Learning Hidden Markov Models with Hidden Markov Trees as Observation Distributions

Diego H. Milone, Leandro E. Di Persia

Signals and Computational Intelligence Laboratory
Department of Informatics
Faculty of Engineering and Water Sciences
National University of Litoral, CONICET
Ciudad Universitaria, Santa Fe, Argentina
{dmilone,ldipersia}@fich.unl.edu.ar
http://fich.unl.edu.ar/sinc

Abstract

Hidden Markov models have been found very useful for a wide range of applications in artificial intelligence. The wavelet transform arises as a new tool for signal and image analysis, with a special emphasis on nonlinearities and nonstationarities. However, learning models for wavelet coefficients have been mainly based on fixed-length sequences. We propose a novel learning architecture for sequences analyzed on a short-term basis, but not assuming stationarity within each frame. Long-term dependencies are modeled with a hidden Markov model which, in each internal state, deals with the local dynamics in the wavelet domain using a hidden Markov tree. The training algorithms for all the parameters in the composite model are developed using the expectation-maximization framework. This novel learning architecture can be useful for a wide range of applications. We detail experiments with real data for speech recognition. In the results, recognition rates were better than the state of the art technologies for this task.

Keywords: Sequence Learning, EM Algorithm, Hidden Markov Models, Hidden Markov Trees, Wavelets, Speech Recognition.

1 Introduction

In the last two decades, Hidden Markov models (HMM) have been used in many application areas of artificial intelligence [19, 1, 13, 14]. On the other hand, from its early applications the wavelet transform has been a very interesting representation for signal and image analysis [15].

Models for wavelet coefficients were initially based on the traditional assumptions of independence and Gaussianity. Statistical dependence at different scales and non-Gaussian statistics were considered in [6] with the introduction of the hidden Markov trees (HMT). Training algorithms for these models were based in a previous development of an expectation-maximization (EM) algorithm for dependence tree models [18]. In the last years, the HMT model was improved in several ways, for example, using more states within each HMT node and developing more efficient algorithms for initialization and training [10, 9].

A discrete HMM defines a probability distribution over sequences of symbols arriving from a finite set. On the other hand, continuous HMM

ISSN: 1137-3601. ©AEPIA (http://www.aepia.org/revista/)

provides a probability distribution over sequences of continuous data in $\mathbb{R}^N$. A general model for the continuous observation densities is the Gaussian mixture model (GMM) [5]. The HMM-GMM architecture was a widely used model, for example, in speech recognition [17]. Nevertheless, more accurate models have been proposed for the observation densities [3]. In both, discrete and continuous models, the most important advantage of the HMM lies in that they can deal with sequences of variable length. However, if the whole sequence is analyzed by the standard discrete wavelet transform (DWT), like in the case of HMT, a representation whose structure is dependent on the sequence length is obtained. Therefore, the learning architecture should be trained and used only for this sequence length or, otherwise, a warping preprocessing is required (to fit the sequence length to the model structure). On the other hand, in HMM modeling, stationarity is generally assumed withing each observation in the sequence. This stationarity assumption can be removed when observed features are extracted by the DWT, but a suitable statistical model for learning this features in the wavelet domain would be needed.

Combining the advantages of the HMM to deal with variable length sequences and the HMT to model DWT representations, in this paper we propose an EM algorithm to train a composite model in which each state of the HMM uses the observation density provided by an HMT. In this HMM-HMT composite model, the HMM handle the long term dynamics in the sequence while the local dynamics are well captured in the wavelet domain by the set of HMT models.

Fine et al. [11] proposed a recursive hierarchical generalization of discrete HMM. They apply the model to learn the multiresolution structure of natural English text and cursive handwriting. Some years later, Murphy and Paskin [16] derived a simpler inference algorithm by formulating the hierarchical HMM as a special kind of dynamic Bayesian network. A wide review about multiresolution Markov models was provided in [22], with special emphasis on applications to signal and image processing. Dasgupta et al. [7] proposed a dual-Markov architecture, similar to the one proposed in the present work. This model is trained by means of an iterative process where the most probable sequence of states is identified, and then each internal model is adapted with the selected observations. However, in this case the model consists of two separated and independent entities, that are forced to work in a coupled way. By the contrary, Bengio et al. derived an EM algorithm for the full model [2], composed of an external HMM in which for each state an internal HMM provides the observation probability distribution [21]. In the following, we derive an EM algorithm for a composite HMM-HMT architecture that observes sequences of DWTs in $\mathbb{R}^N$. This algorithm can be easy generalized to sequences in $\mathbb{R}^{N \times N}$ with 2-D HMTs like the used in [4] or [12].

