A Comparison Between GMM and non-GMM models applied in Voice Conversion

Alejandro Uriz[1], Pablo Agüero[1], Juan Carlos Tulli[1], Esteban González[1], and Antonio Bonafonte Cávez[2]

[1] Facultad de Ingeniería, Universidad Nacional de Mar del Plata, Mar del Plata, Argentina
[2] Universitat Polytècnica de Catalunya, Barcelona, Spain
ajuriz@fi.mdp.edu.ar
http://www.fi.mdp.edu.ar

Abstract. Clustering data is a commonly used technique to decompose a space of numerical or categorical data into a number of classes. In particular, clustering categorical data is an integral part of the data mining area and has received special attention until today. In this paper, it is made a comparison between a novel algorithm for clustering categorical data (k-histograms) and a well know algorithm for clustering numerical data (k-means) in the context of voice conversion systems. It is proposed an adaptation of the k-histograms techniques to deal with numerical data through quantization. The resulting statistical tool can model non-gaussian distributions for its use in voice conversion. Objective and subjective results support the proposed idea.

Key words: k-means, k-histograms, Gaussian Mixture Models, voice conversion.

1 Introduction

The main goal of clustering a space D is to divide the data X ($X \in D$) into groups named clusters. The process aims to obtain clusters whose distance between adjacents clusters is maximized while the radio of each cluster is minimized. There are many studies about clustering techniques and they are widely applied in differents fields such as image processing, voice recognition, economics, biology, etc.

The data can be divided in two classes: *numerical data* and *categorical data*. The numerical data has inherent geometric properties, which can be used to define distance functions between data points. Due to their properties it is easy to cluster a numerical data space. An example of a clustering method of numerical data is the *K-means* algorithm [1].

On the other hand, in many process the majority of data is categorical: it can be divided into classes and its attributes are not the same than for numerical data. For example, a categorical attribute is *phoneme*, whose values include */a/* (e.g.: **a**uto) , */b/* (e.g.: **b**ut), */e/* (e.g.:**e**nconding), */Ts/* (e.g.:**ch**at), etc. Due

to the properties of categorical attributes, the clustering of categorical data is more complex than numerical data. Examples of clustering of a categorical data space are *k-histograms* [2] and *k-modes* [3].

In this paper we make a comparison between k-means and k-histograms algorithm applied to a specific area of signal processing: *voice conversion*. The primary goal of voice conversion systems is to modify the voice of a source speaker in order to be perceived as if it had been uttered by another specific speaker: the target speaker. For this purpose, relevant features of the source speaker are identified and replaced by the corresponding features of the target speaker.

Several voice conversion techniques have been proposed since the problem was first formulated in 1988. In this year Abe et al. [4] proposed to convert voices through mapping codebooks created from a parallel training corpus . Since then, many authors tried to avoid spectral discontinuities caused by the hard partition of the acoustic space by means of fuzzy classification [5] or frequency axis warping functions [6].

The appearance of statistical methods based on *k-means* algorithm, such as *gaussian mixture models (GMM)* for spectral envelope transformation was an important breakthrough in voice conversion [7, 8], because the acoustic space of speakers was partitioned into overlapping classes and the weighted contribution of all the classes was considered when transforming acoustic vectors. The spectral envelopes were successfully converted without discontinuities, but in exchange the quality of the converted speech was degraded by over-smoothing. This problem was faced in further works [9–11], while the usage of GMM-based techniques became almost standard, up to the point that the research was focused on increasing the resolution of GMM-based systems through residual prediction [8, 12, 13] in order to improve both the quality scores and the converted-to-target similarity.

Nevertheless, the problem of creating high-quality voice conversion systems that could be used in real-life applications has not been completely solved. At present, there is still a tradeoff between the similarity of converted voices to target voices and the quality achieved by the different conversion methods.

