# Assembling an Expressive Facial Animation System

Alice Wang[2] Michael Emmi[1] Petros Faloutsos[1]
[1] University of California Los Angeles
[2]Northrop Grumman

## Abstract

In this paper we investigate the development of an expressive facial animation system from publicly available components. There is a great body of work on face modeling, facial animation and conversational agents. However, most of the current research either targets a specific aspect of a conversational agent or is tailored to systems that are not publicly available. We propose a high quality facial animation system that can be easily built based on affordable off-the-shelf components. The proposed system is modular, extensible, efficient and suitable for a wide range of applications that require expressive speaking avatars. We demonstrate the effectiveness of the system with two applications: (a) a text-to-speech synthesizer with expression control and (b) a conversational agent that can react to simple phrases.

**CR Categories:** I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

**Keywords:** Facial animation, text-to-visual speech synthesis, expressive visual speech.

## 1 INTRODUCTION

Facial animation is one of the most important aspects of a realistic virtual actor. Expressions convey important information about the emotional state of a person and about the content of speech, making the face the most complex and effective tool of human communication. The study of facial motion has been an active area of research in many disciplines, such as computer graphics, linguistics, psychology et al. Building an animatable, moderately sophisticated human face for research or entertainment purposes requires significant engineering effort. Researchers painstakingly create custom face models because existing ones are either not available, or not suitable for their purposes. To the best of our knowledge, there is currently no easy way to develop a relatively high quality facial animation system without significant effort.

In this paper we propose a facial animation system that is built from robust, off-the-shelf components. Its modular structure makes the system extensible, intuitive and easy to replicate. We demonstrate the versatility and ease of use of the proposed system by implementing two applications. (a) *A visual speech synthesis module*: given an input string of text annotated with emotion-tags, the proposed system can produce the corresponding lip-synchronized motion of a high quality 3D face model with appropriate expressions. Synthesizing visual speech is challenging because of the coarticulation effect: the shape of the mouth that corresponds to a phoneme in a speech sequence depends not only on the current phoneme, but also on phonemes that occur before or after the current phoneme. (b) *A conversational agent*: this application implements an autonomous agent that can understand simple phrases and react to them.

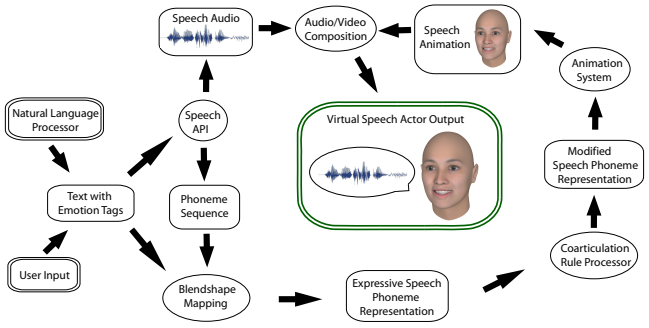


**Figure 1:** *System Overview*

Figure 1 shows an overview of the proposed system. Given input text annotated with expression tags, our *text-to-speech (Speech API)* module produces the corresponding speech signal along with a phoneme sequence. The phoneme sequence goes through the *blendshape mapping* module that produces facial motion that matches the input text and expression tags. The resulting facial motion and corresponding phoneme sequence is fed to the *coarticulation processor* that applies the appropriate rules and produces new sequences. The resulting facial motion is smoothly applied on the facial model by the *animation system*. Finally, the audio and the motion are composited and displayed on the screen. The text can come from any source; for example, directly provided by a user, or through a natural language processing and synthesis module.

Compared to the numerous approaches to facial animation problem, our system has a number of interesting properties. First, it is easy to implement because it integrates APIs and software that are affordable and publicly available. Second, the coarticulation model is rule-based and can be easily and intuitively adjusted, in contrast with statistical models that are not easy to tamper with. Third, the face and motion model is based on blending intuitive *key-shapes* derived from the increasingly popular blendshape approach. Our blendshapes are provided by commercial system FaceGen[Singular Inversion, ] which enables us to model a sizable range of different faces that differ in parameters such as age, complexion, and ethnicity. Lastly, the system is modular, allowing developers or user to replace outdated components with more technologically advanced (or more expensive) modules. For example, improvements in the coarticulation rules and enhanced phoneme timing sequences can easily be introduced into our current system.

We believe that this work is of interest both in its entirety, and in terms of its individual pieces for many applications, including computer games. Furthermore, as operating systems become more sophisticated, even the built-in text-to-speech synthesizers will produce game-quality audio results.

