SPEECH SYNCHRONIZATION FOR PHYSICS-BASED FACIAL ANIMATION

Irene Albrecht, Jörg Haber, Hans-Peter Seidel

Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken
Germany
{albrecht, haberj, hpseidel}@mpi-sb.mpg.de

ABSTRACT

We present a method for generating realistic speech-synchronized facial animations using a physics-based approach and support for coarticulation, i.e. the coloring of a speech segment by surrounding segments. We have implemented several extensions to the original coarticulation algorithm of Cohen and Massaro [Cohen93]. The enhancements include an optimization to improve performance as well as special treatment of closure and release phase of bilabial stops and other phonemes. Furthermore, for phonemes that are shorter than the sampling intervals of the algorithm and might therefore be missed, additional key frames are created to ensure their impact onto the animation.

Keywords: facial animation, lip sync, coarticulation

1 INTRODUCTION

For convincing animations of a talking head two things are important: first, tight synchronization between audible and visible speech is required. Human observers are very sensitive to slight misalignments due to the fact that important speech events like lip closure for /m/ may last as short as 5 msec. Second, the effect of coarticulation needs to be addressed. The term *coarticulation* refers to the influence of surrounding segments on the vocal tract shape of a phoneme. For example, the phoneme /k/ in 'coin' /kɔin/ and the phoneme /k/ in 'cow' /k*a*ʊ/ are quite distinct, because in 'coin', the lip rounding for the /ɔ/ does not begin with the /ɔ/ but already with the /k/. The influencing phonemes may be several segments apart and may even be separated by syllable or word boundaries (see [Blado76]). Therefore it is not sufficient to determine a fixed mapping from mouth positions to phonemes and simply interpolate between consecutive mouth positions. This paper presents an approach to generate visual speech including coarticulation based on the work by Cohen and Massaro ([Cohen93]). Our main contribution is the addition of special treatment of stops like /p/ and /b/ and of other phonemes for that a certain lip position is vital. Furthermore, phonemes that would be missed by the sampling due to their extremely short duration are especially provided for. The process of generating the animation is also sped up.

2 PREVIOUS WORK

Since the advent of talking heads in the early 70's, a lot of work has been done in the area of facial animation. The underlying head models can be classified crudely into three main categories: performance-based, parameterized, and muscle-based models. For the performance-based technique, either a model consisting of a data base of pictures or short movies of the head is created. The entries of the data base are sequenced for animation (e.g. [Bregl97]). Hence the quality of the animation depends on the size of the data base. In parameterized models (e.g. [Cohen93]), the nodes of a face mesh are moved directly via parameters that bundle the movement of several nodes together in order to produce a certain effect as e.g. the opening of the mouth. *Muscle-based approaches* (e.g. [Lee95]) actually model the most important facial muscles, and in addition take into account the facial tissue, i.e. they are not only concerned with the movements and characteristics of the facial surface but with the underlying dynamics as well. The

Figure 1: Snapshots showing in the upper part from left to right the neutral face and the mouth positions for the vowel /ɔ/ and below the fricatives /f/ and /ð/ as in the English word 'the'. Note the lip protrusion and rounding for /ɔ/. To produce an /f/, the lower lip is tucked under the upper incisors. For /ð/, the tongue protrudes slightly between the teeth.

main advantage of this method over the parameterized approach is the greater physical accuracy and hence naturalness of the animations. In contrast to the performance-based approach, this naturalness can be accomplished with fewer expenses and is more flexible. Therefore we decided to use muscle-based animation. We employ the muscle model proposed in ([Kähle01]). It is composed of a mass-spring model which includes skull and jaw, a muscle layer, and a facial tissue/skin layer. Bulging of skin resulting from muscle contractions is modeled in a convincing and natural way. The muscles may be attached to one another causing a muscle to be moved along with the one that attaches to it when this muscle contracts. The muscle encircling the mouth which is responsible for e.g. lip rounding (called *orbicularis oris*) is modeled as an upper and a lower part to allow for separate movement of the upper and lower lip. This makes a very flexible animation of the mouth possible.

2.1 Speech Animation for Talking Heads

Facial animation can be synchronized to speech in several ways. The employed method depends mainly on the kind of speech data which is available for synchronization, e.g. whether the audio signal, the phonemes and timing of an utterance is available or only a text representation.

