An Improved Voice Conversion Method Using Segmental GMMs and Automatic GMM Selection

Hung-Yan Gu and Sung-Fung Tsai
Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology
Taipei, Taiwan

***Abstract*—In this paper, the idea of segmental GMMs is proposed for voice conversion. Also, to apply this idea to on-line voice conversion, we have developed an automatic GMM selection algorithm based on dynamic programming. In addition, to map a vector of DCC (discrete cepstrum coefficients) with only one Gaussian mixture, we have designed a mixture selection algorithm. For evaluating the performance of the idea, segmental GMMs, three voice conversion system are constructed and used to conduct listening tests. The results of the listening tests show that segmental GMMs proposed here can indeed help to improve the performances in both timbre similarity and voice quality.**

Keywords-voice conversion; discrete cepstrum; Gaussian mixture model; timbre similarity; harmonic plus noise model

I. INTRODUCTION

The GMM based voice conversion method was introduced by Stylianou [1]. Afterward, many researches had tried to improve this method by considering one or several related issues [2-5]. Nevertheless, some serious problems still exist when applying the GMM based voice conversion method. The most noticeable one is that the converted spectrums are often over smoothed [2-4]. As a result, the converted voice is perceived with apparent distortion, i.e. the voice quality is significantly decreased. In addition, another noticeable problem is that two adjacent frames' converted spectrums may become discontinuous when the over smoothing problem is tried to solve by using just the most probable Gaussian mixture to map the source spectral coefficients [4, 6].

In this paper, we study to solve the over smoothing problem with a different approach. Note that the cause results to over smoothing is the summation across many Gaussian mixtures (usually 128 mixtures) in the GMM based mapping function,

$$y = F(x; \mu, \Psi) = \sum_{m=1}^{M} \left[\frac{w_m \cdot N(x; \mu_m^x, \Psi_m^{xx})}{\sum_{m=1}^{M} w_m \cdot N(x; \mu_m^x, \Psi_m^{xx})} \left(\mu_m^y + \left(\Psi_m^{yx}\right) \cdot \left(\Psi_m^{xx}\right)^{-1} \cdot (x - \mu_m^x) \right) \right] \quad (1)$$

where x denotes a feature vector of the source speaker, y denotes the converted feature vector for the target speaker, M is the number of Gaussian mixtures, and μ and Ψ represent the sets of mean vectors and covariance matrices, respectively. To solve the problem of over smoothing, we think reducing the number of Gaussian mixtures, M, in the mapping function is necessary. Nevertheless, the probability density function (PDF) of the trained GMM would become coarse when the number of mixtures is directly decreased. Therefore, we consider to segment each of the training sentences into a sequence of speech segments, and to group these speech segments into several classes. For example, a speech segment may be a phoneme or a syllable. After segmentation, the signal frames grouped to a class are taken to train a corresponding GMM with fewer mixtures (e.g. 16 mixtures). Then, this GMM is dedicated to convert the source frames recognized to belong to the corresponding class. In this way, the GMM based mapping function, i.e. (1), can be applied with fewer mixtures. That is, a complicated GMM is now replaced with multiple simpler GMMs, and each GMM is dedicated for converting the signal frames recognized to belong to its corresponding class.

In this paper, we study voice conversion for Mandarin, and Mandarin is a syllable prominent language. Therefore, we take each syllable of a labeled training sentence as a speech segment. Next, each segment is grouped to one of the 37 classes according to its syllable final. For each of the 37 syllable-final classes, a corresponding GMM is then trained. After training, the 37 GMMs are used for on-line voice conversion. Nevertheless, there is a problem that must be solved beforehand. That is, how can the right class that an input frame belongs to be picked out? For this problem, we have developed an automatic selection algorithm based on dynamic programming. This algorithm will be described in Subsection III.A.

Besides using multiple segmental GMMs to reduce the number of mixtures, we advanced furthermore to use only one Gaussian mixture for mapping a source spectrum into its converted spectrum in order to solve the problem of over-smoothed converted spectrum. Nevertheless, two adjacent source frames' converted spectrums may become discontinuous and result in artifact sounds. Therefore, we studied to design a dynamic programming based algorithm to consider both the likelihood (when using a particular Gaussian mixture) and spectral continuity simultaneously for a sequence of signal frames. This algorithm will be described in Subsection III.B. In addition, we have integrated the two solution methods mentioned to build an on-line voice conversion system. Then, this system is used to conduct the listening tests.

