A HYBRID PHYSICAL AND STATISTICAL DYNAMIC ARTICULATORY FRAMEWORK INCORPORATING ANALYSIS-BY-SYNTHESIS FOR IMPROVED PHONE CLASSIFICATION

Ziad Al Bawab[1]*, Bhiksha Raj*[2]*, and Richard M. Stern*[3]

Department of Electrical and Computer Engineering and Language Technologies Institute, Carnegie Mellon University, Pittsburgh PA 15213
ziada@cs.cmu.edu[1], bhiksha@cs.cmu.edu[2], and rms@cs.cmu.edu[3]

ABSTRACT

In this paper, we present a dynamic articulatory model for phone classification. The model integrates real articulatory information derived from ElectroMagnetic Articulograph (EMA) data into its inner states. It maps from the articulatory space to the acoustic one using an adapted vocal tract model for each speaker and a physiologically-motivated articulatory synthesis approach. We apply the analysis-by-synthesis paradigm in a statistical fashion. We first present a fast approach for deriving analysis-by-synthesis distortion features. Next, the distortion between the speech synthesized from the articulatory states and the incoming speech signal is used to compute the output observation probabilities of the Hidden Markov Model (HMM) used for classification. Experiments with the novel framework show improvements over baseline in phone classification accuracy.

Index Terms— Dynamic articulatory modeling, analysis-by-synthesis, articulatory synthesis for recognition, physical model of the vocal tract, hybrid physical and statistical models for classification

1. INTRODUCTION

Articulatory modeling [1] is used to incorporate speech production information into automatic speech recognition (ASR) systems. It is believed that solutions to the problems of co-articulation, pronunciation variations, and other speaking style related phenomena rest in how accurately we capture the production process.

Our goal in this paper is a dynamic articulatory framework for speech recognition where the model states are collections of possible vocal tract shapes. In previous work we presented two key components that enable us to address this goal. In [2] we proposed new features that convey articulatory information. Using a physically-motivated codebook of vocal tract shapes to derive analysis-by-synthesis distortion features was shown to provide improvements in phone classification accuracy. In our recent work [3], we showed how to derive realistic vocal tract shapes from the EMA data in the MOCHA database. We solely relied on the EMA data to perform speech synthesis, in contrast to the more common approach of learning a statistical mapping between the EMA and acoustic recordings from parallel recordings of the two [4, 5]. EMA measurements are insufficient to describe the overall vocal tract shape. The EMA sensors are located on indefinite locations on the lips, tongue, and velum which vary from a speaker to another and even from one recording session to another. We used Maeda's geometric vocal tract model [6] and adjusted its parameters to superpose vocal tract shapes onto the EMA measurements and to provide a continuous contour of the vocal tract. The resulting vocal tract shapes are defined by seven Maeda parameters. The adapted Maeda model captures the geometry of the vocal tract of each speaker explicitly. The combination of the adapted vocal tract models and a physiologically-motivated articulatory synthesizer, *e.g.* the Sondhi and Schroeter synthesizer [7], models the physical speech production process for each speaker.

Previous and current approaches to incorporating articulatory models into speech recognition [1, 8] have used phonological features representation derived from the transcript through linguistic expert knowledge. This representation may not represent the actual underlying articulatory phenomena that produced the speech signal. The same speech may be produced differently. In the work reported in this paper, we use EMA measurements as means for capturing the ground truth articulatory phenomena. EMA provides exact information about the articulators' movements. Our aim here is to build upon our previous work [2, 3] and incorporate the distortion features in a dynamic framework whose inner states are vocal tract shapes. These vocal tract shapes are derived in a principled geometric fashion as described in [3]. We synthesize speech using the adapted vocal tract models for each speaker to closely mimic the incoming speech signal. The distortion between the incoming speech and the speech synthesized from the articulatory states is used to dynamically traverse the articulator space. This framework not only constrains the set of possible vocal tract shapes for each phone, but is also capable of modeling the articulatory dynamics and imposing further constraints in a probabilistic fashion.

The set of all possible vocal tract shapes is quantized into a codebook represented by vectors of Maeda parameters. For a given phone, only a restricted region in the space of vocal tract shapes, represented by a subset of the codewords, is active. Hence we would only need the distortion features associated with these codewords. In this paper, we show how we can learn this subset by estimating weights for each codeword. We use two approaches, one that uses the EMA data (*i.e.* ground truth) and another that is audio driven. Both approaches yield a solution that zeros out the weights associated with codewords not relevant to the phone in study.

