Appears in Proceedings of the Computer Animation Conference, Philadelphia, Pennsylvania, June 1998

MikeTalk: A Talking Facial Display Based on Morphing Visemes

Tony Ezzat tonebone@ai.mit.edu
Tomaso Poggio tp-temp@ai.mit.edu
MIT Center for Biological and Computational Learning
45 Carleton St. E25-204
Cambridge, MA 02141

Abstract

We present MikeTalk, a text-to-audiovisual speech synthesizer which converts input text into an audiovisual speech stream. MikeTalk is built using visemes*, which are a set of images spanning a large range of mouth shapes. The visemes are acquired from a recorded visual corpus of a human subject which is specifically designed to elicit one instantiation of each viseme. Using optical flow methods,* correspondence *from every viseme to every other viseme is computed automatically. By* morphing *along this correspondence, a smooth transition between viseme images may be generated. A complete visual utterance is constructed by* concatenating *viseme transitions. Finally, phoneme and timing information extracted from a text-to-speech synthesizer is exploited to determine which viseme transitions to use, and the rate at which the morphing process should occur. In this manner, we are able to synchronize the visual speech stream with the audio speech stream, and hence give the impression of a photorealistic talking face.*

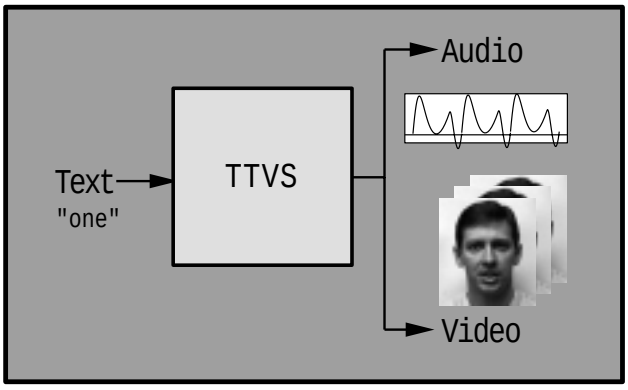

Figure 1. Overview of the MikeTalk TTVS system.

1. Introduction

The goal of the work described in this paper is to develop a text-to-audiovisual speech (TTVS) synthesizer called MikeTalk. MikeTalk is similar to a standard text-to-speech synthesizer in that it converts typed text into an audio speech stream. However, MikeTalk also produces an accompanying visual stream composed of a talking face enunciating that text. An overview of our system is shown in Figure 1.

TTVS systems are attracting an increased amount of interest in the recent years, and this interest is driven by the possible deployment of these systems as visual desktop agents, digital actors, and virtual avatars. In addition, they may also have potential uses in very low bandwidth videoconferencing and special effects, and would also be of interest to psychologists who wish to study visual speech production and perception.

The main research issue underlying the construction of a TTVS visual stream is the nature of the facial model to use. One approach is to model the face using traditional *3D modeling* methods. Parke [17] was one of the earliest to adopt such an approach by creating a polygonal facial model. Recent work on TTVS systems that is based on Parke's models include the work of Cohen & Massaro [7] and LeGoff & Benoit [13]. Extending this 3D modeling approach, Terzopoulos & Waters [19] built a facial model which also includes muscle and bone structures. To improve the *photorealism* of the facial model, Lee, Terzopoulos, et al. [12] resorted to Cyberware scanning. The Cyberware scanner produces accurate 3D structure and texture maps of the scanned face, and these maps are animated by overlaying them on top of the muscle-based models.

At the other end of the facial modeling spectrum

are a group of *image-based* approaches where the face is modeled using images alone. One such early approach was that of Beymer, Shashua, and Poggio [3], who utilized a *morphing* technique to synthesize novel, intermediate images of a face from example endpoints. In doing so, their algorithm was capable of modeling rigid facial transformations such as pose changes, as well as non-rigid transformations such as smiles.

In a similar vein, Scott, Kagels, et al. [18] also employed an image-based morphing method in their work. Their facial model is composed of a set of images which capture a large range of the mouth shapes occurring during speech. To animate the face, a morphing algorithm is developed which is capable of transitioning between the various mouth shapes in a smooth and realistic manner.

