A Study on Bag of Gaussian Model with Application to Voice Conversion

Yu Qiao[1,2], Tong Tong[1,3], and Nobuaki Minematsu[4]

[1]Shenzhen Institutes of Advanced Technology, Chinese Academy of Science, Shenzhen, China
[2]The Chinese University of Hong Kong, Hong Kong, China
[3]University of Science and Technology of China, Hefei, China
[4]The University of Tokyo, Hongo, Bunkyo-ku, Tokyo, Japan

yu.qiao@siat.ac.cn, ttravel@mail.ustc.edu.cn, mine@gavo.t.u-tokyo.ac.jp

Abstract

The GMM based mapping techniques proved to be an efficient method to find nonlinear regression function between two spaces, and found success in voice conversion. In these methods, a linear transformation is estimated for each Guassian component, and the final conversion function is a weighted summation of all linear transformations. These linear transformations fit well for the samples near to the center of at least one Guassian component, but may not deal well with the samples far from the centers of all Gaussian distributions. To overcome this problem, this paper proposes Bag of Gaussian Model (BGM). BGM model consists of two types of Gaussian distributions, namely basic and complex distributions. Compared with classical GMM, BGM is adaptive for samples. That is for a sample, BGM can select a set of Guassian distributions which fit the sample best. We develop a data-driven method to construct BGM model and show how to estimate regression function with BGM. We carry out experiment on voice conversion tasks. The experimental results exhibit the usefulness of BGM based methods.

Index Terms: GMM, bag of Gaussian model, voice conversion, linear regression.

1. Introduction

Gaussian Mixture Model is a popular probabilistic model for representing the presence of sub-populations within an overall population. Mathematically, GMM is a weighted summation of Gaussian functions whose means and covariances are different. The parameters of GMM can be estimated by Expectation Maximization (EM) algorithm. GMM can be seen as a soft clustering method, where each Gaussian component corresponds to a cluster and the posterior probability denotes the degree of a sample belonging to a Gaussian component.

Among its many success, GMM has been used to design a mapping function between two spaces. This technique has been widely used for woice conversion (VC) [1, 2, 3, 4]. VC aims at transforming a speaker's voice to make it sound like another speaker's without changing the linguistic contents. These researches made use of GMM to model the densities of source cepstral vectors [2] or joint cepstral vectors [1]. The mapping function is a weighted summation of linear transformations for each Gaussian component while the weights are calculated as posterior probabilities of source vectors. The parameters of the linear transformations are estimated by minimizing the conversion errors. The efficiency of GMM-based mapping and its advantage to other spectral conversion methods such as mapping codebooks[1] and artificial neural network [5], have been demonstrated in many previous studies [2, 1, 3, 6]. In a previous work, we proposed a method called mixture of probabilistic linear regressions (MPLR) which unifies the GMM based mapping techniques.

One key advantage of GMM based conversion technique comes from its mixture nature. In the training phase, a local linear regression is estimated for each cluster (Gaussian). In the conversion phase, the converted vector is calculated as a weighted summation of the transformed vectors from each linear regressions. And the weights are determined as the posterior probabilities of input vector being generated by different Gaussian components. This process allows it to deal with the nonlinear regression between two spaces. We found in experiments that the posterior probabilities are always sparse, thus only one linear regression has a large weight in conversion usually.

However, the GMM based mapping techniques has a limitation. The linear regressions are estimated locally for each Gaussian component, respectively. It may work well for the samples near to the mean (center) of Gaussian, but may not fit the samples far from the centers of all Gaussian components. For each sample to be converted, we hope there exists a Gaussian component in GMM whose center is near to the sample. This fact cannot be satisfied by normal GMM. Once a GMM is trained, the number of Gaussian components and their parameters are fixed. There always exist samples which cannot be covered well by GMM. To overcome this problem, this paper introduced the Bag of Gaussian Model (BGM). BGM is built from GMM, and consists of Guassian distributions. Unlike GMM, BGM includes two types of Guassian distributions. The first is called basic distributions which are the same as those in GMM. The second is called complex distributions which are summarization of a subset of basic distributions. We introduce a data driven method to estimate the weight and parameters of new Guassian distributions, and show how to construct the mapping function for BGM. To examine the performance of the proposed method, we carried out voice conversion experiments with a ATR-503 corpus. The results indicate the usefulness of BGM-based methods.

