Available online at www.sciencedirect.com
SCIENCE DIRECT®
Speech Communication 48 (2006) 161–175
SPEECH COMMUNICATION
www.elsevier.com/locate/specom

Integration of articulatory and spectrum features based on the hybrid HMM/BN modeling framework

Konstantin Markov [a,*], Jianwu Dang [b], Satoshi Nakamura [a]

[a] *ATR Spoken Language Translation Research Laboratories, Hikaridai 2-2-2, Keihanna Science City, Kyoto 619-0288, Japan*
[b] *Japan Advanced Institute of Science and Technology, Asahidai 1-1, Tatsunokuchi, Nomi, Ishikawa 923-1292, Japan*

Received 14 May 2004; received in revised form 8 June 2005; accepted 14 July 2005

Abstract

Most of the current state-of-the-art speech recognition systems are based on speech signal parametrizations that crudely model the behavior of the human auditory system. However, little or no use is usually made of the knowledge on the human speech production system. A data-driven statistical approach to incorporate this knowledge into ASR would require a substantial amount of data, which are not widely available since their acquisition is difficult and expensive. Furthermore, during recognition, it is nearly impossible to obtain observations of articulators movement. Thus, research on speech production mechanisms in ASR has largely focused on modeling the hidden articulatory trajectories and using prior phonetic and phonological knowledge. Nevertheless, it has been shown that combining the acoustic and articulatory information can lead to improved speech recognition performance. The approach taken in this study is to integrate features extracted from actual articulatory data with acoustic MFCC features in a way that allows recognition using MFCC only. Rather than trying to map articulatory features to the corresponding acoustic features, we use the probabilistic dependency between them. Bayesian Networks (BN) are ideally suited for this purpose. They can model complex joint probability distributions with many discrete and continuous variables and have great flexibility in representing their dependencies. Our speech recognition system is based on the hybrid HMM/BN acoustic model where the BN is used to describe the HMM states' probability distributions. HMM transitions, on the other hand, model the temporal speech characteristics. Articulatory and acoustic features are represented by different variables of the BN. Dependencies are learned from the observable articulatory and acoustic training data. During recognition, when only the acoustic observations are available, articulatory variables are assumed hidden. We have evaluated our ASR system by using a small database consisting of articulatory and acoustic data recorded from three speakers. The articulatory data are actual measurements of articulators position at several points. In all experiments involving both speaker-dependent and multi-speaker acoustic models, the HMM/BN system outperformed the baseline HMM system trained on acoustic data only. In experimenting with different BN topologies, we found that integrating the velocity and

* Corresponding author. Tel.: +81 774 95 1369; fax: +81 774 95 1308.
E-mail addresses: konstantin.markov@atr.jp (K. Markov), dang@jaist.ac.jp (J. Dang), satoshi.nakamura@atr.jp (S. Nakamura).

0167-6393/$ - see front matter
doi:10.1016/j.specom.2005.07.003

acceleration coefficients calculated as first and second derivatives of the articulatory position data can further improve recognition performance.

PACS: 43.72.Ne

Keywords: HMM/BN; Multiple feature integration; Articulatory modeling

1. Introduction

The past few years have seen considerable advances in speech recognition technology. The falling error rates achieved by the state-of-the-art speech recognition systems on some tasks have enabled development of various applications ranging from dictation software for personal computers to automated telephone inquiry services and interactive voice-controlled machinery. However, as the speech recognition tasks become less and less constrained in terms of operating environments and application specifics, researchers are facing increasingly difficult real-world problems. Most of these problems come from a wide range of mismatch factors caused by the variations in speaking styles, talkers, contexts, and noises. In order to find solutions to these problems, recent engineering research has taken various directions, but little or no use has been made of the existing knowledge on the mechanisms of speech production. Indeed, the most widely used speech signal parametrization, MFCC features,[1] has been chosen to crudely model the behavior of the human auditory system. Current statistical speech models are generally unstructured, and parameter learning is done in a blind, data-driven fashion with little attention paid to how the speech data is produced.

There are several motivations to use the knowledge on human speech production in speech recognition. Traditional HMM-based recognizers model speech as a sequence of non-overlapping phonetic units while implicitly assuming that speech can be decomposed into disjoint acoustic segments. Transition from one unit to another is done in an abrupt, discrete way at fixed time steps. On the other hand, speech is formed through continuous movements of articulatory organs from one configuration of "phonetic" targets to the next. In fluent speech, various articulators achieve their target positions at different points in time due to anticipation or preservation of the adjacent phonetic units. This asynchrony causes significant overlap of the articulatory targets and results in modifications of the acoustic segments known as co-articulation phenomena. Since current ASR models are not well suited to model the co-articulation effects,[2] a representation that directly reflects articulatory movements could allow better modeling of transitional regions and more accurate recovery of the original phonetic sequence. Another potential advantage of using speech production information is the fact that articulatory movements are much less affected by the environmental conditions than their acoustic representations. Background noises and room reverberation are factors that have disastrous effects on speech recognition performance. Incorporating articulatory movement representation into ASR systems could make them more robust in non-stationary acoustic environments. Some articulatory targets are also speaker-independent. For example, lip rounding does not depend on such speaker characteristics as vocal tract length and pitch. Finally, articulatory movements can form an information source that preserves some of the information lost during the extraction of

[1] We have to note that the well known LPCC parametrization is based on the Linear Predictive speech model, which crudely represents the human speech production system.

[2] Part of the solution to this problem is to use context-dependent models. Triphones are one popular choice. However, it is still assumed that speech is a sequence of discrete, non-overlapping segments.

speech acoustic features. Thus, by combining articulatory and acoustic parameters, we can increase the separability of phonetic classes and achieve higher recognition accuracy.

