JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 31, 609-626 (2015)

A Voice Conversion Method Combining Segmental GMM Mapping with Target Frame Selection*

HUNG-YAN GU AND SUNG-FENG TSAI
Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology
Taipei, 106 Taiwan

In this paper, a voice conversion approach that combines two distinct ideas is proposed to improve the converted-voice quality. The first idea is to map spectral features, *e.g.* discrete cepstrum coefficients (DCC), with segmental Gaussian mixture models (GMMs). That is, a single GMM of a large number of mixture components is replaced here with several voice-content specific GMMs each consisting of much fewer mixture components. In addition, the second idea is to find a frame, of spectral features near to the mapped feature vector, from the target-speaker frame pool corresponding to the segment class as the input frame belongs to. Both ideas are intended to alleviate the problem encountered by a traditional GMM based conversion method, *i.e.* converted spectral envelopes are usually over smoothed. To apply the first idea to implement an on-line voice conversion system, we have proposed an automatic GMM selection algorithm based on dynamic programming (DP). Furthermore, as pointed out by previous researchers, mapping with a single selected Gaussian probability density function (PDF) instead of a combination of several Gaussian PDFs is helpful to obtain better converted-voice quality. Therefore, we have also proposed a Gaussian PDF selection algorithm and integrated it into our system. As to the implementation of the second idea, an algorithm based on DP is adopted which will consider both frame matching and connecting distances. For evaluating the performance of the two ideas studied here, three voice conversion systems are constructed, and used to conduct listening tests. The results of the tests show that the system with the two ideas combined can indeed obtain much improved voice quality besides improvement in timbre similarity.

Keywords: voice conversion, Gaussian mixture model, frame selection, discrete cepstrum coefficients, dynamic programming

1. INTRODUCTION

The research of voice conversion is to develop an effective method for converting one person's voice to a voice that resembles a particular person [1, 2]. Historically, the GMM based voice conversion method was first introduced by Stylianou [3]. Afterward, many researches had tried to improve this method by considering one or two related issues. The related issues include spectral over-smoothing found in converted spectrums [4-7], spectral discontinuities between some adjacently converted frames [4, 5, 7], prosody conversion [8, 9], and other minor issues. Although previous researchers had already proposed their methods to improve voice-conversion performances, these issues, however, need more investigations in order to have various kinds of solution methods to satisfy different requirements by different application developers. Possible requirements include

Received May 29, 2013; revised August 6 & September 17, 2013; accepted October 13, 2013.
Communicated by Chung-Hsien Wu.
* This work was supported by National Science Council, Taiwan, under the contract number, NSC 99-2628-E-011-107.

(a) voice quality first with acceptable similarity; (b) voice similarity first with acceptable quality; (c) voce quality compromised with implementation cost, *etc.* We know that the issue, spectral over-smoothing, had been tackled with at least two kinds of methods, global variance (GV) [6, 7] and dynamic frequency warping (DFW) [4, 5]. Additionally, the methods based on DFW are intended to remedy a weak point of the GV based methods, *i.e.* the correlation between the source and target parameters is low [5]. In this paper, we also study the issue, spectral over-smoothing, but with a different approach, segmental GMM plus target frame selection. The advantages of our approach include (a) simpler in concept; (b) easier to implement (hence saving efforts or money); (c) compromised processing-time latency (*e.g.* 30 frames) between DFW (1 frame) and GV (utterance level); (d) effective for improving the converted-voice quality (the signal quality of the converted voice).

If the converted spectrums are over smoothed, the converted voice will be perceived of some distortions and the voice quality will be decreased apparently. In addition, some adjacent source frames' converted spectra may become discontinuous when the issue of spectral over-smoothing is tackled by using only the most probable Gaussian PDF (or mixture component) to map the source spectral coefficients [10]. What is spectral over-smoothing? We illustrate it by the two curves of magnitude-spectrum envelopes (spectral envelopes) shown in Fig. 1. The dot-lined curve represents an envelope of a recorded target (target-speaker) frame whereas the solid-lined curve represents an envelope of a converted frame. Apparently, the formants, F2, F4 and F6, of the converted envelope become much broader (*i.e.* bandwidth become larger) as compared with the ones of the target envelope. Also, the depths of peak-to-left-valley of the mentioned formants are considerably decreased for the converted envelope. Therefore, a converted envelope that has the two phenomena observed is attributed as over-smoothed.

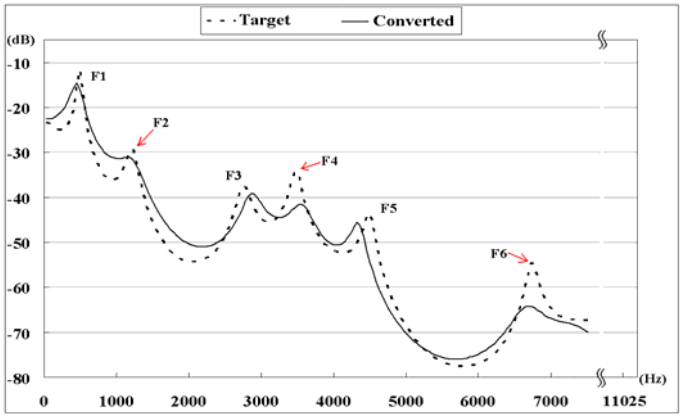

Fig. 1. An example of an over-smoothed spectral envelope.