In order to incorporate the distortion into the probabilistic framework, we need to convert it to a form of probability. The key point here is to apply a density function that penalizes higher distortions (*e.g.* exponential density). The lower the distortion from a given codeword, the more likely it is to be the codeword that has generated the incoming frame of speech. Another way of looking at this is saying that we only care about the codewords that reflect the true articulatory dynamics of the phone in study. We refer to this as the "OR" approach. In the "AND" approach [2], we included the distortion from all the codewords, whether relevant to the phone or not, and that helped provide better discrimination and classification accuracy.

Using a subset of distortion features for a particular phone is a way for applying articulatory knowledge to constrain the recogni-

This work was partially supported by NSF (Grants IIS-0420866 and IIS-0916918).

tion problem. It also reduces the amount of computations involved instead of using all the distortion features as we did in [2]. The sparsity in the estimated weights of the codewords for each phone reflects the amount of computational reduction. Since each HMM state is a collection of articulatory states, then the state itself has an articulatory meaning reflected in the weights attributed to the codewords. The transition from a state to another then reflects articulatory movements. This framework can be easily expanded to incorporate articulatory dynamics in different ways.

In this paper, we present our design of the dynamic framework and the observation probability model used. We present different ways for model training and initialization. We analyze the sparsity of the solutions the algorithms converge to and present preliminary phone classification results.

2. FAST ANALYSIS-BY-SYNTHESIS DISTORTION FEATURES

In [2], we use the analysis-by-synthesis distortion features derived from a codebook of Maeda parameters. For each frame of incoming speech we use Maeda's model to convert the codeword to area function and then the Sondhi and Schroeter chain matrices approach to convert the area function to vocal tract transfer function. We also use the source information in the frame to synthesize speech. We made two main modifications that improved the computations with a small degradation in classification accuracy.

The first modification is to decouple the source model from the vocal tract transfer function as we explained in [3]. The second modification is to use a codebook of transfer functions rather than a codebook of Maeda parameters. In an off-line procedure, we use Maeda's model and Sondhi and Schroeter chain matrices approach to convert the codebook of Maeda parameters to a codebook of transfer functions *h_Tract, h_Frication* for each codeword. The codebook stores the vocal tract impulse response of $h\_Tract = \{h_1^T, h_2^T, \ldots, h_L^T\}$ and the frication impulse response $h\_Frication = \{h_1^F, h_2^F, \ldots, h_L^F\}$. L is the length of the impulse response and M is the number of codewords. In computing the analysis-by-synthesis distortion features, we use these transfer functions in the manner shown in Figure 1. This saves a lot of unnecessary computations at run-time. The impulses are converted to frequency domain using the fast Fourier transform (FFT) and multiplied by the generated source signals in the "Fast Synthesis" block using the overlap-add approach to synthesis speech from each codeword. The Mel-frequency cepstral coefficients (MFCCs) are extracted from the real and the synthesized speech. The Mel-cepstral distortion (MCD) between them is computed for each codeword and frame respectively.

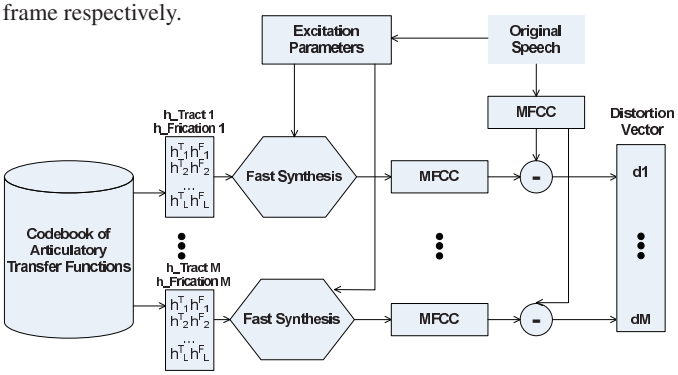

Fig. 1. *A fast analysis-by-synthesis distortion framework.*

3. GENERATING A REALISTIC ARTICULATORY CODEBOOK AND ARTICULATORY TRANSFER FUNCTIONS

In [3], we adapt Maeda's geometric model of the vocal tract to the EMA data of each speaker in the MOCHA database. We then search, on a frame-by-frame basis, a codebook of Maeda parameters for vocal tract shapes that fit each frame of EMA data. In this paper, we sample each phone at five positions: the beginning, middle, end, between beginning and middle, and between middle and end, and read the corresponding Maeda parameter vectors found in the geometric search process. We also add the nasal tract opening area as an additional parameter to the Maeda vector to account for nasal sounds as described in [3].