The research on human speech production in ASR has taken different directions ranging from simple combinations of articulatory and acoustic features to complex hidden dynamic models of articulatory movements and complete articulatory based systems. Since databases of articulatory movement observations are not widely available, in many studies discrete knowledge-based features are adopted for articulatory parametrization (Kirchhoff, 1998; Kirchhoff et al., 2000; Erler et al., 1995; Gao et al., 2000; Liu, 1996). They usually describe articulation, e.g. *voiced*, *fricative*, *nasal*, etc. and biomechanics, e.g. *positions of tongue*, *lips*, *jaw* and so on. In Kirchhoff, 1998, such articulatory features are extracted from the parametrized speech signal by means of Neural Networks (NN) trained on rule-based mapping tables. Since standard recognizers built on these articulatory features have not performed sufficiently well, the combination of acoustic and articulatory input at different levels (frame, state or word level) has been studied, and promising results have been obtained (Kirchhoff et al., 2000). Knowledge-based features can be used to define the HMM state space, as in the Articulatory Feature Model (AFM) (Erler et al., 1995). In this model, each articulatory feature vector corresponds to one HMM state. In this way, it is possible to cover the entire acoustic space with one large HMM. Different phonetic units are specified as different paths through this HMM corresponding to the respective articulatory configuration sequences. A common disadvantage of such approaches is the quantization of the continuous articulatory parameters, where much of the dynamic information is lost. In order to model the co-articulation effect better and to account for the continuous articulatory movement, the discrete articulatory vectors can be regarded as "targets" of trajectory-based models. In Gao et al., 2000, a non-causal Kalman filter is used to smooth target positions and generate "realized" articulations that are further transformed into cepstrum vectors by NN. Such a model can be directly applied in speech synthesis or used for *N*-best re-scoring in speech recognition. A stochastic target model is discussed in Deng et al., 1996, where articulatory trajectories are represented by a linear state-space system. Targets, however, are drawn from distributions depending on the current state of the semi-Markov chain representing a given phonological sequence. In the so-called task-dynamic model, articulatory dynamics are described in terms of a task-variable that represents vocal tract (VT) construction degrees and locations of VT resonances (Deng, 1998). A second-order dynamic system defines the movement of the task-variable from one target to the next. Implementation of this task-dynamic model into speech recognition systems can be realized by trended HMM (Deng, 1992) with a specific trend function. An essential issue in building articulatory models with knowledge-based features or targets is the selection of the feature set. Certain features deemed necessary from a phonetic point of view might in practice turn out to be strongly correlated and not optimal in terms of discrimination performance. The size of the feature set is also important. Too few features may result in a very crude and simplistic model. On the other hand, more features would allow for greater precision in trajectory generation, but the complexity of the model and its implementation cost might be prohibitive in practice.

In contrast to the models based on discrete features, the MALCOM algorithm introduces the concept of continuous articulatory space (Hogden and Valdez, 2000; Hogden and Valdez, 2001). The underlying assumption is that data representing speech acoustics are produced by objects moving smoothly through an abstract hidden space called the Continuity Map (CM). A stochastic model resembling the HMM is used to represent the probabilistic mapping between discrete acoustic data and continuous CM positions. Even though the MALCOM algorithm may look unconventional, it resembles the other methods based on discrete articulatory features in the sense that in modeling the non-linear mapping between two continuous spaces, both approaches require one of the spaces to be discrete.

Relatively few studies involve physically recorded articulatory data (Papcun et al., 1992;

Zacks and Thomas, 1994). Such data are usually collected using X-ray filming or some more advanced techniques like magnetic resonance imaging (MRI), electromagnetic articulography (EMA), and electropalatography (EPG). Articulatory parameters obtained from actual measurements describe articulation in a more fine-grained manner. The main problem with such data, however, is that direct observations are usually not available during recognition. A common approach is to estimate articulatory test data from the acoustic signal using Neural Networks. Although this technique faces the same difficulties as knowledge-based feature methods, acoustic-articulatory mapping performed by an NN is trained on actual data and is able to capture co-articulation effects more precisely. Unfortunately, large-scale articulatory databases are not widely available, and this approach has only been applied so far for small tasks like vowel identification (Zacks and Thomas, 1994).

In this study, we also make use of actual articulatory data. Our database consists of simultaneous recordings of speech signal and articulatory positions obtained with an EMA system. Rather than trying to learn the mapping between acoustic and articulatory data, we consider them as random variables and model their probabilistic dependencies. Bayesian Networks (BN) (Jensen, 1998) are ideally suited for this purpose. They can model complex joint probability distributions with many discrete and continuous variables and have great flexibility in representing their dependencies. Since conventional BN cannot handle temporal processes like speech, Dynamic Bayesian Networks (DBN) have been developed (Dean and Kanazawa, 1988). Unfortunately, because of their high complexity, so far DBNs have only been applied for small tasks like isolated word or continuous digit recognition (Stephenson et al., 2001; Daoudi et al., 2001). An alternative to DBN is the hybrid HMM/BN model (Markov and Nakamura, 2003a), which is used in our speech recognition system. In this model, BN represents the states output probability distributions and HMM governs the temporal speech behavior. In this way, the size of the BN is kept very small and as we show later, the overall model complexity is similar to that of the traditional HMM. Furthermore, in most cases, the state output probability function can be reduced to the form of a Gaussian mixture allowing the HMM/BN model to be used directly in the standard HMM based decoders. Articulatory and acoustic parameters are represented by different variables of the BN. Dependencies are learned from the available training data. During recognition, however, articulatory variables are assumed hidden. Therefore, only acoustic observations are needed to perform the recognition task. In our first experiments, we integrated only articulatory position parameters using simple BN topology (Markov et al., 2003). Evaluation results showed that combined acoustic and articulatory features perform much better than acoustic features alone. Next, we extended these experiments to include articulatory velocity and acceleration parameters. Various BN topologies combining these parameters were studied and are described in this paper. As the results suggest, integrating the articulatory dynamic features can achieve positive effect after careful data analysis and selection of appropriate BN topology.

2. Hybrid HMM/BN Model

In this section, we give a brief description of the hybrid HMM/BN model as well as its training algorithm and implementation. We show that the conventional HMM is actually a HMM/BN with a particular BN topology.

2.1. Background

The HMM/BN model is a combination of an HMM and a Bayesian Network. Speech temporal characteristics are modeled by the HMM state transitions while the HMM states' probability distributions are represented by the BN. A block diagram of the HMM/BN is shown in Fig. 1. Structurally, the HMM/BN model is analogous to the hybrid HMM/NN model (Bourlard and Morgan, 1994). The difference is that instead of a Neural Network, the HMM is coupled with a Bayesian Network. The NN is known for its strong classification abilities, but the BN offers

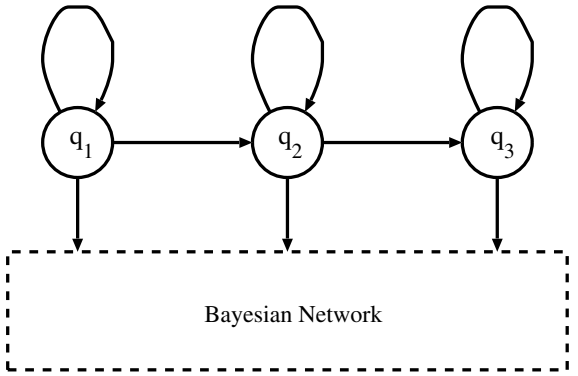

Fig. 1. HMM/BN model structure. HMM transitions model speech temporal characteristics and BN represents states' probability distributions.

greater flexibility in data distribution modeling and can combine different variables (features) in a simple and consistent way. Furthermore, NN estimates state posterior probability rather than the data likelihood needed by the HMM algorithms. In contrast, data likelihood is obtained from the BN directly, which allows for seamless integration with the HMM.

