

# Laughing, Crying, Sneezing and Yawning: Automatic Voice Driven Animation of Non-Speech Articulations\*

Darren Cosker  
Department of Computer Science  
University of Bath  
D.P.Cosker@cs.bath.ac.uk

James Edge  
Centre for Vision, Speech and Signal  
Processing, University of Surrey  
J.Edge@surrey.ac.uk

## Abstract

In this paper a technique is presented for learning audio-visual correlations in non-speech related articulations such as laughs, cries, sneezes and yawns, such that accurate new visual motions may be created given just audio. We demonstrate how performance accuracy in voice driven animation can be related to maximizing the models likelihood, and that new voices with similar temporal and spatial audio distributions to that of the model will consistently provide animation results with the lowest ground truth error. By exploiting this fact we significantly improve performance given voices unfamiliar to the system.

**Keywords:** Voice Driven Facial Animation

## 1 INTRODUCTION

In this paper we propose a data-driven HMM based method for learning correlations between non-speech related audio signals – specifically, laughing, crying, sneezing and yawning – and visual facial parameters. Unlike previous work dealing with the audio-visual modeling of this class of signals (DiLorenzo et al., 2008), our data is observed from recorded motions of real performers as opposed to a pre-defined physical model. Unlike previous audio-driven HMM based synthesis work (e.g. (Brand, 1999)), we also attempt to specifically address person independence in our framework. We concentrate on several common non-speech related actions – laughing, crying, sneezing and yawning. A major challenge when using automatic audio driven systems is that of achieving reliable performance given a variety of voices from new people. We demonstrate our approach in a number of speaker-independent synthesis experiments, and show how animation error in voice driven animation has a relation to the proximity of audio distributions for different people and well as similarities between their temporal behaviour. By exploiting these facts we consistently improve synthesis given voices from new people. We implement this improvement using a pre-synthesis classification step. In sum, our approach potentially increases the reusability of such a model for new applications (e.g. online games), and can reduce the need to retrain the model for new identities. Our approach initially requires example audio-visual performances of the action of interest for training: e.g. several laughs, cries, sneezes or yawns. A HMM framework then encodes this audio-visual information. The framework may be trained using any number of desired non-speech action types.

## 2 AUDIO-VISUAL DATA ACQUISITION

Our data set consisted of four participants (2 male and 2 female) captured performing approximately 6-10 different laughs, cries, sneezes and yawns using a 60Hz Qualysis optical motion-capture system. We captured audio simultaneously at 48KHz. We placed 30 retro-reflective markers on each person in order to capture the visual motion of their face while performing the different

---

\*Thanks to the Royal Academy of Engineering and EPSRC for partially funding this work.

actions. We remove head pose from our data set using a least-squares alignment procedure. We then pick one identity from the data set as the base identity and normalise the remaining three identities such that their mean motion-capture vector is the same as the mean for the base. Finally, we perform PCA on the data to reduce its dimensionality, and use the notation  $\mathbf{V}$  to refer to this data set. We represent audio using Mel-Frequency Cepstral Coefficients (MFCCs), and use the notation  $\mathbf{A}$  to refer to this data.

### 3 MODELLING AUDIO-VISUAL RELATIONSHIPS

Observing audio-visual signals for different non-speech related articulations reveals evidence of a temporal structure. We therefore decided to model this behaviour using HMMs (Rabiner, 1989). We first consider a traditional HMM trained using visual data. Let us consider this data to be a set of example non-speech sounds from  $\mathbf{V}$ . After training, the HMM may be represented using the tuple  $\lambda_v = (\mathbf{Q}, \mathbf{B}, \pi)$ , where  $\mathbf{Q}$  is the state transition probability distribution,  $\mathbf{B}$  is the observation probability distribution, and  $\pi$  is the initial state distribution. In our model, each of the  $K$  states in a HMM are represented as a Gaussian mixture  $G_v = (\mu_v, \sigma_v)$ , where  $\mu_v$  and  $\sigma_v$  are the mean and covariance. Each state therefore represents the probability of observing a visual vector.

Given an example visual data sequence, we may calculate the visual HMM state sequence most likely to have generated this data using the Viterbi algorithm. However, we wish to slightly modify the problem such that we may estimate the visual state sequence given an *audio* observation instead. This is our animation goal, i.e. automatic animation of visual parameters given speech. We can do this by remapping the visual observations to audio ones using the learned HMM parameters, i.e. for each  $G_v$  we calculate the distribution  $G_a = (\mu_a, \sigma_a)$  based on the audio  $\mathbf{A}$  corresponding to the visual vectors  $\mathbf{V}$  used in HMM training.

Using the Viterbi algorithm, we may now estimate the most probable visual state sequence using an audio observation. More formally, we can estimate via the HMM the most probable hidden sequence of Gaussian distribution parameters  $\mu_v$  and  $\sigma_v$  corresponding to the observation sequence of MFCC vectors. We next consider what visual parameters  $\vec{v}_t$  to display at output for each state.