2. GMM based statistical mapping techniques

This section will give a review of the GMM based statistical mapping techniques, which has been widely used in voice conversion. The GMM based mapping technique was originally proposed by Stylianou et al [2]. In [3], Kain et al introduced GMM of joint vectors for this problem, where the transformation parameters can be directly calculated from GMM parame-

ters. After that, many methods have been developed to improve the performance of GMM based mapping techniques. Toda [4, 6] introduced dynamic features to improve the naturalness of speech, and made use of global variance to deal with the over smooth problem. In a previous work [7], we developed a framework called 'mixture of probabilistic linear regressions' to unify the GMM based mapping techniques.

The key problem in voice conversion is to determine a mapping function $y = f(x)$ from source speaker's acoustic vector x to target speaker's acoustic vector y. Linear regression is popular for such problems due to its simplicity and efficiency. However, many real problems include nonlinear transformations which cannot be approximated well by a linear one. The intrinsic idea behind GMM based mapping techniques is to divide the feature space into several components (each corresponds to a Gaussian) in a probabilistic and soft way, and estimate a local linear transformation for each Gaussian component.

Consider the feature space can be divided into K 'virtual spaces' $\{\mathbb{S}_k\}_{k=1}^K$. Each $\mathbb{S}_k$ has the same region as the source feature space, but has different density model of x, denoted by $p(x|\mathbb{S}_k)$ or $p(x|k)$ for short. These densities $\{p(x|k)\}$ yield information for 'soft' division. Then we estimate a linear regressions $y = B_k x + b$ for $\mathbb{S}_k$. For simplicity, let $\hat{x} = [x^\top 1]^\top$. We can rewrite $B_k x + b = A_k \hat{x}$. The final regression function is a weighted combination of all linear regressions,

$$y' = f_{\text{GMM}}(x) = \sum_{k=1}^{K} p(k|x) A_k \hat{x}. \tag{1}$$

Posterior probability $p(k|x)$ can be calculated with the Bayes' theorem,

$$p(k|x) = \frac{w_k p(x|k)}{\sum_j w_j p(x|j)}, \tag{2}$$

where $w_k = p(\mathbb{S}_k)$ denotes a prior probability of the k-th PLR or $\mathbb{S}_k$, and $\sum_k w_k = 1$. It is noted that conversion with Eq. 1 is different from the Mixtures of Linear Regression models (Chapter 14.5 [8]), where the weights are fixed for all training samples.

GMM parameters $p(x|k)$ and $p(k)$ can be obtained by EM algorithm. In the next, we discuss how to estimate the transformation matrix A_k in Eq. 1. Suppose we have a set of training data set $\{x_i, y_i\}_{i=1}^I$, where I is the number of training sample. For convenience, let $p_{i,k} = p(x_i|k)$ and $\gamma_{i,k} = p(k|x_i)$. Define matrix R_k with diagonal as $\text{diag}(R_k) = [\gamma_{1,k}, \gamma_{2,k}, ..., \gamma_{I,k}]$.

The MSSE estimation of Eq. 1 is given by,

$$\begin{aligned} &\arg\min_{\{A_k\}} \sum_i \|y_i - f_{\text{GMM}}(x_i)\|^2 \\ =&\sum_i \|y_i - \sum_k \gamma_{i,k} A_k \hat{x_i}\|^2 \\ =&\sum_i \|\sum_k \gamma_{i,k}(y_i - A_k \hat{x}_i)\|^2, \end{aligned} \tag{3}$$

where $\sum_k \gamma_{i,k} = 1$. This is a linear optimization problem which can be solved directly. Let $\hat{X}_k = [\gamma_{1,k}\hat{x}_1, \gamma_{2,k}\hat{x}_2, ..., \gamma_{I,k}\hat{x}_I]$ and $\hat{\mathbb{X}} = [\hat{X}_1^\top, \hat{X}_2^\top, ..., \hat{X}_K^\top]^\top$. The optimal transform matrices $\{A_k^*\}$ for Eq. 3 are given by

$$[A_1^*, A_2^*, ..., A_K^*] = Y\hat{\mathbb{X}}^\top(\hat{\mathbb{X}}\hat{\mathbb{X}}^\top)^{-1}. \tag{4}$$