By definition, a Bayesian Network represents a joint probability distribution of a set of random variables $Z_1,\ldots,Z_N$ and is expressed by a directed acyclic graph (DAG), where each node corresponds to a unique variable. Arcs between the nodes show the conditional dependencies of the BN variables. Immediate predecessors of variable Z_i are called its *parents* and are referred to as $\mathrm{Pa}(Z_i)$. The BN joint probability distribution function can be factored as (Jensen, 1998):

$$P(Z_1,\ldots,Z_N)=\prod_{i=1}^{N}P(Z_i|\mathrm{Pa}(Z_i)) \tag{1}$$

Let us now consider a simple BN with one discrete variable $Q=\{q_i\}$, $i=1,\ldots,S$ and one continuous multi-dimensional variable X as shown in Fig. 2.[3] X depends on Q, and this dependency is defined by the conditional probability $P(X|Q)$. Since X is continuous, we can use a set of S Gaussian functions (one for each q_i) to express $P(X|Q)$ in parametric form. It is obvious that this BN represents the data distribution of a traditional HMM with S states and a single Gaussian per state. The likelihood of input data x_t with respect to state q_i is simply:

Fig. 2. Simple BN with one discrete variable Q and one continuous variable X.

$$p(x_t|q_i)=P(X=x_t|Q=q_i) \tag{2}$$

In practice, the HMM state distribution is often modeled with a mixture of Gaussian functions. BN topology corresponding to this case is shown in Fig. 3, where a new discrete variable $M=\{m_j\},j=1,\ldots,K$ represents the mixture component index. This variable is hidden since we do not know which Gaussian distribution the input data is drawn from. The data likelihood $p(x_t|q_i)$ is calculated using BN joint probability function (Eq. (1)) as follows:

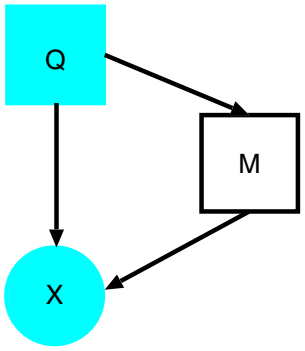

Fig. 3. BN representing mixture of Gaussians.

[3] In this and the following figures, square (circle) node will correspond to a discrete (continuous) variable, while hidden (observable) variables will be shown in clear (shaded) nodes.

$$
\begin{aligned}
P(X=x_t|Q=q_i) &= \frac{P(X=x_t, Q=q_i)}{P(Q=q_i)} \\
&= \frac{\sum_{j=1}^{K} P(X=x_t, M=m_j, Q=q_i)}{P(Q=q_i)} \\
&= \frac{\sum_{j=1}^{K} P(X=x_t|M=m_j, Q=q_i)P(M=m_j|Q=q_i)P(Q=q_i)}{P(Q=q_i)} \\
&= \sum_{j=1}^{K} P(M=m_j|Q=q_i)P(X=x_t|M=m_j, Q=q_i)
\end{aligned}
\tag{3}
$$

If we replace $P(M=m_j|Q=q_i)$ with w_{ji} and $P(X=x_t|M=m_j, Q=q_i)$ with Gaussian function $N(x_t; \mu_{ji}, \Sigma_{ji})$, we get a standard mixture of Gaussians equation:

$$
p(x_t|q_i) = \sum_{j=1}^{K} w_{ji} N(x_t; \mu_{ji}, \Sigma_{ji}) \tag{4}
$$

Fig. 3 allows us to interpret the Gaussian mixture distribution in a different way. It shows that observation variable X depends not only on the state index but also on the variable M. However, M has no physical meaning. In this respect, Gaussian mixture learning is "blind" and does not reflect the way a speech signal is produced or at least does not account for the factors it depends on, such as speaker gender, environmental noises, communication channels, etc. Variable M, for example, could represent articulatory configuration or some other parameter that effects the speech spectrum. The BN can have more variables corresponding to different speech features or variability factors. Dependencies can be set according to prior knowledge or data correlation analysis. In this way, we can impose knowledge-based "structure" on the speech generation process and achieve a more precise speech model. Ideally, the BN structure should be learned automatically from the training data, but this is a very difficult task (Heckerman, 1998) and, usually, BN topology is chosen manually by taking into account the available data and the task at hand (Markov and Nakamura, 2003a,b).

2.2. HMM/BN model training

As in the case of the HMM/NN model, parameter learning of the HMM/BN is based on the Viterbi training paradigm and can be summarized in the following algorithm.

- Step 1. Initialization.
- Step 2. Viterbi alignment.
- Step 3. Update BN parameters.
- Step 4. Update HMM transition probabilities.
- Step 5. Stop or go to Step 2.

First, we choose the HMM/BN state number and the BN topology, and then we initialize their parameters. Since the state variable Q is observable, before BN training we need to obtain its values for each sample of X. This is done by the Viterbi alignment step. For BN parameter estimation, several methods are available. In the simplest case, when all variables are observable, maximum likelihood (ML) estimates can be computed in closed form.[4] In a partially observed case, i.e. when some of the (discrete) variables are hidden, the Expectation–Maximization (EM) algorithm can be applied. After BN is trained and its parameters fixed, the HMM transition probabilities are re-estimated with a standard forward–backward algorithm. All of these steps are repeated until the convergence criterion is met. This can be an increase in data likelihood or simply a fixed number of iterations.

2.3. Recognition with HMM/BN model

For recognition, traditional ASR systems use a decoder that finds the most probable phonetic unit sequence based on the input data likelihood and transition probabilities obtained from the acoustic model.[5] With the HMM/BN, data likelihood $P(X|Q)$ is inferred from the BN, and transition probabilities are available from its HMM part. For simple BN topologies, $P(X|Q)$ can be calculated in closed form. When this is not possible, a number of exact and approximate inference algorithms can be used (Cowell, 1998).

[4] This is true under the condition where continuous variables have no children. Otherwise, we need to model dependency on a continuous parameter which is not a trivial problem.