We first partition the visual parameter distribution used to train the HMM into distinct regions based on the proximity of a visual parameter to each gaussian. Using  $\mu_v$  and  $\sigma_v$ , we calculate the Mahalanobis distance between each observation  $\vec{v}_i$  and each of the  $K$  states and assign a visual parameter to its closest state. This results in  $K$  partitions of the parameter training set, and given an audio observation we may now state that the visual parameter to display at time  $t$  given  $\vec{a}_t$  is taken from the visual parameter partition associated with the state at time  $t$ . In order to find an optimal output visual parameter sequence, we again utilise the Viterbi algorithm.

Figure 1 gives an overview of visual synthesis, and defines it in terms of two levels: High-Level Re-synthesis, and Low-Level Resynthesis. The High-Level stage is concerned with initially selecting the visual state sequence through the HMM given the audio input. This results in a sequence of visual parameter partitions – one for each time  $t$ . The Low-Level stage then uses the Viterbi algorithm to find the most probable path through these partitions given the observed audio. Resulting visual parameters are converted back in to 3D visual motion vectors by projecting back through the PCA model. An RBF mapping approach (Lorenzo et al., 2003) is then used to animate a 3D facial model for output using this data.

#### 3.1 SPEAKER INDEPENDENCE VIA BEST MATCHING PERSON SELECTION

It is often highly desirable for a voice driven system to be robust to a wide range of different voices. Several design options exist in this case, including: (1) a single HMM trained with the knowledge of multiple people, or (2) one of several HMMs where each contains audio-visual data for a specific person. We concentrate on the latter case for now, so our problem is therefore to select one of several HMMs where each encodes information from a specific identity. It turns out that this is equivalent to determining the probability that a specific HMM generated the observation. Calculating this probability may be achieved by estimating the log-likelihood that a HMM could

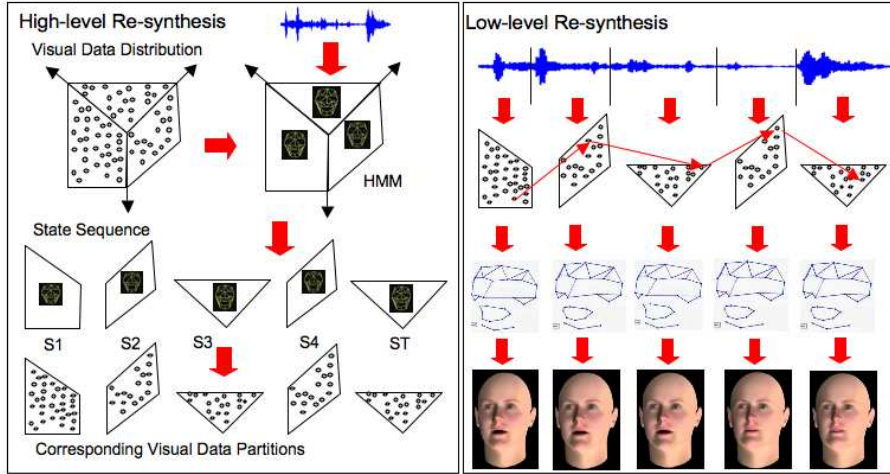


Figure 1: Animation production may be visualised as a high-level state based process followed by a low-level animation frame generation process.

have generated the persons input audio (Rabiner, 1989). We show in our results how selecting a HMM with a higher log-likelihood consistently leads to a lower overall animation error.

#### 4 EXPERIMENTAL RESULTS AND FUTURE DIRECTIONS

We first consider person and action specific synthesis of animations. We trained audio-visual HMMs for a range of specific non-speech actions – laughing, crying, sneezing and yawning – for each of our four performers. Each HMM was trained using approximately 4 different actions, and approximately 4 more were left out for the test cases. Audio corresponding to the test cases was then used to synthesise new 3D animation vectors which were compared to the motion-capture ground truth. Example animations may be found in the video, and RMS errors in millimeters may be found in Table 1.

Person	Laugh			Cry			Sneeze			Yawn		
	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean
P1	0.7	1.42	0.95	0.89	2.3	1.36	0.6	1.5	1.99	1.8	5.1	2.49
P2	2.68	4.7	3.68	1.96	2.42	2.12	3.8	5.6	4.56	1.99	4.13	2.8
P3	0.93	1.49	1.19	1.57	2.25	1.96	0.6	0.92	0.92	ND	ND	ND
P4	1.75	2.16	1.92	1.11	1.4	1.24	1.57	2.52	2	3.74	5.8	4.55

Table 1: Action Specific HMM animation: Min, Max and Mean RMS errors (millimetres) for average synthesised 3D coordinates versus ground truth 3D coordinates.

Person	Min	Laugh	Mean	Min	Cry	Mean	Min	Sneeze	Mean	Min	Yawn	Mean
		Max			Max			Max			Max	
P1+P2+P3+P4	1.15	2.76	1.75	1.29	3.61	2.01	1.6	5.96	3.52	1.77	6.15	3.52

Table 2: Animation with HMMs encoding multiple actions: Min, Max and Mean RMS errors (millimetres) for average synthesised 3D coordinates versus ground truth 3D coordinates.