The *text-driven* approach (e.g. [Water93, VlllB96]) receives a text as input which is transcribed into its phonemic representation. This information is used to generate both synthetic audible speech and synchronized visible speech.

The *speech-driven* method (e.g. [Kshir00]) takes prerecorded speech as input. The audio file is analyzed for phonemes and timing information. This data is used to create the facial animation which is performed synchronously to the audio file playback.

If both text and speech are available, a *text-and-speech-driven hybrid* approach (e.g. [Moris93]) can be applied. The text and its phonemic representation are used to identify segment boundaries and to gain timing information for the animation component, possibly aided by some rules for phoneme durations.

In the above approaches the lip movements are determined using predefined goal lip shapes for the phonemes. These are obtained by measurements or by simple observation. With the *performance-and-speech-driven hybrid approaches* (e.g. [Kurat98]) markers on the face or the lips of a person reciting the script are tracked and the postures are mapped onto the synthetic mouth; i.e. correct timing and coarticulation effects are inherent in the lip movement data.

2.2 Coarticulation

Apart from the approach for modeling coarticulation by Cohen and Massaro ([Cohen93]) that forms the base of our work, several other techniques exist: Pelachaud, Badler, and Steedman (cf. [Pelac96]) proposed a three-step algorithm to compute coarticulation. They cluster the consonants according to the context-dependence of their lip shapes, i.e. their deformability. 's' and 't' for example are highly context-dependent while 'p' and 'f' are not. First Pelachaud et al. apply coarticulation rules to the highly context-dependent groups to adjust the lip shapes of these clusters to be in accordance with the next vowel with low deformability. If the duration of a segment is smaller than the contraction or relaxation time of the muscles, the mouth shape of the previous or following phoneme is modified accordingly. Finally, the initial position from which a movement starts is taken into account, i.e. lip closure is harder to achieve from a wide open mouth than from slightly parted lips. The magnitude of the speech action is rescaled according to its context.

The concatenative approach can capture a person's individual speaking style (e.g. [Bregl97]). A database of several successive phonemes (called polyphones) and the corresponding mouth positions (the visual counterparts of the polyphones called visemes, or polysemes if there is a succession of them) articulated by this person is compiled. To synthesize a sentence, the polysemes are either simply concatenated (e.g. [Hällg98]) or for every three

phonemes, the viseme triple whose context is most similar to the current one is retrieved. In order to stitch the trisemes together, they are first temporally and then spatially aligned and combined with the face. Another approach to coarticulation is to move a sliding window of several phonemes over the phoneme sequence (e.g. [Cosat98]). The viseme for the phoneme in the middle of the window is retrieved from the respective polysemes in the data base. The visemes are then concatenated to form a continuous animation. The main disadvantage of the concatenative approach is the dependence on a data base. Either it is huge or it cannot contain every possible polyseme from every possible context. If polysemes from a wrong context are used because there is no data base entry for the correct context, the result may be irritating.

Of course it is always possible to track the features of a person using e.g. LEDs or other markers or electromyography and to apply the measured movement to the face model (e.g. [Kurat98]). However, this approach is inflexible, because changes in the script require that the features of the person are tracked anew which is a complicated process.

Kshirsagar and Magnenat-Thalmann [Kshir00] consider coarticulatory effects on vowel-to-vowel and vowel-to-consonant transitions. A timed phoneme sequence is obtained from the acoustic speech record using linear prediction analysis ([Lewis91]). In doing so, the average energy of the signal is computed. For the articulation of consonants, the vocal tract is usually constricted at some point, resulting in a diminution of the energy of the speech signal. Therefore the vowel lip shapes and jaw rotations are generated from reference facial parameters using the average energy of the acoustic signal as a weighting function. This approach is able to recognize most phonemes, however it cannot detect explicitly some kinds of stops as well as voiced fricatives, like /ʒ/ in 'azure'.

Furthermore, Hidden Markov Models (HMMs) are used to deal with coarticulation effects. Brand and Shan ([Brand98]) for instance implemented a system that drives facial animation directly from the audio signal using a HMM that learns a mapping between audio and video. Because HMMs learn the characteristics of a single person only a special HMM must be trained for every person.

