Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Speech and Language 20 (2006) 441–467
COMPUTER SPEECH AND LANGUAGE
www.elsevier.com/locate/csl

Robust processing techniques for voice conversion ☆

Oytun Turk [a,b,*], Levent M. Arslan [a,b]

[a] *Electrical and Electronics Engineering Department, Bogazici University, Bebek, Istanbul, Turkey*
[b] *R&D Department, Sestek Inc., ITU Ayazaga Kampusu, ARI-1 Teknopark Binasi, 34469 Maslak, Istanbul, Turkey*

Received 23 November 2004; received in revised form 29 April 2005; accepted 3 June 2005
Available online 12 July 2005

Abstract

Differences in speaker characteristics, recording conditions, and signal processing algorithms affect output quality in voice conversion systems. This study focuses on formulating robust techniques for a codebook mapping based voice conversion algorithm. Three different methods are used to improve voice conversion performance: confidence measures, pre-emphasis, and spectral equalization. Analysis is performed for each method and the implementation details are discussed. The first method employs confidence measures in the training stage to eliminate problematic pairs of source and target speech units that might result from possible misalignments, speaking style differences or pronunciation variations. Four confidence measures are developed based on the spectral distance, fundamental frequency (f_0) distance, energy distance, and duration distance between the source and target speech units. The second method focuses on the importance of pre-emphasis in line-spectral frequency (LSF) based vocal tract modeling and transformation. The last method, spectral equalization, is aimed at reducing the differences in the source and target long-term spectra when the source and target recording conditions are significantly different. The voice conversion algorithm that employs the proposed techniques is compared with the baseline voice conversion algorithm with objective tests as well as three subjective listening tests. First, similarity to the target voice is evaluated in a subjective listening test and it is shown that the proposed algorithm improves similarity to the target voice by 23.0%. An ABX test is performed and the proposed algorithm is preferred over the baseline algorithm by 76.4%. In the third test, the two algorithms are compared in terms of the subjective

☆ Sample voice conversion outputs are available at: http://www.sestek.com.tr/voice_conversion/oytun/rvc_demo.html.

* Corresponding author.Tel.: +90 212 286 25 44x124; fax: +90 212 286 25 47.
E-mail addresses: oytun@sestek.com.tr (O. Turk), arslanle@boun.edu.tr (L.M. Arslan).

0885-2308/$ - see front matter
doi:10.1016/j.csl.2005.06.001

quality of the voice conversion output. The proposed algorithm improves the subjective output quality by 46.8% in terms of mean opinion score (MOS).

1. Introduction

The aim of voice conversion is to modify a source speaker's voice in order to obtain an output that sounds like a specific target speaker's voice. It has been a popular topic in speech processing research for the last two decades (Abe et al., 1988; Arslan and Talkin, 1997; Arslan, 1999; Moulines and Sagisaka, 1995; Stylianou et al., 1998). Existing algorithms employ two common stages: training and transformation. The voice conversion system gathers information from the source and target speaker voices and automatically formulates voice conversion rules at the training stage. The transformation stage employs the conversion rules to modify the source voice in order to match the characteristics of the target voice.

The training stage involves three steps in general: acoustic modeling, alignment, and acoustic mapping. In acoustic modeling stage, speaker-specific parameters are extracted from the speech waveform. These parameters describe the short-term and long-term characteristics of the source and target voices. Vocal tract, glottal source (pitch, spectral tilt, open/closed quotient), duration, and energy characteristics convey important speaker-specific information (Furui, 1986; Itoh and Saito, 1982; Kuwabara and Sagisaka, 1995; Matsumoto et al., 1973; Necioglu et al., 1998). Linear prediction coefficients (LPCs) (Makhoul, 1975), line spectral frequencies (LSFs) (Itakura, 1975a), Mel-frequency cepstral coefficients (MFCCs) (Davis and Mermelstein, 1980), formant frequencies and bandwidths (Holmes et al., 1990), and sinusoidal transform coding (STC) parameters (McAulay and Quatieri, 1995) can be used for modeling the vocal tract characteristics. There has been considerable amount of work on the analysis, modeling and modification of glottal source characteristics in voice quality research (Childers and Lee, 1991; Childers, 1995; Fant et al., 1985). Pitch is one of the most important speaker-specific dimensions among the glottal source characteristics. It can be estimated using the autocorrelation function, average magnitude difference function, Fourier Transform, and harmonic analysis (Rabiner and Schafer, 1978). Dynamic programming is a popular method employed to avoid discontinuities and hence improve the robustness of the pitch detection algorithm (Talkin, 1995).

The second step, alignment, is necessary to determine corresponding units in the source and target voices. This is due to the fact that the durations of sound units (i.e., phonemes or subphonemes) can be quite different among speakers. It is preferable to employ automatic alignment techniques like dynamic time warping (DTW) (Itakura, 1975b), and hidden Markov models (HMMs) (Rabiner, 1989) because manual alignment is time consuming.

The final training step is the estimation of the acoustic mapping function between the source and the target speaker's acoustic spaces using machine learning techniques like vector clustering/quantization (Abe et al., 1988), codebook mapping (Acero, 1993), weighted codebook mapping (Arslan and Talkin, 1997; Arslan, 1999), GMMs (Stylianou et al., 1998), Radial Basis Function Networks (RBFNs) (Drioli, 1999), Artificial Neural Networks (ANNs) (Narendranath et al., 1995), and Self Organizing Maps (SOMs) (Knohl and Rinscheid, 1993). The main distinction between the earlier methods (Abe et al., 1988 and Acero, 1993) and more recent methods

(Arslan and Talkin, 1997; Arslan, 1999; Stylianou et al., 1998) are that smoothing among the mapping units is performed to reduce distortion at frame boundaries. Another distinction of more recent methods is the employment of text and language independent automatic techniques for alignment such as Sentence-HMM and DTW.

The transformation stage employs acoustic analysis techniques similar to the acoustic modeling step in training. Once the parameters of the input waveform are determined, voice conversion rules are employed to obtain the corresponding target parameters. Necessary modifications are performed on the input waveform to match the target speaker characteristics. The modifications include transformation of the vocal tract, glottal source, duration, and energy characteristics. The vocal tract characteristics can be transformed using formant modification (Mizuno and Abe, 1995), interpolation of the line spectral frequencies (Arslan, 1999), and sinusoidal modeling techniques (Laroche et al., 1993). There exists several methods for pitch modification: time-domain pitch synchronous overlap-add algorithm (TD-PSOLA) (Moulines and Charpentier, 1990), frequency-domain pitch synchronous overlap-add algorithm (FD-PSOLA) (Moulines and Verhelst, 1995), sinusoidal synthesis (Quatieri and McAulay, 1992), and phase vocoding (Flanagan and Golden, 1966).