Most recently, Bregler, Covell, el al. [5] also described an image-based approach to facial modeling: in their work, a set of short audiovisual sequences are extracted from a larger audiovisual corpus. Each one of these short sequences is a *triphone* segment, and a large database with all the acquired triphones is built. A new audiovisual sentence is constructed by concatenating the appropriate triphone sequences from the database together.

The approach described in this work falls into the *image-based, morphing* category, and is close in spirit to the work of [3] and [18]. In the following sections, we describe the various aspects of our approach in detail.

2. Corpus and Viseme Data Acquiry

The basic underlying assumption of our facial synthesis approach is that the complete set of mouth shapes associated with human speech may be reasonably spanned by a finite set of *visemes*. The term *viseme* itself was coined initially by Fisher [9] as an amalgamation of the words "visual" and "phoneme". To date, there has been no precise definition for the term, but in general it has come to refer to a speech segment that is *visually* contrastive from another. In this work, a viseme will be defined to be a *static lip shape image that is visually contrastive from another.*

Given the assumption that visual speech is spanned by a set of visemes, we would like to design a particular visual corpus which elicits one instantiation for each viseme. The simplest approach to take is to assume a *one-to-one mapping* between the set of phonemes and the set of visemes, and design the corpus so that there is at least one word uttered which instantiates each phoneme.

This one-to-one strategy is a reasonable approach in light of the fact that our ultimate goal in this work is to use an underlying TTS system to produce an audiovisual sequence. In doing so, the TTS will produce a stream of phonemes corresponding to the input text. Consequently, we will need to *map* from the set of phonemes used by the TTS to a set of visemes so as to produce the visual stream. The one-to-one mapping strategy is also a good idea because most speech textbooks and dictionaries contain a list of phonemes and example words which instantiate them, and the corpus may thus be limited to those example words.

However, current literature on viseme research indicates that the mapping between phonemes and visemes is *many-to-one*: there are many phonemes which look alike visually, and hence they fall into the same visemic category. This is particularly true, for example, in cases where two sounds are identical in manner and place of articulation, but differ only in voicing characteristics. For example, \b\ and \p\ are two bilabial stops which differ only in the fact that the former is voiced while the latter is voiceless. This difference, however, does not manifest itself visually, and hence the two phonemes should be placed in the same visemic category. The reader is referred to Owens and Blasek [16] for a discussion of the consonantal visemic categories, and to Montgomery and Jackson [14] for a discussion of the vocalic visemic categories.

Conversely, the literature points out that the map from phonemes to visemes is also *one-to-many*: the same phoneme can have many different visual forms. This phenomenon is termed *coarticulation*, and it occurs because the neighboring phonemic context in which a sound is uttered influences the lip shape for that sound. For example, the viseme associated with \t\ differs depending on whether the speaker is uttering the word `two` or the word `tea`. In the former case, the \t\ viseme assumes a rounded shape in anticipation of the upcoming \uu\ sound, while the latter assumes a more spread shape in anticipation of the upcoming \ii\ sound. The reader is referred to Cohen and Massaro [7] for an in-depth discussion on the theories behind coarticulation, and to Owens and Blasek [16] for a study on consonantal perception in various vocalic contexts.

At the present stage of our work, we have decided for the sake of simplicity to ignore coarticulation effects. Consequently, the recorded corpus, which is shown in Figure 2, assumes a one-to-one map from phonemes to visemes, and hence one word is uttered for every phoneme. The example words uttered are obtained from Olive, Greenwood, et al. [15], and are generally categorized into example words which instantiate consonantal, monophthong vocalic, and diphthong vocalic phonemes.