II. Training Procedure

As an overview, the processing flow for the training stage of our voice conversion system is as that drawn in Fig. 1. Three persons are invited to record 375 parallel sentences in a soundproof room. The sampling rate is 22,050Hz. Among the three persons, two are males, denoted as M1 and M2, and the other one is a female, denoted as F1. In this study, M1 is treated as the source speaker whereas M2 and F1 are treated as the target speakers, respectively. Therefore, the two voice conversion tasks here are converting the voice of M1 into the voice of M2 or F1.

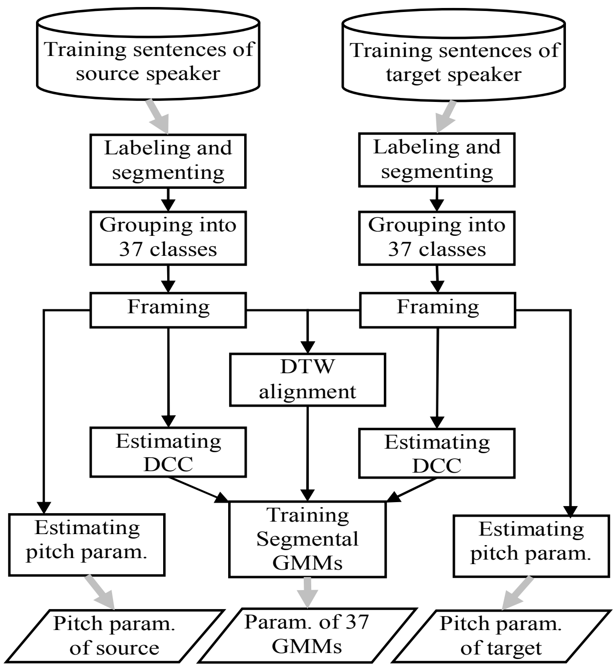

Figure 1. Processing flow for the training stage.

A. Labeling and Grouping

First, the software package, HTK, was used to do forced alignment, i.e. automatic labeling. Here, the speech unit is syllable. Since many errors are found in the labeled syllable boundaries, manual checking and correcting of the syllable boundaries are thus required. Here, we used the software, WaveSurfer, to edit the labels and boundaries. Then, according to the information of syllable boundaries and phonetic symbol, each syllable's signal was extracted and saved into a separate file which is named with sentence number, syllable number, and phonetic symbol. As a total, 2,926 syllables were extracted from the 375 recorded sentences of a speaker. Next, the syllables from the first 350 sentences are grouped into 37 classes according to the syllable-final symbol parsed from the filename of each saved signal file.

B. DCC Estimation

There are several methods proposed for estimating a signal frame's magnitude-spectrum envelope (spectral envelope). The method, STRAIGHT, is very accurate in its estimated spectral envelope but it requires a large amount of computations and cannot be used to implement a real-time system currently.

Therefore, in this study, we adopt the spectral envelope estimation method, discrete cepstrum [7, 8], and use the estimated discrete cepstrum coefficients (DCC) as the spectral parameters. For each signal frame, the DCC estimation scheme developed previously [8] is executed to obtain 40 DCC. Here, a frame's width is 512 sample points, and adjacent frames are placed 110 points (about 5ms) apart.

C. Training of Segmental GMM

After the block, "grouping into 37 classes", in Fig. 1 is executed, there would be 37 classes of syllable segments. For each class, a GMM of 16 mixtures is trained from those syllable signals grouped to the class. The GMM obtained is hence termed a segmental GMM.

Here, a parallel corpus is used. Each source syllable and its corresponding target syllable are time aligned first with DTW as indicated in the block, "DTW alignment". Then, the DCC computed from a source frame is jointed with the DCC computed from the aligned target frame. With the jointed vectors of DCC, the training method based on maximum likelihood estimate is used to train a GMM for each class [9].

D. Pitch Parameters

A pitch detection method based on both autocorrelation and AMDF is used to detect the pitch frequency of a signal frame [10]. Then, the pitch frequencies detected from a speaker's utterances are collected to compute their average and standard deviation, which are the pitch parameters used in this study.