In the next section we introduce the notation for HMM and HMT. Using this notation we present the proposed model, defining the joint likelihood and then deriving the training formulas for single observation sequences (the generalization to multiple observation sequences being straightforward). In Section 3, the experimental results for speech recognition using real data are presented and discussed. In the last section we present the main conclusions and many ideas that are opened to future works using this novel learning architecture.

2 The Model

The architecture proposed in this work is a composition of two Markov models: the long term dependencies are modeled with an external HMM and each pattern in the local context is modeled with an HMT.

2.1 Basic Definitions

To model a sequence $\mathbf{W} = \mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^T$, with $\mathbf{w}^t \in \mathbb{R}^N$, a continuous HMM is defined with the structure $\vartheta = \langle \mathcal{Q}, \mathbf{A}, \boldsymbol{\pi}, \mathcal{B} \rangle$, where:

i) $\mathcal{Q} = \{Q \in [1 \ldots N_Q]\}$ is the set of states.

ii) $\mathbf{A} = \left[a_{ij} = \Pr\left(Q^t = j \left| Q^{t-1} = i\right.\right)\right], \forall i, j \in \mathcal{Q}$, is the matrix of transition probabilities, where $Q^t \in \mathcal{Q}$ is the model state at time $t \in [1 \ldots T]$, $a_{ij} \geq 0 \; \forall i, j$ and $\sum_j a_{ij} \doteq 1 \; \forall i$.

iii) $\boldsymbol{\pi} = [\pi_j = \Pr(Q^1 = j)]$ is the initial state probability vector. In the case of left-to-right HMM this vector is $\boldsymbol{\pi} = \boldsymbol{\delta}_1$.

iv) $\mathcal{B} = \left\{b_k\left(\mathbf{w}^t\right) = \Pr\left(\mathbf{W}^t = \mathbf{w}^t \left| Q^t = k\right.\right)\right\}, \forall k \in \mathcal{Q}$, is the set of observation (or emission) probability distributions.

Let be $\mathbf{w} = [w_1, w_2, \ldots, w_N]$ resulting of a DWT analysis with J scales and without including w_0, the approximation coefficient at the coarsest scale (that is, $N = 2^J - 1$). The HMT can be defined with the structure $\theta = \langle \mathcal{U}, \mathcal{R}, \boldsymbol{\pi}, \boldsymbol{\epsilon}, \mathcal{F} \rangle$, where:

i) $\mathcal{U} = \{u \in [1 \ldots N]\}$ is the set of nodes in the tree.

ii) $\mathcal{R} = \{R \in [1 \ldots NM]\}$ is the set of states in all the nodes of the tree, denoting with $\mathcal{R}_u = \{R_u \in [1 \ldots M]\}$ the set of states in the node u.

iii) $\boldsymbol{\epsilon} = \left[\epsilon_{u,mn} = \Pr(R_u = m | R_{\rho(u)} = n)\right]$, $\forall m \in \mathcal{R}_u, \forall n \in \mathcal{R}_{\rho(u)}$, is the array whose elements hold the conditional probability of node u being in state m given that the state in its parent node $\rho(u)$ is n, where $\sum_m \epsilon_{u,mn} \overset{\circ}{=} 1$.

iv) $\boldsymbol{\pi} = [\pi_p = \Pr(R_1 = p)]$, $\forall p \in \mathcal{R}_1$ are the probabilities for the root node being on state p.

v) $\mathcal{F} = \{f_{u,m}(w_u) = \Pr(W_u = w_u | R_u = m)\}$ are the observation probability distributions. This is, $f_{u,m}(w_u)$ is the probability of observing the wavelet coefficient w_u with the state m (in the node u).

In the following, we will simplify the notation for random variables. For example, we write $\Pr(w_u | r_u)$ rather than $\Pr(W_u = w_u | R_u = r_u)$.