[5] Assuming no language model is used.

There is a special class of BNs for which the data likelihood inference can be reduced to a Gaussian mixture calculation. This is practically useful because in this case, the HMM/BN model is computationally equivalent to the HMM, and there is no need to modify the decoder in order to use it. BNs belonging to this special class satisfy the following three conditions:

- All variables except X are discrete.
- All variables except Q and X are hidden.
- Variable Q has no parents and variable X has no children.

Indeed, for an arbitrary BN satisfying the above conditions and having joint pdf $P(X, Q, Z_1, \ldots, Z_K)$, from Eqs. (1) and (3), we have:

$$P(X = x_t|Q = q_i) = \sum_{z_1} \cdots \sum_{z_K} \prod_{i=1}^{K} P(Z_i = z_i|\mathrm{Pa}(Z_i)) \times P(X = x_t|\mathrm{Pa}(X)) \quad (5)$$

Since all Z_i and their parents are discrete, the product $\prod_{i=1}^{K} P(Z_i = z_i|\mathrm{Pa}(Z_i))$ is a simple number. The $P(X|\mathrm{Pa}(X))$ is a Gaussian function and therefore the above equation represents a Gaussian mixture.

The second condition, however, is not required for the BN training. In fact, when we train the BN, all variables can be observable. By making a variable hidden during recognition only, we eliminate the need for the corresponding observations. This scenario is very well suited for the task of this study, since the articulatory data may be available for training but are difficult to obtain during recognition.

3. Articulatory data and baseline HMM system

The articulatory data used in this study were collected by using the Electromagnetic Midsagittal Articulographic (EMA) system at NTT, Japan (Okadome and Honda, 2001; Hiroya and Honda, 2004). In the EMA system, a number of miniature coils are attached to points in the vocal tract. The subject's head is then placed in an electromagnetic field, allowing the movement of the coils to be inferred from the corresponding induced voltages. The output of the system is a set of x and y traces for articulatory movement. Fig. 4 shows the placement scheme of the coils used in the data collection. Four coils were placed on the tongue surface in the midsagittal plane, named T1–T4, and one coil each for the upper lip, lower lip, maxilla incisor, mandible incisor (LJ), and velum. The maxilla incisor was chosen as the origin of the coordinate system as shown in the figure. Acoustic signal and articulatory traces were recorded simultaneously. The sampling rate was 250 Hz for the articulatory channels and 12 kHz for the acoustic channel. All articulatory data were subsequently corrected for head movements and rotated to bring the occlusal plane into coincidence with the horizontal axis. The speech material consisted of 350 randomly selected Japanese sentences (about 25 min) that were read at normal speed by three male subjects (MH, TO and TM).

Our baseline system uses the conventional HMM. The speech model consists of 29 context-independent phonemes, and each of them is represented by a 3-state left-to-right HMM. As training data, we chose 300 utterances from each speaker, and the remaining 50 we used as test data. In total, we had 900 training and 150 test sentences. We trained three types of baseline models with three different speech parametrizations. The first one is MFCC extracted from the acoustic speech signal

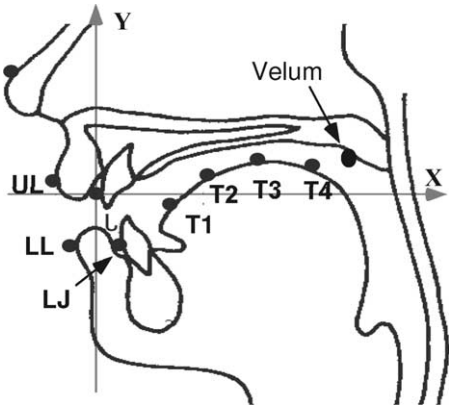

Fig. 4. Placement of the coils in the EMA recording, and the coordinate system used in this study.

at a 8 ms frame rate and a 20 ms frame window. The feature vector consists of 16 MFCC coefficients and their first and second derivatives. The second parameter type includes articulatory position, velocity, and acceleration data. Traces from the eight observation points are time-varying vectors with 16 components (x- and y-coordinates) obtained every 4 ms. Since the MFCC frame rate is 8 ms, we used every second position vector in order to have time synchronous pairs of acoustic and articulatory parameters. Articulatory velocity and acceleration coefficients were obtained as first and second derivatives of the position data in the same way as in the MFCC case. For the third parameter type, we combined the acoustic and articulatory data into one feature vector. Such features are of no practical use, since articulatory observations are needed during recognition, but they can show the potential effect of combining the acoustic and articulatory data. In order to keep the same vector dimension, we used MFCC static and first delta coefficients and articulatory position data. Up to 10 iterations of the Baum–Welch algorithm were performed in all baseline models training. The software tool used in both recognition and training was the HTK toolkit (Young, 1999).

Phoneme recognition rates obtained from the three different types of features using both speaker-dependent and multi-speaker models are summarized in Table 1. Simple phoneme pair grammar was used as the language model. All HMM states have 12 Gaussian mixture components. The results show that articulatory data alone is not a good candidate for speech representation, but when combined with the acoustic data, performance clearly improves. We also investigated the effect of articulatory velocity and acceleration parameters. Table 2 shows the results when delta (velocity) and delta–delta (acceleration) coefficients are gradually added in the feature vector for both MFCC and articulatory position features. As can be seen from this table, articulatory velocity and acceleration parameters are quite effective, and in comparison with the MFCC delta and delta–delta coefficients, their contribution to system performance is much bigger.

Table 2
Phoneme recognition accuracy obtained using static (S), delta (Δ) and delta–delta ($\Delta\Delta$) coefficients of the acoustic MFCC and articulatory position data

	Acoustic MFCC			Articulatory position		
Speaker	MH	TM	TO	MH	TM	TO
S	79.55	81.15	73.20	68.83	63.21	58.13
S + Δ	86.44	86.93	77.63	75.74	73.15	66.44
S + Δ + ΔΔ	86.55	87.03	78.55	80.44	79.52	75.31

4. Articulatory and acoustic feature integration

Since both the acoustic and articulatory features are real valued vectors, direct integration using the HMM/BN model is quite difficult. In order to make this task feasible, we transform the articulatory parameters into discrete data by using Vector Quantization (VQ). Of course, some information will be lost, but this is a trade-off between the model's accuracy and its complexity. If we were to integrate articulatory position parame-

Table 1
Phoneme recognition accuracy obtained with three different feature vectors and two types of models

Feature	Speaker-dependent model			Multi-speaker model		
	AC	ART	AC + ART	AC	ART	AC + ART
Speaker MH	86.55	80.44	89.09	81.96	67.69	86.82
Speaker TM	87.03	79.52	88.6	83.09	71.53	84.55
Speaker TO	78.55	75.31	82.77	73.10	66.61	80.88
Average	84.04	78.42	86.82	79.38	68.61	84.08

AC: MFCC + ΔMFCC + ΔΔMFCC; ART: articulatory position + velocity + acceleration; AC + ART: MFCC + ΔMFCC + articulatory position.