Erro et al. [14] presented a new voice conversion technique named Weighted Frequency Warping (WFW), which combined the conversion capabilities of GMM-based systems and the quality of frequency-warping transformations. The aim of WFW was to obtain a better balance between similarity and quality scores than previous existing methods. At the same time, other authors tried to improve conventional GMM-based systems by applying frequency-warping functions to residuals [15]. Both kinds of systems resulted in significant quality improvements and a slight decrement in the converted-to-target similarity scores, although they were conceptually different.

In this paper it is implemented a voice conversion system based on *k-histograms*. This technique can be employed because the acoustic space of each speaker can be divided into acoustic classes (phonemes). The main goal is to maximize the similarity to target speaker without any discontinuities in the sinthesized voice, reducing the over-smoothing found **in** other techniques. These techniques de-

scribed in the literature, produce target features that may not be uttered by the target speaker. In order to compare the system's performances, it is included GMM in the experiments, a well known state-of-the-art technique.

This paper is organized as follows. In Section 2, the cluster algorithms under study are explained in detail, emphasising the differences. In Section 3, both techniques are implemented. In Section, the results of the objective and subjective tests are presented and discussed. Finally, the main conclusions are summarized in Section 5.

2 Description of the Methods Under Study

In this section k-means and k-histograms are described (both clustering techniques), and their use for voice conversion. The first technique is widely used for numerical data, while the second is a novel approach for categorical data.

2.1 Gaussian Mixture Model (GMM)

This model uses a representation of space through a number of m gaussian distributions, with their corresponding parameters (mean and standard deviation). The space is partitioned using the k-means algorithm. Given a set of numeric objects $X_i \in D$ and an integer number k, the k-means algorithm searches for a partition of D into k clusters that minimizes the within groups sum of squared errors (WGSS). This process can be formulated as the minimization of the function $P(W, Q)$ with respect to W and Q, as shown in equations 1 and 2.

$$\text{Minimize} P(W,Q) = \sum_{l=1}^{m}\sum_{i=i}^{n} w_{i,l} d(X_i, Q_l) \tag{1}$$

$$\text{Subject to} \sum_{l=1}^{k} w_{i,l} = 1, 1 \leq i \leq n \tag{2}$$

$$w_{i,l} \in \{0,1\}, 1 \leq i \leq n, 1 \leq l \leq k$$

where W is an $n \times k$ partition matrix which assigns each vector X_i to one cluster, $Q = \{Q_1, Q_2, ..., Q_k\}$ is a set of objects in the same object domain (usually known as centroids of the clusters), and $d(\cdot, \cdot)$ is the definition of distance between vectors.

Then each vector of the space is modeled using a weighted sum of parameters. GMM is a weighted sum of m component gaussian densities. For the training stage we need at least two spaces: *source* and *target* space. Assuming that a parallel training corpus of data is available, the vectors of the source space (x_t) and those of the target space (y_t) may be aligned in pairs. Then, a joint-density GMM may be estimated from vectors z_t by means of the EM algorithm [1] , where z_t is obtained by concatenating x_t and y_t. The resulting model is given by

the weights p_i, the mean vectors μ_i and the covariance matrices Σ_i of its m gaussian components. Individual models for each space can be extracted from these parameters, since the mean vectors and covariance matrices can be decomposed into

$$\mu_i = \begin{bmatrix} \mu_i^x \\ \mu_i^y \end{bmatrix} \tag{3}$$

$$\mu_i = \begin{bmatrix} \Sigma_i^{xx} & \Sigma_i^{xy} \\ \Sigma_i^{yx} & \Sigma_i^{yy} \end{bmatrix} \tag{4}$$

Once the model is trained, it is possible to calculate the probability that a source vector x belongs to the i^{th} class (each gaussian component represents one of the m overlapping classes):

$$p_i(x) = \frac{\alpha N(x, \mu_i^x, \Sigma_i^{xx})}{\sum_{j=1}^{m} \alpha N(x, \mu_j^x, \Sigma_j^{xx})} \tag{5}$$

where $N(\cdot)$ denotes a gaussian distribution. In conventional GMM-based methods, for each gaussian component is assigned a statistical transformation function, so for a given input vector x to be converted, the m probabilities $p_i(x)$ are used as weights for combining the contribution of all classes:

$$F(x) = \sum_{i=1}^{m} p_i(x)|\mu_i^y + \Sigma_i^{yx}\Sigma_i^{xx\,-1}(x - \mu_i^x)| \tag{6}$$

More information about GMMs can be found in [7, 8], with studies about the dimension of the matrices involved in training. In those papers some simplifications are proposed to reduce the number of parameters and the estimation error, such as diagonal covariance matrices.