We then perform k-means clustering over the set of parameter vectors obtained in this manner. We designate the vector closest to the mean of each cluster as the codeword representing the cluster. This is done to guarantee that the codeword is a legitimate articulatory configuration. The set of codewords obtained in this manner is expected to span the space of realistic articulatory configurations and also accounts for velum information.

Once we compute the codebook, we convert it to articulatory transfer functions to be used to derive the analysis-by-synthesis distortion features described in Section 2. We have the option of using the adapted Maeda's model to map the codewords to area functions then to transfer functions or to use the unadapted model.

4. MIXTURE-DENSITY FUNCTION FOR MODELING THE STATE OUTPUT OBSERVATION PROBABILITIES

In [2], the dimensionality of the analysis-by-synthesis distortion features was the same as the number of codewords (*i.e.* 1024). We then used linear discriminant analysis (LDA) to reduce the dimensionality of the features (to 20). Each of the new LDA features is no more related to a particular codeword, but related to a collection or even all of codewords due to the transformation. This is the "AND" approach referred to above.

In order to build a dynamic articulatory framework with the codewords as its sub-states we follow the "OR" approach. We seek to find the path of least distortions and most probable articulatory trajectory. Hence, we cannot use the LDA features used in the "AND" approach and instead use the original distortion features in a larger topology framework. In this section, we model the set of codewords as a mixture probability density function. We show how we can learn the subset of relevant codewords for a given phone.

For state S_1 the acoustic distortion between the speech synthesized from each of the codewords $CD = \{cd_1, cd_2, \ldots, cd_M\}$ and the incoming speech x is $D = \{d_1, d_2, \ldots, d_M\}$. We follow a soft decision approach in which we estimate a set of weights for each phone $\{w_1, w_2, \ldots, w_M\}$ that defines the contribution of each codeword as follows:

$$\begin{aligned} P(x|S_1) &= \sum_{j=1}^{M} P(x, cd_j|S_1) \\ &= \sum_{j=1}^{M} P(cd_j|S_1)P(x|cd_j, S_1) \\ &= \sum_{j=1}^{M} w_{1j}P(x|cd_j, S_1) \end{aligned} \quad (1)$$

4.1. Weight Estimation from Audio

We use the EM algorithm to derive the weights for Equation 1 for a given *phone C* . The EM derivations for HMMs can be found in

Table 1. *Phone error rates for the two speakers using different features, topologies, and initialization procedures.*

Experiment	Features (dimension)	Adaptation	Topology	Obser Prob	Initialization	Sparsity	α	fsew0	msak0	Both	Improvement
Baseline	MFCC + CMN (13)		3S-128M-HMM	Gaussian	VQ	0%	1	61.6%	55.9%	58.8%	
Exp HMM 1	Fast Dist (1024)	NO	3S-1024M-HMM	Exponential	Flat	21%	0.2	57.6%	53.7%	55.7%	5.3%
Exp HMM 2	Fast Dist (1024)	NO	3S-1024M-HMM	Exponential	EMA	51%	0.2	58.3%	53.9%	56.1%	4.6%
Exp HMM 3	Fast Dist (1024)	YES	3S-1024M-HMM	Exponential	EMA	51%	0.25	58.4%	53.1%	55.7%	5.3%
Gaus HMM	Fast Dist + LDA + CMN (20)	NO	3S-128M-HMM	Gaussian	VQ	0%	0.6	54.9%	49.8%	52.4%	10.9%

Bilmes [9]. In our setup, the set of model parameters to estimate is $\boldsymbol{\phi} = \{w_1, w_2, \ldots, w_M, \theta_1, \theta_2, \ldots, \theta_M\}$. The exact set of parameters $\{\theta_j, j = 1 : M\}$ depends on the observation probability used. We assume derivation from a set of $\{x_u, u = 1 : U\}$ data points belonging to *phone C* and drop the phone identity from the equations. From Bilmes [9], the maximum likelihood solution for the weights is:

$$\begin{aligned} w_j^t &= \frac{1}{U}\sum_{u=1}^{U} P(cd_j|x_u, \theta_j^{t-1}) \\ &= \frac{1}{U}\sum_{u=1}^{U} \frac{w_j^{t-1} P(x_u|cd_j, \theta_j^{t-1})}{\sum_{k=1}^{M} w_k^{t-1} P(x_u|cd_k, \theta_k^{t-1})} \end{aligned} \quad (2)$$

Starting with a flat initialization of $w_j = \frac{1}{M}, (j = 1 : M)$ we can iterate until the values of w_j converge.