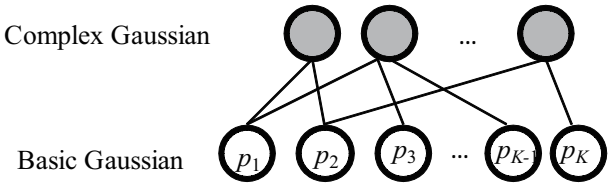

Figure 1: Bag of Gaussian Model.

However, this is computationally expensive, since matrix $\hat{\mathbb{X}}$ has a size $K(d+1) \times I$, where d is the dimension of x. Here we consider a fast and approximate calculation by decomposing Eq. 3. Remind $\sum_k \gamma_{i,k} = 1$ and $\gamma_{i,k} > 0$. According to Jensen's inequality, we have $|\sum_k \gamma_{i,k}(y_i - A_k\hat{x}_i)|^2 \leq \sum_i \gamma_{i,k}|y_i - A_k\hat{x}_i|^2$. Therefore Eq. 3 can be approximated by the following upper bound,

$$\arg\min_{\{A_k\}} \sum_k \sum_i \gamma_{i,k}\|y_i - A_k\hat{x}_i\|^2. \tag{5}$$

This can be further decomposed into K linear optimization problems, $\arg\min_{A_k} \sum_i \gamma_{i,k}\|y_i - A_k\hat{x}_i\|^2$.

3. Bag of Gaussian model

In GMM based conversion techniques, a linear transformation $A_k\hat{x}$ is estimated for each Gaussian component. In the training phase, the samples that are near to the mean of k-th Gaussian component have higher posterior probabilities $p(k|x_i)$ and thus are more important in Eq. 3 and Eq. 5 than those with lower posterior probabilities. For this reason, in the testing phase, samples near to the means of Gaussian components will have better conversion performance, while samples far from the Gaussian means will not.

Perhaps one idea to this problem is to increase the mixture number in GMM thus more samples will be covered precisely by Gaussian components. But large mixture number leads to smaller variance (or covariance), and makes training and conversion unstable. Experiments have shown that very large mixture number can harm conversion performance too. Ideally, we hope a large number of Gaussian distributions which cover the feature space densely and whose variance (or covariance) are relatively large. These two constraints cannot be achieved together by classical GMMs.

This paper introduces the Bag of Gaussian Model (BGM) to deal with the problem discussed above. Like GMM, BGM is composed by a number of Gaussian distributions and their weights. Unlike GMM, there are two types of Gaussian distributions in BGM, basic distributions and complex distributions. Basic distributions are the same as those in GMM, while complex distributions are summarization of a subset of basic distributions (Fig. 1). The number of complex distributions is flexible. One can design different complex distributions for different applications. Since complex distributions summarize the basic ones, they are generally stable. In the reminder of this section, we will discuss how to build BGM at first. Then we will show how use BGM for conversion.

3.1. Construction of Bag of Gaussian Model

Bag of Gaussian Model consists of basic and complex Gaussian distributions, together their weights. The diagram of BGM is shown in Fig. 1. Let $P = \{p_1, p_2, ..., p_K\}$ denote the set of basic Gaussian distributions, $Q = \{p_{K+1}, p_{K+2}, ..., p_M\}$