2.2 K-Histograms (KH)

K-histograms is an interesting approach to cluster categorical data. It is an expansion of k-means algorithm to cope with categorical data using histograms and a different cost measure.

Notation Let $A_1, ..., A_m$ a set of categorical attributes with domains $D_1, ..., D_m$ respectively. Let the dataset $S = X_1, X_2, ..., X_n$ be a set of objects described by m categorical attributes, $A_1, ...A_m$. The value set V_i of A_i is a set of values of A_i that are represented in S. For each $v \in V_i$, the frequency f_v, denoted as f_v, is the number of objects $O \in X$ with O. Suppose the number of distinct attribute values of A_i is p_i, we define the histogram of A_i as the set of pairs $h_i = (v_1, f_1), (v_2, f_2), ..., (v_{p_i}, f_{p_i})$. The histograms of the data set S is defined as: $H = h_1, h_2, ..., h_m$.

Let X, Y be two categorical objects described by m categorical attributes. The dissimilarity measure between X and Y can be defined by the total mismatches of the corresponding attribute values of the two objects. The smaller the number of mismatches are, the more similar the two objects. Formally,

$$d_1(X,Y) = \sum_{j=1}^{m} \delta(x_j, y_j) \tag{7}$$

where

$$\delta(x_j, y_j) = \begin{cases} 0(x_j = y_j) \\ 1(x_j \neq y_j) \end{cases} \tag{8}$$

Given the dataset $S = X_1, X_2, ...X_n$ and an object Y. The dissimilarity measure between X and Y can be defined by the average of the sum of the distances between X_i and Y.

$$d_2(D,Y) = \frac{\sum_{j=1}^{n} \delta(X_j, Y)}{n} \tag{9}$$

If we take the histograms $H = h_1, h_2, ...h_m$ as the compact representation of the data set S, Eq. 9 can be refined as Eq. 10.

$$d_3(H,Y) = \frac{\sum_{j=1}^{m} \phi(h_j, y_j)}{n} \tag{10}$$

where

$$\phi(h_j, y_j) = \sum_{l=1}^{p_j} f_l * \delta(v_l, y_j) \tag{11}$$

From a viewpoint of implementation efficiency, Eq. 10 can be presented in form of Eq. 12.

$$d_4(H,Y) = \frac{\sum_{j=1}^{m} \psi(h_j, y_j)}{n} \tag{12}$$

where

$$\psi(h_j, y_j) = \sum_{l=1}^{p_j} f_l * (1 - \delta(v_l, y_j)) \tag{13}$$

Algorithm Equation 13 can be efficiently computed because it requires only the frequencies of matched attribute value pairs. The previous equations will be used to explain the clustering algorithm named k-histograms. The main idea of this method is to replace the means in the k-means algorithm for histograms, defining a dissimilarity measure between categorical object and histogram. When Eq. 12 is applied, the cost function 14 used in the k-means algorithm is transformed into:

$$\text{Minimize} P(W,H) = \sum_{l=1}^{m} \sum_{i=i}^{n} w_{i,l} d(X_i, H_l) \tag{14}$$

where $w_{i,l} \in W$ and $H_l = \{h_{l,1}, h_{l,2}, ..., h_{l,m}\}$ are the counts of each category (1 to m) in the histogram l.

While the GMM algorithm is initialized using random values, the k-histograms training stage algorithm is initialized using real values. The initial point of each class is a real vector. In this way it is possible to obtain a faster convergence than GMM algorithm.

