Lip-Synced Character Speech Animation with Dominated Animeme Models

Shuen-Huei Guan*[1,2], Yu-Mei Chen†[2], Fu-Chun Huang‡[3], and Bing-Yu Chen§[2]

[1]Digimax Inc.
[2]National Taiwan University
[3]University of California at Berkeley

Abstract

One of the holy grails of computer graphics is the generation of photorealistic images with motion data. To re-generate convincing human animations might not be the most challenging part, but it is definitely one of ultimate goals for computer graphics. Amongst full-body human animations, facial animation is the challenging part because of its subtlety and familarity to human beings.

In this paper, we like to share the work of lip-sync animation, part of facial animations, as a framework for synthesizing lip-sync character speech animation in real time from a given speech sequence and its corresponding texts.

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

Keywords: lip synchronization, speech animation, character animation, dominated animeme model, animeme modeling, coarticulation

1 Introduction

With the technological advance of computer graphics, and the popularity of 3D animations and vidoe games, realistic character animation is becoming more important. Facial and speech animation is still difficult to sculpt because of complexity of the correlation and interaction on the face. It is more challenging to have a character model's lips synchronized to the spoken speech, such that it is still a labor-consuming process, requiring even millisecond-precise key-framing.

Although several performance-driven approaches were proposed [Guenter et al. 1998][Ma et al. 2008][Weise et al. 2011], the captured performance is hard to re-use. Furthermore, the transitions, a.k.a. *coarticulation*, between words or phonemes, play a major role in speech animation and need to be adjusted carefully. *Coarticulation* indicates the situation in which a phoneme or speech sound is influenced by a preceeding or following ones. The mouth shape depends on not only the current phoneme, but also its context. Therefore, it is unavoidable to manual adjust the captured performance.

In this paper, a framework to synthesize lip-sync character speech animation in real time is proposed. For each phoneme, one or multiple **dominated animeme models (DAMs)** are first learned from a training set of speech-to-animation control signal (e.g. the character controls used in Autodesk Maya or cross-mapped mocap lip-motions). Then, in the synthesis phase, given a novel speech sequence, the **DAMs** are used to synthesize the corresponding speech-to-animation control signals, which in turn are used to generate the lip-sync character speech animation.

To summarize the contributions of this paper:

1. A framework is proposed to synthesize lip-sync character speech animation in real time.
2. DAM is presented to model *coarticulation* better.
3. Multiple DAMs are used to handle larger intra-animeme variations.
4. Instead of generating hard-to-fine-tune vertex deformations, high-level control signal of 3D characters is synthesized. That makes it easier to integrate into operating animation production pipeline.

2 Related Work

The advantage of the physically-based methods [Choe et al. 2001][Sifakis et al. 2005] over the parameterized/blend-shape ones is extensibility: the faces can be animated more realistically. However, the muscle-simulation is very expensive, and hence reduces the applicability to interactive controller. Data-driven approaches [Deng and Neumann 2006][Cao et al. 2004] form a graph for searching the given sentences. Nevertheless, they still suffer from missing data or duplicate occurrence. Parameterized techniques[Chuang and Bregler 2005] for speech animation are the most popular methods because of simplicity. Sifakis[Sifakis et al. 2006] presented a physically-based approach which can interact with objects while simulating, but the simulation cost is really high. Machine-learning based methods [Chang and Ezzat 2005][Deng et al. 2006][Kim and Ko 2007][Wampler et al. 2007] learn the statistics for phoneme-to-animation correspondences, to connect animation up to speech directly and reduce these searching efforts.

Some recent methods [Sifakis et al. 2006][Kim and Ko 2007][Wampler et al. 2007] used the concept of **animeme**, a shape function, to model the sub-viseme signal to increase the accuracy of phoneme fitting. Kim and Ko models the **viseme** within a smaller sub-phoneme range with a data-driven approach. *Coarticulation* is modeled via a smooth function in their regularization with the parameters found empirically. However, it has to resolve conflicting and insufficient records in the training set. Sifakiscite[Sifakis

*drake@cmlab.csie.ntu.edu.tw
†yumeiohya@gmail.com
‡jonash@eecs.berkeley.edu
§robin@ntu.edu.tw

et al. 2006] modeled the muscle-control-signal animeme (**physeme** in their work) for each phoneme, and concatenate these animemes for words. They found that each phoneme has various similar animemes with slight variations due to *coarticulation*, which is modeled with linear cross-fade weighting in a diphone or triphone fashion.