2.3 The Approach by Cohen and Massaro

Based on the articulatory gesture model by Löfqvist ([Löfqv90]), Cohen and Massaro derived a method to compute the behavior of the articulators (e.g. lips or vocal chords) during speech that includes coarticulation ([Cohen93]). The approach takes a phoneme transcript and the timing information as input and computes key frames at specified regular intervals. This approach is still state of the art and has been included into a teaching aid for profoundly deaf children ([Cole99]). The central idea is that each speech segment has a certain time-varying degree of dominance over every articulator. This dominance is expressed by means of a *dominance function* for every articulator-phoneme pair. It describes the influence of the segment on the behavior of the articulator at any point in time. The function is used as a weight in determining how close an articulator gets to reaching its goal position and which position it takes at a given time. The articulators are not represented directly but by so-called *facial control parameters* as 'lip rounding' or 'tongue tip position'.

Because segments can influence each other (which results in coarticulation), the dominance of a segment does not automatically cease at the segment boundary but can well reach into other segments. Thus gestures of neighboring segments may overlap and so may the corresponding dominance functions.

According to Cohen and Massaro's method, the behavior of a facial control parameter p over time can be determined as follows: if $D_{s,p}$ is the function describing the dominance of segment s over p and $T_{s,p}$ is the goal position of parameter p for s, the function describing the behavior F_p of p over time is given as the weighted average of the targets of p during the whole utterance:

$$F_p(t) = \frac{\sum_{s=1}^{N}(D_{s,p}(t) \cdot T_{s,p})}{\sum_{s=1}^{N} D_{s,p}(t)}, \qquad (1)$$

where N is the number of segments in the utterance.

A possible dominance function is e. g. the negative exponential function

$$D_{s,p}(t) = \alpha_{s,p} e^{-\theta_{\pm,s,p}|\tau(t)|^c}, \qquad (2)$$

where

$$\tau(t) = t^s_{\text{start}} + \frac{t^s_{\text{dur}}}{2} - t^{s,p}_{\text{off}} - t \qquad (3)$$

t^s_{start} indicates the starting time of segment s and t^s_{dur} is its duration, i.e. $t^s_{\text{start}} + \frac{t^s_{\text{dur}}}{2}$ is the center of s. As the peak of the dominance function of s over facial control parameter p need not necessarily be at the center of the segment, a parameter $t^{s,p}_{\text{off}}$ describing the time offset from the center to the peak of domination can be specified. Therefore at $t^s_{\text{start}} + \frac{t^s_{\text{dur}}}{2} - t^{s,p}_{\text{off}}$, the influence of the segment on the articulatory parameter is most pronounced. Hence $\tau(t)$ denotes the time distance from the peak of the dominance function.

The parameter $\alpha_{s,p}$ determines the magnitude of the dominance function. It describes how important a parameter is in comparison to the others.

The dominance function's increase and decrease are modified by the parameter $\theta_{\pm,s,p}$, which describes the rate at which the function rises and falls. $\theta_{\pm,s,p}$ can have distinct values $\theta_{+,s,p}$ and $\theta_{-,s,p}$ for increase ($\tau(t) \geq 0$) and decrease ($\tau(t) < 0$), respectively. Therefore, the function can have a different slope for each case, allowing for differences in forward (the segment is colored by a successive phoneme) and backward (the segment is colored by a preceding phoneme) coarticulation.

Variations in parameter c change the characteristics of the transition between adjacent segments. When c increases, the values for the articulators are more likely to hit their goal positions while at the same time there is an overall decrease of coarticulatory effects. Moreover, the transitions between the segments become more abrupt.

