In: Proceedings of Computer Animation '91 (CA '91)*, Geneva, Switzerland, May 22-24 1991. Springer, pp 89-99.

[121] Parent, R., King, S. and Fujimura, O. (2002).Issues with lip synch animation: can you read my lips?, *In: Proceedings of Computer Animation 2002 (CA 2002)*, Geneva, Switzerland, June 19-21 2002. IEEE Computer Society, pp 3-10.

[122] Parke, F.I. (1972). Computer generated animation of faces. *In: Proceedings of the ACM annual conference - Volume 1*, Boston, USA, August 14 1972. ACM Press, pp 451-457.

[123] Parke F.I. and Waters, K. (1996). *Computer facial animation*. AK Peters.

[124] Pasquariello, S. and Pelechaud, C. (2001). Greta: a simple facial animation engine. *In:* Roy, R., Koppen, M. Ovaska, S., Furuhashi, T. and Hoffmann, F., eds. *Soft Computing and Industry*. Springer, pp 511-526.

[125] Pease, A. (1981). *Body language*. London: Sheldon Press.

[126] Pei, Y. and Zha, H. (2006). Vision based speech animation transferring with underlying anatomical structure. *In: Proceedings of 7th Asian Conference on Computer Vision (ACCV 2006)*, Hyderabad, India, January 13-16 2006. Birkhäuser, pp 591–600.

[127] Pelachaud, C. (1991). *Communication and coarticulation in facial animation*. PhD thesis. University of Pennsylvania.

[128] Pelachaud, C., Badler, N. I. and Steedman, M. (1991) Linguistic issues in facial animation. *In: Proceedings of Computer Animation '91 (CA '91)*, Geneva, Switzerland, May 22-24 1991. Springer, pp 15-30.

[129] Pelachaud, C. , van Overveld, C. and Seah, C. (1994). Modelling and animating the human tongue during speech production. *In: Proceedings of Computer Animation '94*, Geneva, Switzerland, May 25-28 1994. IEEE Computer Society, pp 40-49.

[130] Pelachaud, C., Badler, N. I. and Steedman, M. (1996). Generating facial expressions for speech. *Cognitive Science*, **20**(1), pp 1-46.

[131] Pelachaud, C. (2005). Multimodal expressive embodied conversational agents. *In: Proceedings of the 13th ACM International Conference on Multimedia*, November 6-11 2005, Singapore. ACM Press, pp 683-689.

[132] Pennsylvania State University. (2001). *Human biology*. [online]. Available from: http://www.bmb.psu.edu/courses/bisci004a. [Accessed 16 November 2008].

[133] Pighin, F., Hecker, J., Lichinski, D., Szeliski, R. and Salesin, D.H. (1998). Synthesizing realistic facial expressions from photographs. *In: Proceedings of SIGGRAPH 1998*, Orlando, USA, July 19-24 1998. ACM Press, pp 75-84.

[134] *Tin Toy* (1988). Short animated film. PIXAR, San Raphael, USA.

[135] Platt, S. M. and Badler, N. I. (1981). Animating facial expressions. *Computer Graphics*, **15**(3), pp 245-252.

[136] Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J. and Koch, R. (2004). Visual modelling with a hand-held camera. International Journal of Computer Vision, **59**(3), pp 207–232.

[137] Radovan, M. and Pretorius, L. (2006). Facial animation in a nutshell: past, present and future. *In: Proceedings of SAICSIT 2006*, Somerset West, South Africa, October 9-11 2006. ACM Press, pp 71 – 79.

[138] Radovan, M., Pretorius, L. and Kotze, A.E. (2007). Towards a Northern Sotho talking head. *In: Proceedings of AFRIGRAPH 2007*, Grahamstown, South Africa, October 29-31 2007. ACM Press, pp 125-133.

[139] Ravyise, I. (2006). Facial analysis and synthesis. *PhD Thesis*. Free University of Brussels.