We learned from previous methods and improved the deficiencies in them. The analysis in the **animeme** space has significant improvements over the viseme analysis. In addition, we also solve for the hidden dominance functions, and extend *coarticulation* beyond the simpler diphone or triphone model. Moreover, the synthesis process is much simpler and faster because the models used for generating the results are trained in offline phase.

3 Dominated Animeme Models (DAMs)

Firstly, we need to model the relationship between phonemes (from a given text script) and the corresponding animation control signal $C(t)$, a.k.a. **animeme**, the animation representation of the phoneme. Due to *coarticulation*, we model the animation control signal $C(t)$ with the product of two functions: the animeme function and its dominance function. The animeme function $A(t)$ controls the intrinsic mouth shapes when used alone, and the dominance function $D(t)$ controls their individual influence and fall-off for a sequence of phonemes, the *coarticulation*. **Dominated animeme model (DAM)** is modeled as:

$$C(t) = D(t)A(t), \quad t \in [-\infty, \infty]$$

$$D(t) = \begin{cases} 1, & t \in [0,1] \\ \exp\left(\dfrac{-t^2}{\sigma^2 + \varepsilon}\right), & t < 0 \\ \exp\left(\dfrac{-(t-1)^2}{\sigma^2 + \varepsilon}\right), & t > 1 \end{cases} \tag{1}$$

Inspired by [Cohen and Massaro 1993], the dominance function is modeled as Eq. 2 where σ is the phoneme specific parameter affecting the range of influence, and ε is a small constant to prevent dividing by zero. Putting multiple phonemes together to get the full sequence of animation control signal, we simply concatenate these DAMs with the summation of their normalized values:

$$C^*(t) = \sum_{j=1}^{J} C_j(t_j) = \sum_{j} D_j(t_j)A_j(t_j), \tag{2}$$

where $j = 1, 2, ..., J$ indicates the j-th phoneme in the given phoneme sequence, and $t_j = (t - s_j)/d_j$ is the normalized local time for each phoneme activation, where s_j is the starting time-stamp of the j-th phoneme and d_j is its duration.

4 System Overview

The whole system is illustrated in Figure. 1 with two phases: training (left) and synthesis (right). In the training phase, the system takes as input the captured lip-motions or the animation control signal (e.g. animation data from Autodesk Maya) If inputs are from a speech video or 3D lip-motions captured by a mocap facility, the data in the vertex domain will be first cross-mapped to the control signal domain. Then, the speech and its corresponding texts are aligned with SPHINX-II [Huang et al. 1993] to obtain the aligned scripts (phoneme sequence), which contain phonemes with

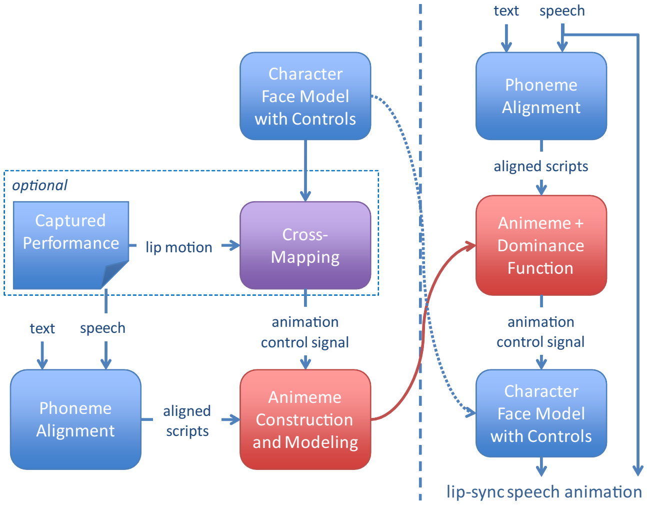

Figure 1: *System flowchart.*

their starting time-stamps and durations in the speech. The aligned scripts and animation control signal $C(t)$ are used as training examples to construct the DAMs for future novel speech animation synthesis.

In the synthesis phase, we take as input a novel speech and its corresponding texts, and use SPHINX-II again to obtain the aligned scripts. From the scripts, the DAMs are concatenated to generate the animation control signal C^*. Finally, the animation control signal C^* is used to animate the character (face) to generate the lip-sync character speech animation.