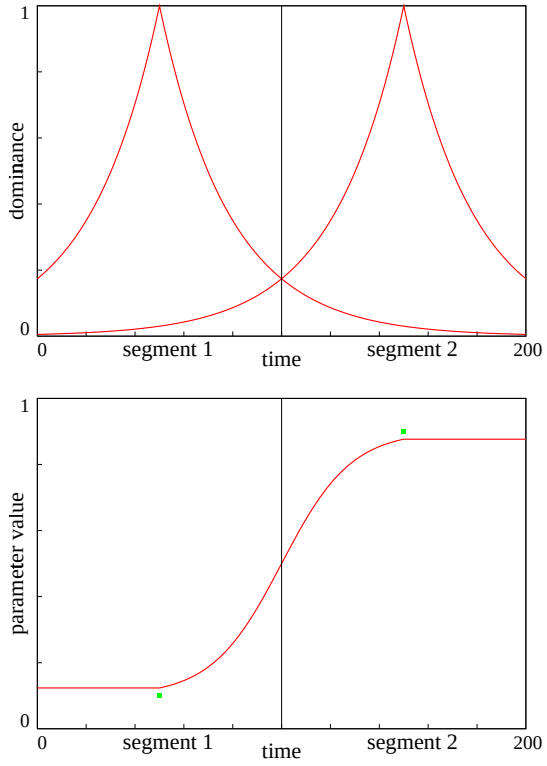

Figure 2: Dominance functions (Eq. 2) of a facial parameter for two neighboring segments (top) and the resulting behavior of the parameter computed using Eq. 1. The squares denote the target values for the parameter.

Fig. 2 shows an example for the dominance functions in Eq. 2: the upper part contains the dominance functions of a parameter for two consecutive segments. Below is the resulting parameter behavior for target values of 0.1 for the first and 0.9 for the second segment.

This algorithm is easy to implement and the results are convincing. Memory usage is low, no training of neural networks is required, and generation of the animations is fast.

3 OUR APPROACH

As input our lip sync system accepts a timed phoneme string for synchronization and computation of the animation. The phonemes and their timings are extracted from an audio file manually by an expert, because today's automated labeling software is not yet completely reliable. In order to generate the animation we implemented several extensions to the coarticulation algorithm by Cohen and Massaro ([Cohen93]). The animation is then synched to the audio.

In order to adjust the original coarticulation algorithm by Cohen and Massaro to the muscle-based approach proposed in [Kähle01] we use the muscle contraction, jaw and tongue movement values directly instead of the abstract facial control parameters employed by Cohen and Massaro. Therefore, the dominance functions now describe the influence of a segment on the amount of muscle contraction or jaw and tongue movement instead of on the facial control parameters.

3.1 Optimizing the Computation of Coarticulation

Coarticulation can affect movements belonging to phonemes up to six segments away ([Kent77]). In theory, one can imagine coarticulation to stretch over even longer periods than seven segments, but this is the greatest expansion known so far for coarticulatory effects. Yet the original algorithm by Cohen and Massaro considers the effects of coarticulation of every segment on all other segments in the whole utterance. Especially for longer utterances comprising several sentences this is unnecessary computational overhead. Therefore in our implementation, the algorithm only considers the six following and preceding segments of a segment s, $s_{-6}, ..., s_{-1}, s, s_{+1}, ..., s_{+6}$, plus two additional segments s_{-7} and s_{+7}. In the seventh segment, the function is led to 0 using cubic Hermite interpolation in order to restrict the support of the dominance functions to seven segments on each side of s and to ensure continuity. The dominance function describing the influence of segment s over muscle m then assumes the following shape:

$$D_{s,m}(t) = \begin{cases} h_0^1(t), & \text{if } t \in s_{-7}, \\ \alpha_{s,m} e^{-\theta_{\pm,s,m}|\tau(t)|^c}, & \text{if } t \in \bigcup_{i=-6}^{+6} s_i, \\ h_2^3(t), & \text{if } t \in s_{+7}, \\ 0, & \text{else,} \end{cases} \tag{4}$$

where

$$\begin{aligned} h_i^j(t) \quad = \quad & H_0^0(t) \cdot P_i + H_0^1(t) \cdot P_i' \\ & + H_1^0(t) \cdot P_j' + H_1^1(t) \cdot P_j. \end{aligned}$$

P_0, P'_0 and P_1, P'_1 are the dominance values and their derivatives at the left and right border of segment s_{-7}. In order to smoothly fade $D_{s,m}$ to zero at the left border of s_{-7}, we set $P_0 = P'_0 = 0$ and $P_1 = D_{s,m}(s^{\text{left}}_{-6})$, $P'_1 = D'_{s,m}(s^{\text{left}}_{-6})$. Respectively, P_2, P'_2 and P_3, P'_3 are the values and derivatives at the borders of segment s_{+7}. Here, $P_2 = D_{s,m}(s^{\text{right}}_{+6})$, $P'_2 = D_{s,m}(s^{\text{right}}_{+6})$ and $P_3 = P'_3 = 0$. s^{left}_x and s^{right}_x are the left and right boundary of segment s_x.

