REAL-TIME LIP-SYNCH FACE ANIMATION DRIVEN BY HUMAN VOICE

Fu Jie Huang and Tsuhan Chen
Dept. of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA
Email: {jhuangfu,tsuhan}@ece.cmu.edu

Abstract - In this demo, we present a technique for synthesizing the mouth movement from acoustic speech information. The algorithm maps the audio parameter set to the visual parameter set using the Gaussian Mixture Model and the Hidden Markov Model. With this technique, we can create smooth and realistic lip movements.

INTRODUCTION

Techniques for converting the human voice into visual parameters of mouth movements have applications in face animation, human-computer interfaces, and joint audio-visual speech recognition [1]. The problem of mapping from the audio feature space to the visual feature space can be solved at several different levels, according to the speech analysis being used.

At the first level, frame level, a universal mapping can be derived to map one frame of audio to one frame of visual parameters. This method uses a large set of audio-visual parameters to train the mapping. Such mapping could be extracted by method like Vector Quantization, the Neural Network [4], the Gaussian Mixture Model (GMM) [2], etc. In our demo, we use the GMM to map the acoustic feature set to the visual feature set.

At the second level, phoneme level, the mapping could be found for each phoneme in the speech signal. The first step of mapping from audio to visual parameters is to segment the speech sequence phonetically. Then we use a lookup-table to find out the sequence of visual features. The look-up table is predefined for the whole set of phonemes. In this table, each phoneme is associated with one visual feature set.

At the third level, word level, we can explore the context cues in the speech signals. First we use a speech recognizer to segment the speech into words, like “one”, “four”. For each word, we can create a Hidden Markov Model (HMM) to represent the acoustic state transition in the word. For each state in the HMM model, we can use the methods as in the first level to model the mapping from acoustic to visual feature frame by frame [3]. Since this mapping is tailored to individual words, better results could be achieved than the frame-lever and phoneme-level approaches. In our demo, we will explore the mapping for a small vocabulary composed of 10 digits and 26 English letters.[1]

---

[1] Published in IEEE Multimedia Signal Processing Workshop, Los Angeles, California, 1998

BACKGROUND

Gaussian Mixture Model and EM Algorithm

We use the GMM to model the probability distribution of the audio-visual vectors. To collect the training data of audio-visual vectors, we use a lip-tracking program to extract the lip shape parameters from the video images. Here we take the two most important parameters: the width and height of the outer contour of the mouth. In the mean time, the acoustic speech is analyzed to yield 13 cepstrum coefficients. Then we cascade the cepstrum coefficients with the visual parameters to compose the joint feature vector $O = [a^T, v^T]^T$, where a is the acoustic vector and v is the visual vector.

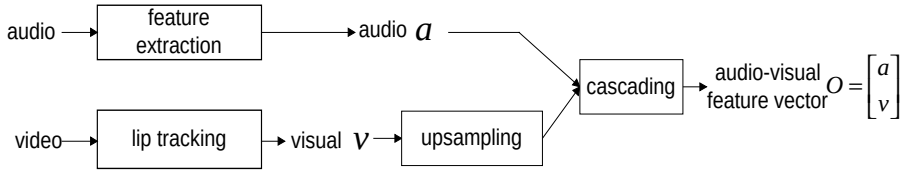

Figure 1. Extraction of Joint audio-visual features

The probability distribution of the audio-visual vectors O is modeled using GMM. The GMM is a weighted sum of k Gaussian functions.

$$p(O) = \sum_{i=1}^{k} w_i g[\mu_i, \Sigma_i](O)$$

where w_i is the mixture weight, $g[\mu_i, \Sigma_i](O)$ is the Gaussian function with mean μ_i, and covariance matrix Σ_i

$$g[\mu_i, \Sigma_i](O) = \frac{1}{\sqrt{(2\pi)^{15}|\Sigma_i|}} \exp\{-\frac{1}{2}(O-\mu_i)^T \Sigma^{-1}(O-\mu_i)\}$$

The GMM is parameterized by a set of triple parameters: the mean vector μ_i, covariance matrix Σ_i and mixture weight w_i

$$\lambda = \{\mu_i, \Sigma_i, w_i\} \qquad i = 1,2,\ldots,k$$

We train the GMM on the training data set with the Expectation-Maximization (EM) algorithm [5]. After the model λ is initialized, the EM algorithm iterates to update the model with an update function, and replace the old model parameter with the new parameter λ'. It has been proven that with the EM algorithm, the product of the likelihood for each data point will increase after each iteration, and converge to the maximum [5].

$$\prod_{O \in \mathbf{O}} p(O \mid \lambda') \geq \prod_{O \in \mathbf{O}} p(O \mid \lambda)$$

where $\mathbf{O}$ is the training data set of O.