4.1 Estimating DAMs in training phase

According to the aligned scripts (phoneme sequence), every phoneme can have many corresponding animation control signals. Based on these training examples, we can construct the phoneme's DAM(s). However, we found it is difficult to decouple the animeme function and its dominance function gracefully if we construct a single DAM for each phoneme due to large intra-animeme variations. Instead, for each phoneme, multiple DAMs, or modes, are used.

Assuming each mode of each phoneme appears in the sequence exactly only once and denoting the j-th dominance function $D_j(i)$ at time i as a fixed value D_j^i, the estimation of the polynomial function $A_j(t)$ can be reduced to find the polynomial coefficients $a_j^0, a_j^1, ..., a_j^M$. Then, Eq. (2) can be rewritten as:

$$C(i) = \sum_{j=1}^{J} D_j^i \left[\sum_{m=0}^{M} a_j^m (t_j^i)^m \right], \tag{3}$$

where $t_j^i = (i - s_j)/d_j$ is the normalized local time-stamp from the activation of the j-th phoneme. In a regression manner, we can set the partial derivative of regression error $\mathbf{R}$ with respect to the m-th coefficient a_j^m for the j-th phoneme to zero. The least square fitting for regression is:

$$\begin{aligned} f_i &= C(i) - \sum_{j=1}^{J} D_j^i \left[\sum_{m=0}^{M} a_j^m (t_j^i)^m \right], \\ \mathbf{R} = \mathbf{F}^T\mathbf{F} &= \sum_{i=0}^{n} \left(C(i) - \sum_{j=1}^{J} D_j^i \left[\sum_{m=0}^{M} a_j^m (t_j^i)^m \right] \right)^2 , \end{aligned} \tag{4}$$

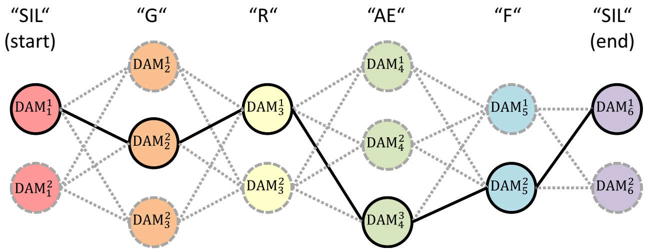

Figure 2: *An animeme-graph example for synthesizing "Graph". There are multiple DAMs (modes) for one phoneme (with the same color). The suitable sequence (denoted by solid circles and lines) is selected by A* algorithm.*

where $\mathbf{F}$ is the column-concatenated vector formed for each element f_i. Since the unknowns a_j^m are linear in $\mathbf{F}$, the problem is essentially a linear least-square fitting. By setting all partial derivatives to zero and arranging Eq. (4), we can obtain the following matrix representation:

$$\mathbf{D} = \begin{bmatrix} D_1^1(t_1^1)^0 & \cdots & D_1^1(t_1^1)^M & \cdots & D_J^1 & \cdots & D_J^1(t_J^1)^M \\ D_1^2(t_1^2)^0 & \cdots & D_1^2(t_1^2)^M & \cdots & D_J^2 & \cdots & D_J^2(t_J^2)^M \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ D_1^n(t_1^n)^0 & \cdots & D_1^n(t_1^n)^M & \cdots & D_J^n & \cdots & D_J^n(t_J^n)^M \end{bmatrix},$$

$$\mathbf{A} = \begin{bmatrix} a_1^0 & \cdots & a_1^M & \cdots & a_J^0 & \cdots & a_J^M \end{bmatrix}^T,$$

$$\mathbf{C} = \begin{bmatrix} C^0 & C^1 & C^2 & \cdots & C^n \end{bmatrix}^T,$$

where $\mathbf{D}$ is the dominance matrix, $\mathbf{A}$ is the coefficient vector we want to solve, and $\mathbf{C}$ is the observed values at each time i, so the minimum error to the regression fitting can be written in the standard normal equation with the following matrix form:

$$(\mathbf{D}^T\mathbf{D})\mathbf{A} = \mathbf{D}^T\mathbf{C}, \qquad (5)$$

Here, we can minimize the regression Eq. (4) with the same mechanism. Since the parameter σ_j for regression is non-linear, a Standard Gauss-Newton iterative solver is used to approach the minimum of the regression error $\mathbf{R}$. An EM-style strategy is employed to iterates between the estimation of the animeme function $A_j(t)$ and the optimization for the dominance function $D_j(t)$.