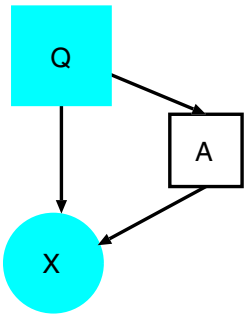

Fig. 5. Simple BN integrating continuous acoustic (X) and discrete articulatory (A) data. Variable A is observable in training but hidden during recognition.

ters only, then the simplest way would be to use the BN shown in Fig. 5. This BN is similar to the BN from Fig. 3 discussed in Section 2.1. Instead of the mixture variable M, we have a new variable A representing articulatory position data, which is observable in training but hidden during recognition. Articulatory variable A depends on state Q because sub-phonetic units represented by the state variable are realized by different articulatory configurations and therefore different values of A. The probabilistic dependency of states and articulatory positions is expressed by the arc between them.

As our baseline system's results suggest, articulatory velocity and acceleration coefficients could also be helpful. The most straightforward approach is to concatenate the articulatory position feature vector with velocity and acceleration parameters as we did for our baseline system, then apply vector quantization and combine it with the acoustic MFCC by using the same BN as in Fig. 5. The acoustic data likelihood in this case is simply mixture of Gaussians,[6] and mixture weights are the conditional probabilities of the articulatory variable given the state index:

$$p(x_t|q_i) = \sum_{j=1}^{K} P(A = a_j|Q = q_i)P(X = x_t|A = a_j, Q = q_i) \tag{6}$$

where K is the size of the articulatory VQ codebook and x_t is the acoustic feature vector consisting of MFCC static, delta, and delta–delta coefficients. This method, however, does not make use of the BN's flexibility and power in modeling data dependencies. Indeed, we have every reason to believe that articulatory velocity has a much bigger effect on the spectral change than on the spectrum itself. In other words, we can reasonably assume that MFCC delta coefficients (mostly) depend on articulatory velocity parameters. The same holds for the dependency between MFCC delta–delta coefficients and articulatory acceleration parameters. A BN that expresses these dependencies is shown in Fig. 6, where variables X_s, X_v and X_a correspond to MFCC static, delta, and delta–delta components. Variables A_s, A_v and A_a represent articulatory position, velocity and acceleration parameters. Vector quantization of these parameters can be done independently using codebooks of different sizes: K_s, K_v and K_a. Again, articulatory variables are observable in training but assumed hidden during recognition. According to this BN, the likelihood of x_t is calculated as:

$$\begin{aligned} p(x_t|q_i) &= \prod_{n\in\{s,v,a\}} \sum_{j=1}^{K_n} P(A_n = a_j^n|Q = q_i) \\ &\quad \times P(X_n = x_t^n|A_n = a_j^n, Q = q_i) \\ &= \prod_{n\in\{s,v,a\}} P(X_n = x_t^n|Q = q_i) \end{aligned} \tag{7}$$

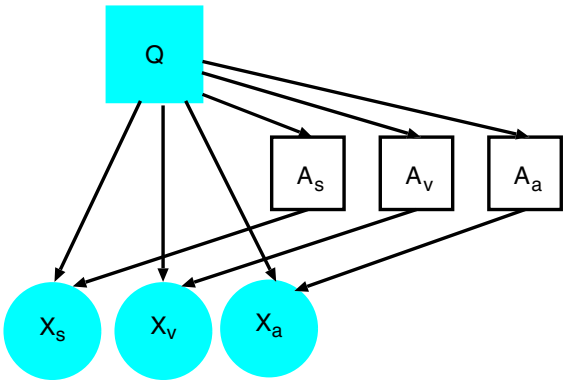

Fig. 6. BN structure modeling corresponding dependencies between MFCC static, delta, and delta–delta coefficients and articulatory position, velocity and acceleration parameters.

[6] In this and the following equations, it is implicitly assumed that conditional probabilities of X are modeled with Gaussian functions.

The above equation is simply a product of the MFCC static x_t^s, delta x_t^v and delta–delta x_t^a vector likelihoods, each of which is computed as a Gaussian mixture. This is actually the same as the well known case of multi-stream data likelihood calculation and is supported by many ASR decoders. One serious drawback of the multi-stream method is that any useful correlation between data streams is lost, and this often has a negative effect on system performance. A BN structure free from this problem is shown in Fig. 7, where concatenated MFCC static, delta, and delta–delta coefficients are represented by X. This is similar to the BN from Fig. 5, but now X depends explicitly on the three articulatory variables. In addition, the possible correlation between articulatory position, velocity and acceleration is taken into account by making them dependent on each other. The output likelihood obtained from this BN structure is as follows:

$$\begin{aligned} p(x_t|q_i) = \sum_{j=1}^{K_s}\sum_{n=1}^{K_v}\sum_{m=1}^{K_a} & P(A_s = a_j^s|Q = q_i) \\ & \cdot P(A_v = a_n^v|A_s = a_j^s, Q = q_i) \\ & \cdot P(A_a = a_m^a|A_v = a_n^v, Q = q_i) \\ & \cdot P(X = x_t|A_s = a_j^s, A_v = a_n^v, A_a = a_m^a, Q = q_i) \end{aligned} \tag{8}$$

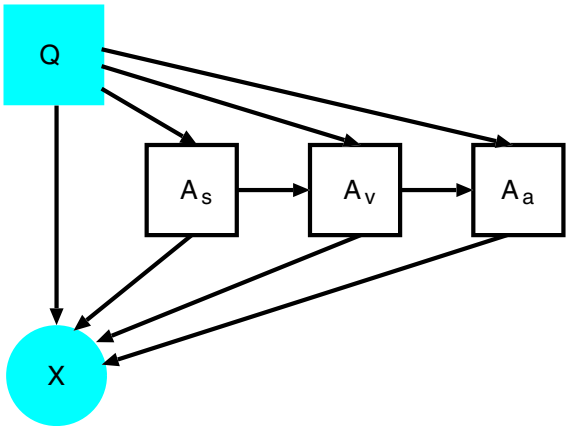

Fig. 7. BN structure explicitly modeling dependencies between acoustic, articulatory position, velocity and acceleration variables.