The cause resulting to over-smoothing we think is the summation across too many Gaussian PDFs (usually 128 PDFs) in a GMM based mapping function [3],

$$y = F(x; \mu, \Psi) =$$

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

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 PDFs, w_m is the weight of the mth mixture component, and μ and Ψ represent the sets of mean vectors and covariance matrices, respectively. To solve the problem of spectral over-smoothing, we think reducing the number of Gaussian PDFs, M, in the mapping function is necessary. Nevertheless, the probability density function (PDF) of the trained GMM would become coarse when the number of mixture components 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, a syllable, or an acoustically defined sub-syllable [11]. After segmentation, the signal frames grouped to a same class are taken to train a corresponding GMM with fewer Gaussian PDFs (*e.g.* 8 PDFs). Then, this GMM is dedicated to convert the source signal frames recognized to belong to its corresponding class. In this way, the GMM based mapping function, *i.e.* Eq. (1), can be applied with fewer mixture components. 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 Chinese, and Mandarin Chinese is a syllable prominent language. Therefore, we treat each syllable of a labeled training sentence as one segment if the syllable has no initial consonant or has just unvoiced initial consonant, or as two segments (*i.e.* the voiced initial consonant and syllable final) if the syllable is started with a voiced consonant. Next, each segment is grouped to one of the 39 classes, including 4 classes of voiced initial consonants (*i.e.* /m/, /n/, /l/, /r/) and 35 classes of syllable finals. In Mandarin Chinese, a syllable final is a vowel nucleus consisting of one to three vowels plus a possible nasal ending. In details, the 35 types of syllable finals are listed in Table 1 for reference. For each of the 39 classes, a corresponding GMM will be trained from the segments grouped to. After training, the 39 GMMs are used for on-line voice conversion. Nevertheless, there is a problem that must be solved beforehand. The problem is how the right class that an input frame belongs to can be picked out? For this problem, we have developed an automatic selection algorithm based on dynamic programming. This algorithm will be described in section 3.1.

Table 1. The 35 types of syllable finals in Mandarin Chinese.

Structures	Members							
Single vowel	a	o	ə	u	i	y	ɯ	
Diphthong	ua	au	ia	ai	uo	ou	ie	ei
Triphthong	uai	iau	iou	uei				ye
Nasal ended (*n*)	an	uan	ən	uən	ien	yen	in	yn
Nasal ended (*ŋ*)	aŋ	iaŋ	uaŋ	əŋ	oŋ	yoŋ	iŋ	

Besides using multiple segmental GMMs to reduce the number of mixture components, we advanced furthermore to use only one Gaussian PDF for mapping a source spectrum into its converted spectrum in order to help alleviate 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 DP based algorithm to consider both the likelihood (when taking a particular Gaussian PDF) and the spectral continuity (between two adjacent frames) simultaneously for a sequence of signal frames. This algorithm will be described in section 3.2.

In this paper, we not only improve the method, segmental GMMs, presented in our previous work [12], but also extend it by adding an essential processing step, frame selection, to further alleviate the problem of spectral over smoothing. By frame selection, each converted feature vector is replaced with a real (*i.e.* not converted) feature vector analyzed from a target frame in order to improve the converted-voice quality. In fact, the idea of frame selection is proposed previously by Dutoit, *et al.* [13]. In that paper [13], the feature vector of a source frame is mapped with a conventional GMM, and then a target frame is searched, in terms of the mapped feature vector, with a DP based algorithm. Here, we map the feature vector of a source frame with a segmental GMM, and then search for a target frame with a developed DP algorithm. The detail of this algorithm will be explained in section 3.3. We think that the two steps, spectral mapping and frame selection, are not independent. A better spectral mapping method would help the module, frame selection, to find out a more appropriate target frame. By cascading the two steps, segmental GMM based spectral mapping and target frame selection, we have built an on-line voice conversion system. Then, this system and two other systems with different option setting are used to conduct listening tests.

This paper is structured as follows. Section 2 first describes the voice data recorded and the steps of the training stage to build a voice conversion system according to our approach. Section 3 then describes the steps of the voice conversion stage for our system to convert a source speaker's utterance. Section 4 presents results from subjective and objective evaluation experiments, demonstrating that our approach can provide significantly improved voice quality. Finally, the main conclusions of this work are summarized in Section 5.

2. TRAINING PROCEDURE

As an overview, the processing flow for the training stage of our voice conversion system is as that drawn in Fig. 2. 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 MA and MB, and the other one is a female, denoted as FA. In this study, MA is treated as the source speaker whereas MB and FA are treated as the target speakers, respectively. Therefore, the two voice conversion tasks here are to convert the voice of MA into the voice of MB or FA.

2.1 Labeling and Grouping

First, the software package, HTK (HMM tool kit) [14], was used to execute forced alignment, *i.e.* automatically labeling the syllable boundaries. Since many errors are

found in the labeled results, manual checking and correcting of the syllable boundaries are thus required. Here, we used the software, WaveSurfer [15], to edit the labels and boundaries. Also, a boundary between syllable initial and final is placed for a syllable of a voiced initial (consonant). 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 39 classes according to the phonetic symbol and boundary information from the filename of each saved signal file.