Text-to-speech synthesis (TTS) quality has increased by the employment of large databases and unit-selection techniques (Hunt and Black, 1996; Dutoit, 1997). As voice conversion requires less training data (5–10 min of voice recordings), it is advantageous to employ voice conversion for creating new TTS voices out of the existing ones. Therefore, TTS has been considered as the primary application field for voice conversion in the literature (Kain and Macon, 1998; Zhang et al., 2001). With the development of high-quality voice conversion systems, many other applications can be implemented some of which were demonstrated in our previous work. We have reported a demonstration for dubbing movies by employing only several dubbers, generating the voice of famous actresses/actors in a foreign language which they cannot speak, and generating the voices of actresses/actors who are not alive (Turk and Arslan, 2002, 2003). Other dubbing applications might be to regenerate the voices of actresses/actors who have lost their voice characteristics due to old age and to perform dubbing for radio broadcasts.

The factors that influence the performance of a voice conversion algorithm and the problems related to those factors are as follows:

(i) The accuracy of acoustic parameter extraction algorithms: Inaccurate acoustic parameter extraction results in reduction in output quality and similarity to target voice. As an example, errors in the pitch detection algorithm result in distortion in the PSOLA output.
(ii) The amount of available training data from the source and target speakers: Requirement of a large speech database for voice conversion limits the practical applicability of the voice conversion algorithm. However, it might be difficult to estimate model parameters reliably in the case of a restricted database.
(iii) Differences in the accent, prosody, gender, and voice quality of speakers: When there is significant spectral mismatch between source and target speakers automatic alignment algorithm may fail and result in less accurate mapping. When the pitch ranges are significantly different between source and target speakers, more signal processing distortion results due to inherent problems with PSOLA. If the accents are different, we encounter more

one-to-many mappings in the transformations. For example, the source speaker's /a/ in one context might map to target speaker's /a/, while in another context it might map to target speaker's /ae/.

(iv) Differences in the recording equipment and environment: The acoustic mapping performance may degrade when the source and the target recordings are collected in significantly different conditions. For example, if the target recordings contain significant background noise, the transformation will attempt to produce this noise that will result in lower quality transformed speech despite the fact that the source may have been recorded under ideal conditions.

(v) Acoustic alignment performance: The acoustic mapping algorithm may fail and incorrect source and target acoustic features can be matched in the case of poor alignment.

(vi) Processing distortion introduced by the transformation algorithm: Processing distortion reduces subjective quality and naturalness of voice conversion output.

(vii) Evaluation methods and criteria employed: Objective evaluations do not reflect practical performance, i.e., how humans would perceive the voice conversion output. There is currently no standard database for the performance evaluation of voice conversion methods. It is difficult to design and perform subjective tests in multilingual voice conversion applications.

Due to the problems mentioned above, most systems developed for voice conversion have not been put into wide practical use. This study addresses the factors (i)–(v), proposes solutions for the problems related to these factors, and reports the evaluation results. We focus on confidence measures in obtaining the source-to-target mapping, and pre-emphasis to improve acoustic parameter extraction and acoustic mapping accuracy for a codebook mapping based voice conversion algorithm. Reliable estimation of source and target codebook parameters by employing confidence measures and pre-emphasis helps to improve robustness in different source–target speaker combinations. Spectral equalization reduces the differences in the long-term average spectrum of the source and target voices due to differences in the recording conditions.

Section 2 gives a brief description of the baseline voice conversion method entitled speaker transformation algorithm using segmental codebooks (STASC) (Arslan, 1999). Section 3 describes three methods for improving robustness in voice conversion: confidence measures in constructing the codebooks, spectral equalization, and pre-emphasis. The results of subjective tests for the new methods are described in Section 4. The study is concluded with a discussion in Section 5.

2. Baseline voice conversion method

STASC is a two-stage codebook mapping method for voice conversion (Arslan, 1999). In the training stage, STASC determines the corresponding acoustic parameters of the source and target speakers automatically and collects them in codebooks. In the transformation stage, the source speaker acoustic parameters are matched with the source speaker codebook on a frame-by-frame basis and the corresponding target parameters are determined. The transformed utterance is thus obtained by applying a time-varying filter on the source speaker utterance to match the target speaker's acoustic characteristics. Sections 2.1 and 2.2 describe the training and transformation stages briefly.

2.1. Training

STASC uses the recordings of the same phrases from source and target speakers in the training stage. A left-to-right HMM with no skip is trained for each source speaker utterance and both the source and the target speaker utterances are force-aligned with this HMM. The number of states for each utterance is directly proportional to the duration of the utterance. For every 40 ms a new state is added to the HMM topology. With this model, neither the text nor the language of the utterance needs to be known. This automatic alignment procedure is named as the Sentence-HMM method. Fig. 1 shows the flowchart of the STASC training algorithm. In the acoustic feature extraction step, MFCCs are calculated for the source and target speaker utterances. Seven cepstral coefficients derived from a mel-frequency filterbank of 14 bands, log energy, and probability of voicing are combined to form the acoustic feature vector for each frame when the sampling rate is 16 kHz. Delta coefficients are also appended to the feature vector to model temporal variations in the speech signal. Therefore, the final acoustic feature vector has 18 dimensions. An HMM is initialized using the segmental K-means algorithm and trained using the Baum-Welch algorithm for each source speaker utterance using the acoustic feature vectors obtained. Next, source and target speaker utterances are force-aligned with the corresponding source HMM using the Viterbi algorithm. One may also consider using speaker independent models, such that the sentence HMM is trained from both the source and target utterances at the same time. However, this method resulted in less accurate alignments in our trials, therefore we decided to use speaker dependent models. After Sentence-HMM based alignment, LSF vectors, fundamental frequency values, durations and energy values are calculated in the corresponding source and target HMM states. The state arithmetic means of those acoustic features are computed and stored in source and target speaker codebooks.

2.2. Transformation

Fig. 2 shows the flowchart for the transformation algorithm. The vocal tract and excitation spectra are modified separately. First, linear prediction (LP) analysis for the input frame is performed pitch-synchronously. Next, LP parameters are converted to LSFs. The distance between

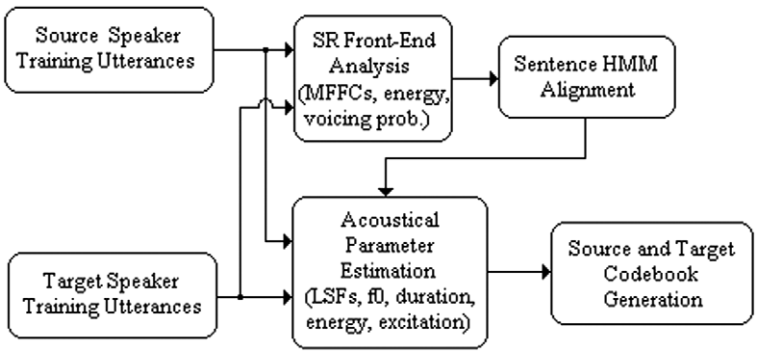

Fig. 1. Flowchart for the STASC training algorithm.