- The **E-step** involves estimating the polynomial coefficients a_j^m for each animeme function $A_j(t)$ by solving a linear regression using the standard normal equation.
- The **M-step** tries minimizing the regression error to estimate the non-linear dominance function $D_j(t)$.

4.2 Synthesizing with DAMs

In the synthesis phase, we want to generate the control signals according to the input phoneme sequence. Since some phonemes may have multiple modes, we have to decide which mode should be used for each phoneme. To construct the output animation control signal requires selecting the most suitable mode for each phoneme, and then directly use Eq. (2) to concatenate the DAMs in the sequence.

Giving a phoneme sequence $j = 1, 2, ..., J$ and possible modes DAM_j^g ($g = 1, ..., G_j$, where G_j is the number of modes) for each phoneme j, the animemes can form an animeme-graph as shown in Figure. 2. The selection of suitable modes for the phoneme sequence can be treated as a graph search problem, and A* algorithm is used in our implementation. Since we want to find a compromise between the likelihood of the modes and the smoothness in the animation, the cost of each node in the animeme-graph is set as:

$$E = w_c E_c + w_s E_s, \qquad (6)$$

where E_c is a data term, which represents the likelihood of the mode DAM_j^g in the training set linked with its previous and next phonemes, E_s is the smoothness term computing the C^2 smoothness on the joint frame of every DAM_j^g ($g = 1, ..., G_j$) of the current phoneme j and every DAM_{j-1}^g ($g = 1, ..., G_{j-1}$) of its previous phoneme $j-1$, and w_c and w_s are the weights of the error terms. We used $w_c = 1000$ and $w_s = 1$ for all results in this paper.

Table 1: *The models used in this paper and the accompanying video.*

model	vertex#	face#	control#
Afro-woman	5,234	5,075	7
Boy	6,775	6,736	7
Child	6,991	6,954	16
Old-hero	8,883	8,738	8
Court-lady	1,306	1,307	7

5 Result

The training set involves 80 sentences and about 10 minutes of speech context with unbiased content. In the training phase, constructing the DAMs costs about 50∼60 minutes per control on a desktop PC with an Intel Core2 Quad Q9400 2.66GHz CPU and 4GB memory. For synthesizing a lip-sync speech animation, the animation control signal formed by our DAMs are generated in real time (i.e. 0.8 ms. per phoneme on average). Table. 1 shows geometrical data used in this paper.

Figure. 4 shows a part of signal fitting for these results by continuous lip motions from left to right. According to the training data, the lips should be closed during the phoneme "P" and opened for other phonemes appropriately. At the last frame of the sequence, the mouth closes to prepare for the following sentence. The reconstruction result of the Cohen-Massaro model is too smooth at some parts, such that consecutive phonemes are greatly influenced, i.e. they span too much. The MMM formulates the fitting and synthesis as a regulation problem. by fitting each phoneme as a multidimensional Gaussian distribution. It soloves lip-sync problem by minimizing an energy function. The reconstructed speech by MMM has good timing but lack prominent features, while our results by DAM reach closer to the peaks of the training data.

6 Conclusion

With pre-processed learning phase, given a phoneme sequence, the DAMs are used to generate animation control signals, which can then be fed directly into Autodesk Maya or similar packages in real time. Even though the synthesized results may not be perfect, they can be easily fine-tuned interactively. Our current work focuses on stylized characters and our next step is to make it work for realistic talking heads.

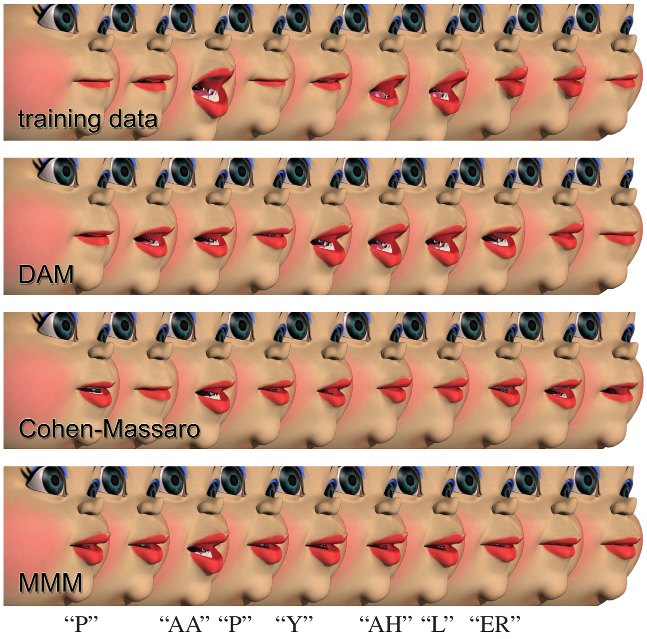

Figure 3: *The comparison of training data (the 1st raw) and the synthesized results of DAM (the 2nd raw), Cohen-Massaro model (the 3rd raw), and MMM (the 4th raw), while speaking “popular” by Afro-woman.*