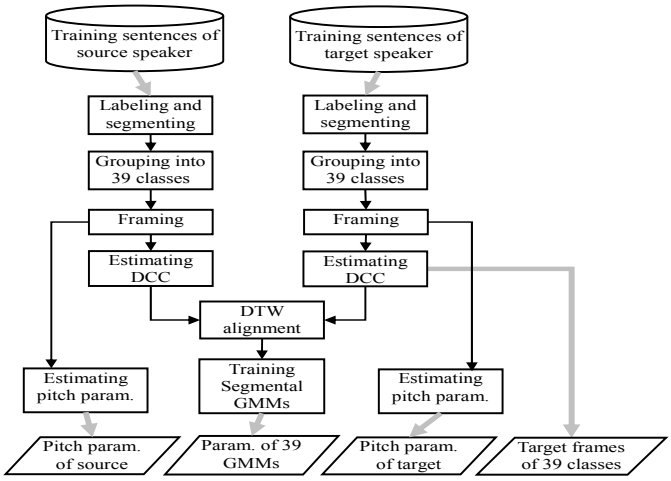

Fig. 2. Processing flow for the training stage.

2.2 Estimation of Discrete Cepstrum Coefficients

There are several methods proposed for estimating a signal frame's spectral envelope. The method, STRAIGHT [16], 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 [17, 18], and use the estimated discrete cepstrum coefficients (DCC) as the spectral features. For each signal frame, the DCC estimation scheme proposed in a previous work [18] is used to calculate 40 DCC. In that scheme, a mel-like frequency scale is adopted. Here, a frame's width is 512 sample points, and adjacent frames are placed 110 points (5 ms) apart. In addition, the estimated DCC of each target frame are stored with its frame-sequence number to one of the 39 target-frame pools according to the segment class that this frame belongs to.

2.3 Training of Segmental GMMs

After the block, "grouping into 39 classes", in Fig. 2 is executed, there would be 39

classes of segments. For each class, a GMM of 8 mixture components was trained from those speech segments grouped to that class. Such a GMM obtained is hence termed a segmental GMM.

Here, a parallel corpus is used. Each source (source speaker) syllable and its corresponding target syllable were time aligned first with dynamic time warping (DTW) as indicated in the block, "DTW alignment". Then, the DCC computed from a source frame was jointed with the DCC computed from the aligned target frame. With the jointed vectors of DCC, the training method based on maximum likelihood estimate was used to train a GMM for each class [19].

2.4 Pitch Parameters

A pitch detection method based on both autocorrelation and absolute magnitude difference function (AMDF) is used to detect the pitch frequency of a signal frame [20]. To prevent some unvoiced frames from being incorrectly detected as voiced, we also calculate the value of zero-crossing rate (ZCR) for each frame and use ZCR to determine if a frame is unvoiced beforehand. 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.

3. CONVERSION PROCEDURE

The procedure proposed here for converting voice is as the processing flow drawn in Fig. 3. 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 section 2.2. Then, the pitch frequency of each frame is detected in the left flow of Fig. 3 with the method mentioned in section 2.4. When a frame is detected to be unvoiced, the four gray colored blocks in Fig. 3 are bypassed directly, which means that 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 with the equation,

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

where p_t is the detected pitch frequency, $\mu^{(x)}$ and $\sigma^{(x)}$ are the average and standard deviation of the source speaker's pitch frequencies, and $\mu^{(y)}$ and $\sigma^{(y)}$ are the average and standard deviation of the target speaker's pitch frequencies.

In the right flow of Fig. 3, the input frames are processed one after another basically. Nevertheless, in the block, "Selecting a GMM", we propose a selection algorithm that processes every 30 voiced frames in a batch. With this algorithm, the correct GMM (or its nearby GMM sometimes) can be picked out from the 39 GMMs for each frame. Then, in the block, "Mapping with single PDF", only one Gaussian PDF of the selected GMM is used to map the DCC in order to alleviate the problem of spectral over-smoothing. Nevertheless, the Gaussian PDF 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 Gaussian PDF selection, we have developed a DP based algorithm that is different from the one studied by previous researchers [6]. Hence, in this block, a sequence of voiced frames bounded with left and right unvoiced frames are processed in a batch. Next, in the block, "selecting target frames", the sequence of voiced frames is also processed in a batch with another developed DP algorithm. Similarly, spectral continuity between adjacently selected target frames must also be considered besides the spectral matching distance between the feature vector of the converted input frame and the feature vector of a target frame. Finally, in the jointed block, "HNM based speech synthesis", speech signals are re-synthesized using a harmonic plus noise model (HNM) based method [21, 22].

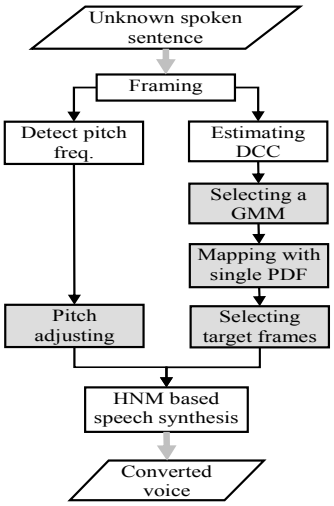

Fig. 3. Processing flow for the conversion stage.

3.1 GMM Selection

Since the content of the input speech is unknown, which one of the 39 GMMs should be selected for mapping each input 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 nearby GMMs are tolerable.