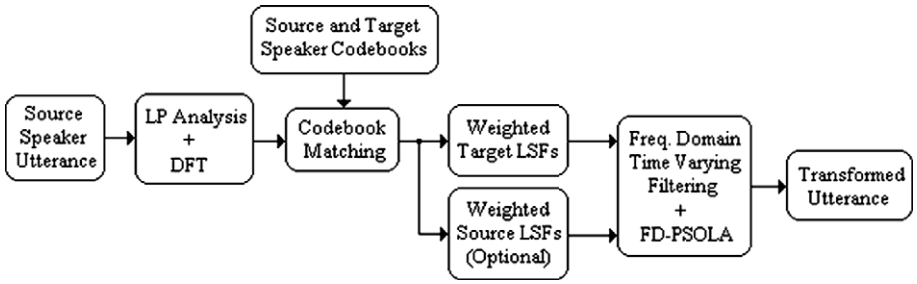

Fig. 2. Flowchart for the STASC transformation algorithm.

the source input LSF vector and each LSF vector in the source codebook is computed using the following equations:

$$d_m = \sum_{n=1}^{P} k_n |u_n - C^{\mathrm{s}}_{mn}| \quad \text{for } m = 1, \ldots, M, \tag{1}$$

$$k_n = \frac{1}{\operatorname{argmin}(|u_n - u_{n-1}|, |u_n - u_{n+1}|)} \quad \text{for } n = 1, \ldots, P, \tag{2}$$

where m is the codebook entry index, M is the codebook size, n is the index of LSF vector entries, P is the dimension of LSF vectors (order of LP analysis), u_n is the nth entry of the LSF vector for the input source frame, C^{s}_{mn} is the nth entry of the mth source codebook LSF vector, d_m is the weighted distance between the input source frame LSF vector u and the mth source codebook LSF vector. LSF weights, k_n, are estimated using Eq. (2). LSFs with closer values are assigned higher weights since closely spaced LSFs are more likely to correspond to formant frequency locations (Crosmer, 1985). Normalized codebook weights, v_m, are obtained by Eq. (3) where using $\gamma = 1.0$ works well in practice

$$v_m = \frac{\mathrm{e}^{-\gamma d_m}}{\sum_{l=1}^{M} \mathrm{e}^{-\gamma d_l}}. \tag{3}$$

Target speaker's vocal tract spectrum is estimated using Eqs. (4) and (5) where $\hat{y}_n$ is the nth entry of the estimated target LSF vector. In our notation here and onwards, circumflex represents that the feature is obtained by weighted averaging of codebook entries. In Eq. (4), C^{t}_{mn} is the nth entry of the mth target codebook LSF vector. The estimated target LSF vector $\hat{y}$ is converted to target LP coefficients, $\hat{a}^{\mathrm{t}}_n$'s. Target vocal tract spectrum, $H^{\mathrm{t}}(w)$, is obtained using Eq. (5) where w is the angular frequency in radians and $\hat{a}^{\mathrm{t}}_n$ is the nth entry of the target LP coefficients vector $\hat{a}^{\mathrm{t}}$,

$$\hat{y}_n = \sum_{m=1}^{M} v^m C^{\mathrm{t}}_{mn} \quad \text{for } n = 1, \ldots, P, \tag{4}$$

$$H^{\mathrm{t}}(w) = \left| \frac{1}{1 - \sum_{n=1}^{P} \hat{a}^{\mathrm{t}}_n \mathrm{e}^{-\mathrm{j}nw}} \right|. \tag{5}$$

$H^{VT}(w)$, the frequency response of the time varying vocal tract filter for the current frame, is given by the following equation:

$$H^{VT}(w) = \frac{H^{t}(w)}{H^{s}(w)} \quad \text{or} \quad H^{VT}(w) = \frac{H^{t}(w)}{\hat{H}^{s}(w)}. \tag{6}$$

Note that the source vocal tract spectrum can be obtained in two different ways to give two different versions of the time varying vocal tract filter:

(i) using the original LP coefficients, a_k^s, of the input speech frame as in Eq. (7), or
(ii) using the LP coefficients $\hat{a}_n^s$ that are obtained from $\hat{u}_n$'s estimated by weighted averaging of the source codebook LSF vectors as in Eq. (9).

In the latter case, the estimate of the source input frame LSF vector, $\hat{u}$, is obtained as a weighted average of the source codebook LSF vectors using Eq. (8). LSF entries, $\hat{u}_n$, are converted to LP coefficients, $\hat{a}_n^s$'s and the source speaker vocal tract spectrum, $\hat{H}^s(w)$, is estimated using Eq. (9). In our simulations, we observed that using Eq. (9) resulted in more natural and higher quality transformation output. This was mainly because the same type of averaging in both the numerator and denominator of the filter transfer function resulted in a smoother and balanced filter function across frames. However, there has been slight similarity degradation from the target speaker since $\hat{H}^s(w)$, in this case was not able to filter out all the effects of the source vocal tract.

$$H^{s}(w) = \left| \frac{1}{1 - \sum_{n=1}^{P} a_n^s e^{-jnw}} \right|, \tag{7}$$

$$\hat{u}_n = \sum_{m=1}^{M} v_m C_{mn}^s \quad \text{for } n = 1, \ldots, P, \tag{8}$$

$$\hat{H}^{s}(w) = \left| \frac{1}{1 - \sum_{n=1}^{P} \hat{a}_n^s e^{-jnw}} \right|. \tag{9}$$

Prosodic modifications are performed on the excitation signal to match the target characteristics using FD-PSOLA algorithm. FD-PSOLA algorithm operates on a pitch-synchronous manner and first removes the vocal tract estimate from the spectrum and then applies necessary pitch modifications on the magnitude of the excitation spectrum either by compression or expansion in the frequency domain. Finally, it overlays the original spectrum on top of the modified excitation magnitude spectrum and the original phase spectrum (Moulines and Verhelst, 1995).

3. Robust processing techniques for voice conversion

The standard STASC algorithm described in the previous section suffers from problems in the modeling and alignment stages. In this section, we describe several techniques to improve the performance of the voice conversion algorithm by focusing on the factors discussed in Section 1. The first technique employs confidence measures to eliminate the source and target acoustic feature pairs that were matched by the training algorithm but poses significant differences due to speaker

accent, prosody, gender, or voice quality. Therefore, it helps to reduce alignment mismatches and improve training performance. The alignment and acoustic mapping performance may also degrade when the source and target recording conditions are different. The second technique, spectral equalization, reduces these differences. The last technique, pre-emphasis, improves the accuracy of the estimation of vocal tract parameters. In general, all techniques discussed in this section help to reduce misalignments and facilitate the extraction of sufficient information for voice conversion from a reasonable amount of source and target training data.