[140] Reutersward, K., Flynn, J., Roble, D and Museth, K. (2005). Model flowing: capturing and tracking of deformable geometry, *In: Proceedings of SIGGRAPH 2005 Sketches*, Los Angeles, USA, July 31-August 4 2005. ACM Press.

[141] Rizvic, S. and Avdagic, Z. (2004). Phoneme reduction in automated speech for computer animation. *In: Proceedings of the 20th Spring Conference on Computer Graphics (SCCG 2004)*, Budmerice, Slovakia, April 22-24 2004. ACM Press, pp 89-96.

[142] Rogers, D. F. (2001). *An introduction to NURBS with historical perspective*. Academic Press, Morgan Kaufmann.

[143] Roux, J. C. (1979). *Labialization in Sesotho: the role of phonetic data in phonological analyses*. PhD Thesis. University of Stellenbosch.

[144] Rubin, M. B. and Bodner, S. R. (2002). A three-dimensional nonlinear model for dissipative response of soft tissue. *International Journal of Solids and Structures*, **39**(19), pp 5081-5099.

[145] Second Reality. (1998). *Subdivision Head Model*. [online]. Available from: http://www.arildwiro.com/tutorials/modelling/head/head.html. [Accessed 18 November 2008].

[146] Sederberg, T. W. and Parry, S. R. (1986). Free-form deformation of solid geometric models. *In: Proceedings of SIGGRAPH 1986*, Dallas, USA, August 18-22 1986. ACM Press, pp 151-160.

[147] Sifakis, E., Neverov, I. and Fedkiw, R. (2005). Automatic determination of facial muscle activations from sparse motion capture marker data. *In: Proceedings of SIGGRAPH 2005,* Los Angeles, USA, July 31-August 4 2005. ACM Press, pp 417 – 425.

[148] Sifakis, E., Selle, A., Robinson-Mosher, A. and Fedkiw, R.(2006) Simulating speech with a physics-based facial muscle model. *ACM SIGGRAPH/ Eurographics Symposium on Computer Animation (SCA 2006)*, Vienna, Austria, September 2-4 2006. A K Peters, 261-270.

[149] SIL International. (2004). *Glossary of linguistic terms*. [online]. Available from: http://www.sil.org/linguistics/GlossaryOfLinguisticTerms. [Accessed 18 November 2008].

[150] Steinke, F., Scholkpf, B. and Blanz, V. (2005). Support vector machines for 3D shape processing. *Computer Graphics forum*, **24**(3), pp 285-294.

[151] Tang, S., Liew. A.W.C. and Yan, H. (2004) Lip-sync in human face animation based on video analysis and spline models, *In: Proceedings of the 10th International Multimedia Modelling Conference (MMM 2004)*, Brisbane, Australia, January 5-7 2004. IEEE Computer Society, pp 102-108.

[152] Teran, J., Sifakis, E., Salinas-Blemker, S., Ng-Thow-Hing, V., Lau, C., and Fedkiw, R. (2005). Creating and simulating skeletal muscle from the visible human data set. *IEEE Transactions on Visualisation and Computer Graphics*, **11**(3), pp 317–328.

[153] Terzopoulos, D. (1988). The computation of visible-surface representations. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, **10**(4), pp 417-438.

[154] Terzopoulos, D. and Vasilescu, M. (1991). Sampling and reconstruction with adaptive mashes. *In: Proceedings of the IEEE Computer Vision and Pattern Recognition Conference (CVPR'91)*, Lahaina, Hawaii, June 3-6 1991. IEEE Computer Society, pp 70–75.

[155] Terzopoulos, D., Lee, Y. and Vasilesco, M.A.O. (2004). Model-based and image-based methods for facial synthesis, analysis and recognition. *In: Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition (FG`04)*, Seoul, Korea, May 17-19 2004. IEEE Computer Society, pp 3-10.

[156] Terzopoulos, D. and Waters, K. (1990). Physically-based facial modelling, analysis and animation. *Journal of Visualization and Computer Animation*. **1**(1-4), pp 73-80.

[157] Tucker, A.N. (1929). *The comparative phonetics of the Suto-Chuana group of Bantu languages*. Longmans, Green and Co.