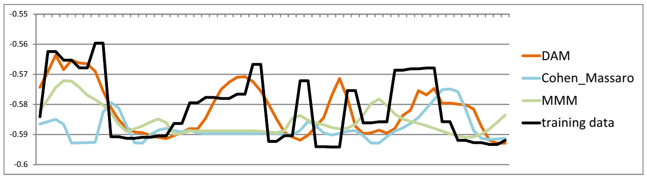

Figure 4: *The comparison of the signal fitted in Figure. 3 by DAM, Cohen-Massaro Model, and MMM with the captured one. The y-axis shows one of the coordinates of a control.*

References

CAO, Y., FALOUTSOS, P., KOHLER, E., AND PIGHIN, F. 2004. Real-time speech motion synthesis from recorded motions. In Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, 345–353.

CHANG, Y.-J., AND EZZAT, T. 2005. Transferable videorealistic speech animation. In Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, 143–151.

CHOE, B., LEE, H., AND KO, H.-S. 2001. Performance-driven muscle-based facial animation. The Journal of Visualization and Computer Animation 12, 2, 67–79.

CHUANG, E., AND BREGLER, C. 2005. Mood Swings: expressive speech animation. ACM Transactions on Graphics 24, 2, 331–347.

COHEN, M. M., AND MASSARO, D. W. 1993. Modeling coarticulation in synthetic visual speech. In Computer Animation 1993 Conference Proceedings, 139–156.

DENG, Z., AND NEUMANN, U. 2006. efase: Expressive facial animation synthesis and editing with phoneme-isomap control. In Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, 251–259.

DENG, Z., NEUMANN, U., LEWIS, J., KIM, T.-Y., BULUT, M., AND NARAYANAN, S. 2006. Expressive facial animation synthesis by learning speech coarticulation and expression spaces. IEEE Transactions on Visualization and Computer Graphics 12, 6, 1523–1534.

GUENTER, B., GRIMM, C., WOOD, D., MALVAR, H., AND PIGHIN, F. 1998. Making faces. In ACM SIGGRAPH 1998 Conference Proceedings, 55–66.

HUANG, X., ALLEVA, F., HON, H.-W., HWANG, M.-Y., LEE, K.-F., AND ROSENFELD, R. 1993. The SPHINX-II speech recognition system: An overview. Computer Speech and Language 7, 2, 137–148.

KIM, I.-J., AND KO, H.-S. 2007. 3d lip-synch generation with data-faithful machine learning. Computer Graphics Forum 26, 3, 295–301. (Eurographics 2007 Conference Proceedings).

MA, W.-C., JONES, A., CHIANG, J.-Y., HAWKINS, T., FREDERIKSEN, S., PEERS, P., VUKOVIC, M., OUHYOUNG, M., AND DEBEVEC, P. 2008. Facial performance synthesis using deformation-driven polynomial displacement maps. ACM Transactions on Graphics 27, 5, Article No.: 121. (SIGGRAPH Asia 2008 Conference Proceedings).

SIFAKIS, E., NEVEROV, I., AND FEDKIW, R. 2005. Automatic determination of facial muscle activations from sparse motion capture marker data. ACM Transactions on Graphics 24, 3, 417–425. (SIGGRAPH 2005 Conference Proceedings).

SIFAKIS, E., SELLE, A., ROBINSON-MOSHER, A., AND FEDKIW, R. 2006. Simulating speech with a physics-based facial muscle model. In Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, 261–270.

WAMPLER, K., SASAKI, D., ZHANG, L., AND POPOVIĆ, Z. 2007. Dynamic, expressive speech animation from a single mesh. In Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, 53–62.

WEISE, T., BOUAZIZ, S., LI, H., AND PAULY, M. 2011. Realtime performance-based facial animation. ACM Transactions on Graphics 30, 4, 77:1–77:10. (SIGGRAPH 2011 Conference Proceedings).