The remainder of this paper is organized as follows. Section 2 re-


Sandbox Symposium 2007, San Diego, CA, August 04-05, 2007.
© 2007 ACM 978-1-59593-749-0/07/0008 $5.00

views related work. Section 3 presents our rule-based coarticulation model. Section 4 describes the components of the system. Section 5 shows our results. Section 6 discusses applications and future work. Section 7 concludes the paper.

## 2 PREVIOUS WORK

There are mainly two ways to model faces. The first class of models, first introduced by [Parke, 1974], is based on 3D polygonal representations. Such models can be constructed by hand, by 3D scanning of real faces, or using image-based rendering techniques [Pighin et al., 1998; Guenter et al., 1998; Blanz and Vetter, 1999]. The muscle-based approach [Lee et al., 1995; Waters, 1987] models the underlying muscles and skin of the face. Polygonal face models need to be accompanied by high quality texture maps which can be extracted by scanners [Lee et al., 1995] and photographs [Pighin et al., 1998; Guenter et al., 1998]. Image-based rendering techniques provide an alternative to 3D face models. Using recorded video sequences of human subjects, such techniques can directly produce novel video sequences of animated faces [Brand, 1999; Ezzat et al., 2002; Saisan et al., 2004].

Independently of the underlying face model, the main problem is the synthesis and control of facial motion.

### 2.1 Face motion synthesis

A facial motion model typically consists of 3 major motion classes: lower face motion (lip and chin), upper face motion (eyes and eyebrows), and rigid head motion. In this work, we focus on lip-synchronization that accurately mimics lip motion that matches an input audio sentence. This problem is challenging because of a need to accurately model human coarticulation: the shape of the mouth corresponding to a phoneme depends on the phonemes before and after the given phoneme. Approaches that attempt to solve this problem can be categorized in three ways.

The physics-based approach uses the laws of physics and muscle forces to drive the motion of the face. Although it is computationally expensive, it has been shown to be quite effective [Lee et al., 1995; Waters, 1987]. Data-driven approaches use a phoneme segmented input speech signal to search within large databases of recorded motion and audio data for the closest matches[Massaro, 1997; Bregler et al., 1997; Cao et al., 2003; Cao et al., 2005]. Within this family of techniques, a similar approach to our system was proposed in [Albrecht et al., 2002], but their work focused on delivering expressive emotions along with the speech. A third class of techniques attempt to eliminate the need for large example databases by creating statistical or dynamic models of face motion[Brook and Scott, 1994; Masuko et al., 1998; Cohen and Massaro, 1993; Brand, 1999; Ezzat et al., 2002; Saisan et al., 2004; Kalberer, 2003; Kalberer et al., 2002]. Most of these approaches use high quality motion or video data to learn compact statistical models of facial motion. These models produce the corresponding facial motion essentially by performing a statistical interpolation of the recorded data. Although such approaches can produce high quality results, they have certain drawbacks. *Efficiency*: the synthesis time is often exponential to the duration of the input speech. *Complexity*: statistical models are not easy to implement and are often hard to adjust. *Training data*: training such models require high fidelity audio and motion (or video) data which can be hard to collect.

There is a lot of work on facial animation within the area of embodied conversational agents . Much of it implements rule-based coarticulation ideas. Revéret [Reveret et al., ] present a talking head which can track facial movements. Lundeberg [Lundeberg and Beskow, ] create a dialogue system, called August, which can convey a rich repertoire of extra-linguistic gestures and expressions. [Pelachaud, 1991; Cassell et al., 1994] develop a complex facial animation system that incorporates emotion, pitch and intonation. They also use a rule-based system to model coarticulation. Ruth [DeCarlo and Stone, ] is one of the few publicly available conversational agents. An interesting approach towards combining gestures and speech is presented in [Stone et al., 2004].

In this paper we propose an expressive prototype facial animation system based on affordable, publicly available components. Our goal is to provide a modular system that researchers can easily use and extend. Upon publication, the system will be available online for free except for the blendshapes that are commerically licensed (FaceGen). The proposed system is interactive, easy to implement, and modular. It is particularly suitable for interactive applications that require affordable, high quality virtual actors that can speak.

## 3 VISUAL SPEECH SYNTHESIS

The main problem in visual speech synthesis is modeling coarticulation. In this paper, we do not attempt to provide a definitive answer to the problem of accurately modeling coarticulation in speech animation. However, our proposed set of rules is an excellent starting point and provides satisfactory results.