III. Conversion Procedure

The procedure proposed here for converting voice is as the processing flow drawn in Fig. 2. When a spoken sentence with unknown content is inputted, it will be sliced into a sequence of frames first with the frame width and shift as given in Subsection II.B. Then, the pitch frequency of each frame is detected in the left flow of Fig. 2 with the method mentioned in Subsection II.D. When a frame is detected to be unvoiced, the three gray colored blocks in Fig. 2 are bypassed directly. That is, pitch adjusting is not needed and the spectral parameters, DCC, are not converted. On the other hand, when a frame is detected to be voiced, its pitch is simply converted as

$$q_t = \mu^y + \frac{\sigma^y}{\sigma^x}(p_t - \mu^x) \tag{2}$$

where p_t is the detected pitch frequency, μ^x and σ^x are the average and standard deviation of the source speaker's pitch frequencies.

As for the right flow of Fig. 2, the input frames are processed one after another basically. Nevertheless, in the block, "Selecting a GMM", we propose a selection algorithm that processes every 20 voiced frames in a batch. With this algorithm, the correct GMM (or its nearby GMM sometimes) can be picked out from the 37 GMMs for each frame. Then, in the block, "Mapping with single mixture", only one mixture of the selected GMM is used to map the DCC in order to avoid spectral over smoothing. Nevertheless, the mixture selected for mapping is not always the most probable one. This is because spectral continuity between adjacent converted frames must also be considered to prevent artifact sounds from being generated. For the problem of mixture selection, we have developed a dynamic programming based algorithm that is different from the one studied by previous researchers [4]. Hence, in this block, a sequence of voiced frames bounded with left and right unvoiced frames are processed in a batch. Finally, in the jointed block, "HNM based speech synthesis", speech signals are re-synthesized using an HNM (harmonic plus noise model) based method [8, 11].

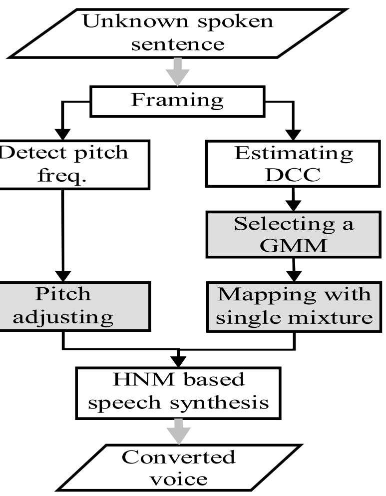

Figure 2. Processing flow for the conversion stage.

A. GMM Selection

Since the content of the input speech is unknown, which one of the 37 GMMs should be selected for mapping each frame's DCC becomes a problem that must be solved. In general, this is a problem of speech recognition. Nevertheless, it is not so serious because some frames are assigned with incorrect but similar GMMs are tolerable.

Here, we intend to use the 37 GMMs trained to take the role of HMM usually used for speech recognition. In addition, we observe that it is impossible for a person to utter more than 2 segments (i.e. syllables here) within a very short time interval, e.g. 100ms. Therefore, we decide to select GMMs for every 20 successive voiced frames (spanning 100ms of time) in a batch. Then, only one or two of the 37 GMMs should be picked out. Here, we have developed a dynamic programming based algorithm that selects one or two GMMs according to the criterion of maximum likelihood.

Let the probability that the t-th input frame's DCC are generated by the s-th GMM be $G_t(s)$. That is,

$$G_t(s) = \sum_{m=1}^{M} w_m(s) \cdot N\left(x_t;\ \mu_m^x(s),\ \Psi_m^{xx}(s)\right). \tag{3}$$

where $w_m(s)$ is the weight of the m-th mixture, and x_t is the vector of DCC for the t-th frame. In addition, let $R(t, s)$ be the logarithmic likelihood that the frames from time 1 to time t are all generated by the s-th GMM. In contrast, let $D(t, s)$ be the logarithmic likelihood that the frames from time 1 to t are generated by two GMMs and the t-th frame is generated by the s-th GMM. In terms of these definitions, we can derive the two recursive formula,

$$R(t,s) = \log\left(G_t(s)\right) + R(t-1, s)\,, \tag{4}$$

$$D(t,s) = \log\left(G_t(s)\right) + \max\left\{ \max_{0 \le v < 37,\, v \ne s} \left[R(t-1,v)\right],\, D(t-1,s) \right\}, \tag{5}$$

where the boundary values are $D(1, s) = 0$ and $R(1, s) = \log(G_1(s))$. Then, the maximum likelihood can be calculated as

$$A(T) = \max\left\{ \max_{0 \le v < 37} \left[R(T,v)\right],\ \max_{0 \le v < 37} \left[D(T,v)\right] \right\}. \tag{6}$$

where the final time T is set to 20 in this study. In terms of (4), (5), and (6), we can calculate the maximum likelihood, $A(20)$, and then back track to find the sequence of GMM indices that are best for assigning to the batch of 20 voiced frames.