Here, we use the 39 segmental GMMs trained to take the role of HMM (hidden Markov model) usually used for speech recognition. In addition, we notice that it is rare for a person to utter more than 2 different segments (syllables) within a very short time interval, *e.g.* 150 ms, under an ordinary speaking rate. Therefore, we decide to select GMMs for every 30 successive voiced frames (spanning 150ms of time) in a batch. Actually, we have experimented to inspect the differences in the numbers of segments selected when setting the batch length to 20, 30, and 40, respectively. It is found that the batch length, 30, is indeed a better choice. Then, only one or two of the 39 GMMs will be picked out for a batch of 30 voiced frames. Here, we have developed a DP based algorithm that selects one or two GMMs according to the criterion of maximum likelihood.

Let the probability that the tth input frame's DCC are generated by the sth GMM be $G_t(s)$. Hence,

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

where $w_m(s)$ is the weight of the mth mixture component, and x_t is the vector of DCC for the tth frame. In addition, let $R(t, s)$ be the logarithmic likelihood (*i.e.* in logarithmic scale) that the frames from time 1 to t are all generated by the sth GMM. Here, $R(t, s)$ is intended to record the probability that the voiced part spanning the frames from time 1 to t is within the time interval occupied by a syllable of the syllable-final type corresponding to the sth 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 tth frame is generated by the sth GMM. Here, $D(t, s)$ is intended to record the probability that the voiced part spanning the frames from time 1 to t is across two adjacent syllables. In terms of these definitions, we can derive the two recursive formula,

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

$$D(t,s) = \log\left(G_t(s)\right) + \max\left\{\max_{0\le v<39,\, v\ne s}\left[R(t-1,v)\right],\ D(t-1,s)\right\}, \tag{5}$$

where the boundary values are $D(1, s) = -10^{30}$ and $R(1, s) = \log(G_1(s))$, $s = 0, 1, \ldots, 38$. In Eq. (4), when the voiced part within a syllable is expanded with a further frame, *i.e.* the tth frame, the probability, $R(t, s)$, of the expanded part can be calculated in terms of the probability, $R(t–1, s)$, of the within-syllable part before the expansion.

In Eq. (5), the cross-syllable voiced part may be formed by concatenating the tth frame as a newly spanned frame by a syllable of the sth syllable-final type which is different from the syllable-final type of the preceding syllable that spans the frames from time 1 to $t–1$. In this scenario, we should have the formula,

$$D(t,s) = \log\left(G_t(s)\right) + \max_{0\le v<39,\, v\ne s}\left[R(t-1,v)\right], \tag{5a}$$

derived. As an improvement to the work [12] for practical implementation, here we place a constraint that the time index t must be of a value greater than 5 and less than $T+2–5$ on applying Eq. (5a) where T is the length of a batch of voiced frames. This constrain is intended to prevent short syllable-segments of lengths less than 5 frames from being selected by the GMM selection algorithm. If this constraint is not placed, it may occur that a sequence of frames coming from a syllable is incorrectly divided into too many short segments, which may possibly cause spectral discontinuities at segment boundaries and lower the quality of the converted speech.

In the other scenario, the cross-syllable voiced part of t frames in length may be obtained by spanning the tth frame with the right-side syllable of an already formed cross-syllable voiced part. Hence, the formula,

$$D(t,s) = \log\left(G_t(s)\right) + D(t-1, s), \tag{5b}$$

is derived. When the two scenarios are put together, the formula, Eq. (5), is thus derived. Next, when we advance to the last frame of a batch of T frames, we can decide whether this batch of T frames is spanned by a single syllable or spanned by two syllables according to their likelihoods. In detail, the maximum likelihood, $A(T)$, is calculated as

$$A(T) = \max\left\{\max_{0\le v<39}\left[R(T,v)\right],\ \max_{0\le v<39}\left[D(T,v)\right]\right\}, \tag{6}$$

where the time length T is set to 30 in this study. In terms of Eqs. (4)-(6), we can calculate the maximum likelihood, $A(30)$, and then back track to find the sequence of GMM indices that are best for assigning to the batch of 30 voiced frames.

3.2 Mapping with Single Gaussian PDF

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

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

where x is the input frame's DCC and $F^k(x)$ denotes the mapping function using the kth Gaussian PDF.

Here, we have improved the DP based algorithm for selecting Gaussian PDFs that is developed in the work [12]. By using the DP based algorithm, we can find a sequence of Gaussian PDFs that will minimize the cumulated inter-frame distances between adjacently converted DCC vectors under a likelihood bound defined by the threshold parameter, H, in Eq. (8). Of a minimized cumulated distance, the selected sequence of Gaussian PDFs is believed to generate converted voice most probably without spectral discontinuity. The details of the improved algorithm are as the following. Let the index of the GMM selected by section 3.1 for the tth frame be $I(t)$. Denote the mapping function using the kth Gaussian PDF 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 Gaussian PDF used at time t be k. Then, the derived recursive formula,

