

Fast GMM-based voice conversion for Text-To-Speech synthesis systems

Taoufik En-Najjary*, Olivier Rosec* and Thierry Chonavel**

* France Telecom R&D, DIH/IPS, 2 avenue Pierre Marzin, 22307 Lannion Cedex, France
{taoufik.ennajjary, olivier.rosec}@francetelecom.com

** ENST Bretagne, Département SC, BP 832, 29285 Brest Cedex, France
Thierry.Chonavel@enst-bretagne.fr

Abstract

Voice conversion (VC) can be seen as a powerful technology for customizing Text-to-Speech (TTS) systems. This paper deals with the integration of a VC method based on Gaussian Mixture Model (GMM) in a TTS system. In this framework, an algorithm that enables complexity reduction of the VC processing is proposed. The main idea is to restrict the conversion function to the most representative components of the GMM for each frame and, if necessary, to store the component indices and their associated weights in the acoustic dictionary. This method is evaluated by comparison to a classical GMM-based transformation function. Tests show that both methods yield comparable results. Furthermore, additional experiments indicate that this new technique leads to a significant decrease of the computational load involved in the conversion process.

1. Introduction

The purpose of VC is to modify a source speaker's speech so that it is perceived as if a target speaker had uttered it. VC has received considerable attention as it is a quick and automatic way to create additional voices for a TTS system, while avoiding the actual recording and processing of large speech databases for each new voice, which is known to be a very tiresome and expensive task. Indeed, given a limited recording of a new target speaker, a learning procedure aims to estimate a transformation function, which will then be applied to the source speaker signal in order to mimic the speech of the target speaker.

In the literature, a variety of techniques has been proposed for converting spectral features, including vector quantization with mapping codebooks [1], dynamic frequency warping [2], speaker interpolation [3], neural networks [4], and GMM [5][6]. It has been shown that GMMs have as good as or superior performance to other transformation approaches [7].

In the framework of speech synthesis, two approaches can be used to apply the voice conversion process: modify the source database and then create the new voice which will be used subsequently in the TTS system, or integrate the transformation function as a post-processing stage in the synthesizer. The latter approach is preferable as it drastically reduces the amount of data to be stored as only one reference voice needs to be stored (which represents a few hundreds Megabytes for corpus-based synthesis) together with the transformation functions associated with each other voice (a few hundreds Kilobytes per voice). So the integration of VC

in a synthesizer is an important topic which will be very useful for example in the case of embedded systems.

However, the current voice conversion systems are complex. For instance, when integrating a GMM-based VC function in a synthesizer based on an Harmonic+Noise Model (HNM) [5], the conversion task costs between 1.5 and 2 times more than the synthesis task itself. So, there is a real need to decrease the computational burden of the VC process.

In this paper, we present a simplified GMM-based VC procedure, which enables reducing the conversion complexity by a factor between 40 and 120. The paper is organized as follows. In section 2, we present the basic principles of VC using the GMM together with a description of the widely used conditional expectation (CE) estimator before introducing our new algorithm. In section 3 tests are carried out to show the relevance of the proposed method and algorithmic issues are also addressed.

2. Voice conversion algorithms

2.1. Gaussian Mixture Model (GMM)

Formally, a GMM model allows the probability distribution of a random variable z to be modeled as the sum of Q multivariate Gaussian components also referred to as classes. Its probability density function can be written as:

$$p(z) = \sum_{i=1}^Q \alpha_i N(z; \mu_i, \Sigma_i), \quad \sum_{i=1}^Q \alpha_i = 1, \quad \alpha_i \geq 0,$$

$N(z; \mu, \Sigma)$ denotes the Gaussian distribution with mean vector μ and covariance matrix Σ and α_i denotes the prior probability that vector z belongs to the i^{th} class. The model parameters (α, μ, Σ) are estimated using the Expectation Maximization (EM) algorithm [8] which is an iterative method for computing maximum likelihood parameter estimates.