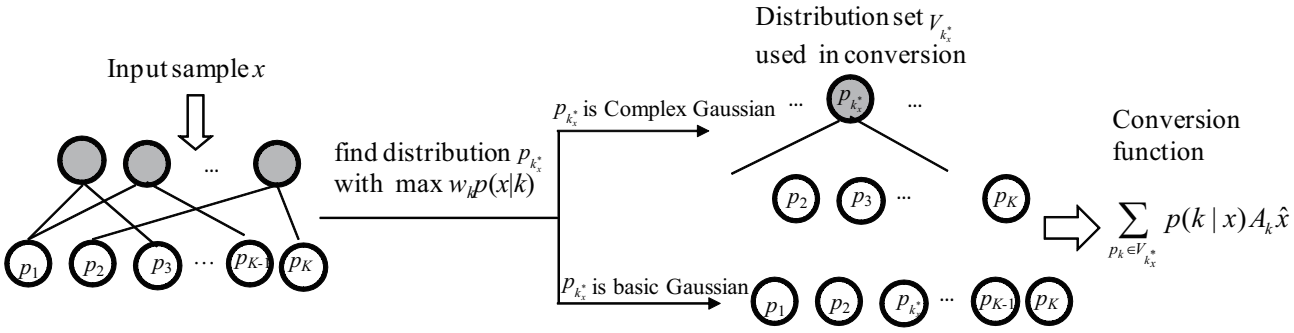

Figure 2: Soft conversion with Bag of Gaussian Model.

the set of complex Gaussian distributions, and $W = \{w_1, w_2, ..., w_M\}$ the set of weights. Suppose we have a set of training data $X = [x_1, x_2, ..., x_I]$. Distributions in P and their associated weights are obtained by EM algorithm for GMM. In the next, we show how to learn the distributions Q and their associated weights.

Each complex distribution p_m $(K+1 \leq m \leq M)$ in Q is a summarization of several basic distributions. Let $S_m = \{p_1^m, p_2^m, ..., p_{K'}^m\}$ be the set of distributions which should be merged to p_m. Here we have two basic questions: 1) how to determine merge sets $\{S_m\}$, and 2) how to calculate the parameters of new merged distribution p_m and its weight w_m.

The answer to the first question is flexible, and depends on the applications to which BGM is applied. In this study, we use a simple approach. For each $p_i \in P$, we find its first n nearest neighbors with the minimum KL-divergence,

$$KL(p_i, p_j) = \frac{1}{2}(\log \frac{|\Sigma_j|}{|\Sigma_i|} + \text{tr}(\Sigma_j^{-1}\Sigma_i) + (\mu_j - \mu_i)^\top \Sigma_j^{-1}(\mu_j - \mu_i)), \quad (6)$$

where $p_i(x) = N(x|\mu_i, \Sigma_i)$ and $p_j(x) = N(x|\mu_j, \Sigma_j)$. Then we take p_i and each of its neighbors p_j to construct one merge set $S_m = \{p_i, p_j\}$. Note a basic distributions can be merged into more than one complex distributions.

We solved the second question in a data-driven manner. At first for each training data x_i and basic distribution $p_k(x)$, we calculate the posterior probability as $\gamma_{k,i} = p(k|x_i)$.

Given S_m, weight w_m for complex distribution p_m can be calculated by

$$w_m = \sum_{p_k \in S_m} w_k. \quad (7)$$

Due to additive property of probabilities, we estimate the posterior probability $\gamma_{m,i} = p(m|x_i)$ for complex distribution as,

$$\gamma_{m,i} = \sum_{p_k \in S_m} \gamma_{k,i} = \sum_{p_k \in S_m} p(k|x_i). \quad (8)$$

With $\{\gamma_{k,i}\}$, we calculate mean μ_m and covariance Σ_m for p_m through maximum likelihood. The likelihood is defined as,

$$P(X|m, \{\gamma_{m,i}\}) = \prod_i p_m(x_i)^{\gamma_{m,i}}. \quad (9)$$

Maximizing the above equation leads to,

$$\mu_m = \frac{1}{\sum_i \gamma_{m,i}} \sum_i \gamma_{m,i} x_i, \quad (10)$$

$$\Sigma_m = \frac{1}{\sum_i \gamma_{m,i}} \sum_i \gamma_{m,i}(x_i - \mu_m)(x_i - \mu_m)^T. \quad (11)$$