In the k-histograms algorithm we need to calculate the total cost P against the whole data set each time when a new H or W is obtained. To make the computation more efficient, the following algorithm, also adopted in k-modes algorithm [3], is used instead:

1. Select k initial histograms, one for each cluster.
2. Allocate an object to the cluster whose histogram is the nearest to it according to Eq. 12. Update the histogram of the cluster after each allocation.
3. After all objects have been allocated to clusters, retest the dissimilarity of objects against the current histograms. If an object is found such that its nearest histogram belongs to another cluster rather than its current one, reallocate the object to that cluster and update the histograms of both clusters.
4. Repeat 3 until no objects have changed clusters after a full cycle test of the whole data set.

In this paper it is proposed the use of k-histograms to partition the vectors of features (LSF parameters) used in voice conversion into sets. The LSF parameters are discretized to estimate the counts in the histograms of each set. The source and target LSF vectors are aligned in the training set, and they are jointly partitioned using k-histograms.

This approach intends to avoid the assumption made in GMM-based voice conversion system about the possibility to approximate the distribution of each LSF coefficient through a mixture of gaussians. In this proposal, no assumption is adopted about a particular distribution of the parameters by estimating it using histograms.

The conversion between source and target parameters using histograms is performed using a non-gaussian to non-gaussian mapping via the cumulative distribution function (CDF) coefficient by coefficient, as shown in Eq. 15.

$$\hat{y_i} = F_{y_j}^{-1}[F_{x_j}(x_i)] \tag{15}$$

As shown in Fig. 1, the LSF parameter x_i of source speaker is mapped into the target LSF parameter $\hat{y_i}$ using the CDF of source and target i^{th} LSF parameter and j^{th} set (F_{x_j} and F_{y_j} respectively). The different available sets are obtained using the partition of the LSF parameter space via the k-histograms clustering technique.

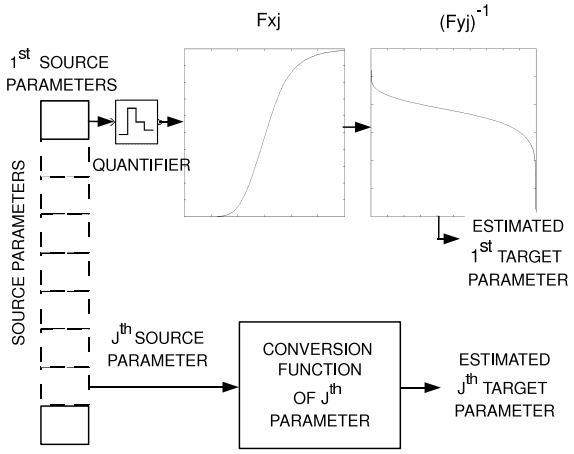

Fig. 1. Example of the conversion of the LSF parameters of source speaker

The decision about the set j used in the transformation of a given source feature vector x is performed calculating the joint probability of each component of the vector for each possible set (Eq. 16).

$$p_j = \sum_{i}^{K} log(f_{x_j}(x_i)) \tag{16}$$

where f_{x_j} is the probability that the coefficient x_i belongs to set j. The vector belongs to the set j with the highest probability p_j.

The parameters estimated using Eq. 15 are used to perform the synthesis of the target speech. In the next section two voice conversion methods will be explained based on the LSF transformation shown in this section.

3 Implementation

As stated in Section 1 the main goal of voice conversion is to modify the voice of a source speaker in order to be perceived as if it had been uttered by another specific speaker: the target speaker. Hence, it is necessary to convert the features of source speaker into the features of target speaker. In this section, the steps in the transformation procedure are explained. This involves to convert the most relevant features of source speaker into the acoustic space of target speaker based on the two algorithms under study: k-means and k-histograms.

The features used to transform the acoustic space of source speaker can vary depending on the used method. Linear Predictor Coefficients (LPC) and the residual signal are the two more used features. LPC are used to model the vocal tract of speakers, formed by lungs, trachea, tongue, etc.