B. Mapping with Single Mixture

Mapping an input frame's DCC with a single Gaussian mixture is meant that the summation and the weighting term of (1) are removed. That is, the converted DCC vector, y, is calculated as,

$$y = F^k(x) = \mu_k^y + \left(\Psi_k^{yx}\right) \cdot \left(\Psi_k^{xx}\right)^{-1} \cdot (x - \mu_k^x)\,, \tag{7}$$

where x is the input frame's DCC and $F^k(x)$ denotes the mapping function using the k-th mixture.

The developed dynamic programming based algorithm for mixture selection is as the following. Let the index of the GMM selected by Subsection III.A for the t-th frame be $I(t)$. Denote the mapping function using the k-th mixture as $F_{I(t)}^k(x_t)$. In addition, let $C(t, k)$ represent the cumulated distance from time 1 to time t and the index of the mixture used at time t be k. Then, we design the recursive formula,

$$C(t,k) = \min_{\substack{0 \le m < M,\\ w_m(I(t-1)) > H}} \left[dist\left(F_{I(t)}^k(x_t),\, F_{I(t-1)}^m(x_{t-1})\right) + C(t-1,m) \right], \tag{8}$$

to realize dynamic programming, where $dist(\bullet,\bullet)$ is a geometric distance measure for DCC, H is a threshold set to 0.3 empirically, and $w_m(s)$ is the weight of the m-th mixture. At time 0, the values of $C(0, k)$ are directly set as $C(0, k) = 0$, $0 \le k < M$. Finally, at time T, the minimum cumulated distance $B(T)$ is computed as

$$B(T) = \min_{0 \le k < M,\, w_k(I(T)) > H} \left[C(T,k)\right]. \tag{9}$$

In terms of (8) and (9), the minimum cumulated distance can be obtained. Also, the sequence of mixture indices for the frames from time 1 to T can be obtained through backtracking.

C. HNM Based Speech Synthesis

In HNM, the spectrum of a voiced frame is divided into the lower-frequency harmonic part and the higher-frequency noise part. The frequency that the two parts are divided according to is termed the maximum voiced frequency (MVF). In the original work [11], a method is provided to dynamically detect each frame's MVF. Here, to simplify the synthesis processing, we just use the static MVF value, 6,000Hz, across all voiced frames.

Suppose the i-th and (i+1)-th frames are both voiced and have L^i and L^{i+1} harmonic partials, respectively. To synthesize a signal sample for the t-th sampling point between the i-th and (i+1)-th frames, we first derive the frequencies, $f_k(t)$, and amplitudes, $a_k(t)$, of the harmonic partials for this sampling point with linear interpolation. That is,

$$\begin{aligned} f_k(t) &= f_k^i + \frac{f_k^{i+1} - f_k^i}{N} t, \quad k = 1,2,...,L, \\ a_k(t) &= a_k^i + \frac{a_k^{i+1} - a_k^i}{N} t, \quad k = 1,2,...,L \end{aligned} \tag{10}$$

where N is the number of sampling points between two adjacent frames, L is the larger one of L^i and L^{i+1}, and f_k^i and a_k^i are the frequency and amplitude for the k-th harmonic partial of the i-th frame. The value of f_k^i is simply computed as $k \times q_i$ where q_i is the converted pitch frequency for the i-th frame. As to a_k^i, its value is derived from the converted vector of DCC. The detail of the derivation is referred to our previous work [8]. Here, we directly set a_k^i = 0, k= L^i +1, …, L^{i+1}, if L^i is less than L^{i+1}. Then, the harmonic signal, $h(t)$, for the t-th sampling point is computed as

$$\begin{aligned} h(t) &= \sum_{k=1}^{L} a_k(t) \cdot \cos(\phi_k(t)), \quad 0 \le t < N, \\ \phi_k(t) &= \phi_k(t-1) + 2\pi \cdot f_k(t) / 22,050 \end{aligned} \tag{11}$$

where $\phi_k(t)$ denotes the cumulated phase on time t for the k-th harmonic partial and 22,050 is the sampling frequency. $\phi_k(-1)$ is defined to be $\phi_k(N\text{-}1)$ of the last frame to keep continuity of phase. If i = 0, *i.e.* there is no last frame, the value of $\phi_k(-1)$ is then set randomly.