3.1. Confidence measures

Although the overall automatic alignment performance might be satisfactory, some of the source and target HMM states that were matched in the training stage might be significantly different in terms of their acoustic features. Consider the case when the accents of the source and target speakers are different but the alignment was performed accurately, i.e., using manual alignment. In this case, some of the source and target states that were matched would sound quite different. Fig. 3 shows an example. The target is a native American-English speaker whereas the source is a speaker with Russian accent. We have recorded utterances in English, performed manual alignment and training to convert the Russian speaker's voice to that of the American speaker's voice. The left-most spectrum in Fig. 3 corresponds to the phoneme /aa/ in the word "w/a/sh" as pronounced by the source and target speakers. The Russian (source) speaker pronounced it as /ao/. Therefore, we have an incorrect match between the source and the target states due to differences in the accent of the speakers. Now consider the example shown in the middle of Fig. 3. In this case, the identical /ao/ phonemes are matched. When the two matched source–target pairs /ao/–/aa/ and /ao/–/ao/ are accepted as the codebook entries, the phoneme /ao/ in any source utterance to be transformed will have a one-to-many mapping in the target acoustic space. The corresponding target vocal tract features will be averaged as in Eq. (4) and the transformed phoneme will be a "mutant" phoneme between /ao/ and /aa/ as shown in Fig. 3 on the right.

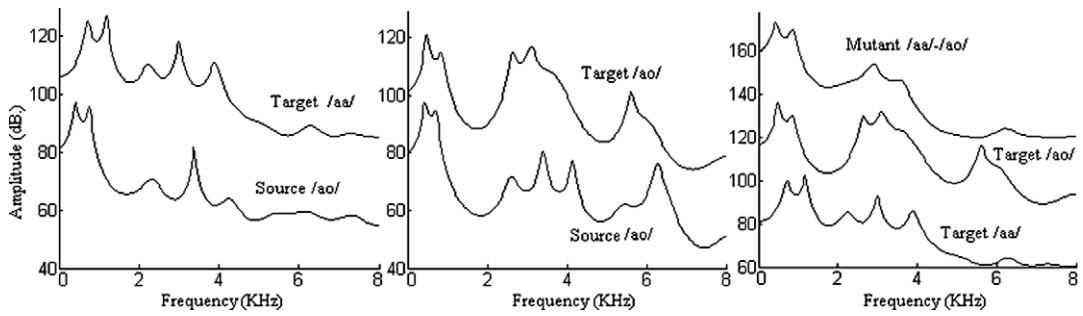

Fig. 3. An example for the results of one-to-many mappings in the codebooks. The utterance is "She had your dark suit in greasy wash water all year". Incorrectly matched phonemes /ao/ and /aa/ due to differences in the pronunciation of phoneme /aa/ in "w/a/sh" (left), correctly matched /ao/ phonemes in "/a/ll" (middle), and mutant phoneme /aa/–/ao/ in "w/a/ter" after transformation because of the one-to-many mapping in the source codebook for the phoneme /ao/ (right). Note that the spectra are shifted along the amplitude axis in appropriate amounts in order to enhance visibility.

The one-to-many mapping problem can be solved in two different ways:

(i) Determining automatically the most appropriate target state to be used when similar source states are matched with significantly different target states.
(ii) Eliminating one-to-many mappings from the codebooks by rejecting the outliers in the distributions of a set of acoustic distance measure differences between the source and the corresponding target states.

In this study, we focus on relatively small training databases (on the order of 20–100 utterances). Although the first solution is ideal to solve the one-to-many mappings problem, it requires an exact description of the source and target accents that can only be made possible by the collection and analysis of large databases. To develop a practical approach for limited databases, we focus on the second solution to eliminate the source and target states that were matched in the alignment but were significantly different. We consider four confidence measures based on spectral distances, f_0 distances, energy distances, and duration differences. The source and target HMM state pairs that are significantly different from each other are eliminated from the codebooks. Although the one-to-many mapping problem still persists, by eliminating significant mismatches from the codebooks we were able to reduce the distortion at the voice conversion output due to this problem.

Another problem arises when significantly different processing is applied to neighboring frames. For example, a source speech frame to be transformed might be matched with codebook entries containing source and target acoustic parameters that are very close to each other. In such a case, the amount of modification applied by the transformation algorithm will be relatively small. If the neighboring source speech frame is matched with significantly different codebook entries, a discontinuity in the output will be produced. Performing less modification in the transformation stage can reduce the transformation distortion. However, the similarity of the conversion output to the target voice will be diminished. In order to balance the trade-off between the similarity to the target voice and the output speech quality, we have also eliminated the source and target pairs that are acoustically similar from the codebooks. The four confidence measures mentioned above are also employed for this purpose. In the following sub-sections, we describe the elimination procedures in detail.

3.1.1. Spectral distance confidence measure

The distance between a source and a target LSF vector, ΔL, is calculated using Eq. (10) where s denotes the source LSF vector and t denotes the target LSF vector. The mean and standard deviation of ΔL values for all pairs of source and target HMM states are estimated. The HMM states which satisfy one of the inequalities in (11) are eliminated from the codebooks where the mean and the standard deviation of ΔL values are denoted as $\mu_{\Delta L}$ and $\sigma_{\Delta L}$, respectively. The first inequality ensures that the source and target states that are very close to each other in terms of spectral distance are eliminated. Similarly, the second inequality ensures the elimination of source and target states that are significantly different in terms of spectral characteristics. Note that we have used $r = 3$ in all the tests. As r is increased, fewer number of states are eliminated from the codebook

$$\Delta L = \sum_{n=1}^{P} \frac{|s_n - t_n|}{\operatorname{argmin}(|s_n - s_{n-1}|, |s_n - s_{n+1}|)}, \tag{10}$$

$$\Delta L < \mu_{\Delta L} - 0.5r\sigma_{\Delta L}, \tag{11a}$$

$$\Delta L > \mu_{\Delta L} + 0.5r\sigma_{\Delta L}. \tag{11b}$$

Fig. 4 shows the histogram of ΔL values for two different source–target speaker pairs. Fig. 5 shows examples of source and target states eliminated from the codebooks due to the spectral distance confidence measure.