To aid in the understanding of our current set of coarticulation rules, the following is a background of the field of coarticulation. It is commonly thought that actual human lip movements during the pronunciation of a word is sensitive to the context of the syllables that occur near the phoneme that is currently being enunciated. This is termed as coarticulation. A simple solution to coarticulation attempts to subdivide a given word into its constituent phonemes. However, modern science suggests that the human brain uses a more sophisticated methodology to convert a plan to speak into the physical mechanisms that generate speech. Linguists have theorized that the speech articulators in real humans are moved in an energy efficient manner. Therefore, the components of the mouth will move only when it is necessary to take a different shape in order to correctly pronounce a phoneme. Proper coarticulation requires that the mouth shape used to produce a particular phoneme depends not only on the current phoneme, but also on some set of phonemes before or after the current phoneme. For example, a comparison of the words “strength” and “store” reveal that the shape of the lips at the point of pronunciation of the “s” in the latter word is rounded, whereas in the former it is not. Linguists believe that more energy can be preserved by maintaining the current lip position for as long as possible. This preservation of articulator shape is propagated forwards and backwards through phoneme strings, and it has been found to be preserved at up to six phoneme positions away in some languages [Kent, 1977]. The implication of the work in [Kent, 1977] suggests that a set of coarticulation rules must consider up to six phonemes before and after the current phoneme in order to accurately model human speech.

Our implementation of rule-based coarticulation assumes that the linguistic laws that govern coarticulation exist. No assumption is made about whether the rules are in fact general or specific to each possible combination of visemes, so the necessary flexibility to incorporate both types of rules is allowed. This strategy is inspired by [Kent, 1977], which serves as a survey for models of coarticulation that have been found to contain counterexamples. Figure 2 shows the set of rules that we currently have in our system. Each rule defines a set of *source phonemes*, a set of *target phonemes*, and a *direction*, forward or backward.

In our rule-based system, source phonemes will influence each target phoneme in the specified direction until a phoneme in the

complement of the target set is reached. For instance, using the "strength" and "store" example from above, a backward rule could be made such that the source set consists of the "e" and "o", and the target set contains the "s", "t", and "r". When the rule is processed on each word, the result would be the lip shape of the "e" appearing over the "str" in "strength", and the lip shape of the "o" appearing during the "st" in "store". The validity of this model depends on the assumption that the natural laws of coarticulation may be represented in a specific case by case format, if not a nicer general format. Although the rules used here are assembled from the available linguistics literature, it may not be a complete set. However, our experiments show that the system produces visually satisfactory results.

### 3.1 Emotion Control

A total of six emotions were incorporated in our system: happy, sad, angry, surprised, disgusted, and fearful. This is determined by the available blendshapes generated from FaceGen. We allow the user to tag input text on the granularity of words. Words between tags are interpreted to have the same emotion as the previous tagged word, and any untagged words at the beginning of the text are treated as neutral words. Currently, our system uses linear transition between different emotions and between different weights of the same emotion.

The tags can optionally contain a weight factor that ranges from 0 to 1 and defines the amount of emotion that the tag carries. For example the following text: `<happy*0.5>Hi Susie<neutral>` makes the avatar speak the given text starting with a 50% happy expression and finishing with a neutral one. When no weight is specified, a weight of 1 is used.

## 4 IMPLEMENTATION

Apart from the proposed rule-based coarticulation model, our system is based on off-the-shelf, publicly available components. Figure 1 illustrates the modular high-level overview of the proposed system. The TTS (text-to-speech) component divides the input text into an audio signal and a phonetic sequence. A mapping from phonemes to visemes provides the visemes to the aforementioned coarticulation rules, which in turn supplies the modified sequence of visemes to the audiovisual composition mechanism. The rest of this section describes each component in detail.

### 4.1 Face Modeling

To model 3D facial motion we use a *blendshape* approach [Joshi et al., 2003; Singular Inversion, ]. Given a set of $n$ facial expressions and corresponding polygonal meshes $B = \{B_0, B_1, ..., B_n\}$ called *blendshapes*, we can create new facial expressions by blending different amounts of the original meshes $B_i$:

$$B_{new} = B_0 + \sum_{i=1}^{n} (w_i(B_i - B_0)), \quad (1)$$

where $w_i$ are arbitrary weights and $B_0$ corresponds to a neutral expression. To avoid exaggerated expressions, the weights are typically restricted such that $w_i \in [0, 1]$. By varying the weights $w_i$ over time, we can produce continuous animation. The main challenge of a blendshape approach is constructing an appropriate set of blendshapes.