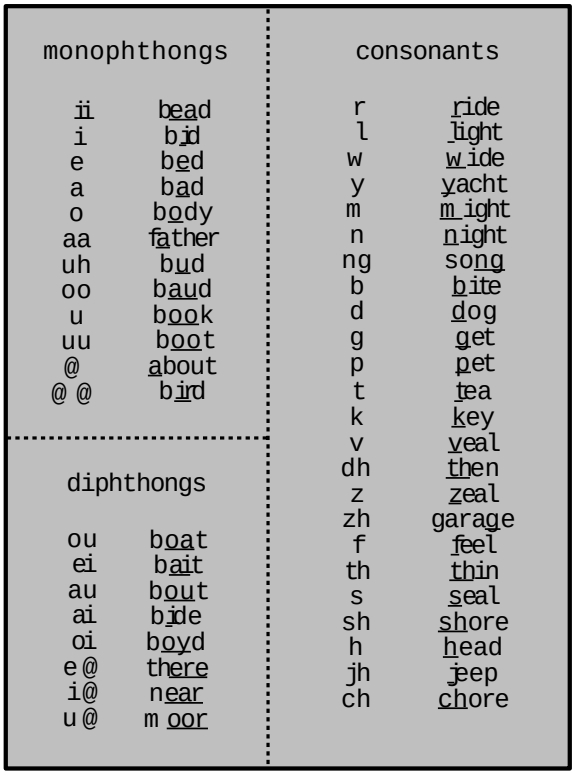

Figure 2. The recorded visual corpus. The underlined portion of each example word identifies the target phoneme being recorded. To the left of each example word is the phonemic transcription label being used.

After the whole corpus is recorded and digitized, one lip image is extracted as an instance of that viseme. This leads to the extraction of 52 viseme images in all: 24 representing the consonants, 12 representing the monophthongs, and 16 representing the diphthongs. Since this is an unnecessarily large number of visemes, it was decided to further reduce the viseme set by grouping together those visemes which looked similar. This was done in a subjective manner, by comparing the viseme images visually to assess their similarity. The authors were also guided in this process by the conclusions in Owens and Blasek [16] for the case of consonantal visemes, and in Montgomery and Jackson [14] for the case of vocalic visemes. *This grouping step is, in effect, a decision to use a many-to-one mapping strategy instead of a one-to-one mapping strategy.*

The final reduced set of visemes are shown in Figures 3 and 4. There were 6 final visemes representing the 24 consonantal phonemes. There were 7 visemes

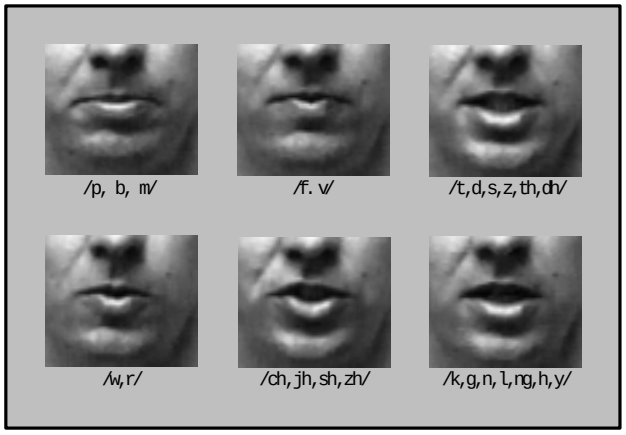

Figure 3. The 6 consonant visemes

representing the 12 monophthong phonemes. In the case of diphthongs, it was found that all vowel nuclei could be represented by corresponding monophthong visemes. The only exception to this occurred in the case of two nuclei: the second nucleus of the \au\ diphthong, which we call the \w-au\ viseme, and the first nucleus of the \o-ou\ diphthong, which we call the \o-ou\ viseme. Finally, one extra viseme was included to represent silence, which we call \#\.

In all, there are 16 final visemes.

3. Morphing Between Visemes

In constructing a visual speech stream, it is not sufficient to simply display the viseme images in sequence. Doing so would create the disturbing illusion of very abrupt mouth movement, since the viseme images differ from each other in shape significantly. Consequently, a mechanism of transitioning from each viseme image to every other viseme image is needed, and this transition must be smooth and realistic. In this work, a *morphing* technique was adopted to create this transition.