3.1.2. f_0 Distance confidence measure

Main motivation behind the application of f_0 confidence measures is to eliminate misalignments based on voiced/unvoiced decision consistency between source and target utterances. The average f_0 value is computed for each source and target HMM state. The state f_0 difference,

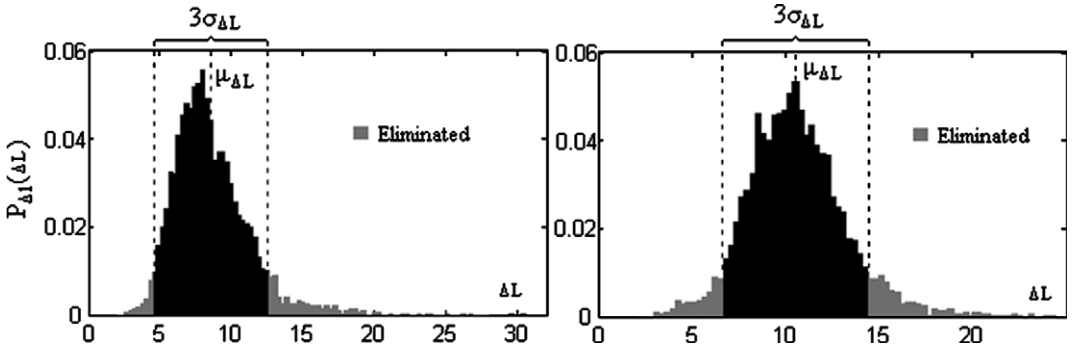

Fig. 4. The histogram of ΔL values for a male–male (left) and male–female (right) source–target speaker pair.

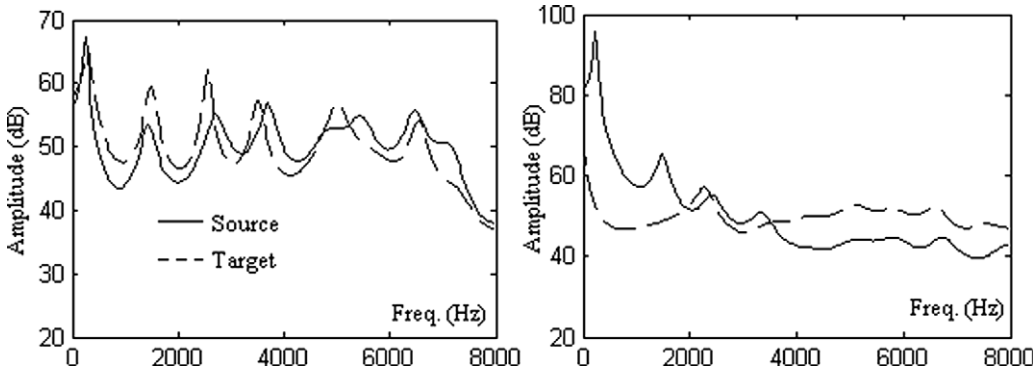

Fig. 5. Spectra for the pairs of source and target states that are eliminated from the codebooks due to the spectral distance confidence measure. Spectra for a source and target state pair that are close in terms of spectral characteristics, i.e., that satisfy 11a (left), and that are significantly different in terms of spectral characteristics, i.e., that satisfy 11b (right).

Δf, is calculated as the absolute difference between the average source state f_0 value, f_0^s, and the average target state f_0 value, f_0^t, as in Eq. (12). The mean of all state f_0 differences, $\mu_{\Delta f}$, and their standard deviation, $\sigma_{\Delta f}$, are computed. The HMM states which satisfy one of the inequalities in (13) are eliminated from the codebooks

$$\Delta f = |f_0^s - f_0^t|, \tag{12}$$

$$\Delta f < \mu_{\Delta L} - 0.5r\sigma_{\Delta f}, \tag{13a}$$

$$\Delta f > \mu_{\Delta L} + 0.5r\sigma_{\Delta f}. \tag{13b}$$

Fig. 6 shows the histogram of Δf values for two different source–target speaker pairs. Fig. 7 shows an example for the source and target states eliminated from the codebooks due to the f_0 distance confidence measure. Most of the states eliminated had voiced frames in target speaker's utterance in place of unvoiced frames in source speaker's utterance or vice versa.

3.1.3. Energy distance confidence measure

The main motivation behind the application of energy based confidence measures is that alignment problems can be identified when there is a high energy mismatch between the corresponding states of the source and the target. For example, we expect to observe high energy values for the same vowel in the same context between the two speakers. Although there might be differences in terms of accent and speaking style, our aim is to eliminate only a very small portion of the states where there is significant energy mismatch.

The energy distance between a pair of source and target HMM states are determined by first computing the average energy within the states. The state energy distance, ΔE, is determined as the absolute difference between the source state average rms energy, E^s, and the target state average rms energy, E^t, using the following equation:

$$\Delta E = |E^s - E^t|. \tag{14}$$

The mean of all state energy differences, $\mu_{\Delta E}$, and their standard deviation, $\sigma_{\Delta E}$, are estimated. If ΔE satisfies one of the inequalities in (15), the corresponding source and target HMM states are eliminated from the codebooks. Fig. 8 shows the histogram of ΔE values for two different

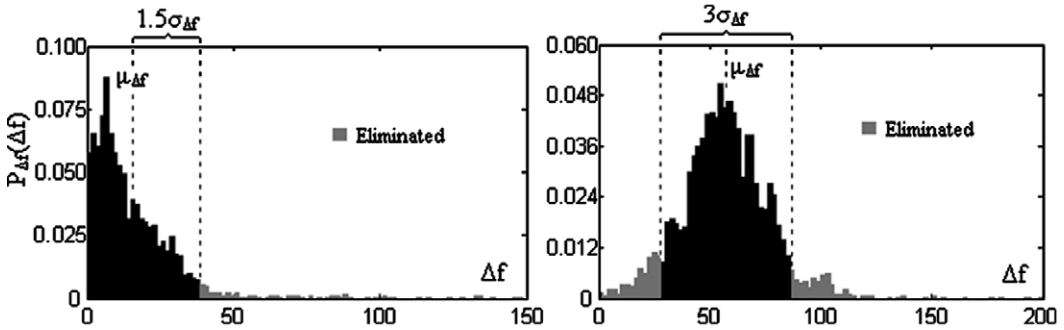

Fig. 6. The histogram of Δf values for a male–male (left) and male–female (right) source–target speaker pair.