After we trained the GMM with the training data set, we can use the model to map the audio feature to the visual feature. If we constrain the covariance matrix Σ_i of each mixture component to be diagonal, we can simplify the optimal estimate of v given a, $\hat{v} = E[v \mid a]$, to be the sum of the mean vector of the visual feature, weighted by the probability that the given acoustic observation belongs to the mixture component

$$\begin{aligned} E[v \mid a] &= \int v \frac{p_{va}(v,a)}{p_a(a)} dv \\ &= \int v \frac{\sum_i w_i \cdot p_{i,va}(v,a)}{p_a(a)} dv \\ &= \sum_i \frac{w_i}{p_a(a)} \int v \cdot p_{i,v}(v) \cdot p_{i,a}(a) dv \\ &= \sum_i \frac{w_i \cdot p_{i,a}(a)}{p_a(a)} \cdot \overline{v}_i \end{aligned}$$

Hidden Markov Model

The mapping method mentioned above is a universal mapping. It ignores the context cues that are inherent in speech. To utilize the context information, we can use a mapping tailored to a specific word. Here we use HMM for the audio to visual parameter conversion.

We use a standard left-right HMM, with 5 states. The parameters describing the model are the transition matrix A, the initial state distribution Π, and the observation symbol probability distribution B for each state [6]. We use the cascaded audio-visual feature vector sequences for each word in the vocabulary to train a HMM for this word.

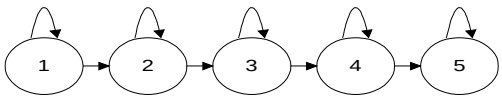

Figure 2. Left-right Hidden Markov Model

Thus we get a joint audio-visual HMM with the parameter of (Π, A, B). As for each state, the observation probability distribution is modeled by the GMM with 3 mixture components.

We also derive an acoustic HMM from the joint HMM. The parameter of the acoustic HMM is (Π, A, B'). Π and A are the same as in the joint HMM, while in B', the observation distribution is derived from B with the equation

$$b_j(a) = \int b_j(o)dv$$

where $b_j(a)$ is the distribution of the acoustic features in the acoustic HMM, and $b_j(o)$ is the distribution of the audio-visual features in the joint HMM.

For each state in the acoustic HMM, we derive an optimal estimator of visual parameter given the acoustic feature based on the same method mentioned in the previous section.

At the conversion phase, we first use the acoustic HMMs to recognize the digits/letters and select the proper HMM to find out the sequence of states using the Viterbi algorithm [6]. Within each state, we can estimate the visual parameters given the acoustic parameters with the associated optimal estimator.

DEMO

Use GMM To Map Continuous Speech To Mouth Movements

To collect the training data, we record some sentences spoken by a male speaker. The acoustic speech is sampled at 11kHz, 8 bits per sample, mono. At the same time video images of the speaker's mouth area are captured at 15 frames per second, 352×288 pixels, 24 bits per pixel. The silence part in the speech and the corresponding video frame is cut out using the end point detection [6].

Then we use our lip-tracking program to extract the lip shape parameters from the video images. Here we take the width and height of the outer contour of the mouth. In the mean time, the acoustic speech is analyzed to yield 13 cepstrum coefficients. The acoustic speech signal is blocked into frames of 256 samples, with adjacent frames being separated by 128 samples. Therefore frame rate of acoustic signal is 86 frames per second. We upsample the visual parameters to the same frame rate as the audio features. Then we cascade the cepstrum coefficients with the visual parameters to compose the audio-visual feature.