2.2. Classical GMM-based voice conversion

Let $X = [x_1 x_2 \dots x_N]$ and $Y = [y_1 y_2 \dots y_N]$ be the sequences of N spectral vectors characterizing a succession of speech produced respectively by the source and target speakers. In the same approach as in [6], the combination of the source and target vectors $z = [x^T y^T]^T$ is used to

estimate the GMM parameters (α, μ, Σ) . Then, the learning step falls down to the estimation of the joint density $p(x, y)$. The conversion function is chosen to be the conditional expectation (CE) estimator whose expression is:

$$\hat{y}_{CE} = F(x) = E[y|x] = \sum_{i=1}^Q h_i(x) \left[\mu_i^y + \Sigma_i^{yy} (\Sigma_i^{xx})^{-1} (x - \mu_i^x) \right], \quad (2)$$

where

$$h_i(x) = \frac{\alpha_i N(x; \mu_i^x, \Sigma_i^{xx})}{\sum_{j=1}^Q \alpha_j N(x; \mu_j^x, \Sigma_j^{xx})}, \quad (3)$$

is the *a posteriori* probability that a given input vector x belongs to the i^{th} class, with

$$\Sigma_i = \begin{bmatrix} \Sigma_i^{xx} & \Sigma_i^{xy} \\ \Sigma_i^{yx} & \Sigma_i^{yy} \end{bmatrix} \text{ and } \mu_i = \begin{bmatrix} \mu_i^x \\ \mu_i^y \end{bmatrix}.$$

μ_i^x and μ_i^y denote mean vectors of class i for the source and target speakers respectively. Σ_i^{xx} denotes the covariance matrix of class i for the source speaker, and Σ_i^{yx} denotes the cross-covariance matrix of class i for the source and target speakers.

2.3. Fast GMM-based voice conversion

The GMM-based conversion methods have the useful property of using a continuous probabilistic modeling of the acoustic space. This continuous transformation reduces drastically the unwanted spectral discontinuities observed with most of other voice conversion algorithms. However, for real-time applications, this transformation remains rather expensive. The determination of a converted vector requires the computation of the *a posteriori* probabilities (3) and the evaluation of the transformed vector according to (2). The computational load is summarized in table 1.

Equation	Cost	
	*	+
(2)	$Q(p^2+p)$	$Q(p^2+2p)$
(3)	$Q(p^2+2p)$	$Q(p^2+2p)$
Total	$Q(2p^2+3p)$	$Q(2p^2+4p)$

Table 1: computational load of the CE conversion method: (*) number of multiplications, (+) number of additions, where p indicates the dimension of the data vectors to be transformed and Q the mixture number.

The conversion function given in (2) can be seen as a "soft" transformation function composed of a sum of linear

functions whose weights are the *a posteriori* probabilities of each class. However, if the GMM clustering has been done correctly, then the weights of each mixture component will be very different. More precisely, only a few components will give rise to significant *a posteriori* probabilities as depicted on figure 1. Thus, it is not useful to calculate the sum upon all the classes in (2). Instead, we propose to retain only the N components having the largest *a posteriori* probabilities, N being fixed for all the frames. Then the transformation function is simply written as:

$$F(x) = \sum_{i \in A} w_i(x) \left[\mu_i^y + \Sigma_i^{yy} (\Sigma_i^{xx})^{-1} (x - \mu_i^x) \right], \quad (4)$$

$$\text{with } w_i(x) = \frac{h_i(x)}{\sum_{i \in A} h_i(x)} \quad (5)$$

and where A is the set of indices of the N selected components. When $N=1$, this method falls down to use the maximum *a posteriori* probability (MAP) estimator.