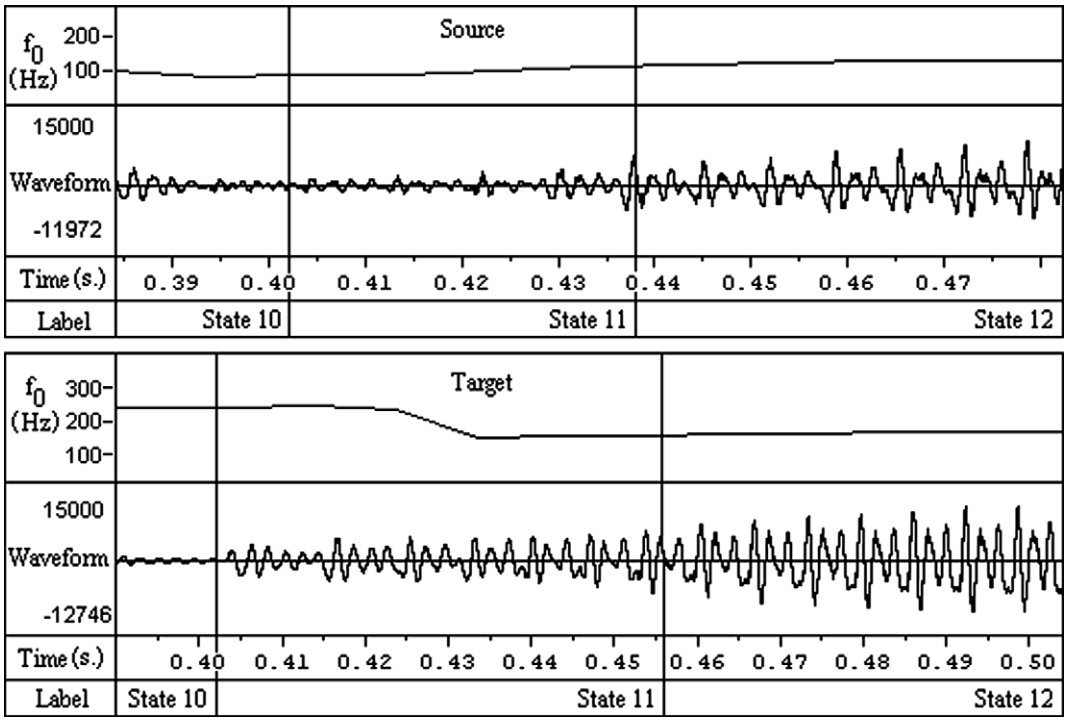

Fig. 7. Waveforms and f_0 contours for a pair of source and target states that are eliminated from the codebooks due to the f_0 distance confidence measure. The eliminated state is labeled as "State 11" and it satisfies 13b, i.e., the f_0 characteristics for the source and the target state are significantly different.

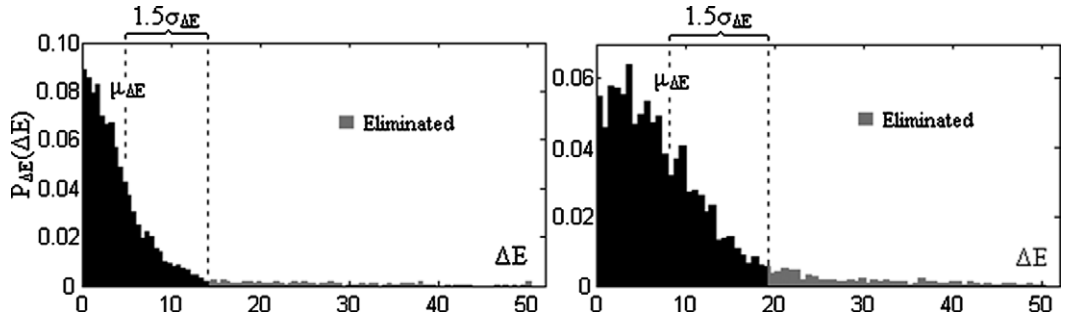

Fig. 8. The histogram of ΔE values for a male–male (left) and male–female (right) source–target speaker pair.

source–target speaker pairs. Fig. 9 shows an example of source and target states eliminated from the codebooks due to the energy distance confidence measure

$$\Delta E < \mu_{\Delta E} - 0.5r\sigma_{\Delta E}, \tag{15a}$$

$$\Delta E > \mu_{\Delta E} + 0.5r\sigma_{\Delta E}. \tag{15b}$$

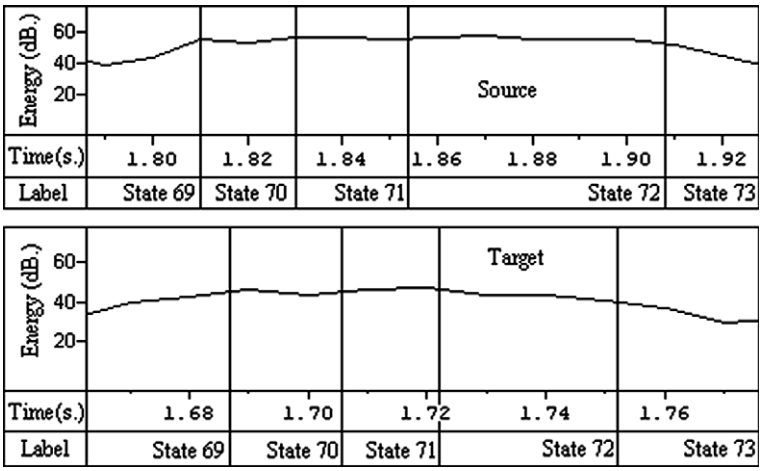

Fig. 9. Energy contours for a pair of source and target states that are eliminated from the codebooks due to the energy distance confidence measure. The pair satisfies Eq. (15b).

3.1.4. Duration difference confidence measure

Large duration difference between a matched source and target HMM state pair is likely to indicate an alignment mismatch. The mismatch might be due to differences in the pronunciations of the speakers, recording conditions, or inaccurate state labeling by the HMMs. The duration difference confidence measure performs two checks on the matched pair of source and target HMM states to eliminate source and target state pairs with large duration differences. First, if the duration of either a source or target HMM state is less than 10 ms or greater than 180 ms, the corresponding states are eliminated from the codebooks. Then, the absolute duration differences between all remaining pairs of source and target HMM states, ΔD's, are determined using:

$$\Delta D = |D^{s} - D^{t}|. \tag{16}$$

The mean of all duration differences, $\mu_{\Delta D}$, and their standard deviation, $\sigma_{\Delta D}$, are computed. The HMM states which satisfy one of the inequalities in (17) are removed from the codebooks. Fig. 10 shows the histogram of ΔD values for two different source–target speaker pairs. Fig. 11 shows an example of source and target states eliminated from the codebooks due to the duration difference confidence measure.

$$\Delta D < \mu_{\Delta D} - 0.5r\sigma_{\Delta D}, \tag{17a}$$

$$\Delta D > \mu_{\Delta D} + 0.5r\sigma_{\Delta D}. \tag{17b}$$

3.1.5. Objective evaluation of confidence measures

In order to investigate the contribution of confidence measures in voice conversion performance we used a speaker pair where both the gender and language origin of the source and target speakers were different. The target speaker was a female native American-English speaker whereas the source speaker was a male Turkish speaker. We have recorded 60 TIMIT utterances from both speakers at 16 kHz. We used 50 utterances for the training and we set aside the remaining 10 utterances for the tests. In our experiments, we performed training and transformation sessions using