4.2. Weight Estimation using EMA

Forced-alignment of the audio and the transcript provides the phonetic segmentation for the MOCHA database. For each frame we know which *phone C* it corresponds to. From the EMA data, we can also know which codeword it corresponds to. Hence we can count the codewords for each phone and estimate the probability of being in one codeword for this phone. This estimate can be used as a prior to estimating the weights from audio and for initialization purposes in other databases where the EMA data are not available.

$$w_j = \frac{count\_frames(phone = C, truecode = cd_j)}{total\_frames(phone = C)} \quad (3)$$

4.3. Output Distortion Probabilities

For each frame of speech x_u, we compute the distortion $\{d_{uj}, j = 1 : M\}$ for each codeword $\{cd_j, j = 1 : M\}$. The lower the distortion the more likely is the codeword in producing the speech. This is reflected in our choice of the exponential density functions to model the output observation probabilities of the HMM:

$$P(x_u|cd_j, \theta_j) = \lambda_j \exp^{-\lambda_j d_{uj}^2} \quad (4)$$

4.4. Estimating the Lambdas of the Exponential Distributions from Audio

The set of parameters to estimate is $\{\lambda_1, \lambda_2, \ldots, \lambda_M\}$. Including the estimation of lambdas in the EM:

$$\begin{aligned} \lambda_j^t &= \frac{\sum_{u=1}^{U} P(cd_j|x_u, \theta_j^{t-1})}{\sum_{u=1}^{U} d_{uj}^2 P(cd_j|x_u, \theta_j^{t-1})} \\ &= \frac{\sum_{u=1}^{U} \frac{w_j^{t-1} P(x_u|\lambda_j^{t-1})}{\sum_{k=1}^{M} w_k^{t-1} P(x_u|\lambda_k^{t-1})}}{\sum_{u=1}^{U} d_{uj}^2 \frac{w_j^{t-1} P(x_u|\lambda_j^{t-1})}{\sum_{k=1}^{M} w_k^{t-1} P(x_u|\lambda_k^{t-1})}} \end{aligned} \quad (5)$$

Starting with a flat initialization of $\lambda_j = \frac{1}{mean(d_{uj}^2|phone=C)}$ we can iterate until the values of λ_j converge.

4.5. Estimating the Lambdas of the Exponential Distributions from EMA

As mentioned in Subsection 4.2, using forced-alignment and EMA, we get the codewords identities. With ground truth information about the codewords we can estimate $\{\lambda_1, \lambda_2, \ldots, \lambda_M\}$ directly for each codeword without polluting the estimation with data generated from other codewords like when using EM. This estimate can be used as a prior to estimating $\{\lambda_1, \lambda_2, \ldots, \lambda_M\}$ from audio and for initialization purposes in other databases where the EMA data are not available.

$$\lambda_j = \frac{1}{mean(d_{uj}^2|phone = C, truecode = cd_j)} \quad (6)$$

4.6. HMM Formulation for Exponential Observation Probabilities

To model dynamics, we use a three-state $S = \{S_1, S_2, S_3\}$ left-to-right HMM for each phone. The basic formulation of HMM parameters we use is as in Bilmes [9], with some modification to reflect the observation density used to model the distortion features. To enforce the left-right three state topology we initialize the vector of initial state probabilities $\{\pi_i, i = 1 : N\}$, where N is the number of states ($N = 3$), as $\pi = [1, 0, 0]$. We also initialize the transition probabilities matrix $\{a_{ij}, i, j = 1 : N\}$ to $a_{ij} = [0.5, 0.5, 0; 0, 0.5, 0.5; 0, 0, 1]$. The EM parameters we define for each segment here are $bd_i(t), \gamma_i(t)$, and $\gamma_{ik}(t)$, where k is the mixture index $\{k = 1 : M\}$. The HMM model parameters computed from all the segments are w_{ik}, and λ_{ik}. Each segment $\mathbf{X}$ of a given phone is made of observations $\{x(t), t = 1 : T\}$, where T is the number of frames in each segment. The output observation probability from each state and $\gamma_{ik}(t)$ are described by Equation 7. $\gamma_i(t)$ is as described in [9]. This formulation is integrated in the forward-backward code for HMM model estimation.