This leads to the following function describing the behavior of the muscle m over time:

$$F_m(t) = \frac{\sum_{s=min(0,c-7)}^{max(N,c+7)} (D_{s,m}(t) \cdot T_{s,m})}{\sum_{s=min(0,c-7)}^{max(N,c+7)} D_{s,m}(t)} \qquad (5)$$

instead of Eq. 1. Again, N is the total number of phonemes in the utterance, and c denotes the position of the phoneme belonging to the current instance of time, t.

The possible point of discontinuity at the peak of the original dominance function which depends on the dominance function used is not a problem, because the derivative is only computed at the outward segment boundaries. It is very unlikely that the peak of the dominance function should coincide with either segment boundary, because the maximum is usually reached during the duration of the segment. If the peak *should* coincide with the boundary in question, the derivative of the descending part of the function is chosen as P'_1 in case of a left boundary, and the derivative of the ascending part as P'_2 in case of a right boundary. This is possible because the respective other part of the function will not come into play at all, and because both parts of the function taken separately are differentiable.

The original method computes for each sample the dominance function of every phoneme in the utterance. The modified method computes at most the dominance functions of 15 phonemes for each sample, i.e. of the current phoneme and the 7 preceding and following phonemes. If for example s samples are taken per second, the utterance is t seconds long and contains n segments, then the original algorithm computes $s{\cdot}t{\cdot}n$ dominance functions, whereas the modified version computes less than $s{\cdot}t{\cdot}15$ dominance functions. For longer utterances this makes a noticeable difference.

3.2 Closure and Release Phases

For the production of the bilabial stops /p/ and /b/ it is vital that the lips are fully closed. This is also the case for the nasal /m/. For the fricatives /f/ and /v/, the lower lip must touch the teeth. That these target positions are reached exactly is not only important for the production of the sound but also for its visual perception. For the stops /b/ and /p/, the lips are held closed for a certain (usually very short) time interval, the so-called *closure* phase. During the following *release*, the lips burst open to let the retained air rush out. Although the closure can be as short as 5 msec, it is used as a visual cue during speech. If the timing of the audible and visible speech is not perfectly tuned, this is detected quite easily and considered irritating. It is similar for the two fricatives mentioned above. The exact timing of the lips touching the front teeth is essential for a good lip sync. But here the "closure" is not complete, the air is pressed through the space between the teeth. Therefore, a release phase does not exist. For the production of /m/, the lips are fully closed but the velum is lowered, thus opening the connection between the oral and nasal cavity. Through this passage, the air can escape, again obviating the need for a release phase. Our system considers these facts by modeling the closure and release phase of the stops /b/ and /p/ separately. Because we do not model air pressure, the fricatives /f/ and /v/, and the nasal /m/ are handled in the same way as the bilabial closure.

Usually, samples are taken at intervals of uniform size. At the beginning of the closure of a phoneme, however, a key frame is always generated. By assigning a high magnitude to the dominance function of the closure phoneme, the lips come sufficiently close to their targets. The next key frame is computed at the beginning of the release phase, if existent, or at the start of the next phoneme. Here, the normal procedure with equidistant time steps is resumed (Fig. 3).

The peak of the dominance function of the release visemes is always set to the beginning of the corresponding interval in order to simulate the rapid opening of the lips after a closure.

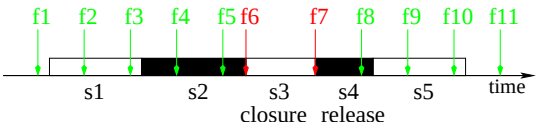

Figure 3: Distribution of key frames (f_n) in a word containing both closure (s_3) and release (s_4) phase of a phoneme. To capture the complete closure of the mouth during the closure phase, a sample is taken at the beginning of the closure (*f6*) and also at its end (*f7*). Afterwards, the normal equidistant sampling process is resumed (*f8* ff.). Complete lip closure is not only vital for the production of these phonemes, but also for their perception.

3.3 Handling of Very Short Visemes

In practice, we found a sampling rate of 20 frames per second to be reasonable, but the sampling frequency can be chosen for every animation anew.

As phonemes can last considerably shorter than 0.05 sec, the duration interval of a viseme can be shorter than the sampling intervals. If no sample is taken during a segment, this viseme would only appear indirectly via its coarticulation effects on the surrounding visemes, but make no impact as a single phoneme. This is undesirable, because important speech events might be missed. In the case of a viseme that is not a closure or a release, a frame is generated at the peak of its dominance function (Fig. 4). This is where the influence of the segment is most pronounced. Therefore even very short visemes make their appearance in the animation, and not only as coarticulatory trouble makers for their neighbors. If the phoneme is a closure or release it is treated as described in Section 3.2.