3.1. Background

Morphing was first popularized by Beier & Neely [1] in the context of generating transitions between different faces for Michael Jackson's *Black or White* music video. The transformations between images occur as a *warp* of the first image into the second, a similar *inverse warp* of the second image into the first, and a final *cross-dissolve* or *blend* of the warped images. It should be noted that both Beymer, Shashua, Poggio [3] and Scott, Kagels, et al. [18] noticed the viability of using

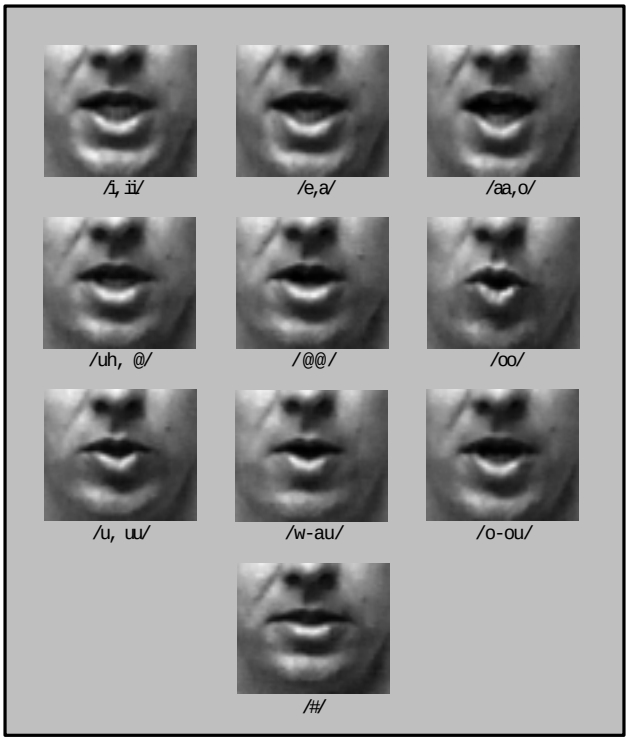

Figure 4. The 7 monophthong visemes, 2 diphthong visemes, and the silence viseme.

morphing as a method of transitioning between various facial pose, expression, and mouth position imagery.

The difficulty with traditional morphing approaches is that the specification of the warp between the images requires the definition of a set of high-level *features*. These features serve to ensure that the warping process preserves the desired *correspondence* between the geometric attributes of the objects to be morphed. For example, if we were morphing between two faces, we would want the eyes in one face to map to the eyes in the other face, the mouth in one face to map to the mouth in the other face, and so on. Consequently, the correspondence between these eyes and mouth features would need to be specified.

When it is done by hand, however, this feature specification process can become quite tedious and complicated, especially in cases when a large amount of imagery is involved. In addition, the process of specifying the feature regions usually requires hand-coding a large number of ad-hoc geometric primitives such as line segments, cornerpoints, arcs, circles, and meshes. Beier & Neely [1] in fact, make the explicit statement that the specification of the correspondence between images constitutes the most time-consuming aspect of the morph.

As a result, we have resorted to *optical flow methods* to alleviate these problems.

3.2. Optical Flow

Optical flow was originally formulated by Horn & Schunck [10] in the context of measuring the motion of objects in images. This motion is captured as a two-dimensional vector field $\{d_x, d_y\}$ that describes how each pixel has moved between the viseme images. From our perspective, optical flow is important because it allows for the *automatic* determination of correspondence between images. In addition, since each pixel is effectively a feature point, optical flow allows us to bypass the need for hand-coding any ad-hoc feature primitives.

In general, determining optical flow is a highly under-constrained problem, and an additional set of assumptions about the underlying motion need to be made. In the particular case of the optical flow algorithm used in this work (Bergen & Hingorani [2]), one such assumption made is that the motion between images is *small*. This *small motion assumption* is extremely detrimental, however, because in many cases it prevents the optical flow algorithm from computing correct correspondence between viseme images that exhibit large differences in motion between each other. Consequently, *direct* application of our optical flow algorithm only succeeds when the motion between any two viseme images is small.

However, we have found that a *flow concatenation* procedure (Ezzat [8]) overcomes the problems which occur when the small motion assumption fails to apply. Since the original visual corpus is digitized at 30 fps, there are many intermediate frames that lie between the chosen viseme images. The pixel motions between these consecutive frames are small, and hence the small motion assumption is not violated. Consequently, we compute a series of consecutive optical flow vectors between each intermediate image and its predecessor, and then *concatenate* them all into one large flow vector that defines the global transformation between the chosen visemes.