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

$$U_m(t,s) = \frac{w_m(s)\times N(x_t;\,\mu_m^{(x)}(s),\,\Psi_m^{(xx)}(s))}{\sum_{i=1}^{M} w_i(s)\times N(x_t;\,\mu_i^{(x)}(s),\,\Psi_i^{(xx)}(s))}$$

is used to execute dynamic programming, where $dist(\bullet, \bullet)$ is a geometric distance measure for DCC, H is a probability threshold who's value is experimentally set to 0.2, and $U_m(t, s)$ denotes the posterior probability of the mth component among the mixture components of the GMM indexed with s given that the DCC vector observed at time t is x_t. The probability term, $U_m(t, s)$, is introduced here (not used in the work [12]) to measure

the percentage that the *m*th mixture component will contribute to the probability that x_t is seen under the *s*th GMM. If no Gaussian PDF is found to have probability greater than *H* in Eq. (8) (this situation seldom occurs), the Gaussian PDF belonging to the *s*th GMM and having the maximal probability is then selected instead to ensure that Eq. (8) can proceed. Notice that we have experimented to measure average cepstral distances for many combinations of *M* and *H* values (*M* varied from 8 to 16 and *H* varied from 0.1 to 0.5). Then, appropriate values for *M* and *H* are taken accordingly. At time 0, $C(0, k)$, $0 \le k < M$, are all set to have the value, 0. Finally, at time *T*, the minimum cumulated distance $B(T)$ is computed as

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

In terms of Eqs. (8) and (9), the minimum cumulated distance can be obtained. Also, the sequence of Gaussian PDF indices for the frames from time 1 to *T* can be obtained through backtracking. Here, *T* is the time length of a sequence of voiced input frames.

3.3 Target Frame Selection

Let $y_1, y_2, \ldots, y_T$ be a sequence of converted DCC vectors obtained from mapping with singe Gaussian PDF, *i.e.* the method presented in Section 3.2. Notice that each vector, y_t, of the sequence may be somehow distorted during the mapping from x_t to y_t. To improve the quality of the converted voice, we are thus motivated, by Dutoit, *et al.* [13], to replace y_t with a real (not converted) DCC vector, z_t, analyzed from a target frame belonging to the segment class indexed as $I(t)$. To select a frame, z_t, from a group of frames corresponding to a segment class, we should consider not only the matching distance, $dist(y_t, z_t)$, but also the connection distance, $dist(z_{t-1}, z_t)$, in order to prevent spectral discontinuity from occurring. Besides the connection distance adopted in the work by Dutoit, *et al.* [13], we add another term of dynamic-spectral distance to reflect a dynamic-spectral change, $\Delta y_t = y_t - y_{t-1}$, between a pair of adjacently converted frames, to its corresponding pair of real target frames, $\Delta z_t = z_t - z_{t-1}$. This dynamic-spectral distance is useful to slightly improve the quality of the converted speech according to our experiments. Consequently, we have developed another DP based algorithm to do frame selection.

As the first step, for each converted DCC vector, y_t, *K* target-frame DCC of the least distances to y_t are found by fully searching the frame group corresponding to the segment class indexed as $I(t)$. Here, *K* is set to 24 according to the results of the experiments measuring VR (variance ratio defined in Eq. (14)) values with *K* varied from 12 to 36. Next, let $Q(t, i)$ denote the best cumulated distance from time 1 to *t* and the index of the target-frame DCC selected at time *t* be *i*, *i.e.* the *i*th frame of the *K* found frames for replacing y_t. Then, the recursive formula,

$$Q(t,i) = \min_{0 \le j < K} \left[Q(t-1, j) + \alpha \cdot dist\left(z_{t-1}^{j}, z_t^{i}\right) + \alpha \cdot dist\left(y_t - y_{t-1}, z_t^{i} - z_{t-1}^{j}\right) \right] + dist\left(y_t, z_t^{i}\right), \tag{10}$$

is used to execute dynamic programming, where α is a weighting factor for both connection and dynamic-spectral distances, and z_t^i denotes the *i*th target-frame DCC candidate

of the K found candidates at time t for replacing y_t. Here, α is set to 1.5 according to the results of the experiments measuring VR values with α varied from 0.25 to 6. As mentioned in the work [13], a trick to obtain more natural spectral connection is to dynamically reset the value of α to 0 if z_{t-1}^j and z_t^i are checked to be adjacent frames coming from a same utterance. This trick is also adopted here, and is extended by accepting the case that z_{t-1}^j and z_t^i come from a same utterance and have just one another frame in between. Finally, at time T, the minimum cumulated distance $W(T)$ is computed as

$$W(T) = \min_{0 \le j < K} \left[Q(T, j) \right]. \tag{11}$$

In terms of Eqs. (10) and (11), the minimum cumulated distance can be obtained. Also, the sequence of target-frame indices from time 1 to T can be backtracked. Then, the real target-frame DCC, z_t^i, corresponding to the indices backtracked are taken to replace the converted-frame DCC, y_t, t = 1, 2, …, T. Here, T is the time length of a sequence of voiced frames.

3.4 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 [21-23]. The frequency that the two parts are divided according to is termed the maximum voiced frequency (MVF). In the original work [21], 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 ith 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 tth sampling point between the ith 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. The detail is,

$$\begin{aligned} f_k(t) &= f_k^i + \frac{f_k^{i+1} - f_k^i}{N} t, \quad k = 1, 2, ..., L, \\ a_k(t) &= a_k^i + \frac{a_k^{i+1} - a_k^i}{N} t, \quad k = 1, 2, ..., L, \end{aligned} \tag{12}$$

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 kth harmonic partial of the ith 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 ith frame. As to a_k^i, its value is derived from the converted vector of DCC. The detail of the derivation is referred to a previous work [18]. 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 tth sampling point is computed as

$$\begin{aligned} h(t) &= \sum_{k=1}^{L} a_k(t) \cdot \cos(\phi_k(t)), \ 0 \le t < N, \\ \phi_k(t) &= \phi_k(t-1) + 2\pi \cdot f_k(t) / 22,050, \end{aligned} \tag{13}$$

where $\phi_k(t)$ denotes the cumulated phase on time t for the kth harmonic partial and 22,050 is the sampling frequency. $\phi_k(-1)$ is defined to be $\phi_k(N-1)$ 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.