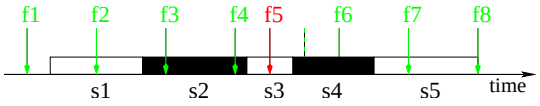

Figure 4: Distribution of key frames in a word containing a viseme (s_3) that would be missed using regular sampling. To avoid this, a key frame (*f5*) is inserted at the center of the phoneme. Subsequent samples are again taken at regular intervals (*f6* ff.).

4 RESULTS

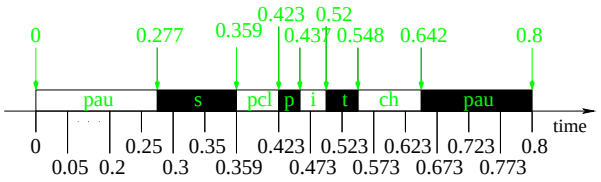

Figure 5: Time line for the utterance 'speech' /s pcl p i tcl tʃ/ at a sampling frequency of 20 fps. The black numbers below the time axis show the sampling points. The numbers above the time axis and phonemes denote the segmentation. The uniform sampling process is interrupted by a sample at 0.359 at the beginning of the closure interval /pcl/. The next sample is taken at the end of /pcl/ at 0.423. After that, the 0.05s sampling intervals are resumed.

Using our method, we are able to generate convincing animation for a natural looking face model that can be played back at 40 fps animation key frame rate and 120 fps rendering frame rate on an off-the-shelf dual processor PC (2x 1.7 GHz, GeForce 3 graphics card).

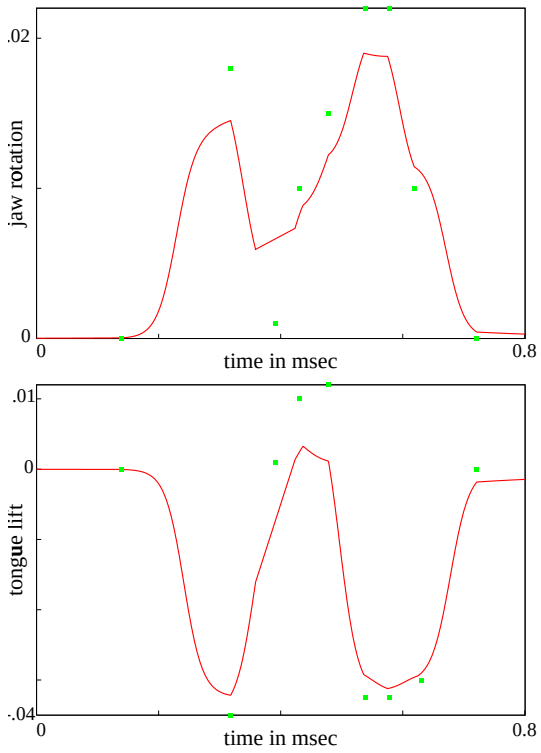

Figure 6: Up and down motion of jaw and tongue for the utterance 'speech' /s pcl p i tcl tʃ/. The target positions for each phoneme are marked by squares. The tongue moves up both for /t/ and /s/. Otherwise its movement is oriented at the jaw behavior. Although for the fifth segment, /i/, the tongue should be rotated downwards as the jaw, this is not the case due to coarticulation effects from the following /t/.

Due to the fact that the muscle model we use ([Kähle01]) allows the user to specify a set of muscles for one head model and then to adapt it semi-automatically to a different face, the parameters required for the coarticulation algorithm need to be specified only for the original head and can then either be adopted to the new one without modifications or adapted to it with only tiny modifications.

The timebar in Fig. 5 is intended to clarify the placement of key frames generated by the lip sync for a sequence of snapshots for the word 'speech' /s pcl p i tcl tʃ/ in Fig. 7. The corresponding parameter values for the movement of jaw and tongue are plotted in Fig. 6. Due to the fact that the closure for /tcl/ takes place inside the mouth and is hence not visible to the observer, we model /tcl/ like /t/.