2.2 Joint Likelihood

Let be Θ an HMM like the one defined above but using a set of HMTs to model the observation densities within each HMM state:

$$b_{q^t}(\mathbf{w}^t) = \sum_{\forall \mathbf{r}} \prod_{\forall u} \epsilon^{q^t}_{u, r_u r_{\rho(u)}} f^{q^t}_{u, r_u}(w^t_u), \qquad (1)$$

with $\mathbf{r} = [r_1, r_2, \ldots, r_N]$ a combination of hidden states in the HMT nodes. To extend the notation in the composite model, we have added a superscript in the HMT variables to make reference to the state in the external HMM. For example, $\epsilon^k_{u,mn}$ will be the conditional probability that, in the state k of the external HMM, the node u is in state m given that the state of its parent node $\rho(u)$ is n.

Thus, the complete joint likelihood for the HMM-HMT can be obtained as

$$\mathcal{L}_\Theta(\mathbf{W}) =$$

$$\begin{aligned} &= \sum_{\forall \mathbf{q}} \prod_t \left(a_{q^{t-1} q^t} \sum_{\forall \mathbf{r}} \prod_{\forall u} \epsilon^{q^t}_{u, r_u r_{\rho(u)}} f^{q^t}_{u, r_u}(w^t_u) \right) \\ &= \sum_{\forall \mathbf{q}} \sum_{\forall \mathbf{R}} \prod_t a_{q^{t-1} q^t} \prod_{\forall u} \epsilon^{q^t}_{u, r^t_u r^t_{\rho(u)}} f^{q^t}_{u, r^t_u}(w^t_u) \\ &\triangleq \sum_{\forall \mathbf{q}} \sum_{\forall \mathbf{R}} \mathcal{L}_\Theta(\mathbf{W}, \mathbf{q}, \mathbf{R}), \end{aligned} \qquad (2)$$

where we simplify $a_{01} = \pi_1 = 1$, $\forall \mathbf{q}$ is over all possible state sequences $\mathbf{q} = q^1, q^2, \ldots, q^T$ and $\forall \mathbf{R}$ are all the possible sequences of all the possible combinations of hidden states $\mathbf{r}^1, \mathbf{r}^2, \ldots, \mathbf{r}^T$ in the nodes of each tree.

2.3 Training Formulas

In this section we will obtain the maximum likelihood estimation of the model parameters. For the optimization, the auxiliary function can be defined as

$$\mathcal{D}(\Theta, \bar{\Theta}) \triangleq \sum_{\forall \mathbf{q}} \sum_{\forall \mathbf{R}} \mathcal{L}_\Theta(\mathbf{W}, \mathbf{q}, \mathbf{R}) \log \left(\mathcal{L}_{\bar{\Theta}}(\mathbf{W}, \mathbf{q}, \mathbf{R}) \right) \qquad (3)$$

and using (2)

$$\mathcal{D}(\Theta, \bar{\Theta}) =$$

$$= \textstyle\sum_{\forall \mathbf{q}} \sum_{\forall \mathbf{R}} \mathcal{L}_\Theta(\mathbf{W}, \mathbf{q}, \mathbf{R}) \cdot \{ \sum_t \log(a_{q^{t-1} q^t}) +$$

$$+ \sum_t \sum_{\forall u} \left[\log \left(\epsilon^{q^t}_{u, r^t_u r^t_{\rho(u)}} \right) + \log \left(f^{q^t}_{u, r^t_u}(w^t_u) \right) \right] \Bigg\}. \qquad (4)$$

For the estimation of the transition probabilities in the HMM, a_{ij}, no changes from the standard formulas will be needed. However, on each internal HMT we hope that the estimation of the model parameters will be affected by the probability of being in the HMM state k at time t.

Let be $q^t = k$, $r^t_u = m$ and $r^t_{\rho(u)} = n$. To obtain the learning rule for $\epsilon^k_{u,mn}$ the restriction $\sum_m \epsilon^k_{u,mn} \overset{\circ}{=} 1$ should be satisfied. If we use

$$\hat{\mathcal{D}}(\Theta, \bar{\Theta}) \triangleq \mathcal{D}(\Theta, \bar{\Theta}) + \sum_n \lambda_n \left(\sum_m \epsilon^k_{u,mn} - 1 \right), \qquad (5)$$

the learning rule results

$$\epsilon^k_{u,mn} = \frac{\sum\limits_t \gamma^t(k)\xi^{tk}_u(m,n)}{\sum\limits_t \gamma^t(k)\gamma^{tk}_{\rho(u)}(n)}, \tag{6}$$

where $\gamma^t(k)$ is computed as usual for HMM and $\gamma^{tk}_{\rho(u)}(n)$ and $\xi^{tk}_u(m,n)$ can be estimated with the upward-downward algorithm [9].

