SOUND MORPHING WITH GAUSSIAN MIXTURE MODELS

Federico Boccardi, Carlo Drioli

Centro di Sonologia Computazionale
Dept. of Electronics and Informatics
University of Padova, Italy
feboccardi@tin.it
drioli@dei.unipd.it

ABSTRACT

In this work a sound transformation model based on Gaussian Mixture Models is introduced and evaluated for audio morphing. To this aim, the GMM is used to build the acoustic model of the source sound, and a set of conversion functions, which rely on the acoustic model, is used to transform the source sound. The method is experimented on a set of monophonic sounds and results show that it provides promising features.

1. INTRODUCTION

Gaussian Mixture Models have been widely used in the field of speech processing, mostly for speech recognition, speaker identification, and voice conversion [1, 2]. Their capability to model arbitrary densities and to represent general spectral features motivates the use of GMMs as part of the acoustical front-end for further processing tasks, such as the ones mentioned.

In this work a sound transformation model based on Gaussian Mixture Models is introduced and evaluated for audio morphing, defined here as modifying the time-varying spectrum of a source sound to match the time- varying spectrum of a given number of target sounds. To this aim, the GMM is used to build the acoustic model of the source sound, and a set of conversion functions, which rely on the acoustic model, is used to transform the source sound.

The paper is organized as follows. In Section 2 we recall the properties of GMMs and introduce the spectral conversion framework. In Section 3 the design of the acoustic model and conversion functions for sound morphing purposes is addressed. In Section 4 the method is experimented on a set of monophonic sounds and the results are discussed.

2. DESCRIPTION OF THE SPECTRAL CONVERSION MODEL

The GMM approach assumes that the density of an observed process can be modelled as a weighted sum of component densities and given by the equation

$$f(\vec{x}|\Lambda) = \sum_{i=1}^{M} \alpha_i N(\vec{x}; \vec{\mu_i}, \mathbf{\Sigma}_i) \tag{1}$$

where $\vec{x}$ is a P-dimensional input vector, $N(\vec{x}; \vec{\mu_i}, \mathbf{\Sigma}_i)$ are the component densities, and α_i are the mixture weights. Each component density is a P-variate gaussian function of the form

$$N(\vec{x}; \vec{\mu_i}, \mathbf{\Sigma}_i) = \frac{1}{(2\pi)^{p/2}}(\mid \mathbf{\Sigma}_i \mid)^{-1/2} e^{-\frac{1}{2}(\vec{x}-\vec{\mu_i})^T \mathbf{\Sigma}_i{}^{-1}(\vec{x}-\vec{\mu_i})} \tag{2}$$

with mean vector $\vec{\mu_i}$ and covariance matrix $\mathbf{\Sigma}_i$. The weights α_i satisfy the constraints $\sum_i^M \alpha_i = 1$ and $\alpha_i \geq 0$.

The gaussian mixture is completely specified by the mean vectors, covariance matrix and mixture weights, and can be represented by

$$\Lambda = \{\alpha_i, \vec{\mu_i}, \mathbf{\Sigma_i}\} \; i = 1 \ldots M \tag{3}$$

An interesting feature of the GMM for sound processing applications is that the component densities of the mixture may represent a partition of the underlying sound process in a set of acoustic classes. The probability that an observed input vector $\vec{x}$ belongs to the class $\lambda_i = (\alpha_i, \vec{\mu_i}, \mathbf{\Sigma_i})$ is given, in terms of density, by the formula

$$p(\lambda_i|\vec{x}) = \frac{f(\vec{x}|\lambda_i)p(\lambda_i)}{f(\vec{x}|\Lambda)} = \alpha_i \frac{N(\vec{x}; \vec{\mu_i}, \mathbf{\Sigma}_i)}{\sum_{j=1}^{M} \alpha_j N(\vec{x}; \vec{\mu_i}, \mathbf{\Sigma}_i)} \tag{4}$$

where $f(\vec{x}|\lambda_i) = N(\vec{x}; \vec{\mu_i}, \mathbf{\Sigma}_i)$, $p(\lambda_i) = \alpha_i$, and $f(\vec{x}|\Lambda)$ is given by Eq. (1).

When used to model speech, the components of the GMM represent different phonetic events. When used to model spectra of a sound from a musical instrument, say a single sustained note, we may say that the components of the GMM represent different portions of the sound (e.g., frames from the attack, the sustain, or the release portion). However, depending on the data the model is trained with, it may represent the notes from the same instrument played with different intensities, or notes from different instruments, and so on. In other words, a conversion function which relies on this model is in principle able to classify the input sound frame to be transformed and to perform the transformation required for that frame.