For the noise signal, we adopt a synthesis method recommended in the literature of HNM [21]. The method is to synthesize the noise signal also as a summation of sinusoidal components as expressed in Eq. (12). Nevertheless, the sinusoids here are all of frequencies greater than the MVF value, the phase increment of each sinusoidal is fixed and not changed with time, and the gap between two adjacent sinusoids is fixed to 100Hz. For a detailed description of the method, the previous works [21, 22] are referred to.

4. EXPERIMENTAL EVALUATIONS

For evaluating the conversion method proposed here, we have constructed three kinds of voice conversion systems, named SOG, SLG, and SLGF, respectively. In the system SOG (system using original GMM for mapping), a single GMM of 128 mixture components are trained with the 350 training sentences. Then, the mapping function, Eq. (1), is used to convert the DCC of each input frame. In the system SLG (system using selected GMM and selected single Gaussian PDF for mapping), we trained 39 segmental GMMs instead of a single GMM. The number of mixture components for each segmental GMM is set to 8, and the value for the probability threshold H is set to 0.2. Then, the methods presented in Sections 3.1 and 3.2 are used to select segmental GMM and single Gaussian PDF, respectively. As to the system SLGF (adding target frame selection upon the system SLG), the method presented in Section 3.3 is used to select target-frame DCC vectors to replace the converted DCC vectors.

By 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, 15 persons are invited to listen to the voice files and give relative scores. The 15 persons are undergraduate and graduate students. Among the 15 persons, 5 of them are not familiar with the research field of voice conversion.

4.1 Timbre Similarity Tests

For timbre similarity tests, 5 voice files are prepared first, which are named VS (uttered by the source speaker), VT (uttered by the target speaker), VXA (converted by the system SOG), VXB (converted by the system SLG), and VXC (converted by the system SLGF). Among the 5 files, VS and VT are of same content whereas VXA, VXB, VXC are of same content but different from VS and VT. These 5 files can be downloaded from the web page, http://guhy.csie.ntust.edu.tw/VoiceConv/. During listening tests with the method ABX, 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 VXA, VXB, and VXC. The test method, ABX, was adopted by many researchers in studying voice conversion [2, 3, 6, 7, 9, 24]. Each time that three files, ABX, are played, the participant is requested to give a score.

Here, the score range is from 1 to 5. The score 5 (1) means the timbre of X is sure to be that of B (A), the score 4 (2) means the timbre of X is more like that of B (A), and the score 3 means the timbre of X cannot be judged.

After listening tests, the scores given by the 15 persons are collected to compute average scores (AVG) and standard deviations (STD) for the three systems respectively. The results are as those values listed in Table 2. From this table, it can be seen that the average scores for voice conversion between different genders (*i.e.* from MA to FA) are higher than those for voice conversion between same genders (*i.e.* from MA to MB). In addition, when the average scores of the three systems are compared, it can be found that the average scores of the system SLGF are much better than those of the system SLG whereas the average scores of SLG are slightly better than those of SOG. Therefore, mapping with segmental GMM and target-frame selection can indeed help to improve the timbre similarity of the converted voices.

Table 2. Average scores and standard deviations for timbre similarity tests.

		SOG	SLG	SLGF
MA=>MB	AVG (STD)	4.18 (0.74)	4.33 (0.49)	4.67 (0.49)
MA=>FA	AVG (STD)	4.53 (0.52)	4.80 (0.41)	4.93 (0.26)

4.2 Voice Quality Tests

In the tests of voice quality, the three converted voice files, VXA, VXB, and VXC are used. These files are played in the order AX where A is fixed to VXA and X is randomly selected from VXB and VXC. Each time that two files, AX, are played, the participant is requested to give a score. Here, the score range is from 1 to 5. The score 5 (1) means the quality of X is much better (worse) than A, the score 4 (2) means the quality of X is slightly better (worse) than A, and the score 3 means the quality of X cannot be distinguished from that of A.

After listening tests, the scores given by the 15 persons are collected to compute average scores and standard deviations for the two systems, SLG and SLGF, respectively, when compared with the system SOG. The results are as those values listed in Table 3. From Table 3, it can be found that the average scores for voice conversions from MA to MB (*i.e.* same gender) is about 0.3 better than the average scores for voice conversion from MA to FA. 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, SLG and SLGF, are compared, it can be found that the scores of SLGF are both better

Table 3. Average scores and standard deviations for voice quality tests.

		SOG vs SLG	SOG vs SLGF
MA=>MB	AVG (STD)	3.73 (0.59)	4.33 (0.62)
MA=>FA	AVG (STD)	3.53 (0.52)	3.93 (0.59)

than those of SLG. Therefore, the idea of cascading automatic segmental GMM selection with automatic frame selection can indeed help to improve the quality of the converted voice.