LPC are the coefficients of a polynomial of orden n that models a filter of the vocal tract of the speaker. The residual signal is the signal resulting of

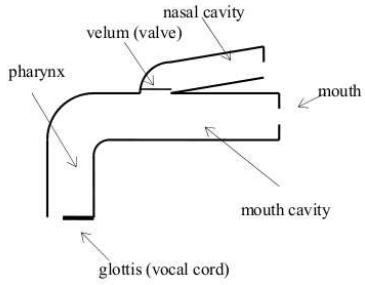

Fig. 2. Simplified view of a vocal tract

taking the voice signal and perform the inverse filtering through the LPC filter. Figure 3 shows the LPC model. Although, the presented features are the most important features for representing the acoustic space of each speaker, there are other features that must be taken into account, i.e. fundamental frequency, voiced/unvoiced frame, energy of each frame, etc.

Fig. 3. Model Proposed of vocal tract

The most widely used model to convert voice consists of taking the LPC parameters of source speaker to obtain an equivalent representation more robust to linear transformation: Line Spectral Frequencies (LSF) [16]. LSF coefficients are transformed into target parameters [7, 17] [1]. Other algorithms also converts the residual signal to obtain a better transformed signal [18].

The idea of our proposal is the use of the methods under study to convert the LSF source parameters into estimated LSF target parameters. The model proposed is shown in the Fig. 4. In this model, the source signal is processed with a pitch-synchronous analysis block [19]. The windowing is made using an assimetrical Hanning window with a length of two periods of pitch. Next, LPC parameters are obtained from each frame. The residual signal is obtained using the inverse filter. On the other hand, the LPC coefficients are transformed to obtain the LSF source parameters.

Then, the LSF source parameters are converted using the corresponding algorithm (in this case GMM or k-Histograms). A set of estimated LSF target parameters are obtained. These parameters are used to obtain a set of estimated

[1] Linear transformation of LPC coefficients may result in unstable converted LPC coefficients. LSF coefficients do not suffer such effects and ensure stability

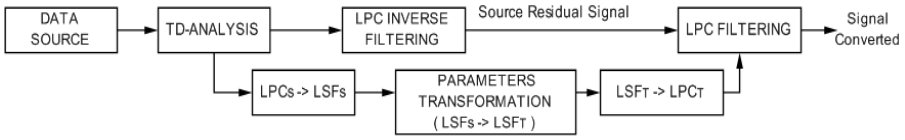

Fig. 4. System Proposed

LPC target parameters LSF_T. The converted voice is calculated by filtering the source residual signal using LSF_T.

3.1 Voice conversion using Gaussian Mixtures Model

The GMM voice conversion algorithm uses k-means and has four steps in the proposed experiments: windowing and parameterization, inverse filtering, parameter transformation and resynthesis. This proposal is very similar to the works of Stylianou and Kain [7, 8].

Each utterance is divided into overlapping pitch synchronous frames with a width of two periods. An assimetrical Hanning window is used to minimize boundary effects. The parameterization consists of a 20^{th} order LSF vector. The source excitation (the residual of LPC estimation) is calculated via inverse filtering with the LPC parameters obtained in each frame.

During the training process source and target LSF parameter vectors are aligned to obtain the mapping function using the k-means algorithm explained in Section 2.1. The alignment information is extracted from phone boundaries provided by a speech recognizer. Inside the boundaries of a frame, the alignment is linear.

The LSF parameters are transformed using the method proposed in Section 2.1. The transformed LSF parameters are converted into LPC coefficients, and they are used to obtain the target converted voice by filtering the source excitation. The fundamental frequency is transformed using a mean and standard deviation normalization and the signal is resynthesized using PSOLA [19].

Figure 5 shows the scheme of the proposed model. In this case, the target excitation is preferred to study the accuracy of LSF parameter conversion without the influence of an inaccurate excitation estimation.