For the observation distributions we use $f^{q^t}_{u,r^t_u}(w^t_u) = \mathcal{N}\left(w^t_u, \mu^{q^t}_{u,r^t_u}, \sigma^{q^t}_{u,r^t_u}\right)$. From (4) we have

$$\begin{aligned}
&\mathcal{D}(\Theta,\bar{\Theta}) = \\
&= \sum_{\forall \mathbf{q}} \sum_{\forall \mathbf{R}} \mathcal{L}_\Theta(\mathbf{W},\mathbf{q},\mathbf{R}) \cdot \Bigg[\sum_t \log(a_{q^{t-1}q^t}) + \\
&\quad + \sum_t \sum_{\forall u} \log\left(\epsilon^{q^t}_{u,r^t_u r^t_{\rho(u)}}\right) + \\
&\quad + \sum_t \sum_{\forall u} \Bigg(-\frac{\log(2\pi)}{2} - \log\left(\sigma^{q^t}_{u,r^t_u}\right) - \\
&\qquad - \frac{\left(w^t_u - \mu^{q^t}_{u,r^t_u}\right)^2}{2\left(\sigma^{q^t}_{u,r^t_u}\right)^2} \Bigg) \Bigg] . \qquad (7)
\end{aligned}$$

Thus, the training formulas result

$$\mu^k_{u,m} = \frac{\sum\limits_t \gamma^t(k)\gamma^{tk}_u(m) w^t_u}{\sum\limits_t \gamma^t(k)\gamma^{tk}_u(m)} \tag{8}$$

and

$$(\sigma^k_{u,m})^2 = \frac{\sum\limits_t \gamma^t(k)\gamma^{tk}_u(m)\left(w^t_u - \mu^k_{u,m}\right)^2}{\sum\limits_t \gamma^t(k)\gamma^{tk}_u(m)}.$$

These results can be easily extended to multiple observation sequences.

3 Experimental Results and Discussion

In this section we test the proposed model in the context of automatic speech recognition with the TIMIT corpus [23]. TIMIT is a well known corpus that has been used extensively for research in automatic speech recognition. From this corpus five phonemes that are difficult to classify were selected. The voiced stops */b/* and */d/* have a very similar articulation (bilabial/alveolar) and different phonetic variants according to the context (allophones). Vowels */eh/* and */ih/* were selected because their formants are very close. Thus, these phonemes are very confusable. To complete the selected phonemes, the affricate phoneme */jh/* was added as representative of the voiceless group [20]. In the next experiments we used 1145 patterns for train and 338 for test, that is, all the phoneme realizations in the geographical region 1 of the TIMIT corpus.

Regarding practical issues, the training formulas were implemented in logarithmic scale to make a more efficient computation of products and to avoid underflow errors in the probability accumulators [9]. In addition, underflow errors are reduced because in the HMM-HMT architecture each DWT is in a lower dimension than the dimension resulting from a unique HMT for the whole sequence. All learning algorithms and transforms used in the experiments were implemented in C++ from scratch[1].

Frame by frame, each local feature is extracted using a Hamming window of width N_w, shifted in steps of N_s samples [17]. The first window begins N_o samples out (with zero padding) to avoid the information loss at the beginning of the sequence. The same procedure is used to avoid information loss at the end of the sequence.

Then a DWT is applied to each windowed frame. The DWT was implemented by the fast pyramidal algorithm [15], using periodic convolutions and the Daubechies-8 wavelet [8]. Preliminary tests were carried out with other wavelets of the Daubechies and Splines families but not important differences in results were found.

A separate model is trained for each phoneme and the recognition is made by the conventional maximum-likelihood classifier.

In the first study, different recognition architectures are compared, but setting they to have the total number of trainable parameters in the same order of magnitude. Table 1 shows the recognition rates (RR) for: GMM with 4 Gaussians in the mixture (2052 trainable parameters),

[1] The source code can be downloaded from http://fich.unl.edu.ar/sinc

$N_w \rightarrow$	128		256	
$N_s \rightarrow$	64	128	64	128
GMM	36.98	28.99	34.64	29.88
HMT	**40.53**	31.36	**44.08**	36.39
HMM-GMM	25.44	35.21	24.85	37.87
HMM-HMT	39.64	**47.34**	42.90	**39.64**

Table 1. Recognition rates (RR%) for TIMIT phonems using models with a similar number of trainable parameters (see more details in the text). Each learning architecture was trained and tested with two frame sizes, N_w, and two frame steps, N_s, in the feature extraction.