We use the 15 dimensional audio-visual features extracted from the video as the training data set to train the GMM with 20 mixture components. We first use Vector Quantization (VQ) [6] to cluster the data set into 20 classes. We use the center vector and covariance matrix of each cluster as the initial mean and covariance, and use the normalized number of the data in each class as the initial weight of that

component. Then we use the EM algorithm to optimize the parameters for the mixture components to find the optimal GMM. Since we constrain each Gaussian component to have a diagonal covariance matrix, we get the final model with 20 component weights, 20 component mean vector of 15 dimensions, 20 component covariance matrix of 15 dimensions.

At the conversion phase, the speaker can record a sequence of continuous speech, then the sequence of talking mouth synchronized with the recorded speech will be given as the output.

The first step to convert the audio signal to the visual parameters is to detect the silence part in the speech. Since with the silence part of the speech, there is no information to predict the visual parameters, we simply set the parameters as the mouth is closed.

For the non-silence part in the speech, we extract the cepstrum coefficients of each frame, calculate the probability of this acoustic feature belonging to the mixture components, and use these probabilities as the weights to sum up the visual means of the components to get the visual estimate, $\hat{v} = E[v \mid a]$, as described in the last section.

Use HMM With GMM To Map Isolated Words To Mouth Movements

We record sequences of isolated digits from 0 to 9 and English letters from A to Z, each for 10 times, spoken by a male speaker, as the training data. At the same time video images of the speaker's mouth area are captured as before. The acoustic speech is sampled at 11KHz, 8 bits per sample. This video stream is segmented by hand according to the speech to be used as the training data. As before, we extract the visual feature and the acoustic feature and combine them to compose the audio-visual feature $O = [a^T, v^T]^T$.

We use this joint audio-visual feature vector to train a 5-state, left-right, Hidden Markov Model. The densities with each state of the HMM is modeled with 3 GMM components. We derive an acoustic HMM from this join HMM model, and derive an optimal estimator of visual parameter given acoustic parameter for each state in this HMM. Since we constrain the HMM to have diagonal covariance matrices for each mixture component, the visual parameters are the sum of the mean visual parameters of each mixture component weighted by the probability that the acoustic observation belongs to the mixture component.

At the conversion stage, when the user speaks several words, the system splits them into isolated words. Then for each word, extract the cepstrum coefficients and feed them into the acoustic HMM models. After recognized correctly, take the proper HMM model and segment the sequence of acoustic parameters into optimal state sequence using Viterbi algorithm.

At each state, we estimate the visual parameters given the acoustic parameters with the optimal estimator. The estimated visual parameters for each digit are then concatenated together to get the whole sequence of the visual parameters of the input audio. For the silence part in the acoustic speech, we simply put fixed mouth parameters. The length of the estimated sequence of the visual parameters is the same with the input audio sequence, and they can be played back with perfect lip synchronization.

CONCLUSION AND FUTURE WORK

In this demo, we implemented a real-time audio-to-visual mapping algorithm. With the GMM, we could predict the visual parameters given the acoustic parameters. We used the method to map continuous speech to synchronized lip movements. With the HMM, we also explored the context cue to achieve better mapping performance for isolated words.

For the future work, we will explore the possibility of extending the use of HMM/GMM with the mapping from continuous speech to smooth lip movements. We will also improve the lip animation model to render realistic lip motions such as in explosives, fricatives, and lip protrusion.

REFERENCE

[1] T. Chen and R. Rao, "Audio-visual integration in multimodal communication," *Proceedings of IEEE, Special Issue on Multimedia Signal Processing*, pp. 837-852, May 1998.

[2] R. Rao and T. Chen, "Exploiting audio-visual correlation in coding of talking head sequences" in *Picutre Coding Symposium' 96*, Melbourne, Australia, March, 1996.

[3] R. Rao and T. Chen, "Using HMM's for Audio-to-Visual Conversion" in *IEEE '97 Workshop on Multimedia Signal Processing*

[4] R. Lippmann, "An introduction to computing with neural networks," *IEEE ASSP Magazine*, vol. 4, no. 2, pages 4-22, April 1987.

[5] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM algorightm," *J. R. Stat. Soc. Lond*, Vol. 39, pp. 1-38, 1977.

[6] L. Rabiner, and B. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, Englewood Cliffs, NJ, 1993.