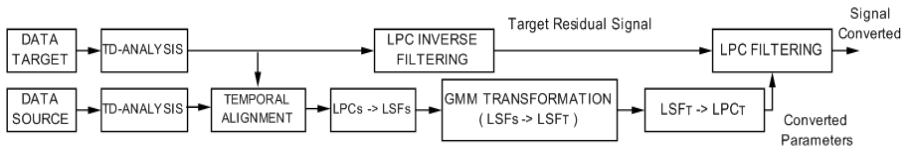

Fig. 5. System based on GMM

3.2 Voice conversion using K-Histograms

The voice conversion algorithm using k-histograms has the same four steps than the previous conversion model: windowing and parameterization, inverse filtering, parameter transformation and resynthesis.

During the training process source and target LSF parameter vectors are aligned to obtain the mapping function using k-histograms. The alignment information is extracted from phone boundaries provided by a speech recognizer. Inside the boundaries of a frame, the alignment is linear.

In this approach, the LSF parameters are transformed using the CDF estimated for the set with the highest probability calculated as shown in Eq. 16. The transformation includes a discretization of the LSF parameters that span from 0 to π. The degree of discretization is an adjustable parameter and it is directly related to the amount of available data to estimate the counts of the histograms.

Figure 6 shows the scheme of our proposal. In this implementation the residual signal of target speaker is also used for resynthesis.

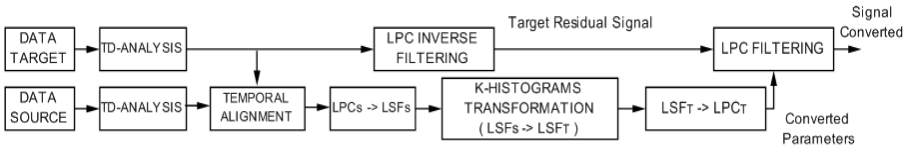

Fig. 6. System based on K-Histograms

Although the proposal is an approximation that uses statistical tools likewise the GMM model [7], we expect to obtain a better conversion with this non-gaussian approach, without introducing assumptions about the distribution of the LSF coefficients. The main drawback of our proposal is the discretization of LSF parameters that introduces noise in the estimation. Subjective experiments will show the influence of such quantization.

4 Experiments

The audio database used for the experiments contains 200 sentences in Spanish, uttered by two male and two female speakers. The sampling frequency was 16 KHz and the average duration of the sentences was 4 seconds. 50% of the sentences were used to train the conversion functions, while 30% were kept as development set (to tune model parameters) and 20% were used to perform the objective test.

One male and one female speaker were chosen as source, and the other two speakers were used as target, so four different conversion directions were considered: male to male (m2m), female to female (f2f), male to female (m2f) and

female to male (f2m). 38 sentences unseen during training were converted and resynthesized for all methods.

A third voice conversion method was included in the experiments. It consists of finding the closest feature vector of target speaker in training data to the real feature vector of target speaker. This voice conversion method based on frame selection that uses privileged information is named **FSOPT**. It is a measure of the highest achievable quality and identity by using the units in the training set (Optimal Frame Selection: FSOPT).

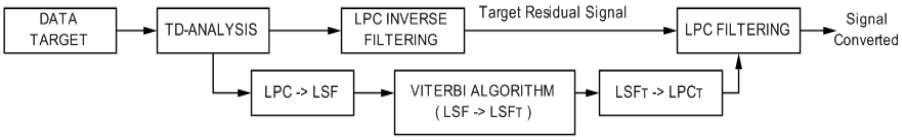

Fig. 7. Architecture of FSOPT

Some results will be shown using box-plots. This representation is an useful statistical tool to compare several statistical distributions. In our case we will use it to compare the distribution of the scores of different systems to study the significance of the differences.

4.1 Objective results

In this work the P distance (see Eq. 17) was used to measure the closeness of the converted voice to the target voice using the six voice conversion methods included in the experiments. The P distance was already used in several works about voice conversion [7].

$$P = 1 - \frac{d(y, \hat{y})}{d(x, y)} \tag{17}$$

The closer the converted parameters ($\hat{y}$) to the parameters of the target speaker (y) produces that P approaches to one. The distance between source parameters (x) and target parameters (y) allows to scale the P distance in the virtual path that goes from source to target parameters.