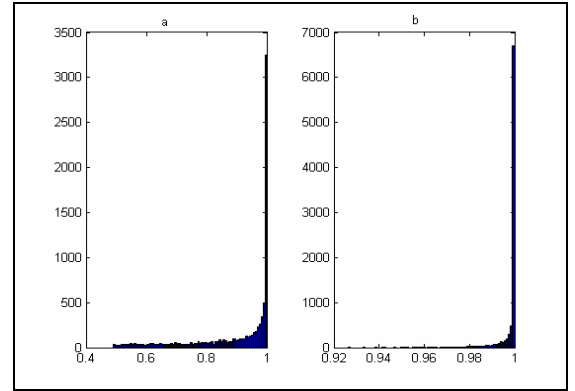


Figure 1: (a) histogram of the largest *a posteriori* probability, (b) histogram of the sum of the three largest *a posteriori* probability

Moreover, when VC is applied to recorded data, which is the case for example in the framework of TTS synthesis, the method described above can be implemented off-line. Indeed, in this case, for reasons of algorithmic complexity, it is judicious to carry out off-line classification and component selection as it avoids evaluating (3) for each GMM component during the conversion step. This presents the disadvantage of storing the indices and the weights of the selected GMM components for each analysis frame in the acoustic dictionary, but in the case of an HNM speech synthesizer, this information requires far less memory than the HNM parameters themselves. Furthermore, using such a procedure, the conversion operation is limited to $N(p^2+p)$ multiplications and $N(p^2+2p)$ additions. Without taking into account exponential computations, the achieved complexity factor reduction is thus about $2Q/N$.

3. Experiments

3.1. Datasets

The database used in order to train and test the conversion function consists of 962 short-paired utterances from one male and one female speakers, which corresponds for each speaker to about 15 minutes of speech sampled at 16kHz. The phonetic transcription and segmentation have been checked and manually corrected. The training set was obtained by holding out 300 paired utterances of the total dataset. The 662 remaining utterances are used for tests.

3.2. Analysis and Training

This conversion system was tested using the HNM, which allows high quality prosodic as well as spectral modifications [5]. However, as in [7], a constant maximum voicing frequency was used and set to half the sampling frequency. This enables the whole spectrum to be represented by the regularized discrete cepstrum method presented in [9]. The analysis is performed asynchronously with a constant frame rate of 10 msec. Note that the asynchronous mode is used only for the data needed to train the conversion function. For the subsequent voice conversion, the analysis is carried out pitch-synchronously because it allows higher quality prosodic modifications. The order of the cepstral representation is set to 20 and the first cepstral coefficient ($c(0)$) is not included in the parameters to be transformed.

In order to train the speech conversion function, the first step consists in aligning the source and the target speech feature streams in such a way that they describe the same phonetic content frame by frame. This is achieved by a Dynamic Time Warping (DTW) algorithm. The phonetic boundaries have been considered as anchor points for the DTW procedure. The pairs of aligned speech frames where one speaker's speech was voiced and the other speaker's was unvoiced were rejected from the training sets. The remaining time aligned data comprise 30,000 spectral vectors.

The second step of the training procedure estimates the GMM parameters. Different GMM were tested where the number of components Q was varied as a power of 2 between 8 and up to 64. The estimation of the GMM parameters is achieved by mean of an EM algorithm, initialized by a classical vector quantization (VQ) procedure, and run until either the likelihood increase is below a given threshold, or after 20 iterations. To prevent singularities, a small value was added to the diagonal elements of the covariance matrices after each iteration.

3.3. Conversion

To obtain a converted utterance, spectral features are extracted in a pitch-synchronous way, and then mapped to new features by the conversion function whose parameters were estimated during the training process.

In this paper, we do not consider the problem of matching the prosodic characteristics of both speakers. However, in order to evaluate our method through listening tests, it is necessary to combine spectral transformation with prosodic modification. The prosodic modifications are generally performed in order to match the average fundamental frequency and articulation rhythm of both speakers. However, when the same sentence uttered by the two speakers is available, another possibility is to impose the pitch and time

contours of the target speaker to the converted speech signal. This latter method is preferable and will be used for the listening tests.