4.3 Cepstral Distance and Variance Ratio

There are 25 remaining parallel sentences that are not used in the training stage. Hence, the 25 sentences uttered by the source speaker, MA, 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 4. From this table, it is seen that the system SOG obtains 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 SLGF in voice quality. Therefore, the average cepstral distances measured are inconsistent with the results of the listening tests. Such a situation, *i.e.* inconsistency between cepstral distance and voice quality, is also reported in several works by others [5, 6, 24, 25]. Therefore, another objective measure, variance ratio (VR), is used in [5, 6], which is consistent in general with the converted-voice quality.

Table 4. Average distances for the three systems.

	SOG	SLG	SLGF
MA=>MB	0.5440	0.6312	0.6889
MA=>FA	0.5237	0.5252	0.5951

The formula of variance ratio adopted here is

$$VR = \sum_{v=1}^{V} \frac{N_v}{NT} \cdot \left(\frac{1}{D} \cdot \sum_{d=1}^{D} \frac{(\hat{\sigma}_v^d)^2}{(\sigma_v^d)^2} \right), \tag{14}$$

where V is the number of segment classes (V = 39 here), N_v is the number of voiced test frames belonging to the vth class, NT is the total number of voiced test frames, D is the dimensionality of DCC (D = 40 here), $\hat{\sigma}_v^d$ and σ_v^d are the standard deviations of the dth dimension for the converted and target DCC, respectively. According to Eq. (14), we measure the values of VR for the three systems, SOG, SLG, and SLGF, by using the 25 parallel sentences. As a result, the measured values are listed in Table 5. From Table 5, it

Table 5. The values of VR measured for the three systems.

	SOG	SLG	SLGF
MA=>MB	0.2223	0.2578	0.6248
MA=>FA	0.1783	0.2058	0.5793
Average	0.2003	0.2318	0.6021

can be seen that the average VR value, 0.2318, of SLG is greater than the one, 0.2003, of SOG, and the average VR value, 0.6021, of SLGF is much greater than the other two systems' values. Therefore, the measured VR values are consistent with the perceived qualities of the three systems' converted voices.

On the other hand, to investigate why the inconsistent situation may occur, we take some frames of target DCC vectors and their corresponding converted DCC vectors to draw spectral envelope curves. Then, the spectral envelope curves of each target DCC vector and its two converted DCC vectors by the two systems, SLG and SLGF, are compared to see if something strange can be found. As a result, we find a noticeable phenomenon. The spectral envelopes converted by the system SLG are still over-smoothed frequently. This spectral over-smoothing makes the spectral envelope converted by SLG closer to the spectral envelope of the target DCC and hence results in smaller distances be calculated when compared to the spectral envelope converted by SLGF.

An example of a spectral envelope converted by SLG is taken from a source frame of /li/, and drawn in Fig. 4 as the light solid-lined curve. In contrast, the spectral envelope converted by SLGF is obtained from a selected real target-frame, and drawn in Fig. 4 as the dark solid-lined curve. The other curve in Fig. 4, *i.e.* the dash-lined curve, is the spectral envelope of the target frame aligned (with DTW) to the source frame. Inspecting the three curves in the frequency range from 5,000 to 8,000 Hz, we find that the light solid-lined curve almost goes between the other two curves. This explains why the distance between the converted DCC by SLG and the target DCC will be smaller than the distance between the converted DCC by SLGF and the target DCC. In addition, inspecting the three curves in the frequency range from 2,000 to 4,000 Hz, we find that the third and fourth formants on the dark solid-lined curve are much sharper than the corresponding formants on the light solid-lined curve. This may explain why the quality of the converted voice by SLGF is perceived as better than the converted voice by SLG.

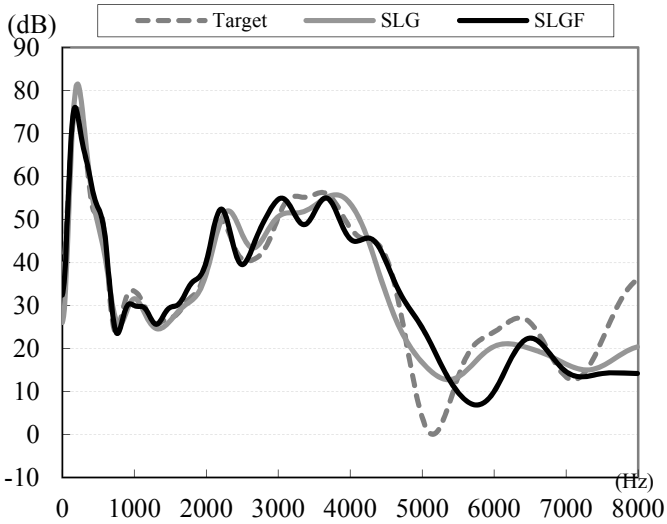

Fig. 4. Spectral envelopes converted by the systems SLG and SLGF for a source frame from /li/.