3.2. Conversion with BGM

In this section, we will discuss how to use Bag of Gaussian model to estimate a mapping relation between x and y. Let $X = [x_1, x_2, ..., x_I]$ and $Y = [y_1, y_2, ..., y_I]$ denote the training set. For the beginning, we estimate a linear transformation $y = A_k \hat{x}$ for each distribution $p_k \in P \bigcup Q$. Here, $\hat{x} = [x; 1]$ and A_k is $d \times (d+1)$ matrix. For sample x_i and Guassian distribution p_k, we can estimate the posterior probability $p(k|x_i)$ by Eq. 2 if p_k is a basic distribution, or by Eq 8 if p_k is a complex one. A_k is obtained by minimizing the following error function,

$$\arg\min_{A_k} \sum_i \gamma_{k,i} \|y_i - A_k \hat{x}_i\|^2. \quad (12)$$

The above equation can be solved directly [9]. The optimal A_k is

$$A_k = Y \Gamma_k \hat{X}^\top (\hat{X} \Gamma_k \hat{X}^\top)^{-1}, \quad (13)$$

where Γ_k is a diagonal matrix with diagonal part as $[\gamma_{k,1}, \gamma_{k,2}, ..., \gamma_{k,N}]$, and $\hat{X} = [\hat{x}_1, \hat{x}_2, ..., \hat{x}_n]$.

Now we consider how to transform a new sample x with Bag of Gaussian Model. There are two ways: one is to chose one Gaussian component for transformation (hard method), the other is to use the weighted summation of the transformations of all components (soft method). For a training sample x, we can find the distribution which fits x best,

$$k_x^* = \arg\max_k p(k|x) = \arg\max_k w_k p_k(x). \quad (14)$$

In the hard method, we just transform x with the associated linear transformation of k_x^*-th distribution,

$$f_{hard}(x) = A_{k_x^*} \hat{x}. \quad (15)$$

The above hard mapping is in spirit similar to vector quantization based conversion technique proposed in [1]. One difference is that [1] used k-mean for clustering.

In the soft method, we consider a weighted summation of transformation matrices. Introduce $V_{k_x^*}$ to denote the set of

nodes which need to be considered in transformation. $V_{k_x^*}$ consists of node k_x^*-th distribution and the basic nodes not in $S_{k_x^*}$,

$$V_{k_x^*} = \{p_{k_x^*} \bigcup P/S_{k_x^*}\}, \tag{16}$$

where $P/S_{k_x^*}$ means subtract $S_{k_x^*}$ from P. With $V_{k_x^*}$, the soft transformation is given by,

$$f_{soft}(x) = \sum_{p_k \in V_{k_x^*}} p(k|x) A_k \hat{x}. \tag{17}$$

If p_k is a basic distribution, the transformation is the same as those of GMM based conversion Eq. 1. If p_k is a complex distribution, we used $p_{k_x^*}$ to replace the basic distributions summarized by it in the transformation function. The diagram of conversion with BGM is shown in Fig. 2.

4. Experiments

We carried out experiments to evaluate the performances of the propose method on voice conversion task. We used the ATR-503 phoneme balanced sentences. The data set used contains 503 utterances from a male speaker and another 503 utterances from a female speaker. The sampling frequency is 16k Hz. For each utterance, we calculated its 24-D cepstrum sequence. We converted the female voice to the male voice. For conversion, the training utterances of the source speaker and the target speaker are aligned by dynamic time warping. We used 50 utterances for training the conversion model, and another 40 utterances for testing. The mixture numbers are set as 4, 8, 16, 24, 32, respectively.

We made comparison among the following four methods, 1) GMM based conversion, 2) GMM based hard conversion [1], 3)BGM based soft conversion (Eq. 16), and 4)BGM based hard conversion (Eq. 15). The average cepstrum distortion between target vector y^t amd converted vector y_c, CD[dB] $= \frac{10}{\ln 10}\sqrt{2\sum_d (y_t^d - y_c^d)^2}$ is used for evaluation. The results are shown in Fig. 3. It can be seen that GMM-based soft conversion method outperforms the BGM based one when the mixture number is very few i.e. 4. But generally, BGM based methods achieve lower cepstrum distortions than GMM based methods for other mixture numbers. Among all the methods compared, BGM based soft conversion has the best performance. It is noted that we use a simple approach to merge basic Gaussian distributions with their two nearest neighbors for merge. There may exist better methods to select Gaussian distributions for merge, which can improve the performance.