[158] University of Iowa. (1996). *Voxel processing in a nutshell*. [online]. Available from: http://www.uiowa.edu/~image/iaf/concepts/voxels/solidblock.jpeg [Accessed 18 November 2008].

[159] University of Pittsburgh Voice Center. (1996). University of Pittsburgh Voice Center. [online]. Available from: http://www.pitt.edu/~crosen/voice/upvchome.html [Accessed 18 November 2008].

[160] US National Library of Medicine. (1994). *The visible human project*. [online]. Available from: http://www.nlm.nih.gov/research/visible/. [Accessed 18 November 2008].

[161] Van Wyk, E.B. (1977). *Praktiese fonetiek vir taalstudente*. Butterworth.

[162] Vasilescu, M. and Terzopoulos, D. (1992). Adaptive meshes and shells: irregular triangulation, discontinuities, and hierarchical subdivision. *In: Proceedings of the IEEE Computer Vision and Pattern Recognition Conference (CVPR'92)*, Champaign, USA, June 15-18 1992. IEEE Computer Society, pp 829–832.

[163] Vinayagamoorthy, V., Garau, M., Steed, A and Slater, M. (2004). An eye gaze model for dyadic interaction in an immersive virtual environment: practice and experience. *Computer Graphic Forum*, **23**(1), pp 1-11.

[164] Wang, C. L. (1993). *Langwidere: Hierarchical spline based facial animation system with simulated muscles*. MSc Thesis. University of Calgary.

[165] Waters, K. (1987). A muscle model for animating three-dimensional facial expressions. *In: Proceedings of SIGGRAPH 1987*, Anaheim, USA, July 27-31 1987. ACM Press, pp 17-24.

[166] Waters, K. and Levergood, T. M. (1993). *DECface: an automatic lip synchronisation algorithm for synthetic faces*. DEC Cambridge Research Lab., Cambridge, MA, Tech. Rep. CRL 93/4, 1993.

[167] Waters, K. and Levergood, T. M. (1994). An automatic lip synchronisation algorithm for synthetic faces. *In: Proceedings of the Second ACM International Conference on Multimedia '94*, San Francisco, USA, October 15-20 1994. ACM Press, pp 149-156.

[168] Williams, L. (1990a). Performance-driven facial animation. *In: Proceedings of SIGGRAPH 1990*, Dallas, USA, August 6-10 1990. ACM Press, pp 235-242.

[169] Williams, L. (1990b). 3D paint. *In:Proceedings of the 1990 Symposium on Interactive 3D Graphics*, Snowbird, USA, March 25-28 1990. ACM Press, pp 225-233.

[170] Yong, C., Tien, W. C., Faloutsos, P. and Pighin, F. (2005). Expressive speech-driven facial animation. *ACM Transactions on Graphics*, **24**(4), pp 1283–1302.

[171] Zhang, Y., Prakash, E.C. and Sung, E. (2001) Real-time physically-based facial expression animation using mass-spring system. *In: Proceedings of Computer Graphics International 2001*, Hong Kong, China, July 3-6 2001. IEEE Computer Society, pp. 347-350.

[172] Zhang, Y., Sim, T. and Tan, C. L. (2004). Rapid modelling of 3D faces for animation using an efficient adaptation algorithm. *In: Proceedings of 2nd International Conference Computer Graphics and Interactive Techniques in Australasia and South East Asia*, Singapore, June 15-18, 2004. ACM Press, pp 173-181.

[173] Zhang, L., Snavely, N., Curless, B. and Seitz, S. (2004). Spacetime faces: high resolution capture for modelling and animation. *In: Proceedings of SIGGRAPH 2004,* Los Angeles, USA, August 8-12 2004. ACM Press, pp 548–558.

[174] Zhang, Q., Liu, Z., Guo, B., Terzopoulos, D. and Shum, H. (2006). Geometry-driven photorealistic facial expression synthesis. *IEEE Transactions on Visualisation and Computer Graphics*, **12**(1), pp 48-60.