The P column of Table 1 shows that **FSOPT** is not as close as expected to the target voice, due to missing data in the limited training data. The GMM systems has the highest P score for voice conversion systems, followed by the proposed systems.

In order to validate the results of the P-score, a second experiment using a small speaker verification system (SVS) based on a GMM model [20] is made. MFCC coefficients are used to code the voice signal using a framing rate of $50Hz$. Two GMM models were trained using evaluation data to build source and target models. Given an utterance of a converted voice, these models may

	P	SVS	MOS-S	MOS-Q
TARGET	1.000	199.15	–	–
FSOPT	0.439	94.96	3.6	2.8
GMM	0.346	100.28	2.7	2.1
KH3140	0.197	86.04	3.6	3.0
KH314	0.194	84.64	3.6	2.8
SOURCE	$-\infty$	-189.64	–	–

Table 1. P, SVS, MOS-S and MOS-Q scores for all systems under evaluation, target and source voices.

be used to establish the closeness to source and target. The subtraction of the log-likelihood of source and target models is an indicator (score) of the performance of the conversion. A positive score indicates a good conversion, while a negative score is an indicator of closeness to source voice model.

The SVS column of Table 1 shows that GMM voice conversion system has the highest score using the speaker verification system while the other systems (KH and FSOPT) have the lower scores using this objective score.

The objective results seem poor for the voice conversion systems based on k-histograms and frame selection. However, the analysis of P and SVS scores using box-plots and the Wilcoxon test shown no statistically relevant differences between all voice conversion systems under evaluation except FSOPT. A subjective analysis is necessary to analyze the real perceptual differences between the methods under study.

4.2 Subjective Results

The subjective test was conducted with 35 sentences unseen during training. 15 volunteers were asked to listen to the converted-target sentence in random order. Listeners were asked to judge the similarity of the voices to the target using a 5-point scale, from 1 (totally different to target) to 5 (totally identical to target). On the other hand, the listeners were also asked to rate the quality of the converted sentences from 1 point (bad) to 5 points (excellent). The resulting scores for similarity are shown in Fig. 8.

The MOS of similarity shows that the methods based on k-histograms have a better similarity to target voice than GMM and FSOPT methods, as shown columns MOS-S and MOS-Q of Table 1.

However, the use of frame selection improves the performance of GMM voice conversion, as shown by Dutoit's proposal in our experiments. The similarity improves in 0.7 and quality in 0.3.

The Wilcoxon test of Table 2 shows only statistical relevant differences (black box for $p < 0.01$) in similarity scores of FSOPT, KH314 and KH3140 with respect to GMM. The Wilcoxon test for quality scores of Table 3 shows that all methods have statistical relevant differences ($p < 0.01$), except between FSOPT, KH314 and KH3140.

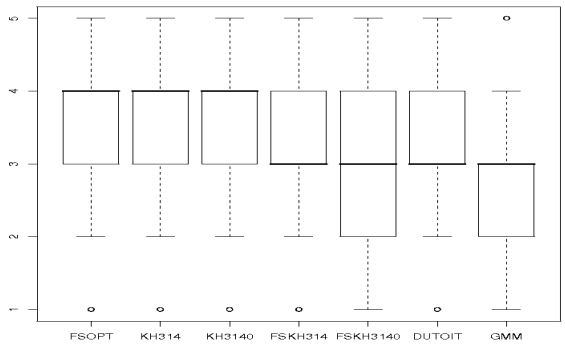

Fig. 8. MOS of similarity to target voice

	KH314	KH3140	GMM
FSOPT	□	□	■
KH314		□	■
KH3140			■

Table 2. Wilcoxon test for MOS-S ($p < 0.01$)

5 Conclusions

In this paper we presented a voice conversion algorithm based on a novel approach using a non-gaussian statistical transformation function.