5 CONCLUSIONS AND FUTURE WORK

All in all, the enhancements we propose for the coarticulation algorithm of Cohen and Massaro enable us to generate convincing lip sync in reasonable time, as can be seen in some example movies

at `http://www.mpi-sb.mpg.de/resources/FAM/`. However, in order to increase the realism of our synthetic visual speech, the incorporation of a more flexible tongue model is desirable. For /g/, for example, the tongue body must touch the palate. Because the current tongue model is rigid this is not yet possible. Even with an enhanced tongue model, the animation as a whole might still appear unnatural due to the lack of movement in other parts of the face. We are currently working to remedy this by automatically deducing prosody related eyebrow and head movement from the speech signal and including it into the animation. Preliminary results are very encouraging. In addition, we plan to link our lip sync system to a text-to-speech system.

REFERENCES

[Blado76] R. A. W. Bladon and A. Al-Bamerni. Coarticulation resistance in English /l/. *Journal of Phonetics*, 4:135–150, 1976.

[Brand98] M. Brand and K. Shan. Voice-Driven Animation. In *Proc. Workshop on Perceptual User Interfaces*, 1998.

[Bregl97] Ch. Bregler, M. Covell, and M. Slaney. Video Rewrite: Driving Visual Speech with Audio. In *Proc. SIGGRAPH '97*, pages 353–360, 1997.

[Cohen93] M. Cohen and D. Massaro. Modeling Coarticulation in Synthetic Visual Speech. In *Models and Techniques in Computer Animation.*, pages 139–156. Springer Verlag, Tokyo, 1993.

[Cole99] R. Cole, D. Massaro, J. de Villiers, B. Rundle, K. Shobaki, J. Wouters, M. Cohen, J. Beskow, P. Stone, P. Connors, A. Tarachow, and D. Solcher. New Tools for interactive speech and language training: Using animated conversational agents in the classrooms of profoundly deaf children. In *Proc. ESCA/SOCRATES Workshop on Method and Tool Innovations for Speech Science Education*, 1999.

[Cosat98] E. Cosatto and H. Graf. Sample-Based Synthesis of Photo-Realistic Talking Heads. In *Computer Animation*, pages 103–110, 1998.

[Hällg98] Å. Hällgren and B. Lyberg. Visual Speech Synthesis with Concatenative Speech. In *Proc. AVSP '98*, 1998.

[Kähle01] K. Kähler, J. Haber, and H.-P. Seidel. Geometry-Based Muscle Modeling for Facial Animation. In *Proc. GI '01*, pages 37–46, 2001.

[Kent77] R. D. Kent and F. D. Minifie. Coarticulation in Recent Speech Production Models. *Journal of Phonetics*, 5:115–133, 1977.

[Kshir00] S. Kshirsagar and N. Magnenat-Thalmann. Lip Synchronization Using Linear Predicitve Analysis. In *Proc. IEEE International Conference on Multimedia and Expo*, August 2000.

[Kurat98] T. Kuratate, H. Yehia, and E. Vatikiotis-Bateson. Kinematics-Based Synthesis of Realistic Talking Faces. In D. Burnham, J. Robert-Ribes, and E. Vatikiotis-Bateson, editors, *Proc. AVSP'98*, pages 185–190, 1998.

[Lee95] Y. Lee, D. Terzopoulos, and K. Waters. Realistic face modeling for animation. In *Proc. SIGGRAPH '95*, pages 55–62, 1995.

[Lewis91] J. Lewis. Automated Lip-Sync: Background and Techniques. *Journal of Vizualisation and Computer Animation*, 2(4):118–122, 1991.

[Löfqv90] A. Löfqvist. Speech as Audible Gestures. In W. J. Hardcastle and A. Marchal, editors, *Speech Production and Speech Modelling*, pages 289–322. Kluwer Academic Publishers, 1990.

[Moris93] S. Morishima and H. Harashima. Facial Animation Synthesis for Human-Machine Communication Systems. In *Proc. 5^{th} International Conference on Human-Computer Interaction*, pages 1085–1090, 1993.

[Pelac96] C. Pelachaud, N. I. Badler, and M. Steedman. Generating Facial Expressions for Speech. *Cognitive Science*, 20(1):1–46, 1996.

[VlllB96] E. Vatikiotis-Bateson, K. G. Munhall, M. Hirayama, Y. Kasahara, and H. Yehia. Physiology-based synthesis of audiovisual speech. In *4^{th} Speech Production Seminar: Models and Data*, pages 241–244, 1996.