$$\begin{aligned} bd_i(t) &= P(x(t)|S(t) = i) \\ &= \sum_{k=1}^{M} w_{ik}\lambda_{ik} \exp^{-\lambda_{ik} d_k^2(t)} \\ \gamma_i(t) &= P(S(t) = i|\mathbf{X}, \boldsymbol{\phi}) \\ \gamma_{ik}(t) &= P(S(t) = i, mixture(t) = k|\mathbf{X}, \boldsymbol{\phi}) \\ &= \gamma_i(t) \frac{w_{ik}\lambda_{ik} \exp^{-\lambda_{ik} d_k^2(t)}}{bd_i(t)} \end{aligned} \quad (7)$$

4.7. HMM Classification using Estimated Parameters

For scoring each segment, we calculate the likelihood probability using the sum of α's of the forward-backward algorithm. The sum of α's over all states at the end of segment e is the likelihood of the segment as shown in [9]: $P(\mathbf{X}|C, \boldsymbol{\phi}) = \sum_{i=1}^{N} \alpha_i(Te)$. Te is the length of each segment e.

5. EXPERIMENTAL RESULTS

We conduct a number of experiments to evaluate the usefulness of the proposed articulatory framework for speech recognition. In order to avoid obfuscating our results with the effect of lexical and linguistic constraints that are inherent in a continuous speech recognition system, we evaluate our features on a simple phone classification task, where the boundaries of phones are assumed to be known.

We choose as our data set the audio recordings from the MOCHA database itself, since it permits us to use the exact articulatory configurations for any segment of sound. We use the data from nine speakers for our work: "faet0", "falh0", "ffes0", "fjmw0", "fsew0", "maps0", "mjjn0", "msak0", and "ss2404". Five of the speakers are females and four are males. We choose to test on the female speaker "fsew0" and the male speaker "msak0" and train on the rest. All experiments are speaker independent. The amount of training utterances is 2569 and testing is 918 composed of 14352 phone segments from speaker "fsew0" and 14302 from speaker "msak0" and 28654 in total. Only EMA data from the training speakers were used to compute the articulatory codebook and to initialize the model parameters. The codebook consisted of 1024 codewords.

The phone $\hat{C}$ for each segment is estimated as:

$$\hat{C} = \text{argmax}_C P(C) P(MFCC|C)^{\alpha} P(FastDist|C)^{(1-\alpha)} \quad (8)$$

where C represents an arbitrary phone, and $MFCC$ and $FastDist$ represent the set of MFCC features and fast analysis-by-synthesis distortion features for the segment respectively. α is a positive number between 0 and 1 that indicates the relative contributions of the two features to classification. We vary the value of α between 0 and 1.0 in steps of 0.05, and choose the value that resulted in the best classification in the form of phone error rate (PER). The classification results and the optimal value of α are shown in Table 1.

The *Baseline* experiment reports phone error rates for the two speakers and both of them using 13-dimensional MFCC features with cepstral mean normalization (CMN). We use a three-state HMM with left-to-right topology where the observation probabilities are a mixture of 128 Gaussian densities. We use vector quantization (VQ) to initialize the means of the mixtures.

In experiment *Exp HMM 1* we use the distortion features derived as explained in Section 2. We use the articulatory synthesis model without adaptation to derive these features. We apply a three-state HMM and mixtures of 1024 exponential density functions for the output probabilities. We initialize the weights and lambdas of the exponential distribution from the distortion features (audio only) as described in Subsections 4.1 and 4.4. Using $\alpha = 0.2$ and combining the probabilities of the baseline system with this system yields an reduction of 5.3% in PER. This shows that our new framework does indeed improve the classification performance. The sparsity of the weights is defined as the percent of them that are zeros over the three states, computed over all the codewords and phones. The codewords that have zero weights over the three HMM states do not need to be considered during classification, i.e. there is no need to synthesize speech from these codewords when considering a particular phone. Initializing from the distortion features (audio only) results in 21% of the weights to be zero. This is the "OR" approach we described above.