4. Evaluation

First, the proposed algorithm was objectively evaluated on the entire test dataset. For that purpose, the chosen objective measure was the Normalized Spectral Distortion (NSD) defined by:

$$NSD = \frac{\sum_{n=1}^N \|P_{dB}(y_n) - P_{dB}(\hat{y}_n)\|_2}{\sum_{n=1}^N \|P_{dB}(y_n) - P_{dB}(x_n)\|_2}, \quad (6)$$

where $P_{dB}(x)$ denotes the spectral envelope resulting from x , expressed in dB (sampled at 512 point), y_n , \hat{y}_n , and x_n denote respectively target, converted and source spectral parameters. As the source and the target speaker had different energy levels, it is judicious to normalize the energy. This was done by setting $c(0) = 0$.

The results depicted on Figure 2 show that the NSD curve obtained with our method using only 3 components ($N3$) is close to the curve corresponding to the CE estimator. When using a MAP estimator, the NSD slightly increases. It is worth noting that with $Q=64$ GMM components, the complexity is reduced by a factor of about 45 with $N=3$ and by a factor of about 130 in the case of the MAP estimator.

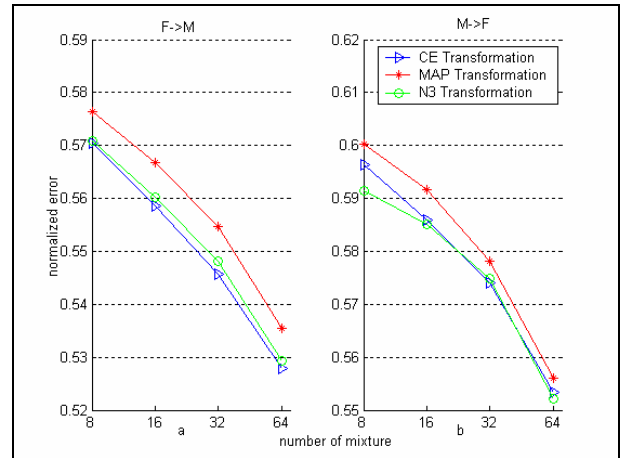


Figure 2: (a) Normalized conversion errors for female-to-male conversion (b) Normalized conversion error for male-to-female conversion

Note that the NSD measure only gives an idea of the global behavior of the transformation function. In order to get further insight of the performance of the MAP method, the Relative Spectral Distortion (RSD) defined by

$$RSD = \frac{\|P_{dB}(y_n) - P_{dB}(\hat{y}_n)\|_2}{\|P_{dB}(y_n) - P_{dB}(x_n)\|_2} \quad (7)$$

was computed for each converted frame. The RSD histograms plotted on figure 3 show that higher errors are more likely to occur when using a MAP estimator. For instance, the probability that the distance between the converted and target spectral envelope is higher than the distance between the source and the target ones (ie the probability that $RSD > 1$) is 3% for the CE, 3.6% for the N3 and 5% for the MAP.

Moreover, informal listening tests reveal that no perceptual difference can be noticed between the speech signals obtained with the CE and the N3 methods. This illustrates the robustness of the N3 method that can be explained by the fact that, the sum of the three largest *a posteriori* probabilities is always larger than 0.9 (see figure 1(b)). However, the MAP conversion method may locally lead to audible artifacts which are perceived as annoying. These are generally observed when the *a posteriori* probability of the selected component is weak (lower than 0.6) which corresponds to cases where a single component cannot appropriately model the spectral envelope of the concerned speech frame.