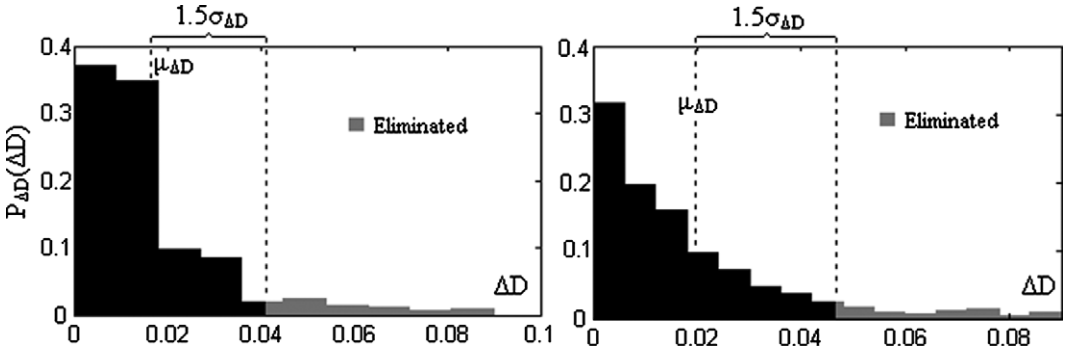

Fig. 10. The histogram of ΔD values for a male–male (left) and male–female (right) source–target speaker pair.

all combinations of confidence measures including the case where no confidence measures were applied. As we have four confidence measures, there are 16 possible combinations for training. Fig. 12 shows the percentage of codebook entries eliminated due to different confidence measures for all combinations. Original codebook size was 8671.

We then manually labeled the test material and computed a frame-by-frame discontinuity measure for the voice conversion outputs. We have used the rms log spectral distortion between the ith speech frame and the jth speech frame to compute the discontinuity measure:

$$E_{\mathrm{dB}}(H_i(w), H_j(w)) = \sqrt{\frac{1}{N}\sum_{w=1}^{N}\left[10\log_{10}\frac{H_i(w)}{H_j(w)}\right]^2}, \tag{18}$$

where N is the DFT size, $H_i(w)$ is the magnitude spectrum of the ith speech frame and $H_j(w)$ is the magnitude spectrum of the jth speech frame. Note that a window size of 20 ms with a skip size of 10 ms is used in the analysis. The rms log spectral distortion values are computed between the ith frame and its six closest neighbors (three predecessors and three successors) by:

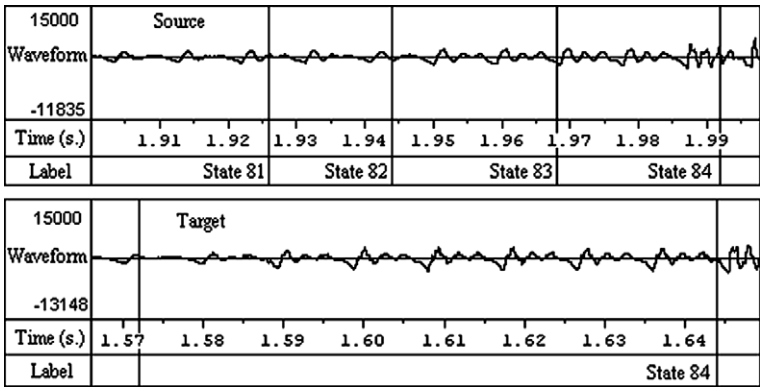

Fig. 11. Waveforms for a pair of source and target states that are removed from the codebooks due to the duration difference confidence measure. The eliminated state is labeled as "State 84" and it satisfies 17b, i.e., the length of the source and target states are significantly different.

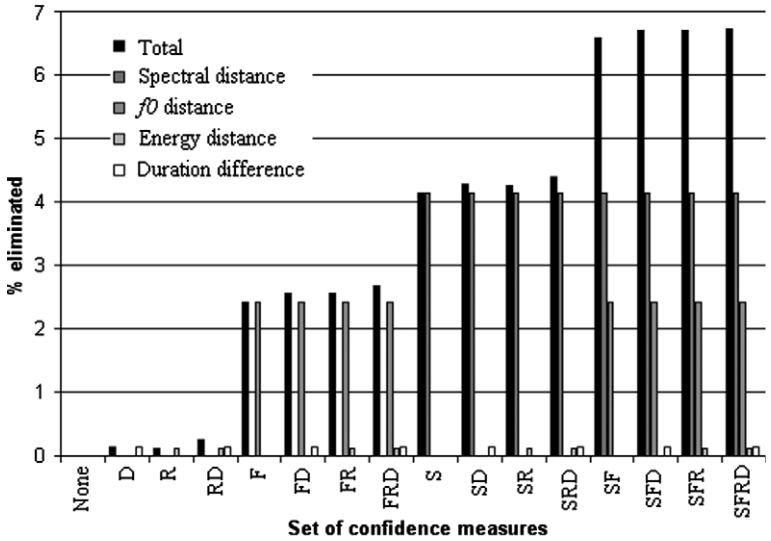

Fig. 12. Percentage of entries eliminated from the codebook using different confidence measures. None: No confidence measure applied in the training. S: Spectral distance confidence measure, F: *f*0 distance confidence measure, R: Energy distance confidence measure, D: Duration difference confidence measure. Each group of bars corresponds to a different combination of confidence measures. Each bar corresponds to the percentage of entries eliminated from the codebooks due to an individual confidence measure.

$$\delta(j) = \begin{cases} E_{\mathrm{dB}}(H_i(w), H_{i+j-3}(w)), & 0 \leqslant j < 3, \\ E_{\mathrm{dB}}(H_i(w), H_{i+j-2}(w)), & 3 \leqslant j < 6. \end{cases} \tag{19}$$

The discontinuity measure, $\alpha(i)$, for speech frame i, is computed as a weighted average of the six rms log spectral distortion values:

$$\alpha(i) = \sum_{j=0}^{5} \beta(j)\delta(j), \quad 0 \leqslant i < T, \tag{20}$$

where $\beta(0) = \beta(5) = 0.0288$, $\beta(1) = \beta(4) = 0.1431$, and $\beta(2) = \beta(3) = 0.3281$ are obtained by normalizing the coefficients of a Hamming window of size six to unity.