5. CONCLUSIONS

According to the results of the listening tests, the system SLGF is the best in both timbre similarity and voice quality among the three systems, SOG, SLG, and SLGF. In addition, the measured values of VR also indicate that the converted-voice quality of SLGF is much better than the qualities of the other two systems. Therefore, the approach that combines automatic segmental-GMM selection with automatic target-frame selection can indeed help to promote the performances of the GMM based voice conversion mechanism. As to the cepstral distances measured, the system SOG indeed obtains the smallest average distance. Nevertheless, according to our observations from Fig. 4, the smaller average distance is obtained in terms of over-smoothed converted spectral envelopes. Therefore, the system SOG (also SLG) suffers the over-smoothed converted spectral envelopes, which will degrade the voice quality and timbre similarity. In the future, we may study to reduce the size of a speech segment from syllable final to vowel nucleus and ending nasal. Also, the idea of global variance may be integrated to our approach to further improve the quality of the converted voice.

REFERENCES

1. M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *Proceedings of International Conference on Acoustics*, *Speech*, *and Signal Processing*, 1988, pp. 655-658.
2. A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proceedings of International Conference on Acoustics*, *Speech*, *and Signal Processing*, 1998, pp. 285-288.
3. Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on Speech and Audio Processing*, Vol. 6, 1998, pp.131-142.
4. D. Erro, A. Moreno, and A. Bonafonte, "Voice conversion based on weighted frequency warping," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 18, 2010, pp. 922-931.
5. E. Godoy, O. Rosec, and T. Chonavel, "Voice conversion using dynamic frequency warping with amplitude scaling, for parallel or nonparallel corpora," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 20, 2012, pp. 1313-1323.
6. H. Benisty and D. Malah, "Voice conversion using GMM with enhanced global variance," in *Proceedings of INTERSPEECH*, 2011, pp. 669-672.
7. T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 15, 2007, pp. 2222-2235.
8. C. H. Wu, C. C. Hsia, C. H. Lee, and M. C. Lin, "Hierarchical prosody conversion using regression-based clustering for emotional speech synthesis," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 18, 2010, pp. 1394-1405.
9. Z. Z. Wu, T. Kinnunen, E. S. Chng, and H. Z. Li, "Text-independent F0 transformation with non-parallel data for voice conversion," in *Proceedings of INTERSPEECH*, 2010, pp. 1732-1735.

10. E. Godoy, O. Rosec, and T. Chonavel, "Alleviating the one-to-many mapping problem in voice conversion with context-dependent modeling," in *Proceedings of INTERSPEECH*, 2009, pp. 1627-1630.
11. J. F. Yeh and C. H. Hsu, "Sub-syllable segment-based voice conversion using spectral block clustering transformation functions," *Journal of the Chinese Institute of Engineers*, Vol. 33, 2010, pp. 1059-1067.
12. H. Y. Gu and S. F. Tsai, "An improved voice conversion method using segmental GMMs and automatic GMM selection," in *Proceedings of International Congress on Image and Signal Processing*, 2011, pp. 2395-2399.
13. T. Dutoit, A. Holzapfel, M. Jottrand, A. Moinet, J. Perez, and Y. Stylianou, "Towards a voice conversion system based on frame selection," in *Proceedings of International Conference on Acoustics*, *Speech*, *and Signal Processing*, 2007, pp. 513-516.
14. S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book* (for HTK version 3.2.1), Engineering Department, Cambridge University, U.K., 2002.
15. K. Sjolander and J. Beskow, "WaveSurfer software package," http://www.speech.kth.se/wavesurfer/index.html.
16. H. Kawahara, I. Masuda-Katsuse, and A. de. Cheveign, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: possible role of a repetitive structure in sounds," *Speech Communication*, Vol. 27, 1999, pp. 187-207.
17. Cappé and E. Moulines, "Regularization techniques for discrete cepstrum estimation," *IEEE Signal Processing Letters*, Vol. 3, 1996, pp. 100-102.
18. 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, 2009, pp. 363-382.
19. R. A. Redner and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Review*, Vol. 26, 1984, pp. 195-239.
20. H. Y. Kim, J. S. Lee, M. W. Sung, K. H. Kim, and K. S. Park, "Pitch detection with average magnitude difference function using adaptive threshold algorithm for estimating shimmer and jitter," in *Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 1998, pp. 3162-3165.
21. 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.
22. H. Y. Gu and H. L. Liao, "Mandarin singing-voice synthesis using an HNM based scheme," *Journal of Information Science and Engineering*, Vol. 27, 2011, pp. 303-317.
23. E. Zavarehe, S. Vaseghi, and Q. Yan, "Noisy speech enhancement using harmonic-noise model and codebook-based post-processing," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 15, 2007, pp. 1194-1203.
24. H. T. Hwang, Y. Tsao, H. M. Wang, Y. R. Wang, and S. H. Chen, "Exploring mutual information for GMM-based spectral conversion," in *Proceedings of International Symposium on Chinese Spoken Language Processing*, 2012, pp. 50-54.

25. D. Erro, E. Navas, and I. Hernáez, "Parametric voice conversion based on bilinar frequency warping plus amplitude scaling," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 21, 2013, pp. 556-566.

Hung-Yan Gu (古鴻炎) received the B.S. and M.S. degrees in Computer Engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1983 and 1985, respectively, and the Ph.D. degree in Computer Science and Information Engineering from National Taiwan University, Taipei, Taiwan, in 1990. Currently, he is a Professor in the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan. His research interests include speech signal processing, computer music synthesis, and data compression.

Sung-Feng Tsai (蔡松峰) was born in 1984. He received the B.S. degree in Mathematics from Fu Jen Catholic University, New Taipei, Taiwan, in 2006, and the M.S. degree in Computer Science and Information Engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, in 2009.