Further details of the flow concatenation procedure itself may be found in Ezzat [8].

3.3. Forward Warping

Given two viseme images `A` and `B`, and the computed correspondence vectors $d_x^{A \to B}$ and $d_y^{A \to B}$ between them, the first step of our morphing algorithm

is to *forward warp* A along $d_x^{A \to B}$ and $d_y^{A \to B}$.

Our forward warping algorithm "pushes" the pixels of A along the flow vectors. By scaling the computed flow vectors uniformly by a constant between 0 and 1, one can produce a series of warped intermediate images which approximate the transformation between visemes A and B. Several such intermediate warps are shown in Figure 5a.

The black *holes* which appear in the intermediate images shown in Figure 5a occur in cases where a destination pixel was not filled in with any source pixel value. One reason for this is that the forward warping algorithm rounds to the nearest integer when it decides which destination pixel to fill. Another reason is that local image expansion involved in the underlying motion of the lips causes the optical flow vectors themselves to diverge.

To remedy this, a hole-filling algorithm first proposed by Chen & Williams [6] was adopted. The algorithm pre-fills the destination images with a special reserved background color. After performing the forward warp, the hole-filling algorithm traverses the destination image in rasterized order and fills in the holes by interpolating linearly between their non-hole endpoints. Figure 5b shows the same set of warped intermediates as in Figure 5a, but with the holes filled in using our algorithm.

3.4. Morphing

Because forward warping can only move pixels around, *it cannot model the appearance of new pixel texture*. As is evident from the sequence in Figure 5b, a forward warp of viseme A along the flow vectors can never produce a final image that looks like viseme B, since viseme B itself contains a large amount of novel texture from the inside of the mouth.

Morphing overcomes this "novel pixel texture" problem because it involves *two* warps, one from the starting viseme to the intermediate point, and another from the ending viseme to the same intermediate point. The two warped images are subsequently scaled by respective *blending parameters* and then added to produce the final morphed image. By interpolating the blending parameters the morph "fades out" the warped versions of the starting viseme and "fades in" the warped versions of the ending viseme. The blending process thus allows the two warps to be effectively combined, and the "new" pixels of the second viseme to become involved in the viseme transition itself. It should be noted that both Beymer, Shashua, & Poggio [3] and Bergen & Hingorani [2] noticed the viability of morphing along optical flow vectors as a means of creating realistic transitions between two images.

Since the second warp in a morph is a warp of viseme B towards viseme A, an *inverse flow* from B to viseme A needs to be computed. In this work, the inverse flow is computed using an algorithm that was first described in Beymer, Shashua, and Poggio [3]. Figure 5c depicts the set of images generated as a result of warping along the inverse flow from viseme B to viseme A.

A final morph sequence is shown in Figure 5d. The blending parameter α is interpolated *linearly* between 0.0 and 1.0.

3.5. Morph Concatenation

To construct a visual stream in which a word or a sentence is uttered, we simply *concatenate* the appropriate viseme transitions together. For example, the word one, which has a phonetic transcription of \w-uh-n\, is composed of the two viseme transitions \w-uh\ and \uh-n\ put together and played seamlessly one right after the other. The transition between viseme transitions is smooth because the \uh\ viseme image is the same image in both viseme transitions.

4. Synthesizing the New Audiovisual Sentence

We have incorporated the Festival TTS system (Black & Taylor [4]), developed at the University of Edinburgh, into our work. A *voice* in the Festival system consists of a set of recorded *diphones*, which are stored as LPC coefficients and corresponding residuals (Hunt, Zwierzynski, et al. [11]). It is interesting to note that the final audio speech stream is constructed by concatenating the appropriate diphones together, in a manner that is completely analogous to our method for concatenating viseme transitions.

The Festival TTS system models speech production using the traditional *source-filter model* in which a pitch impulse train is modulated by a vocal transfer function. This model has been historically important for speech synthesis because it effectively isolates the intonation and duration information, captured by the pitch impulse train, from the phonemic information, captured by the vocal filter.