We next tested combining data from multiple people performing a specific non-speech action inside the same HMM. This assesses the models ability to generalise data for different people within the same model. Again, we left out part of the data for each performer to use as a test-set and calculated RMS errors as shown in Table 2.

	Laugh		B/W L	Cry		B / W L	Sneeze		B / W L	Yawn		B / W L
	B / W E			B / W E			B / W E			B / W E		
P1	<b>2 / 3.3</b>		-1033/-1126	<b>2.08/2.45</b>		-704/-851	<b>2.4/2.46</b>		-749/-1105	<b>2.8/6.6</b>		-607/-1239
P2	<b>2.3/2.5</b>		-422/-777	<b>1.3/2.2</b>		-662/-1130	<b>3.4/3.66</b>		-748/-1006	<b>3.5/5.4</b>		-1081/-1281
P3	<b>1.6/2.8</b>		-763/-2558	<b>1.1/2.1</b>		-857/-3519	<b>2.4/2.9</b>		-1050/-2352	ND		ND
P4	<b>1.7 / 3</b>		-1085/-2039	<b>0.8 / 2.1</b>		-770/-1804	<b>1.5/2.7</b>		-902/-1627	<b>1.8/2.8</b>		1073/1215

Table 3: Average 3D vector animation error (millimeters) given best and worst matching (log-likelihood) HMMs. (B/W E = best/worst error, B/W L = best/worst log-likelihood)

We now test the case where the model has no prior knowledge of a persons voice For each performer we trained four separate HMMs – one for each action. Given input audio for an action, the HMM with the best log-likelihood was selected for synthesis – thus taking into account match between input audio distribution and those of the trained HMMs. Table 3 shows the results, and Figure 2 gives side-by-side comparisons between ground truth video data of a performer, reconstructed 3D vectors, and an animated 3D facial model. Our results clearly show that a HMM with a higher log-likelihood always gives a lower average error reconstructions error. This shows that a high log-likelihood appears correlated with a low animation error. Future work will involve automatically discriminating between non-speech sounds and normal speech, with the eventual aim of animating faces from entirely natural and unconstrained input audio.

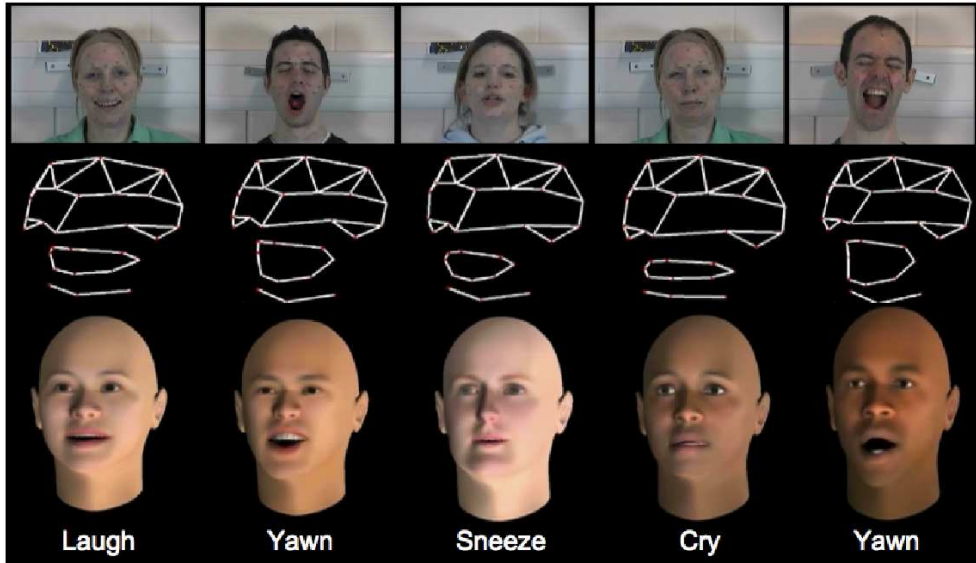


Figure 2: Example Animation Frames. (Top) Ground truth video. (Middle) Corresponding 3D Motion vectors automatically synthesised from speech. (Bottom) A 3D head model animated using the motion vectors using an RBF mapping technique.

## REFERENCES

- Brand, M. (1999). Voice puppetry. In *Proc. of SIGGRAPH*, pages 21–28. ACM Press.
- DiLorenzo, P., Zordan, V., and Sanders, B. (2008). Laughing out loud: Control for modelling anatomically inspired laughter using audio. *ACM Trans. Graphics*, 27(5).
- Lorenzo, M. S., Edge, J., King, S., and Maddock, S. (2003). Use and re-use of facial motion capture data. In *Proc. of Vision, Video and Graphics*, pages 135–142.
- Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–285.