[Water93] K. Waters and T. Levergood. DECface: An Automatic Lip-Synchronization Algorithm for Synthetic Faces. Tech. Report 93-4, Cambridge Research Laboratories, 1993.

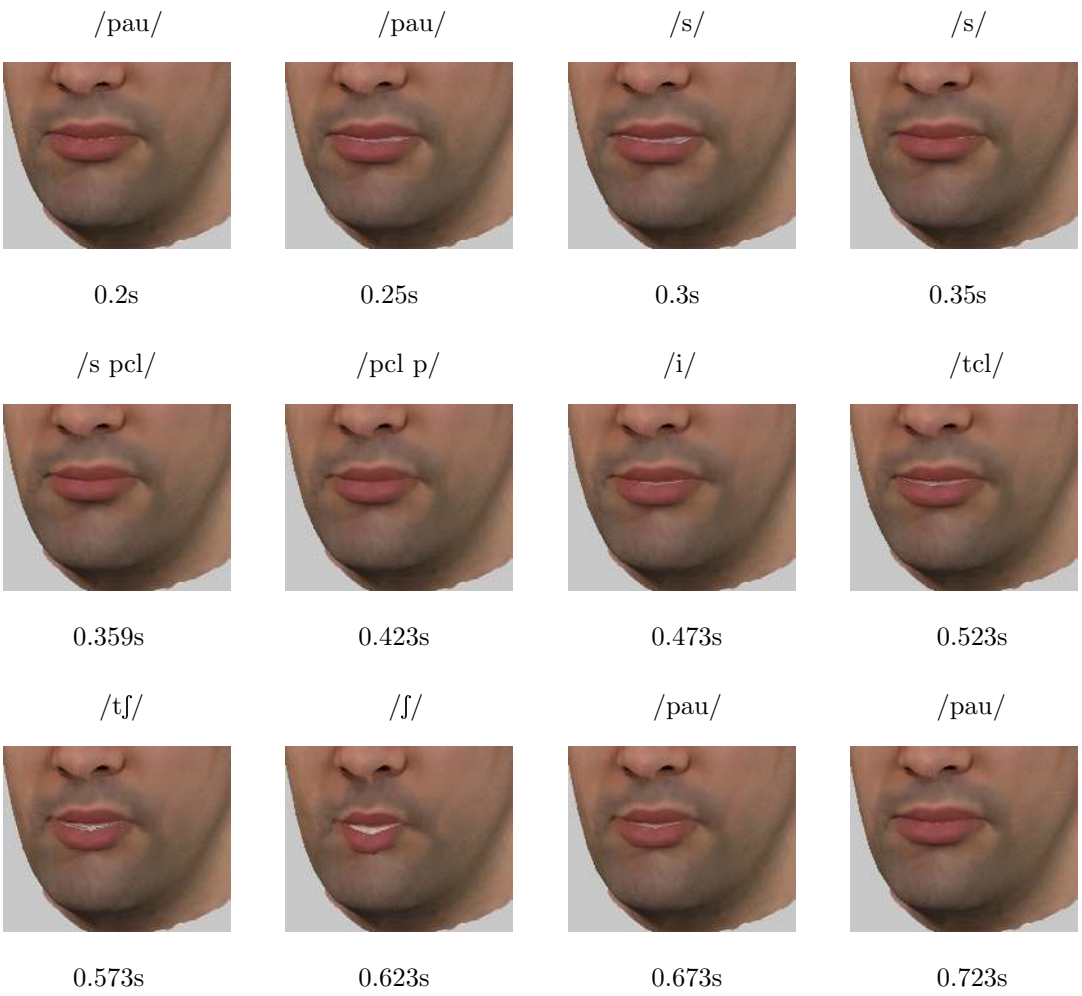

Figure 7: Snapshots of the animation for the utterance 'speech' /s pcl p i tcl tʃ/. /tcl/ is not modeled explicitly but as /t/, because the closure takes place inside the mouth and is not visible from the outside. The mouth opens already during the preceding pause for the following /s/. The effects of the /ʃ/ can also be seen in the subsequent pause due to coarticulation. For the same reason the lips are still puckered slightly during the /i/ because of the preceding /p/. As can be seen, in the frame preceding the closure, the lips are not closed entirely, but during the closure itself they are.