The TTS system thus takes as input a typed sentence and computes as an intermediate representation the desired pitch train with which to excite the vocal transfer function. For each pitch impulse in the train, the TTS system determines its length in samples, and the diphone filter which it will excite. For example, the pitch train for the word bed contains a series of impulses that excite various consecutive portions of the

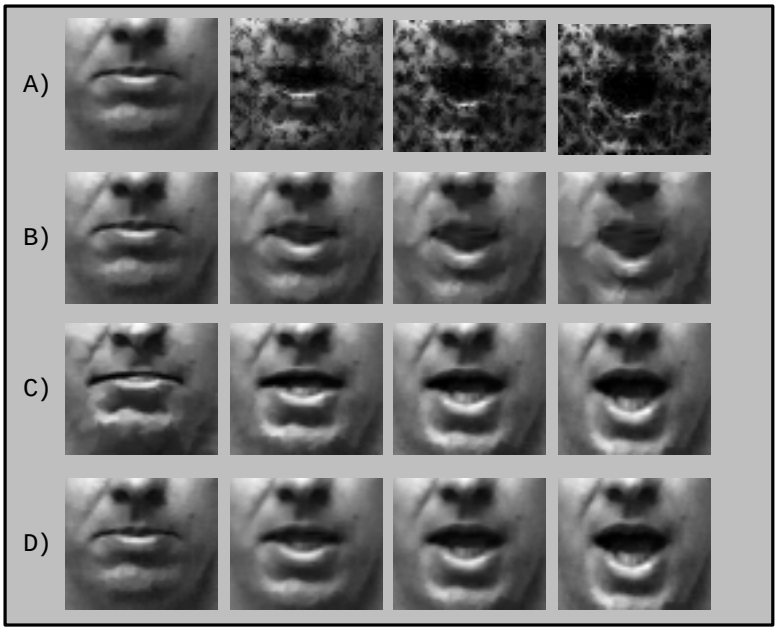

Figure 5. A) Forward warping viseme A (first image) towards B, without hole filling. B) Forward warping viseme A (first image) towards B, with hole filling. C) Forward warping viseme B (last image) towards A, with hole filling. D) A morph between viseme A and viseme B.

diphone \b-e\, followed by a series of impulses that excite various consecutive portions of the diphone \e-d\.

The information contained in the pitch impulse train is sufficient for creating a visual speech stream in close synchrony with the audio stream. We place a new viseme image at every pitch impulse which excites a diphone different from the previous. So the \e\ viseme in the previous example would be placed at the position of the first pitch impulse which transitions between the \b-e\ and \e-d\ diphones.

The number of frames to morph is determined by counting the total length in samples of all the pitch impulses between any two viseme images. We divide this sample total by the audio sampling rate (16kHz) to determine the duration of a viseme transition in seconds. Multiplication by the desired video frame rate (60fps) then determines the number of needed frames.

We have found that the use of TTS timing and phonemic information in this manner produces superb quality lip synchronization between the audio and the video. The drawback of using a TTS system, however, as opposed to a recorded natural speech signal that is manually annotated, is that the audio may have a slightly 'robotic' quality to it. Nevertheless, the flexibility of having our TTVS system produce audiovisual output for *any* typed text has offset any misgivings we might have regarding final audio quality. We also believe that future generations of TTS systems will continue to achieve better levels of audio quality.

5. Summary of Our Approach

In summary, our talking facial model may be viewed as a *collection of viseme imagery and the set of optical flow vectors defining the morph transition paths from every viseme to every other viseme.*

We briefly summarize the individual steps involved in the construction of our facial model:

Recording the Visual Corpus: First, a visual corpus of a subject enunciating a set of key words is recorded. An initial one-to-one mapping between phonemes and visemes is assumed, and the subject is asked to enunciate 40-50 words.

Extracting the Visemes: Next, one single image for each viseme is identified and extracted from the corpus sequence. This is done manually by searching through the recorded frames. The viseme set is then subjectively reduced to a final set of 16 visemes.

Building the Flow Database: Thirdly, we build a database of optical flow vectors that specify the morph transition from each viseme image to every other viseme image. Since there are 16 visemes in our final viseme set, a total of 256 optical flow vectors are computed.