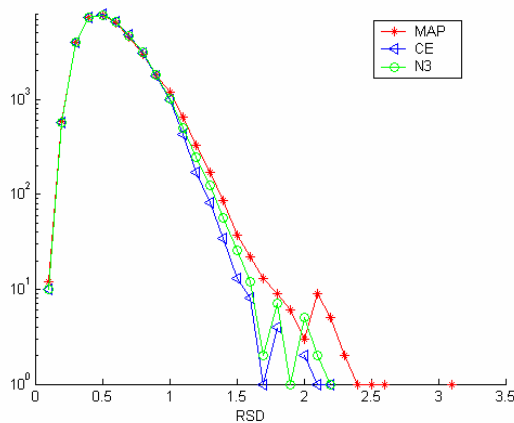


Figure 3: Histogram of the RSD for classical (CE) method as well as for the proposed method with $N=3$ (N3) and $N=1$ (MAP).

In order to optimize the conversion process, one can think of using the MAP method when the *a posteriori* probabilities are sufficiently high (e.g. superior to 0.9) instead of systematically using the N3 method. Thus, a further refinement of the proposed method consists in retaining a minimal set of components whose cumulative *a posteriori* probabilities are higher than a given threshold. For example, with such a threshold fixed at 0.9, this strategy leads to use one, two and three components for respectively 59%, 27% and 14% of the frames of the test dataset. Compared to the CE method, this optimized algorithm offers a complexity reduction factor from 45 up to 130 depending on the number of selected GMM components for a given speech frame. On average, this reduction factor is about 100. Moreover, informal listening tests have shown that this optimal strategy preserved the same quality of the converted speech as with the N3 algorithm. With such a reduction factor the computational load of the conversion process becomes negligible compared to the one of the overall TTS system. Therefore, the

integration of the proposed VC module does not give rise to a noticeable increase of the TTS system computational load.

5. Conclusion

In this paper we have investigated the issue of integrating a voice conversion algorithm in a TTS synthesis system. We proposed a fast GMM-based voice conversion method which consists in restricting the conversion function to the most representative components of the GMM. This method makes it possible to integrate the conversion process in an HNM speech synthesizer without increasing the overall computational load of the system.

This study also reveals that a MAP method is not advisable as it often introduces annoying local artifacts in the converted speech. Then an optimal strategy using a variable number of GMM components was presented. Tests results show that this method performs as good as the classical GMM algorithm while reducing drastically the conversion complexity.

6. References

- [1] M. Abe, S. Nakamura, K. Shikano and H. Kuwabara, "Voice conversion through vector quantization", Proceedings of IEEE ICASSP, pp. 655-658, 1988.
- [2] H. Valbret, E. Moulines and J. P. Tubach, "Voice transformation using PSOLA technique", Speech Communications, vol. 11, pp. 175-187, 1995.
- [3] N. Iwahashi and Y. Sagisaka, "Speech spectrum transformation by speaker interpolation", Proceedings of IEEE ICASSP, vol. 1, pp. 461-465, 1994.
- [4] M. Narendranath, H. A. Murthy, S. Rajendran and B. Yegnanarayana, "Transformation of formants for voice conversion using artificial neural networks", Speech Communication, vol. 16, pp. 207-216, 1995.
- [5] Y. Stylianou, "Harmonic plus Noise Model for speech, combined with statistical methods, for speech and speaker modification", PhD thesis, Ecole Nationale Supérieure des Telecommunications, Paris, France, 1996.
- [6] A. Kain and Mr. Macon, "Text-to-speech voice adaptation from sparse training dated", Proceedings of ICSLP 1998.
- [7] G. Baudoin and Y. Stylianou, "On the transformation of the speech spectrum for voice conversion", Proceedings of ICSLP, 1996.
- [8] A.P. Dempster, N.M. Laird and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", Journal of the Royal Statistical Society Serie B, flight 39, pp. 1-38, 1977.
- [9] O. Cappé, J. Laroche and E. Moulines, "Regularized estimate of cepstrum envelope from discrete frequency points", IEEE ASSP Application of Signal Processing to Audio and Acoustics, pp. 213-216, 1995.