In experiment *Exp HMM 2* we follow the same approach as *Exp HMM 1* except that now we initialize the weights and lambdas from EMA as described in Subsections 4.2 and 4.5. The EM starts from the solution provided by EMA and converges to the most likely solution given the distortion data for each phone. This increases the sparsity to 51% with small degradation in phone accuracy, yet reduces the computations required considerably.

In experiment *Exp HMM 3* we follow the same approach as *Exp HMM 2* except that now we use the articulatory synthesis model with adaptation to derive the distortion features. Table 1 shows the effect on adaptation on the phone classification. Note that especially for speaker "msak0", the adaptation has provided a considerable improvement in classification accuracy. The overall classification accuracy is the same as in *Exp HMM 1* but with the same sparsity as in *Exp HMM 2*. This shows that when the system focuses on a subset of articulatory configurations related to each phone and closely mimics the incoming speech through adaptation, it is most effective in classification. This is more evident in the "msak0" speaker case whose geometric adaptation is more effective on the synthesis quality than the adaptation of speaker "fsew0".

Finally, in experiment *Gaus HMM* we use a similar setup to our previous work [2]. We here use all the distortion features in the "AND" approach. We apply LDA to compress the features to 20 dimensions and then apply CMN to them. We use a three-state HMM with 128 Gaussian Mixtures to model the new features. We combine the probabilities of this system with that of the baseline system. Using $\alpha = 0.6$ yields 10.9% reduction in phone error rate. This is the biggest improvement we achieved and shows that information in all the distortion features is helpful in discriminating among phones. The drawback of this topology is that there is no more an articulatory meaning attributed to the states. Hence, we can not model the dynamics explicitly as we can in the previous experiments.

6. CONCLUSIONS AND FUTURE WORK

We have described a dynamic articulatory model for phone classification that incorporates realistic vocal tract shapes in a statistical HMM framework. We have shown how to incorporate analysis-by-synthesis distortion features in a probabilistic pattern recognition approach. Our new framework attributes articulatory meaning to the states through a set of weights. We have shown how to initialize these weights from ground truth articulatory information and to update them from distortion data. Experiments have provided improvements in phone classification over baseline MFCC features. The framework we presented is a basic prototype of incorporating physical constraints in a statistical framework and can be expanded in the future to incorporate further dynamic constraints. Future work will integrate the trained models into a continuous speech recognition system.

7. REFERENCES

[1] M. Richardson, J. Bilmes, and C. Diorio, "Hidden-articulator markov models for speech recognition," *Speech Communications*, vol. 41(2), October 2003.

[2] Z. Al Bawab, B. Raj, and R. M. Stern, "Analysis-by-synthesis features for speech recognition," in *ICASSP*, Las Vegas, Nevada. USA, April 2008.

[3] Z. Al Bawab, L. Turicchia, R. M. Stern, and B. Raj, "Deriving vocal tract shapes from electromagnetic articulograph data via geometric adaptation and matching," in *Interspeech*, Brighton, UK, September 2009.

[4] C. S. Blackburn and S. J. Young, "Pseudo-articulatory speech synthesis for recognition using automatic feature extraction from X-ray data," in *Proc. ICSLP '96*, Philadelphia, PA, 1996, vol. 2, pp. 969–972.

[5] S. Hiroya and M. Honda, "Estimation of articulatory movements from speech acoustics using an hmm-based speech production model," *IEEE Transactions on Speech and Audio Processing*, vol. 12(2), pp. 175–185, 2004.

[6] S. Maeda, "Compensatory articulation during speech: evidence from the analysis and synthesis of vocal tract shapes using an articulatory model," in *Speech Production and Modellling*. W.J. Hardcastle and A. Marchal, 1990, pp. 131–149.

[7] M. M. Sondhi and J. Schroeter, "A hybrid time-frequency domain articulatory speech synthesizer," *IEEE Transac. ASSP*, vol. 35, pp. 955–967, July 1987.

[8] V. Mitra, H. Nam, C. Espy-Wilson, and E. Saltzman L. Goldstein, "Noise robustness of tract variables and their application to speech recognition," in *Interspeech*, Brighton, UK, September 2009.

[9] J. Bilmes, "A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," in *Technical Report TR-97-021, ICSI*, 1997.