Synthesizing the New Audiovisual Sentence: Finally, we utilize a text-to-speech system (Black & Taylor [4]) to convert unconstrained input text into a string of phonemes, along with duration information for each phoneme. Using this information, we determine the appropriate sequence of viseme transitions to make, as well as the rate of the transformations. The final visual sequence is composed of a *concatenation* of the viseme transitions, played in synchrony with the audio speech signal generated by the TTS system.

6. Results

We have synthesized several audiovisual sentences to test our overall approach for visual speech synthesis and audio synchronization described above. Our results may be viewed by accessing our World Wide Web home page at `http://cuneus.ai.mit.edu:8000/research/miketalk/miketalk.html`. The first author may also be contacted for a video tape which depicts the results of this work.

7. Further Work

Further work involves incorporating a model for coarticulation as well as higher-level mechanisms in visual speech communication such as eye blinks, eye gaze changes, eyebrow movements, and head nods. In addition, a method of synthesizing the face at different poses needs to be explored.

References

[1] T. Beier and S. Neely. Feature-based image metamorphosis. In *SIGGRAPH '92 Proceedings*, pages 35–42, Chicago, IL, 1992.

[2] J. Bergen and R. Hingorani. Hierarchical motion-based frame rate conversion. Technical report, David Sarnoff Research Center, Princeton, New Jersey, Apr. 1990.

[3] D. Beymer, A. Shashua, and T. Poggio. Example based image analysis and synthesis. A.I. Memo No. 1431, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1993.

[4] A. Black and P. Taylor. *The Festival Speech Synthesis System*. University of Edinburgh, 1997.

[5] C. Bregler, M. Covell, and M. Slaney. Video rewrite: Driving visual speech with audio. In *SIGGRAPH '97 Proceedings*, Los Angeles, CA, August 1997.

[6] S. E. Chen and L. Williams. View interpolation for image synthesis. In *SIGGRAPH '93 Proceedings*, pages 279–288, Anaheim, CA, August 1993.

[7] M. M. Cohen and D. W. Massaro. Modeling coarticulation in synthetic visual speech. In N. M. Thalmann and D. Thalmann, editors, *Models and Techniques in Computer Animation*, pages 139–156. Springer-Verlag, Tokyo, 1993.

[8] T. Ezzat. Example-based analysis and synthesis for images of human faces. Master's thesis, Massachusetts Institute of Technology, 1996.

[9] C. G. Fisher. Confusions among visually perceived consonants. *Jour. Speech and Hearing Research*, 11:796–804, 1968.

[10] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[11] M. J. Hunt, D. A. Zwierzynski, and R. Carr. Issues in high quality lpc analysis and synthesis. In *Proceedings of EUROSPEECH*, volume 2, pages 348–351, Paris, France, 1989.

[12] Y. Lee, D. Terzopoulos, and K. Waters. Realistic modeling for facial animation. In *SIGGRAPH '95 Proceedings*, pages 55–62, Los Angeles, CA, August 1995.

[13] B. LeGoff and C. Benoit. A text-to-audiovisual-speech synthesizer for french. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, Philadelphia, USA, October 1996.

[14] A. Montgomery and P. Jackson. Physical characteristics of the lips underlying vowel lipreading performance. *Jour. Acoustical Society of America*, 73(6):2134–2144, 1983.

[15] J. Olive, A. Greenwood, and J. Coleman. *Acoustics of American English Speech: A Dynamic Approach*. Springer-Verlag, New York, USA, 1993.

[16] E. Owens and B. Blazek. Visemes observed by hearing-impaired and normal-hearing adult viewers. *Jour. Speech and Hearing Research*, 28:381–393, September 1985.

[17] F. I. Parke. *A parametric model of human faces*. PhD thesis, University of Utah, 1974.

[18] K. Scott, D. Kagels, S. Watson, H. Rom, J. Wright, M. Lee, and K. Hussey. Synthesis of speaker facial movement to match selected speech sequences. In *Proceedings of the Fifth Australian Conference on Speech Science and Technology*, volume 2, pages 620–625, December 1994.

[19] D. Terzopoulos and K. Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):569–579, 1993.