A closer look at this equation reveals that it is also a mixture of Gaussians equation. Indeed, the first three terms of the right side are discrete probabilities, and their product $P(A_s = a_j^s|Q = q_i)P(A_v = a_n^v|A_s = a_j^s, Q = q_i)P(A_a = a_m^a|A_v = a_n^v, Q = q_i)$ is simply the weight of the corresponding Gaussian mixture component $P(X = x_t|A_s = a_j^s, A_v = a_n^v, A_a = a_m^a, Q = q_i)$, which can be calculated in advance.

5. Experiments and results

Since the BN articulatory variables are discrete, before HMM/BN training, all of the articulatory data had to be quantized. First, we reduced the vector dimension to four by the principal component analysis (PCA) technique. The estimated information loss from this procedure in all cases was less than 15%. Then, for each articulatory parameter type (position, velocity, acceleration as well as concatenation of all three) we trained VQ codebooks of different sizes ranging from 4 to 1024. These codebooks were used to quantize the corresponding type of data, and their VQ labels served as articulatory observations for the BN training. Observations of the state variable Q were obtained using Viterbi alignment as described in Section 2.2. Acoustic feature vectors for variable X were the same as those used in the baseline HMM. Thus, all BNs were fully observable, and ML training was sufficient for the BN parameter estimation. In order to reduce the number of iterations in the HMM/BN training, instead of initializing its parameters randomly, we used the baseline HMM trained on acoustic data only as a bootstrap model. Transition probabilities of this model were taken as initial values of the corresponding HMM/BN state transitions. The bootstrap model was also used in the Viterbi alignment step of the first training iteration to obtain good initial state segmentation. After such initialization, one or two training iterations were performed for all of the HMM/BN models. As explained in Section 4, the HMM/BN state output probability can be reduced to a single- or multi-stream Gaussian mixture form, and since the number of states of both the baseline and HMM/BN models is the

Table 3
Phoneme accuracy (%) for speaker-dependent baseline and HMM/BN models

Speaker MH		Speaker TM		Speaker TO	
HMM	HMM/BN	HMM	HMM/BN	HMM	HMM/BN
84.6(4)	84.76(3.7)	84.55(4)	84.76(4.1)	74.88(4)	75.53(3.9)
85.9(8)	86.76(7.6)	85.58(8)	87.14(7.9)	76.23(8)	77.9(7.8)
86.55(12)	86.44(12.3)	87.03(12)	87.47(12.1)	78.55(12)	78.95(12.0)
84.93(16)	86.61(15.8)	85.25(16)	87.53(16.1)	75.63(16)	79.69(15.9)

Digits in parenthesis indicate the average number of mixture components per state.

same, the only difference between them becomes the number of mixtures and the way they are trained.

The experimental conditions in all HMM/BN performance evaluations were the same as for the baseline HMM models described in Section 3 except that the test data consisted of acoustic observations only (MFCC, ΔMFCC, ΔΔMFCC). In the first series of experiments we used only position data as articulatory features, since we wanted to investigate the contribution of the static and dynamic features separately. Three speaker-dependent HMM/BN models with BN topology from Fig. 5 were trained and their phoneme recognition rates are summarized in Table 3. For comparison, the results of the baseline HMMs trained on acoustic data only are given in the "HMM" columns. Digits in parenthesis indicate the number of Gaussians per state, which for the HMM/BN models is an average number because different states have a different number of mixture components. As the results show, the HMM/BN model is better in almost all cases. This suggests that integration of the articulatory features was effective and that the additional information they provide during training resulted in more precise acoustic models. Interestingly enough, HMM/BN performance keeps improving as the number of Gaussians increases, while HMM models showed the highest results with only 12 mixture components per state.

So far, we have not discussed the impact of the VQ codebook sizes on the HMM/BN model complexity and, indirectly, on its performance. Considering the BN from Fig. 5, we can say that the codebook size K determines each state's mixture component number as evident from Eq. (6). However, this is only the maximum possible number. The actual mixture number for each state depends on the joint probability distribution $P(A, Q)$, which is not uniform. For example, if the joint probability $P(A = a_j, Q = q_i)$ is zero, then state q_i would not have the mixture component $P(X|A = a_j, Q = q_i)$. Therefore, in the HMM/BN model, different states have a different number of Gaussians corresponding to the shape of the data distribution. Furthermore, the amount of acoustic data aligned to each state is roughly proportional to the number of mixture components because $P(A, Q)$ is a marginal distribution of $P(A, Q, X)$.[7] This is very important because in this respect, a nearly optimum mixture number is maintained for each state if the value of K is chosen properly. Unfortunately, there is no principled a priori way to select the codebook size. A small K may result in an under-trained model, while big values can lead to model over-training.

Next, we evaluated the performance of the HMM/BN model using both articulatory static and dynamic features with BNs of different topologies presented in the previous section. For convenience, the model with BN from Fig. 5 will be referred to as HMM/BN1 and those with BNs from Figs. 6 and 7 as HMM/BN2 and HMM/BN3, respectively. In all cases, articulatory features include positions, velocity and acceleration parameters. To illustrate the effect of articulatory dynamic data on model performance, the results of these experiments are shown in Table 4 along with the results of HMM/BN1

[7] Strictly speaking, this is true if the data are drawn randomly from $P(A, Q, X)$ and it represents the actual data distribution. In practice, however, these conditions are met only to a certain extent.

Table 4
HMM/BN model phoneme recognition accuracy (%) obtained with three different BN structures using different articulatory feature sets

	Position data only	Position, velocity and acceleration data		
	HMM/BN1	HMM/BN1	HMM/BN2	HMM/BN3
Speaker MH	86.61	85.75	85.90	87.12
Speaker TM	87.53	85.95	86.20	87.72
Speaker TO	79.69	77.02	77.45	79.85

from the previous tests. The VQ codebook sizes were chosen so that all types of models had roughly the same number of Gaussian mixture components. As the results show, including the articulatory velocity and acceleration parameters has a positive effect only with the HMM/BN3 model. The other two showed a degradation of performance. In the HMM/BN1 case, where all three types of articulatory features are concatenated, the PCA-based dimension reduction retains those components that have the biggest variance. The data analysis we did showed that position parameters had the lowest eigenvalues and therefore could be lost in the transformation. The reason for the low results of the HMM/BN2 model is most probably the fact that the acoustic feature vector is split into static, delta and delta–delta parts. This, usually, leads to performance degradation that, in this case, may have diminished the gain provided by the articulatory dynamic parameters.