There are two ways to construct blendshapes. First, we can digitally scan the face of a human actor while he/she makes different expressions. This approach requires an expensive scanner and typically produces meshes that have different numbers of vertices. To use the simple blending formula in Equation 1, we must establish a correspondence between vertices across blendshapes, which is not practical. Alternatively, we can use a more complex approach such as the one proposed in [Joshi et al., 2003]. Otherwise, we can create a set of blendshapes by manually displacing the vertices of a 3D face model. This approach is tedious and time consuming, and only a skilled animator is capable of producing a quality set of blendshapes.

The set of shapes used in our system are exported from the face modeling tool FaceGen [Singular Inversion, ]. FaceGen allows the user to construct and export a wide range of faces and expressions using an intuitive interface. However, the main advantage is that the exported shapes have the same number of vertices and vertex indexing scheme. In other words, the shapes exported by FaceGen have consistent vertex registration and can be blended simply by using Equation 1. The FaceGen blendshape set includes 17 speech related shapes, which we label $B_1, B_2, ..., B_{16}$. Blendshape $B_0$ corresponds to silence and shows the mouth closed. The second appendix shows a complete set of blendshapes for one of our face models.

A number of other commercial packages can be used to produce desired blendshapes, such as[Curious Labs Inc., ]. Using a different set of blendshapes requires either changes in the code or a mapping between the two sets blendshapes.

### 4.2 Visual Speech Modeling

To animate the motion of the face that corresponds to speech, we use a *viseme* approach. Visemes refers to the mouth shape that corresponds to a phoneme. Typically, more than one phonemes corresponds to a single shape (viseme). Most approaches consider approximately 40 phonemes (including the silence phoneme), and the ratio of visemes to phonemes is about 1:3. In order to keep the system independent of the speech API employed, we make use of a dynamic mapping from phonemes to visemes. Visemes are constructed by blending together one or more blendshapes to mimic the closest possible real-life viseme.

The first appendix shows the mapping from the set of phonemes to linear combinations of blendshapes (visemes). Although most phonemes correspond to a single blendshape, some require a linear combination of two or more blendshapes. In some cases, phonemes require a motion transition between two or more blendshapes. Linguists refer to this latter situation as diphthongs. For example, when pronouncing "bite", the mouth moves while pronouncing the "ai" phoneme corresponding to the letter "i".

Once the sequence of phonemes and the sequence of blendshape weights are computed, a coarticulation rule filter is applied. This filter obeys the rules as described in Section 3. The result of the filter is an enhanced sequence of blendshapes (and corresponding weights) with an increased coarticulation accuracy. To produce smooth final motion, we interpolate between blendshape weights using a function that smoothly accelerates and decelerates between one viseme to the next.

### 4.3 Speech API

Our text-to-speech module is based on the Microsoft Speech SDK Version 5 (MS SDK) [Microsoft, Inc., ], a publicly available toolkit. Given an input string, the MS SDK produces both the corresponding audio and a sequence of phonemes along with their duration. The MS SDK is simple to use and actively developed. The voice used to speak while animating the face can be selected from the

| Rule | Direction | Sources | Targets |
|---|---|---|---|
| Overwrite-H | backward | Vowels | {H} |
| OO-L | backward | {OO} | {L} |
| Round-Vowels-bw | backward | {AO, OW, OY, UW} | {S, Z, T, K, D, G} |
| Round-Vowels-fw | forward | {AO,OW,OY,UW} | {S, Z, T, K, D, G} |
| W | backward | {W} | {AI, AA, AE, K, G, T, D} |

**Figure 2:** *Coarticulation rules used in our system.*

available voices from the MS SDK. However, we have found that the set of voices available from AT&T [ATT Inc., ] provides greater realism.

In addition, the Speech SDK supports natural language processing (NLP). It is fairly easy to process audio input from a microphone and extract the associated text information. The textual content can be further processed based on a user defined grammar. The grammar specification and processing API allows the user to link the grammatical rules to arbitrary functions that can be executed when the rules are succesffuly applied to the input text.

The Microsoft Speech SDK is a free and publicly available option. However, there is a number of available toolkits that can be used as well, ranging from experimental research code to sophisticated commercial products.