Subjective experiments show that the method based on a non-gaussian statistical transformation has a better trade-off of similarity and quality than the other systems under evaluation.

The quantization introduced in the LSF parameters to estimate the histograms and to transform source coefficients into target coefficients did not show an impact in the MOS.

Here we have proved that k-histograms is a very good alternative to transform LSF coefficients in voice conversion. Future work will extend the system with state-of-the-art methods to include excitation, so that the quality of the complete voice conversion system makes it usable.

	KH314	KH3140	GMM
FSOPT	□	□	■
KH314		□	■
KH3140			■

Table 3. Wilcoxon test for MOS-Q ($p < 0.01$)

References

1. MacQueen, J.: Some methods for classification and analysis of multivariate observations. Proc. Fifth Berkeley Symp. on Math. Statist. and Prob. (1967) Vol.1 281–297.
2. He, Z., Xu, X., Deng, S., Dong, B.: K-Histograms: An Efficient Clustering Algorithm for Categorical Dataset. (2005).
3. He, Z.: Aproximation Algorithms for K-Modes Clustering. (2006).
4. Abe, M., Nakamura, S., Shikano, K., Kuwabara, H.: Voice conversion through vector quantization. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. (1988) 655–658.
5. Arslan, L.M.: Speaker transformation algorithm using segmental codebooks (STASC). Speech Communication. (1999) Vol.28 211–226
6. Valbret, H., Moulines, E., Tubach, J.P.: Voice transformation using PSOLA technique. Speech Communication. (1992) Vol.1 145–148.
7. Stylianou, Y., Cappe, O., Moulines, E.: Continuous probabilistic transform for voice conversion. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. (1998) Vol.6 Number 2 131–142.
8. Kain, A.: High resolution voice transformation. PhD thesis, OGI School of Science and Engineering. (2001).
9. Chen, Y., Chu, M., Chang, E., Liu, J., Liu, R.: Voice conversion with smoothed GMM and MAP adaptation. Proceedings of the European Conference on Speech Communications and Technology. (2003) 2413–2416.
10. Toda, T., Saruwatari, H., Shikano, K.: Voice conversion algorithm based on gaussian mixture model with dynamic frequency warping of STRAIGHT spectrum. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. (2001) 841–844.
11. Ye, H., Young, S.: Quality-enhanced voice morphing using maximum likelihood transformations. IEEE Transactions on Audio, Speech and Language Processing. (2006) Vol.14 Number 4 1301–1312.
12. Duxans i Barrobés, H.: Voice Conversion Applied to Text-To-Speech Synthesis. PhD Thesis (Universitat Politècnica de Catalunya). (2006).
13. Sündermann, D., Hoge, H., Bonafonte, A., Duxans, H.: Residual prediction. Proceedings of the IEEE Symposium on Signal Processing and Information Technology. (2005) 512–516.
14. Erro, D.: Intra-lingual and cross-lingual voice conversion using harmonic plus stochastic models. PhD Thesis (Universitat Politècnica de Catalunya). (2008).
15. Sündermann, D., Hoge, H., Bonafonte, A., Ney, H., Black, A., Narayanan, S.: Text-Independent Voice Conversion Based on Unit Selection. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. (2006).
16. Huang, X., Acero, A., Hon, H.W.: Spoken Language Processing. A Guide of Theory, Algorithm, and System Development. (2001).
17. Dutoit, T., Holzapfel, A., Jottrand, M., Moinet, A., Perez, J., Stylianou, Y.: Towards a Voice Conversion System Based on Frame Selection. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. (2007).
18. Erro, D., Moreno, A.: Weighted frequency warping for voice conversion. Proceedings of Interspeech 2007-Eurospeech. (2007) 1965–1968.
19. Moulines, E., Chanpentier, F.: Pitch Synchronous Waveform Processing Techniques for Text-to-Speech Synthesis Using Diphones. Speech Communication. (1990) 453–467.
20. Reynolds, D.A.: Speaker Identification and verification using Gaussian mixture speaker models. Speech Communication. (1995) Vol.17 91–108.