Finally, we investigated the performance of the best of our models, HMM/BN3, as a function of its parameter number. By varying the VQ codebooks sizes, we obtained several models with different numbers of mixture components. Note that in the HMM/BN3 case, the maximum possible Gaussian number is $K_s \times K_v \times K_a$. Phoneme recognition rates for each speaker dependent model are plotted in Figs. 8–10 along with the results obtained from the corresponding baseline HMMs. The acoustic-only HMM is marked as HMM(AC), and the HMM trained using concatenated acoustic and articulatory features is denoted as HMM-(AC + ART). As can be seen, all figures exhibit the same pattern. The HMM/BN3 model always performs better than HMM(AC), but still not as well as HMM(AC + ART). As explained

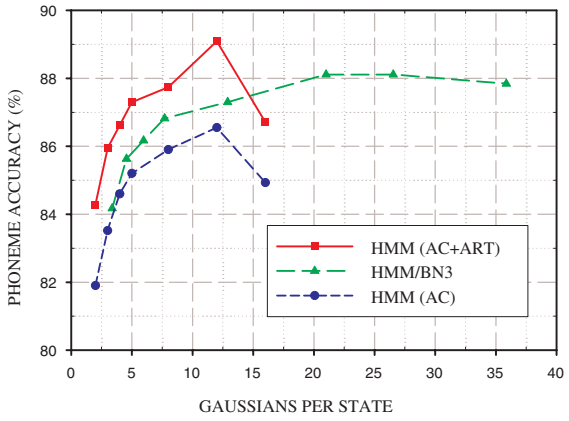

Fig. 8. Performance of HMM/BN3 and two baseline HMMs for speaker MH.

in Section 3, the HMM(AC + ART) model is of no practical use because it requires articulatory observations during recognition; however, we regard its results as a kind of upper bound for the HMM/BN performance. The plots also show that the baseline recognition rates start degrading after the mixture component number reaches 12 Gaussians per state. In contrast, the best HMM/BN3 results were obtained with roughly two times as many model parameters. This is probably because the baseline HMMs have the same mixture number for each state,[8] and given the limited amount of training data, this soon leads to parameter overtraining. In the case of HMM/BN3, however, there is a better balance between the amount of training

[8] Although there are techniques that attempt to optimize the number of Gaussians, such as Chen and Gopalakrishnan (1998), the common approach is to use the same manually set mixture number for each state.

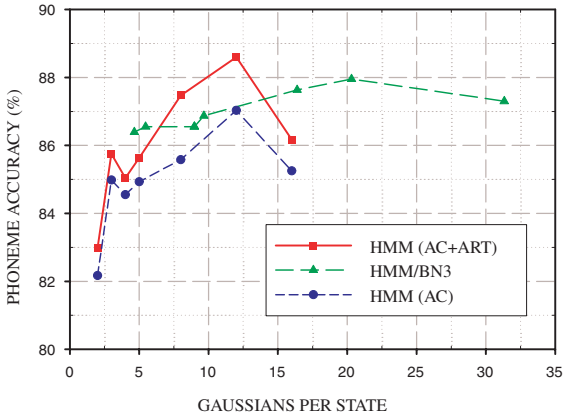

Fig. 9. Performance of HMM/BN3 and two baseline HMMs for speaker TM.

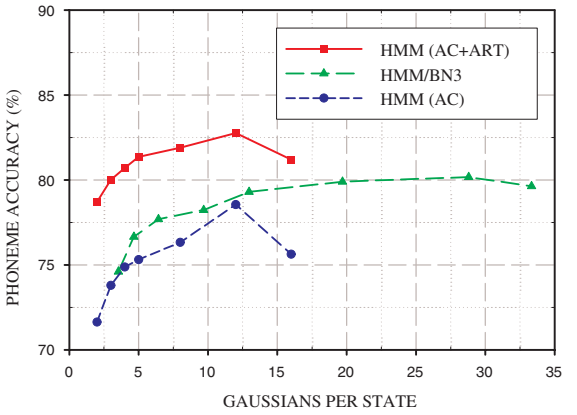

Fig. 10. Performance of HMM/BN3 and two baseline HMMs for speaker TO.

data per state and the number of Gaussians, so the over-training appears at a much larger mixture number. In order to assess the statistical significance of the obtained results, we performed the Wilcoxon matched-pairs signed-ranks test using as input samples the utterance level phoneme accuracy scores of the HMM(AC) and HMM/BN3 for the condition (Gaussians/state) that gives the best overall result for each system. We chose this test because it is non-parametric and takes into account the differences between the samples, which is important in this case. For each speaker, there are 50 pairs of samples (50 test utterances) and the p-value is 0.008, 0.1 and 0.02 for MH, TM and TO respectively. Comparing with the commonly used statistical significance threshold of 0.05, it is clear that for two of the speakers the obtained results are statistically significant. Nevertheless, because of the small number of speakers, the overall p-value (from only three sample pairs) is about 0.25. This suggests that more speakers are needed in order to prove the validity of the proposed approach. We repeated these experiments using models trained on data from all three speakers. The test set consisted of each speaker's test data pooled together. The phoneme recognition results obtained are plotted in Fig. 11. In this case, the HMM/BN3 model performed much better, achieving the same accuracy as HMM(AC + ART). The utterance level results' significance test showed a p-value that was practically zero. As this was not the result we anticipated, we looked for possible factors that could have boosted the HMM/BN3 performance. We made a histogram of the articulatory data's VQ labels and found three almost clearly formed clusters corresponding to the speakers. This means that in BN3 training, most of the Gaussians have been trained with a single speaker's data, and the whole model resembles an interpolation of the speaker-dependent models. Although this situation is a result of the small number of speakers in our database, it suggests that the HMM/BN can better handle the inter-speaker variabilities.

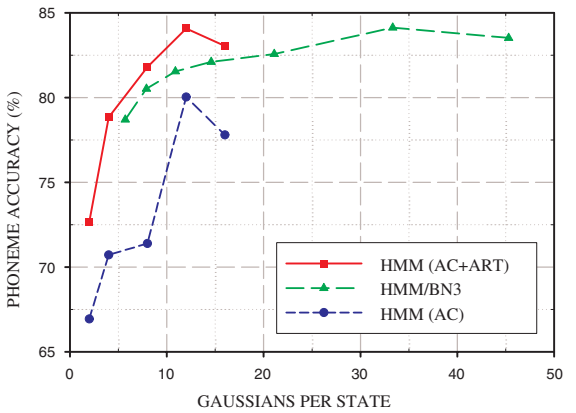

Fig. 11. Performance of HMM/BN3 and two baseline HMMs in multi-speaker case.