Let us suppose that a sequence of P-dimensional column vectors $\{\vec{x}_t\}$, $t = 1, \ldots, T$, which represents the time- varying spectral envelope of a source signal, has been fitted by a GMM. Moreover, let assume that a sequence of P'-dimensional column vectors $\{\vec{y}_t\}$, $t = 1, \ldots, T$, having the same length of the source signal, is the target of the conversion. We define a spectral conversion function as a map $\mathcal{F} : \mathbb{R}^P \rightarrow \mathbb{R}^{P'}$ able to transform each vector in the input sequence into the vector which occupies the same position in the output sequence, thus preserving the time information of the input and output data. Although it is not necessary for the input and the output vectors to have the same dimension, we will assume $P' = P$ in the rest of the paper.

We consider the following parametric form for the spectral conversion function [2]:

$$\mathcal{F}(\vec{x}_t) = \sum_{i=1}^{M} p(\lambda_i|\vec{x}_t)[\vec{\theta}_i + \mathbf{\Gamma}_i \Sigma_i^{-1}(\vec{x}_t - \vec{\mu}_i)]. \qquad (5)$$

This conversion equation is equivalent to the solution of the following set of equations:

$$\vec{y}_t = \sum_{i=1}^{M} p(\lambda_i|\vec{x}_t)[\vec{\theta}_i + \mathbf{\Gamma}_i \Sigma_i^{-1}(\vec{x}_t - \vec{\mu}_i)] \qquad (6)$$

for all $t = 1 \ldots T$. Eq. (6) can be gathered into a single matrix equation by:

$$\mathbf{Y} = \mathbf{P} \cdot \mathbf{\Theta} + \mathbf{\Delta} \cdot \mathbf{\Gamma}, \qquad (7)$$

where

$$\mathbf{Y} = [\vec{y}_1 \cdots \vec{y}_T]^T, \qquad (8)$$

$$\mathbf{P} = \begin{bmatrix} p(\lambda_1|\vec{x}_1) & \cdots & p(\lambda_M|\vec{x}_1) \\ \cdots & \cdots & \cdots \\ p(\lambda_1|\vec{x}_T) & \cdots & p(\lambda_M|\vec{x}_T) \end{bmatrix}, \qquad (9)$$

$\mathbf{\Delta}$ is a matrix that depends on the conditional probabilities,

$$\mathbf{\Theta} = [\vec{\theta}_1 ... \vec{\theta}_M]^T,$$

and

$$\mathbf{\Gamma} = [\mathbf{\Gamma}_1 ... \mathbf{\Gamma}_M]^T$$

are the unknown parameters of the conversion function.

In this work we omit the term $\mathbf{\Gamma}_i \Sigma_i^{-1}(\vec{x}_t - \vec{\mu}_i)$ in (5) and we use the following reduced form of the conversion function [2]

$$\mathcal{F}(\vec{x}_t) = \sum_{i=1}^{M} p(\lambda_i|\vec{x}_t)[\theta_i]. \qquad (10)$$

3. SOUND MORPHING WITH THE GMM

The investigation relies on the well known sinusoidal plus noise model (SMS) of the signal [3]. The analysis algorithm acts on windowed portions (here called *frames*) of the signal, and produces a time-varying representation as sum of sinusoids (here called *partials*). Assuming that the number of partials P is constant for all frames, for the i-th frame the result of the sinusoidal modelling is a set of triples $(f_h(i), a_h(i), \phi_h(i))$ ($h = 1, \ldots, P$) of frequency, magnitude and phase parameters describing each partial, and a residual noise component that will not be considered in this work. We focus here on the transformation of partials magnitude only. We thus omit to model the differences of frequency and phase among the partials of the source and target sounds. For this assumption to be considered reasonable, we also restrict the choice of the source and the target sounds to a set of compatible signals (e.g., morphing among piano notes with different spectral characteristics, morphing among sustained notes of wind or string instruments, etc.).

3.1. Computation of the acoustic model

An observed sound $\mathbf{X}$ is represented by the matrix:

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1N} \\ \cdots & \cdots & \cdots \\ x_{P1} & \cdots & x_{PN} \end{bmatrix}^T = [\vec{x}_1 \quad \cdots \quad \vec{x}_N]^T \qquad (11)$$

where P is the number of partials, N is the number of frames, and $x_{ij} = a_i(j)$ are the magnitudes of the partials. The sound $\mathbf{X}$ is referred to its model Λ by the density $p(\mathbf{X}|\Lambda)$. The Gaussian Mixture density is parameterized by the mean vectors, covariance matrices and mixture weights from all component densities. We adopt the method of maximum likelihood (ML) estimation to compute the parameters of a GMM. For a sequence of T training vectors $\mathcal{S} = \{\vec{s}_1, ...\vec{s}_T\}$ (e.g., a set of columns selected from an observation sound matrix $\mathbf{X}$, see (11)), the ML estimate computed using the expectation-maximization (EM) algorithm maximize the likelihood of the GMM, defined by:

$$p(S|\Lambda) = \prod_{t=1}^{T} p(\vec{s}_t|\Lambda). \qquad (12)$$

The original formulation of the GMM is founded on the assumption that the observation vectors are independent of one another. This simplifying assumption makes the GMM model suited to cases where the sequential aspect of the observations (the time index) is believed to be irrelevant. In speech recognition or conversion tasks the performances of the GMM are fairly satisfactory even with such assumption. However, if we are interested in reaching a high level of accuracy in the target sound reproduction, as is often the case for audio and musical applications, then the sequential aspect may turn out to be a critical factor. Since this was the case in our experience, we managed to include some knowledge on the dynamics of the process in the GMM so to further improve the acoustic model of the source. This can be achieved by augmenting the dimension of the input with one or more delayed versions of the source process. Fig. 3.1 shows this realization for a doubling in the dimension of the input process. If we include the information on the past and we focus on the case were the dimension of each input component is doubled, Eq. (4) becomes:

Figure 1: The canonical GMM (upper figure) and the augmented version if the input is duplicated and delayed (lower figure).

$$p(\tilde{\lambda_i}|\vec{x}_t,\vec{x}_{t-\tau}) = \tilde{\alpha}_i \frac{N(\vec{x}_{t,t-\tau};\tilde{\vec{\mu}}_i,\tilde{\mathbf{\Sigma}}_i)}{\sum_{j=1}^{M}\tilde{\alpha}_j N(\vec{x}_{t,t-\tau};\tilde{\vec{\mu}}_i,\tilde{\mathbf{\Sigma}}_i)} \qquad (13)$$

with $\vec{x}_{t,t-\tau} \triangleq [x_{1,t},x_{1,t-\tau},\ldots,x_{P,t},x_{P,t-\tau}]^T$, and where the component densities $N(\vec{x}_{t,t-\tau};\tilde{\vec{\mu}}_i,\tilde{\mathbf{\Sigma}}_i)$ are now $2P$- variate gaussian functions. The extension to the case where more than one delayed version of each input component is considered is straightforward.

3.2. Conversion functions

Let $\mathcal{X} = \{\mathbf{X}_1,...,\mathbf{X}_K\}$ be the set of K given sounds, each one organized in an observation sound matrix. The number of selected training frames for each sound is assumed to be T. Our idea is to represent only one of the K sounds, say $\mathbf{X}_1$, by his L-dimensional model Λ, defined as a model in which each input is replicated and delayed L times and the component densities are PL-variate gaussian functions. The whole set of data sounds is then achieved by a set of conversion functions $\mathcal{F}_j$, $j = 1,\ldots,K$, of the form introduced in Eq. (5) in which the diagonal matrix elements Γ_i have been omitted:

$$\tilde{\mathbf{X}}_j = \mathcal{F}_j(\mathbf{X}_1) = \mathbf{P}(\Lambda|\mathbf{X}_1)\mathbf{\Theta}_j, j = 1,...,K \qquad (14)$$

where $\mathbf{X}_1$ is the sound represented by the model, i.e. the source sound, $\mathbf{P}(\Lambda|\mathbf{X}_1)$ is a $P \times M$ matrix given by Eq. (4), and $\mathbf{\Theta}_j$ is an $M \times P$ matrix of coefficients computed by:

$$\mathbf{\Theta}_j = (\mathbf{P}(\Lambda|\mathbf{X}_1)'\mathbf{P}(\Lambda|\mathbf{X}_1))^{-1}\mathbf{P}(\Lambda|\mathbf{X}_1)'\mathbf{X}_j, j = 1,...,K. \qquad (15)$$

In particular the sound $\mathbf{X}_1$ is obtained by the equation:

$$\tilde{\mathbf{X}}_1 = \mathcal{F}_1(\mathbf{X}_1) = \sum_{i=1}^{m} P(\lambda_i|\mathbf{X}_1)\mathbf{\Theta}_1. \qquad (16)$$

To perform a morphing whit percentage $\beta_1,...,\beta_K$, we use the equation:

$$\begin{aligned}\mathbf{X}_{morph} &= \mathcal{F}(\mathbf{X}_1) = \sum_{i=1}^{M}(\beta_1 P(\Lambda_i|\mathbf{X}_1)\mathbf{\Theta}_1 + ... \\ &\quad +\beta_K P(\Lambda_i|\mathbf{X}_1)\mathbf{\Theta}_K) \qquad (17)\end{aligned}$$

Figure 2: Diagram of the audio morphing based on the transformation of $\mathbf{X}_1$ by means of the conversion functions

4. RESULTS AND DISCUSSION

Since the investigation is restricted to the conversion of partials magnitude, the method was used to convert sounds with compatible time evolution of frequency tracks. A set of two piano notes, two samples of Fazioli and of Bosendorfer, was considered as the training data, all having the same fundamental frequency and different spectral characteristics. The proposed method was applied to the data set to represent all the sounds by transformation of one note selected to be the source sound. This assumption of limiting the conversion to partials magnitude is not restricting since the method can be extended in order to perform the conversion of frequency tracks as well.