IV. EXPERIMENTAL EVALUATIONS

For evaluating the conversion method proposed here, we have constructed three kinds of voice conversion systems, named SOG, SSG, and SLG, respectively. In the system SOG (system using original GMM for mapping), a single GMM of 256 mixtures are trained with the 350 training sentences, and then the mapping function, (1), is used to convert the DCC of each input frame. In the system SSG (system using single Gaussian mixture for mapping), we still trained a single GMM of 256 mixtures. Nevertheless, in the conversion stage, the mixture selection method as described in Subsection III.B is applied, and then the DCC of a frame is converted with the single Gaussian mixture selected. As to the system SLG (system using selected GMM for mapping), we trained 37 segmental GMMs instead of a single GMM and the number of mixtures for each GMM is 16. Then, in the conversion stage, the GMM selection method as described in Subsection III.A is applied. Next, the mixture selection method as described in Subsection III.B is applied, too.

Using the three systems, we can obtain three different converted voice files for a source voice file. In terms of the converted voice files, we have conducted two types of listening tests. The first type is for timbre similarity whereas the second type is for voice quality. For each type of listening tests, 25 persons are invited to listen to the voice files and give relative scores. Among the 25 persons, 20 of them are not familiar with the research field of voice conversion.

A. Timbre Similarity Tests

In the tests of timbre similarity, 5 voice files are prepared first, which are named VS (uttered by the source speaker), VT (uttered by target speaker), VX1 (converted by SOG), VX2 (converted by SSG), and VX3 (converted by SLG). Among the 5 files, VS and VT are of same content whereas VX1, VX2, VX3 are of same content but different from VS and VT. These 5 files can be downloaded by accessing the web page: http://guhy.csie.ntust.edu.tw/VoiceConv/. During listening tests, these files are played in the order ABX where A is fixed to VS, B is fixed to VT, and X is randomly selected from VX1, VX2, and VX3. Each time that three files, ABX, are played, the participant is requested to give a score. Here, the score range is from 1 to 9. The score 9 (1) means the timbre of X is sure to be that of B (A), the score 7 (3) means the timbre of X is more like that of B (A), and the score 5 means the timbre of X cannot be judged.

After listening tests, the scores given by the 25 persons are collected to compute average scores (AVG) and standard deviations (STD) for the three systems respectively. The results are those values listed in Table I. From this table, it can be seen that the average scores for voice conversion between different genders (i.e. from M1 to F1) are much higher than those for voice conversion between same genders (i.e. from M1 to M2). In addition, when the average scores of the three systems are compared, it can be found that the average scores of the system SLG are much better than those of the system SSG whereas the average scores of SSG are slightly better than those of SOG. Therefore, the idea of segmental GMMs and automatic GMM selection can indeed help to improve the timbre similarity of the converted voices.

TABLE I. AVERAGE SCORES AND STD FOR TIMBRE SIMILARITY TESTS

		SOG	SSG	SLG
M1=>M2	AVG	6.08	6.24	7.05
	STD	1.11	1.09	0.93
M1=>F1	AVG	6.92	7.24	7.60
	STD	1.13	1.07	1.10

B. Voice Quality Tests

In the tests of voice quality, the three converted voice files, VX1, VX2, and VX3 are used. These files are played in the order AX where A is fixed to VX1 and X is randomly selected from VX2 and VX3. Each time that two files, AX, are played, the participant is requested to give a score. Here, the score range is from 1 to 9. The score 9 (1) means the quality of X is much better (worse) than A, the score 7 (3) means the quality of X is slightly better (worse) than A, and the score 5 means the quality of X cannot be distinguished from that of A.

After listening tests, the scores given by the 25 persons are collected to compute average scores and standard deviations for the two systems, SSG and SLG, respectively. The results are those values listed in Table II. From Table II, it can be found that the average scores for conversion from M1 to M2 (i.e. same gender) is about 0.5 better than the average scores for conversion from M1 to F1. This indicates that the quality of the converted voice from different genders is harder to improve. In addition, when the average scores of the two systems, SSG and SLG, are compared, it can be found that the scores of SLG are all higher than 5.0 and are much better than those of SSG. Therefore, the idea of segmental GMMs and automatic GMM selection can indeed help to improve the quality of the converted voice.