6. Conclusions

In this paper, we presented a speech recognition system where actual articulatory data are effectively integrated with the speech acoustic features. In contrast to other methods based on explicit mapping of the acoustic data into articulatory feature space, we use the probabilistic dependency between the two types of speech parameters. This dependency is learned by the hybrid HMM/BN model, where acoustic and articulatory data are represented by different BN variables. Although, observable during training, articulatory variables are assumed hidden in the test phase, allowing us to perform recognition using acoustic data only. The evaluation experiments showed that the HMM/BN model was able to effectively utilize the available articulatory information. Its performance was always better than the baseline HMM trained only on acoustic features. There is still room for improvement, however, as indicated by the comparison with the HMM built on concatenated acoustic and articulatory data. Indeed, the PCA-based dimension reduction and the VQ transformation of the continuous articulatory vectors led to some information loss and, therefore, to sub-optimal performance. Furthermore, we should take into account other factors that have indirect impact on the results. Our database consists of data from three speakers only, and the improvement we achieved, although quite noticeable, is still not statistically significant. Another issue is the effect of the receiving coils on the speech intelligibility during recording by the EMA system. Even though these coils are very small, it is possible that they may cause some people to change their natural pronunciation. As for the model itself, the HMM/BN still does not make full use of the information the articulatory features can provide. For example, the transitional regions are modeled as in standard HMM systems. Also, the knowledge-based articulatory features, which are noise robust and less speaker dependent, are not used, but could be easily integrated in the BN along with the position parameters.

In order to address all of these issues, we are planning further investigations involving more speakers and larger amounts of speech data.

Acknowledgments

The research reported here was supported in part by a contract with the Ministry of Public Management, Home Affairs, Posts and Telecommunications entitled "Multilingual speech-translation systems using mobile terminals". The authors especially thank Dr. M. Honda for allowing us to share the articulatory data.

References

Bourlard, H., Morgan, N., 1994. Connectionist Speech Recognition: A Hybrid Approach. Kluwer Academic Publishers, Boston.

Chen, S., Gopalakrishnan, P., 1998. Clustering via the Bayesian information criterion with applications in speech recognition. In: Proc. ICASSP, May, Vol. 2, pp. 645–648.

Cowell, R., 1998. Introduction to inference for Bayesian Networks. In: Jordan, M. (Ed.), Learning in Graphical Models. Kluwer Academic Publishers, pp. 9–26.

Daoudi, K., Fohr, D., Antoine, C., 2001. Continuous multiband speech recognition using Bayesian Networks. In: Proc. ASRU.

Dean, T., Kanazawa, K., 1988. Probabilistic temporal reasoning. In: AAAI, pp. 524–528.

Deng, L., 1992. A generalized hidden Markov model with state-conditioned trend functions of time for the speech signal. Signal Process. 27, 65–78.

Deng, L., 1998. A dynamic, feature-based approach to the interface between phonology and phonetics for speech modeling and recognition. Speech Comm. 24 (4), 299–323.

Deng, L., Ramsay, G., Sun, D., 1996. Production models as a structural basis for automatic speech recognition. In: ESCA Tutorial and Research Workshop on Speech Production Modeling, pp. 69–80.

Erler, K., Freeman, J., 1995. Using articulatory features for speech recognition. In: Proc. IEEE Conference on Communications, Computers and Signal Processing, pp. 562–566.

Gao, Y., Bakis, R., Huang, J., Xiang, B., 2000. Multistage coarticulation model combining articulatory, formant and cepstral features. In: Proc. ICSLP, pp. 25–28.

Heckerman, D., 1998. A tutorial on learning with Bayesian Networks. In: Jordan, M. (Ed.), Learning in Graphical Models. Kluwer Academic Publishers, pp. 301–354.

Hiroya, S., Honda, M., 2004. Estimation of articulatory movements from speech acoustics using a HMM-based speech production model. IEEE Trans. SAP 12 (2), 175–185.

Hogden, J., Valdez, P., 2000. Bridging the gap between speech production and speech recognition. In: Proc. of the 5th Seminar on Speech Production, Germany.

Hogden, J., Valdez, P., 2001. A stochastic articulatory-to-acoustic mapping as a basis for speech recognition. In: Proc. IEEE IMTC, Vol. 2, pp. 1105–1110.

Jensen, F., 1998. An Introduction to Bayesian Networks. UCL Press.

Kirchhoff, K., 1998. Robust speech recognition using articulatory information, Tech. Rep. TR-98-037, International Computer Science Institute.

Kirchhoff, K., Fink, G., Sagerer, G., 2000. Conversational speech recognition using acoustic and articulatory input. In: Proc. ICASSP, Vol. III, p. 1435.

Liu, S., 1996. Landmark detection for distinctive feature-based speech recognition. J. Acoust. Soc. Amer., 3417–3430.

Markov, K., Nakamura, S., 2003a. A hybrid HMM/BN acoustic model for automatic speech recognition. IEICE Trans. Inf. Systems E86-D (3), 438–445.

Markov, K., Nakamura, S., 2003b. Hybrid HMM/BN LVCSR system integrating multiple acoustic features. In: Proc. ICASSP, Vol. I, pp. 888–891.

Markov, K., Dang, J., Iizuka, Y., Nakamura, S., 2003. Hybrid HMM/BN ASR system integrating spectrum and articulatory features. In: Proc. Eurospeech, pp. 965–968.

Okadome, T., Honda, M., 2001. Generation of articulatory movements by using a kinetic triphone model. J. Acoust. Soc. Amer. 110, 453–463.

Papcun, G., Hochberg, J., Thomas, T., Larouche, F., Zacks, J., Levy, S., 1992. Inferring articulation and recognizing gestures from acoustics with a neural network trained in X-ray microbeam data. J. Acoust. Soc. Amer., 688–700.

Stephenson, T., Mathew, M., Bourlard, H., 2001. Modeling auxiliary information in Bayesian Network based ASR. In: Proc. Eurospeech, pp. 2765–2768.

Young, S. et al., 1999. The HTK Book. Entropic Ltd.

Zacks, J., Thomas, T., 1994. A new neural network for articulatory speech recognition and its application to vowel identification. Computer Speech Lang. 8 (3), 189–209.