### 4.4 Blending Expression with Speech

The original expression shapes from FaceGen are used as the default shapes for their corresponding emotions. However, when adding expression shapes to existing visemes, unwanted results may appear due to different shapes competing to change the same region of the face. For example, the anger blendshape provided by FaceGen has an open mouth with clentched teeth. This shape will conflict with all visemes that require the mouth to be closed when mouthing the phonemes “b”, “m”, and “p”. We resolve this conflict by another set of rules called expression constraints. An expression constraint specifies which blendshapes to use and how much the weights should be for a given pair of phoneme and expression. In the absence of an expression constraint rule, the default shapes and weights will be used. This method gives the user the flexibility to show the same emotion while applying different expression shapes with speech-related shapes. This implementation allows us to use half-shapes which we created from the original expression shapes to obtain an unchanged mouth region ready for visemes integration.

### 4.5 Composition of Audio and Video

Once the input text string has been parsed by the MS SDK, we generate an appropriate viseme sequence (a set of weighted blendshapes) that is enhanced by the coarticulation rule filter. The MS SDK offers important timing information that indicates the duration of each phoneme. Our system produces a viseme sequence that will obey the constraints defined by the timing of the audio sequence. Once the final timing of the viseme sequence is computed, we offer the ability to replay the audio sequence and the synchronized viseme sequence simultaneously.

### 4.6 Animation System

The proposed system is developed on top of the Dynamic Animation and Control Environment (DANCE) which is publicly available [Shapiro et al., 2005]. DANCE offers the standard functionality that one needs to have in a research tool, such as window management etc. Its plug-in API allows complex research projects to link with DANCE and make use of other people’s work. We plan to link our speaking head model to the human model and corresponding simulator provided by DANCE. However, the implementation of our system is not dependent on any DANCE-specific features. It can become part of any animation system.

## 5 RESULTS

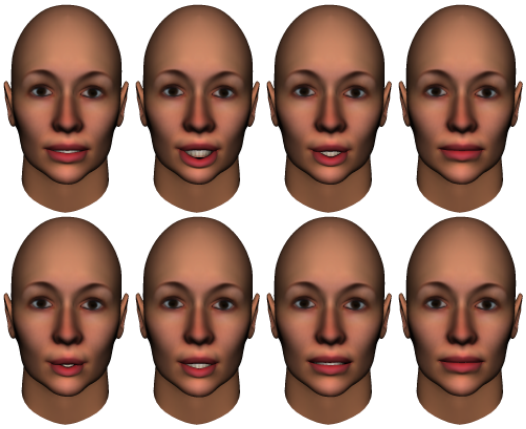

**Figure 3:** *Snapshots of a 3D face model saying “soon” without coarticulation (top) and with coarticulation (bottom). Images in raster order.*

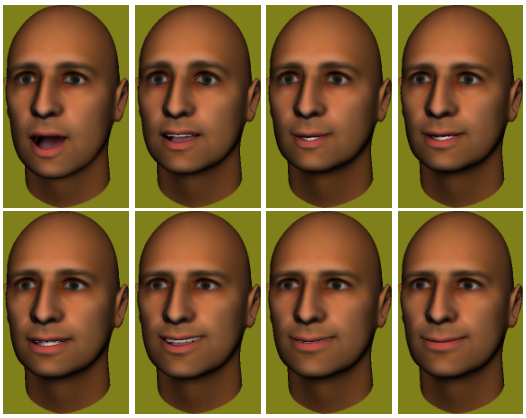

**Figure 4:** *Snapshots of 3D face model saying “Hi Susie” with expression control. Images in raster order.*

We assembled an expressive facial animation system using publically available components, and here we demonstrate its effectiveness with two applications: a) a text-to-speech synthesizer with expression control and b) a conversational agent that can react to simple phrases.

The expressive text-to-speech synthesizer uses a tagged text input

and produces the corresponding facial animation. The virtual actor speaks the input sentence with effects of coarticulation while showing the indicated emotion(s). Examining the results of our coarticulation processor, Figure 3 illustrates the efficacy of the set of rules that we provide with the proposed system. The virtual actor is speaking the word "soon." In this case, the vowel that sounds like "oo" has a backward influence over the "s" phoneme. The coarticulation rules that we have provided will cause the viseme associated with the "s" phoneme to blend with the viseme that is associated with the "oo" phoneme. In Figure 3, note the difference between the animation sequence before and after undergoing the coarticulation processor by comparing the first few frames of the top and bottom row of frames. At the bottom row the mouth properly prepares to form the "oo" viseme while it is in the early stages of the viseme that pronounces the "s" at the beginning of the word.