HMT with 2 states per node and one Gaussian per state (2304 trainable parameters), HMM-GMM with 3 states and 4 Gaussians in each mixture (6165 trainable parameters), HMM-HMT with 3 HMM states, 2 states per HMT node and one Gaussian per node state (6921 trainable parameters)[2]. For HMM-GMM and HMM-HMT the external HMM have connections $i \rightarrow i$, $i \rightarrow (i+1)$ and $i \rightarrow (i+2)$. The last link allows to model the shortest sequences, with less frames than states in the model. In both, GMM and HMM-GMM, the Gaussians in the mixture are modeled with diagonal covariance matrices.

The maximum number of train iterations used for all experiments was 10, but also, as finalization criteria, the training process was stopped if the average (log) probability of the model given the training sequences was improved less than 1%. In most of the cases, the training converges after 4 to 6 iterations, but HMM-GMM models experienced several convergence problems with the DWT data. When a convergence problem was observed, the model corresponding to the last estimation with an improvement in the average probability of the model given the sequences was used for testing.

The results in Table 1 always favor the HMM-HMT model or, when there are more frames in the sequence, results favor the HMT model. This may be due to the use of an HMM with 3 states in the HMM-HMT. However, for a window step of 64 samples, 5 or 6 HMM states could have been better (in any case, when HMT was the best, HMM-HMT was the closest second one). Considering the global maximum and average RR, the HMM-HMT was the best recognition architecture.

	Gaussians per GMM	Total of parameters	Recognition rate [%]
HMM-GMM	2	3087	17.75
	4	6165	37.87
	8	12321	29.59
	16	24633	33.43
	32	49257	27.51
	64	98505	26.92
HMM-HMT	512	6921	39.64

Table 2. Recognition results for TIMIT phonemes. HMM-GMM was trained and tested for different number of Gaussians in the mixtures.

The next experiments were focused to compare the two main models related with this work, that is, HMM using observation probabilities provided for GMMs or HMTs. In this context, the best relative scenario for HMM-GMM is using $N_w = 256$ and $N_s = 128$ (see Table 1). Nevertheless, as we remarked above, several problems of convergence were observed in the training of HMM-GMM. As it is well known, the non-Gaussianity and the correlated information in the wavelet coefficients are important issues.

In Table 2 we present a fine tuning for the HMM-GMM model. Note that comparable architectures for HMM-HMT are HMM-GMM with between 2 to 8 Gaussians in the mixtures, because they have a similar number of trainable parameters. In relation to the HMM architectures, the HMM-HMT is still providing the best recognition rates.

The computational cost is one of the major handicaps of the proposed approach, mainly because of the double Baum-Welch process required in the training. To provide an idea of the computational cost, results reported in Table 1, with $N_w = 256$ and $N_s = 128$, demand 30.20 s of training for the HMM-GMM whereas the same training set demands 240.89 s in the HMM-HMT[3]. In the future it would be interesting to work in the optimization of the algorithms, for example, developing a Viterbi algorithm for the composite model.

4 Conclusions and Future Work

A novel Markov architecture for learning sequences in the wavelet domain was presented.

[2] All these counts are for $N_w = 256$. Recall that Gaussians in GMM and HMM-GMM are in $\mathbb{R}^{N_w}$ while Gaussians in HMT and HMM-HMT are in $\mathbb{R}^1$.

[3] Using a Intel Core 2 Duo E6600 processor (running in one core only).

The proposed architecture is a composite of an HMM in which the observation probabilities are provided by a set of HMTs. With this structure, the HMM captures the long-term dependencies and the HMTs deal with the local dynamics. The HMM-HMT allows learning from long or variable-length sequences, with potential applicability to real-time processing. The training algorithms were derived using the EM framework, resulting in a set of learning rules with a simple structure.

Empirical results were obtained concerning the application of speech recognition with real data. The recognition rates obtained for speech recognition were very competitive, even in comparison with the state-of-the-art technologies in this application domain.