The SMS analysis was performed on ranged sounds, with 60 partials and a reverse analysis direction, for better behavior with attack part. The SMS data set was corrected to run the killed partials [3] and to range the partials of two sounds. The sound of Bosendorfer $\mathbf{X}_b$, see (11), was represented by the 3-dimensional model $\mathbf{\Lambda}_b$, with delays of 100 and 200 samples.

The sound of Fazioli is obtained by the equation:

$$\tilde{\mathbf{X}}_f = \mathbf{P}(\mathbf{\Lambda}_b|\mathbf{X}_b)\mathbf{\Theta}_f,$$

where $\mathbf{\Theta}_f$ is obtained by (15)

Fig. 3 shows the conversion of the source data into the target data (only the 23th partial is shown). The upper figure shows the result for $M = 128$ and 7 iterations of EM the algorithm. The lower figure shows the same conversion performed by augmenting the dimension of the GMM from P to $3P$, with delays of 100 and 200 samples.

To perform a morphing with percentage

$$\beta_b(t), \beta_f(t),$$

where the weight are functions of time, we use (see (17))

$$\mathbf{X} = \sum_{i=1}^{M}(\beta_b(t)\mathbf{P}(\mathbf{\Lambda}_b|\mathbf{X}_b)\mathbf{\Theta}_b + \beta_f(t)\mathbf{P}(\mathbf{\Lambda}_b|\mathbf{X}_b)\mathbf{\Theta}_f).$$

Fig. 4 shows the conversion of the source data into the target data (only the 23th partial is shown), by gradually rising the weight of the conversion. The upper figure shows the result for $M = 128$ and 7 iterations of the EM algorithm. The lower figure shows the same conversion performed by augmenting the dimension of the GMM from P to $3P$, with delays of 100 and 200 samples.

Figure 3: Comparison of the conversion of the 23th partial of the source sound into the 23th partial of the target sound with a 1-dimensional (upper plot) and a 3-dimensional (lower plot) model (time evolution of the amplitudes is shown)

Figure 4: Comparison of morphing which transform the source sound into the target sound with a 1-dimensional (upper plot) and a 3-dimensional (lower plot) model. The morphing is performed by gradually rising the weight of the conversion (time evolution of the amplitude of the 23th partial is shown)

5. CONCLUSIONS

We presented a sound morphing framework based on GMM. The results show that the method is effective in performing spectral transformations while preserving the time evolution of the source sound. The information on the dynamics of the process, obtained by augmenting the model's dimension, improve the quality of conversion because of the improved modelling of time evolution.
With this method we were able to change a source sound into a target sound by the conversion matrices $\mathbf{P}(\mathbf{\Lambda}|\mathbf{X})$ and $\mathbf{\Theta}$, making sound morphing considerably intuitive.

6. ACKNOWLEDGEMENTS

This work was in part supported by the European Commission under the research contract n. IST-1999-20410.

7. REFERENCES

[1] Reynolds, D., Rose, R. ”Robust Text-independent Speaker Identification Using Gaussian Mixture Speaker Models” *IEEE Transanctions on Speech and Audio Processing*, Vol.3 no. 1, 1995, pp. 72-83.

[2] Stylianou, Y., Cappé, O., Moulines, E. ”Continuos Probabilistic Transform for Voice Conversion.” *IEEE Transanctions on Speech and Audio Processing*, Vol.6 no. 2, 1998, pp. 131-142.

[3] Serra, X. ”Musical Sound Modeling with Sinusoids plus Noise”, in *Musical Signal Processing*, pp.497-510, 1997, Swets and Zeitlinger.

[4] Drioli, C. ”Radial Basis function for conversion of sound spectra.” *EURASIP J. on Applied Signal Processing*, vol.2001, n. 1, 2001, pp.36-44.

[5] Deller, J.R., Proakis, J.R. and Hansen, J.H.L. ”Discrete-Time Processing of Speech Signals”, Prentice Hall, 1987.

[6] Iwahashi, N., Sagisaka, Y. ”Speech Spectrum Conversion Based on Speaker Interpolation and Multi-Functional Representation with Weighting by Radial Basis Function Networks”, *Speech Communication*, Vol. 16, 1995, pp. 139-151.