The expression tags in a user input turn into smiles and frowns on the virtual actor's face while he or she is speaking. Figure 4 shows a few snapshots of a face model speaking the phrase "*Hi Susie*" with emotion control. The input to the system is the text and the corresponding expression tags: "`<Surprised>Hi <Happy> Susie`". As expected the character starts with a suprised "Hi" and continues with a happy expresssion. Notice how the smile that corresponds to the happy expression is properly mixed with the speech. The set of expressions along with the method of transition and blending can be expanded or simplified, depending on the user's budget and intent. The modularity of our approach allows for customized components.

The second application we demonstrate is a primitive conversational avatar who reacts to basic greetings and comments. The user types phrases in the text window and the agent responds. The response of the agent depends on the phrase received, as well as her emotional state. For example, she becomes happier as she receives more praises. Since the input is a string of text, a speech-to-text component such as the speech recognition module from Microsoft Speech SDK can be utilized.

For a better demonstration of our results, we refer the reader to the accompanying video.

## 6 DISCUSSION

Our system is based on integrating robust, publicly available components and APIs. The modular architecture of our system allows us to change or upgrade each component separately.

FaceGen allows us to create arbitrary face models, and export a consistent and perfectly registered set of blendshapes for face model. Given a new set of blendshapes, all we have to do is change a few file names in a script and the new face is ready for use. In addition, we could construct blendshapes of higher quality, such as those used in[Joshi et al., 2003]. Publicly available speech APIs or expensive commercial ones can be used to create the audio and the phoneme segmentation, depending on the needs of the application at hand.

The proposed text-to-speech/facial motion system can be used in a wide range of applications. Any application that requires a speaking face can quickly get a prototype solution based on the proposed system. For example, instant messaging applications could use visual representations of the user's face that can speak the typed text. Considering that FaceGen and other software packages allow a user to create a 3D model of his/her face from photographs, we can imagine that such an application can have wide appeal. Our system can be used as a classroom tool for courses in linguistics, physchology, medicine, and computer graphics. For example, by altering the table of coarticulation rules on the fly, an instructor can demonstrate various coarticulation effects.

Although we do not focus on using an audio speech signal to drive the facial animation, it is a feature that could easily be added to the system. Both the Microsoft Speech SDK and open source software such as Festival[Speech Group, ] can segment speech into a phoneme sequence which is the only necessary input to our facial animation system.

Currently, the expression tags do not affect the generated speech signal. The Microsoft Speech SDK supports XML annotation for controling pronunciation, speech rate and emphasis. It is relatively easy to have our expression tags affect the generated speech signal using the provided API.

## 7 CONCLUSION

We have presented an affordable, off-the-shelf system for text-to-speech and 3D facial animation. The power of the system comes from its simplicity, modularity, and the availability of its components. Our system is easy to implement and extend, and furthermore, it offers a quick solution for applications that require affordable virtual actors that can speak from text.

## Acknowledgements

This work was partially supported by the NSF grant CCF-0429983. We would like to thank Yong Cao and the anonymous reviewers for their comments. We would also like to thank Intel Corp., Microsoft Corp., Ageia Corp., and ATI Corp. for their generous support through equipment and software grants.

## References

Albrecht, I., Haber, J., Kähler, K., Schröder, M., and Seidel, H. (2002). May I talk to you? Facial animation from text. In *Proceedings, Tenth Pacific Conference on Computer Graphics and Applications*, pages 77–86.

ATT Inc. Natural Voices. `www.naturalvoices.att.com/`.

Blanz, T. and Vetter, T. (1999). A morphable model for the synthesis of 3d faces. In *SIGGRAPH 99 Conference Proceedings*. ACM SIGGRAPH.

Brand, M. (1999). Voice puppetry. In *Proceedings of ACM SIGGRAPH 1999*, pages 21–28. ACM Press/Addison-Wesley Publishing Co.

Bregler, C., Covell, M., and Slaney, M. (1997). Video rewrite: driving visual speech with audio. In *SIGGRAPH 97 Conference Proceedings*, pages 353–360. ACM SIGGRAPH.

Brook, N. and Scott, S. (1994). Computer graphics animations of talking faces based on stochastic models. In *International Symposium on Speech, Image Processing, and Neural Networkds*.

Cao, Y., Faloutsos, P., and Pighin, F. (2003). Unsupervised learning for speech motion editing. In *Proceedings of Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, pages 225–231.