From this novel architecture, we believe that many topics can be addressed in future works. For example, alternative architectures can be developed with links directly between the HMT nodes (without the external HMM). Moreover, different tying schemes can be used to reduce the total number of trainable parameters, reducing the computational cost and improving the generalization capabilities. More tests would be necessary for the HMT model, with different numbers of states per node and using other observation models within the states (for example, GMM or Laplacian distributions). Concerning to the experiments, in the future we plan to extend our work to continuous speech recognition and using speech contaminated by real non-stationary noises.

Acknowledgment

This work is supported by the National Research Council for Science and Technology (CONICET), the National Agency for the Promotion of Science and Technology (ANPCyT-UNL PICT 11-25984 and ANPCyT-UNER PICT 11-12700), and the National University of Litoral (UNL, project CAID 012-72).

References

[1] P. Baldi and S. Brunak. *Bioinformatics: The Machine Learning Approach.* MIT Press, Cambrige, Masachussets, 2001.

[2] S. Bengio, H. Bourlard, and K. Weber. An EM algorithm for HMMs with emission distributions represented by HMMs. Technical Report IDIAP-RR 11, Institut Dalle Molle d'Intelligence Artificielle Perceptive, Martigny, Switzerland, 2000.

[3] Y. Bengio. Markovian Models for Sequential Data. *Neural Computing Surveys*, 2:129–162, 1999.

[4] P. Bharadwaj and L. Carin. Infrared-image classification using hidden Markov trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(10):1394–1398, October 2002.

[5] C. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.

[6] M. Crouse, R. Nowak, and R. Baraniuk. Wavelet-based statistical signal processing using hidden Markov models. *IEEE Transactions on Signal Processing*, 46(4):886–902, 1998.

[7] N. Dasgupta, P. Runkle, L. Couchman, and L. Carin. Dual hidden Markov model for characterizing wavelet coefficients from multi-aspect scattering data. *Signal Processing*, 81(6):1303–1316, June 2001.

[8] I. Daubechies. *Ten Lectures on Wavelets.* Number 61 in CBMS-NSF Series in Applied Mathematics. SIAM, Philadelphia, 1992.

[9] J.-B. Durand, P. Gonçalvès, and Y. Guédon. Computational methods for hidden Markov trees. *IEEE Transactions on Signal Processing*, 52(9):2551–2560, 2004.

[10] G. Fan and X.-G. Xia. Improved hidden Markov models in the wavelet-domain. *IEEE Transactions on Signal Processing*, 49(1):115–120, January 2001.

[11] S. Fine, Y. Singer, and N. Tishby. The Hierarchical Hidden Markov Model: Analysis and Applications. *Machine Learning*, 32(1):41–62, 1998.

[12] M. Ichir and A. Mohammad-Djafari. Hidden Markov models for wavelet-based blind source separation. *IEEE Transactions on Image Processing*, 15(7):1887– 1899, July 2006.

[13] F. Jelinek. *Statistical Methods for Speech Recognition.* MIT Press, Cambrige, Masachussets, 1999.

[14] S. Kim and P. Smyth. Segmental Hidden Markov Models with Random Effects for Waveform Modeling. *Journal of Machine Learning Research*, 7:945–969, 2006.

[15] S. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.

[16] K. Murphy and M. Paskin. Linear time inference in hierarchical HMMs. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, volume 14, Cambridge, MA, 2002. MIT Press.

[17] L. Rabiner and B. Juang. *Fundamentals of Speech Recognition.* Prentice-Hall, New Jersey, 1993.

[18] O. Ronen, J. Rohlicek, and M. Ostendorf. Parameter estimation of dependance tree models using the EM algorithm. *IEEE Signal Processing Letters*, 2(8):157–159, 1995.

[19] N. Sebe, I. Cohen, A. Garg, and T. Huang. *Machine Learning in Computer Vision.* Springer, November 2005.

[20] K. Stevens. *Acoustic phonetics.* MIT Press, Cambrige, Masachussets, 1998.

[21] K. Weber, S. Ikbal, S. Bengio, and H. Bourlard. Robust speech recognition and feature extraction using HMM2. *Computer Speech & Language*, 17(2-3):195–211, 2003.

[22] A. Willsky. Multiresolution Markov models for signal and image processing. *Proceedings of the IEEE*, 90(8):1396–1458, August 2002.

[23] V. Zue, S. Sneff, and J. Glass. Speech database development: TIMIT and beyond. *Speech Communication*, 9(4):351–356, 1990.