5. Conclusions

This paper proposes the Bag of Gaussian Model (BGM) to estimate a mapping relation between two spaces. BGM is constructed from GMM. The basic distributions of BGM is the same as those in GMM. But different from GMM, BGM also includes complex distributions which are summarizations of basic distributions. We developed a method to calculate the parameters of complex distributions in a data driven manner. We derive the linear transformation for each complex distribution

[1]GMM based hard conversion make uses of the linear transformation associated with the Gaussian component that has the largest posterior probability. It has the same form as Eq. 16 of BGM based soft conversion.

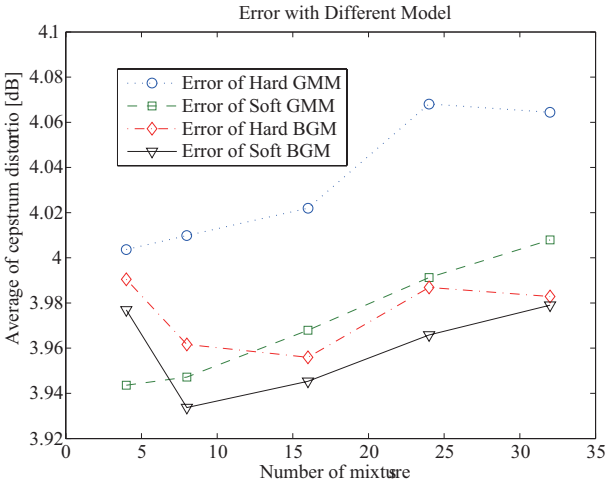

Figure 3: Voice conversion results of GMM and BGM.

in BGM, and show how to use BGM for conversion. Specially, we develop two conversion methods, one is hard conversion which selects one linear transformation associated with the largest probability for conversion, and the other is soft conversion which is a weighted summation of a set of linear transformations. The posterior probabilities of a feature vector being to a Gaussian component play important roles in our method. BGM includes more Gaussian distributions compared with the GMM it is built from. This leads to more linear transformations to fit the area which has not been well fit by the linear transformations obtained from GMM. We executed experiments on the Japanese ATR-503 corpus. The experimental results exhibited that the BGM based conversion methods has better performance on voice conversion. But the improvement is not significant. As future work, we are going to explore how to use BGM on other applications.

6. References

[1] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," *Proc. ICASSP,*, pp. 655–658, 1988.

[2] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. on SAP,*, vol. 6, no. 2, pp. 131–142, 1998.

[3] A. Kain and MW Macon, "Spectral voice conversion for text-to-speech synthesis," *Proc. ICASSP*, 1998.

[4] T. Toda, H. Saruwatari, and K. Shikano, "Voice conversion algorithm based on Gaussian mixture model with dynamic frequency warping of STRAIGHT spectrum," *Proc. ICASSP*, 2001.

[5] M. Narendranath, H.A. Murthy, S. Rajendran, and B. Yegnanarayana, "Transformation of formants for voice conversion using artificial neural networks," *Speech Communication*, vol. 16, no. 2, pp. 207–216, 1995.

[6] T. Toda, A.W. Black, and K. Tokuda, "Voice Conversion Based on Maximum-Likelihood Estimation of Spectral Parameter Trajectory," *IEEE Trans. on ASLP*, vol. 15, no. 8, pp. 2222–2235, 2007.

[7] Y. Qiao and N. Minematsu, "Mixture of probabilistic linear regressions: a unified view of GMM-based mapping techniques," in *ICASSP*, 2009.

[8] C.M. Bishop, *Pattern recognition and machine learning*, vol. 4, Springer New York:, 2006.

[9] Yu Qiao, "Mixture of Probabilistic Linear Regressions," *Technical Report, The university of Tokyo*, 2008.