Cao, Y., Tien, W. C., Faloutsos, P., and Pighin, F. (2005). Expressive speech-driven facial animation. *ACM Trans. Graph.*, 24(4):1283–1302.

Cassell, J., Pelachaud, C., Badler, N., Steedman, M., Achorn, B., Becket, W., Douville, B., Prevost, S., and Stone, M. (1994). Animated conversation: Rule-based generation of facial expression, gesture and spoken intonation for multiple conversational agents. In *Proceedings of ACM SIGGRAPH 1994*.

Cohen, N. and Massaro, D. W. (1993). Modeling coarticulation in synthetic visual speech. In Thalmann, N. M. and Thalmann, D., editors, *Models and Techniques in Computer Animation*, pages 139–156. Springer–Verlang.

Curious Labs Inc. Poser 6. `www.curiouslabs.com/`.

DeCarlo, D. and Stone, M.
`http://www.cs.rutgers.edu/ ~village/ruth`.

Ezzat, T., Geiger, G., and Poggio, T. (2002). Trainable videorealistic speech animation. In *Proceedings of ACM SIGGRPAH 2002*, pages 388–398. ACM Press.

Guenter, B., Grimm, C., Wood, D., Malvar, H., and Pighin, F. (1998). Making faces. In *SIGGRAPH 98 Conference Proceedings*, pages 55–66. ACM SIGGRAPH.

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*, pages 187–192. Eurographics Association.

Kalberer, G., Mueller, P., and Gool, L. V. (2002). Speech animation using viseme space. In *Vision, Modeling and Visualization*.

Kalberer, G. A. (2003). *Realistic Face Animation for Speech*. PhD thesis, Swiss Federal Institute of Technology Zurich.

Kent, R. D. (1977). Coarticulation in recent speech production models. *Journal of Phonetics*, 5:115–133.

Lee, Y., Terzopoulos, D., and Waters, K. (1995). Realistic modeling for facial animation. In *SIGGRAPH 95 Conference Proceedings*, pages 55–62. ACM SIGGRAPH.

Lundeberg, M. and Beskow, J. Developing a 3d-agent for the august dialogue system.

Massaro, D. (1997). *Perceiving Talking Faces*. MIT Press, Cambridge, MA.

Masuko, T., Kobayashi, T., Tamura, M., Masubuchi, J., and k. Tokuda (1998). Text-to-visual speech synthesis based on parameter generation from hmm. In *ICASSP*.

Microsoft, Inc. Speech SDK 5.1. `www.microsoft.com/speech/download/sdk51`.

Parke, F. (1974). *A parametric model for human faces*. PhD thesis, University of Utah, Salt Lake City, Utah. UTEC-CSc-75-047.

Pelachaud, C. (1991). *Realistic Face Animation for Speech*. PhD thesis, University of Pennsylvania.

Pighin, F., Hecker, J., Lischinski, D., Szeliski, R., and Salesin, D. (1998). Synthesizing realistic facial expressions from photographs. In *SIGGRAPH 98 Conference Proceedings*, pages 75–84. ACM SIGGRAPH.

Reveret, L., Bailly, G., and Badin, P. Mother : A new generation of talking heads providing a flexible articulatory control for video-realistic speech animation.

Saisan, P., Bissacco, A., Chiuso, A., and Soatto, S. (2004). Modeling and synthesis of facial motion driven by speech. In *ECCV 2004 (to appear)*.

Shapiro, A., Faloutsos, P., and Ng-Thow-Hing, V. (2005). Dynamic animation and control environment. In *Graphics Interface 2005*.

Singular Inversion, I. FaceGen. `www.facegen.com`.

Speech Group, C. M. U. `www.speech.cs.cmu.edu/festival`.

Stone, M., DeCarlo, D., Oh, I., Rodriguez, C., Stere, A., Lees, A., and Bregler, C. (2004). Speaking with hands: creating animated conversational characters from recordings of human performance. *ACM Transaction on Graphics*, 23(3):506–513.

Waters, K. (1987). A muscle model for animating three-dimensional facial expression. In *SIGGRAPH 87 Conference Proceedings)*, volume 21, pages 17–24. ACM SIGGRAPH.