TABLE II. AVERAGE SCORES AND STD FOR VOICE QUALITY TESTS

		SSG vs SOG	SLG vs SOG
M1=>M2	AVG	5.23	6.04
	STD	1.43	1.45
M1=>F1	AVG	4.89	5.55
	STD	1.50	1.47

C. *Cepstrum Distances Measuring*

There are 25 remaining parallel sentences that are not used in the training stage. Here, these source sentences are fed to the three systems to obtain their corresponding converted sentences, respectively. Then, a geometric distance of DCC is measured between each voiced frame of the converted sentences and its corresponding frame in the target sentences according to the saved DTW alignment data. Next, the measured distances are averaged across all voiced frames. As a result, the average distances obtained for the three systems are as those listed in Table III. From this table, it is seen that the system SOG will obtain the smallest average distances. Nevertheless, the results of listening tests show that the system SOG is the worst in timbre similarity and is worse than SLG in voice quality. Therefore, the average cestrum distances are inconsistent with the results of the listing tests for SOG. On the other hand, the performance improvements of SLG as compared with SSG are reflected in the measured average distances. SLG and SSG both use single Gaussian mixture to map the spectral parameters of DCC but GMM selection is only adopted in SLG.

TABLE III. AVERAGE DISTANCES FOR THE THREE SYSTEMS

	SOG	SSG	SLG
M1=>M2	0.543	0.609	0.601
M1=>F1	0.598	0.634	0.612

V. CONCLUSION

According to the results of the listening tests, the system SLG is the best in both timbre similarity and voice quality among the three systems. This demonstrates that the idea of segmental GMMs and the automatic GMM selection algorithm proposed here can indeed help to improve the performances of the GMM based voice conversion mechanism. As to cepstrum distance measured, the system SOG still obtains the smallest average distance. Nevertheless, it suffers the fault of over-smoothed converted spectrums, which result in degraded voice quality and timbre similarity. In the future, the unit of speech segments can be reduced from syllables to voiced consonant and vowels. It is hopeful that the performance of voice conversion using segmental GMMs is further improved.

ACKNOWLEDGMENT

This study is supported by National Science Council of Taiwan under the contract number, NSC 99-2628-E-011-107.

REFERENCES

[1] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," IEEE trans. Speech and Audio Processing, vol. 6, no. 2, pp.131–142.1998.

[2] T. Toda, H. Saruwatari, and K. Shikano, "Voice conversion algorithm based on Gaussian mixture model with dynamic frequency warping of STRAIGHT Spectrum," Int. Conf. Acoust., Speech, and Signal Processing, Salt Lake City, pp. 841-844, 2001.

[3] T. Toda and A. W. Black and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," IEEE trans. on Audio, Speech, and Language Processing, vol. 15, no. 8, pp. 2222-2235, 2007.

[4] Z. H. Jian and Z. Yang, "Voice Conversion Using Viterbi algorithm based on Gauaaian mixture model", Int. Symposium on Intelligent Signal Processing and Communication Systems, pp. 32-35, Xiamen, China, 2007.

[5] Z. Yue, X. Zou, Y. Jia, and H. Wang, "Voice conversion using HMM combined with GMM", 2008 Congress on Image and Signal Processing, Sanya, China, pp. 366-370, 2008.

[6] E. Godoy, O. Rosec, and T. Chonavel, "Alleviating the one-to-many mapping problem in voice conversion with context-dependent modeling", Proc. INTERSPEECH, pp. 1627-1630, Brighton, UK, 2009.

[7] O. Cappé and E. Moulines, "Regularization techniques for discrete cepstrum estimation," IEEE Signal Processing Letters, vol. 3, no. 4, pp. 100-102, 1996.

[8] H. Y. Gu and S. F. Tsai, "A discrete-cepstrum based spectrum-envelope estimation scheme and its example application of voice transformation," International Journal of Computational Linguistics and Chinese Language Processing, vol. 14, no. 4, pp. 363-382, 2009.

[9] R. A. Redner and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," SIAM Review, vol. 26, no. 2, pp. 195-239, 1984.

[10] H. Y. Kim, *et al.*, "Pitch detection with average magnitude difference function using adaptive threshold algorithm for estimating shimmer and jitter," 20-th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society, Hong Kong, China, 1998.

[11] Y. Stylianou, Harmonic plus noise models for speech, combined with statistical methods, for speech and speaker modification*,* Ph.D. thesis, Ecole Nationale Supèrieure des Télécommunications, Paris, France, 1996.