## PHONEME MAPPING

The mappings from phonemes to blendshapes come in three varieties. In our interpretation of visemes, thirty-one out of the forty phonemes map to a single blendshape. From our experience, some visemes cannot be directly represented with a single blendshape from our set. However, a linear combination of the shapes proved sufficient. A mere four phonemes required the use of a linear combination of blendshapes. The speech articulators change shape during production of dipthongs. To account for these five phonemes we allow transitions between blendshapes, with a relative duration of each segment. For example, they "ay" sound in *bite* is gives a temporal weight of $70\%$ to $B_1$ and $30\%$ to $B_9$. The set of mappings used in our system are outlined in Figure 5.

## THE BLENDSHAPES

The blendshapes, $B_1, \ldots, B_6$ related to speech that FaceGen produces are shown in Figure 6 in raster order. The last row shows examples of facial expressions provided by FaceGen. The blendshape for silence, $B_0$, which shows the mouth closed is not shown.

| Phoneme(s) | Blendshape(s) |
|---|---|
| silence | $B_0$ |
| ah (cut), ax (ago) | $B_1$ |
| b (big), m (mat), p (put) | $B_2$ |
| ch (chin), jh (joy) | $B_4$ |
| sh (she), zh (pleasure) | $B_4$ |
| d (dig), s (sit), t (talk), z (zap) | $B_5$ |
| eh (pet), h (help)* | $B_7$ |
| f (fork), v (vat) | $B_8$ |
| g (gut), k (cut) | $B_{10}$ |
| n (no) | $B_{11}$ |
| ow (go) | $B_{12}$ |
| ee (feel) | $B_9$ |
| l (lid) | $B_{11}$ |
| uw (too) | $B_{13}$ |
| r (red) | $B_{14}$ |
| dh (then), th (thin) | $B_{15}$ |
| w (with) | $B_{16}$ |
| aa (father) | $0.5 \times B_1 + 0.5 \times B_3$ |
| ae (cat) | $0.3 \times B_6 + 0.7 \times B_3$ |
| ao (dog) | $0.7 \times B_1 + 0.3 \times B_{12}$ |
| ih (fill) | $0.2 \times B_7 + 0.8 \times B_9$ |
| ng (sing) | $0.5 \times B_{11} + 0.5 \times B_{10}$ |
| y (yard) | $0.8 \times B_9 + 0.2 \times B_6$ |
| aw (foul) | $(B_1)_{0.6} \rightarrow (B_{13})_{0.4}$ |
| ay (bite) | $(0.5 \times B_2 + 0.5 \times B_3)_{0.6} \rightarrow (B_9)_{0.4}$ |
| er (fur) | $(B_1)_{0.2} \rightarrow (B_{14})_{0.8}$ |
| ey (ate) | $(B_7)_{0.7} \rightarrow (B_9)_{0.3}$ |
| oy (toy) | $(B_{12})_{0.7} \rightarrow (B_9)_{0.3}$ |
| uh (foot) | $(0.5 \times B_1 + 0.5 \times B13)_{0.7} \rightarrow (B_1)_{0.3}$ |

**Figure 5:** *Phoneme to Blendshape mapping.*

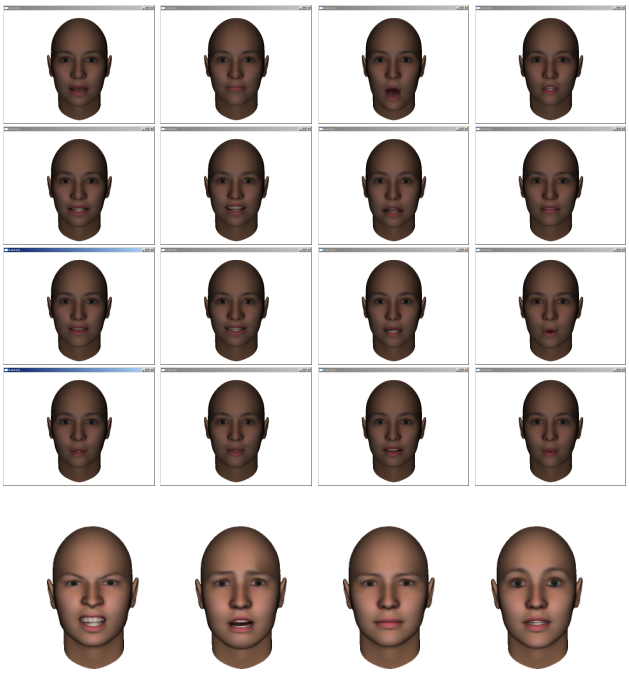

**Figure 6:** *The list of 16 viseme blendshapes, in raster order. The last row shows a few of the expression related blendshapes.*