We have performed paired t-tests between discontinuity measures obtained for all pairs of 16 combinations of spectral, f_0, energy, and duration distance confidence measures with the hypothesis that the mean of the discontinuity measures for using one set of confidence measures is significantly smaller than the mean of the discontinuity measures for another set in the 99% confidence level. Table 1 shows the results where the rows correspond to the first set of confidence measures and the columns correspond to the second set of confidence measures. The null hypothesis is: The mean of discontinuity measures using a set of confidence measures labeled in the leftmost column of Table 1 is less than the mean of discontinuity measures using a set of confidence measures labeled in the top row of Table 1. As an example, the cell in the second row and last column corresponds to the comparison of discontinuity measures obtained when no confidence measures are used (marked as "None" in the first entry of the second row) and all confidence measures are used (marked as "SFRD" at the last column of the first row). The p-value in the cell is

Table 1
Paired t-test analysis results (p-values) for all combinations of confidence measures

	None	D	R	RD	F	FD	FR	FRD	S	SD	SR	SRD	SF	SFD	SFR	SFRD
None	0.50	0.23	0.15	0.32	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
D	0.77	0.50	0.38	0.6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
R	0.85	0.62	0.50	0.71	**0.00**	**0.00**	**0.00**	**0.00**	**0.01**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
RD	0.68	0.40	0.29	0.50	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
F	1.00	1.00	1.00	1.00	0.50	**0.00**	0.59	0.62	0.75	0.49	0.54	0.45	**0.00**	0.07	0.33	**0.00**
FD	1.00	1.00	1.00	1.00	1.00	0.50	1.00	1.00	1.00	1.00	1.00	1.00	0.97	1.00	0.98	**0.00**
FR	1.00	1.00	1.00	1.00	0.41	**0.00**	0.50	0.53	0.67	0.40	0.45	0.36	**0.00**	0.04	0.25	**0.00**
FRD	1.00	1.00	1.00	1.00	0.38	**0.00**	0.47	0.50	0.64	0.37	0.42	0.33	**0.00**	0.04	0.23	**0.00**
S	1.00	1.00	0.99	1.00	0.25	**0.00**	0.33	0.36	0.50	0.24	0.28	0.21	**0.00**	0.02	0.13	**0.00**
SD	1.00	1.00	1.00	1.00	0.51	**0.00**	0.6	0.63	0.76	0.50	0.55	0.46	**0.00**	0.07	0.34	**0.00**
SR	1.00	1.00	1.00	1.00	0.46	**0.00**	0.55	0.58	0.72	0.45	0.50	0.41	**0.00**	0.06	0.29	**0.00**
SRD	1.00	1.00	1.00	1.00	0.55	**0.00**	0.64	0.67	0.79	0.54	0.59	0.50	**0.00**	0.09	0.37	**0.00**
SF	1.00	1.00	1.00	1.00	1.00	0.03	1.00	1.00	1.00	1.00	1.00	1.00	0.50	0.94	0.99	0.02
SFD	1.00	1.00	1.00	1.00	0.93	**0.00**	0.96	0.96	0.98	0.93	0.94	0.91	0.06	0.50	0.85	**0.00**
SFR	1.00	1.00	1.00	1.00	0.67	0.02	0.75	0.77	0.87	0.66	0.71	0.63	**0.01**	0.15	0.50	**0.00**
SFRD	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.98	1.00	1.00	0.50

None: No confidence measure used in training. S: Spectral distance confidence measure, F: $f0$ distance confidence measure, R: Energy distance confidence measure, D: Duration difference confidence measure. Statistically significant p-values are in bold characters (99% confidence).

0.00 indicating that the null hypothesis is rejected. Therefore, the mean of discontinuity measures using no confidence measures is significantly greater than those obtained when all confidence measures are employed. Each case in which the null hypothesis is rejected with $p < 0.01$ are in bold characters. We observe from the last column of Table 1 that using all confidence measures results in significantly less discontinuity as compared to the rest set of combinations. Using the spectral distance and the $f0$ distance measure also results in significantly less discontinuity. A comparison of SF and SFRD show that SFRD results in less discontinuity with probability 0.98 which is also a significant result.

3.2. Spectral equalization

It is not always possible to record source and target utterances for voice conversion in identical environments. For example, in a voice conversion application to produce the voice of an artist who is not alive, it is not possible to replicate the exact recording conditions for the target voice. Performance of STASC may degrade in such a case because:

(i) forced alignment in Sentence-HMM requires a high level of acoustic match between the source and the target speaker utterances,
(ii) the output will not sound as if it was recorded in a similar environment with the target which may limit its acceptability as the target voice.

In this section, we investigate the use of employing spectral equalization in order to match the source and target long-term average spectra. The procedure consists of three steps:

(i) Source speaker's long-term average power spectrum, $P^{s}(w)$, and target speaker's long-term average power spectrum, $P^{t}(w)$, are computed from the training utterances.
(ii) The gain function, $G(w)$, is formulated as follows:

$$G(w) = \frac{P^{t}(w)}{P^{s}(w)}. \tag{21}$$

(iii) $G(w)$ is smoothed by a moving-average filter (i.e., weighted average of the neighboring frequency samples are computed to smooth the filter in the frequency domain) and the spectral equalization filter's frequency response, $\hat{G}(w)$, is obtained. In the transformation stage, Eq. (22) is used instead of Eq. (6) to perform spectral equalization together with vocal tract transformation.

$$H^{VT}(w) = \frac{H^{t}(w)}{H^{s}(w)}\hat{G}(w) \quad \text{or} \quad H^{VT}(w) = \frac{H^{t}(w)}{\hat{H}^{s}(w)}\hat{G}(w). \tag{22}$$

To demonstrate the utilization of spectral equalization in voice conversion, we have collected data under various recording conditions. First, a set of 15 training and four test utterances in Turkish language are recorded at 16 kHz sampling rate from two male speakers (one source, one target) for the evaluations. The recordings were done in a 4.3 m × 9.3 m × 2.8 m office using a Behringer XM2000S cardioid microphone on a desktop PC with a SoundBlaster™ Live Value sound card. We denote these recording conditions as Con-1. Identical utterances are recorded again by only the source speaker in a 5.2 m × 5.3 m × 3.2 m room with a Plantronics Audio 50 analog headset microphone on a laptop computer with an Avance A97 Audio sound card (Con-2). Three training and test sessions are performed as summarized in Table 2. In each session, four test utterances are transformed to the target speaker's voice. The test utterances and transformation outputs are phonetically aligned and the alignments are manually corrected. We have used the rms log spectral distortion $E_{dB}(H^{y}(w),H^{t}(w))$ between $H^{y}(w)$, the magnitude spectrum of a speech frame taken from the transformed utterance, and $H^{t}(w)$, the magnitude spectrum of the corresponding speech frame in the target utterance as defined in Eq. (18) to evaluate the objective performance of spectral equalization. For a given transformed and target utterance pair, the rms log spectral distortion is computed on a frame-by-frame basis using a window size of 20 ms with a skip rate of 10 ms. The mean and the standard deviation of the rms log spectral distortion values obtained from four test utterances are shown in the last two columns of Table 2. We have also performed pairwise analysis of *variance tests to observe whether the group means for each pair

Table 2
Training and test sessions for spectral equalization

Session	Source recording conditions	Target recording conditions	Spectral equalization?	Tfm-to-Tgt	
				μ_E	σ_E
1	Con-1	Con-1	No	5.84	2.19
2	Con-2	Con-1	No	6.18	2.45
3	Con-2	Con-1	Yes	5.96	2.29

μ_E denotes the mean, and σ_E denotes the standard deviation of the rms log spectral distortion, E_{dB}, for the corresponding session.

differed. The ANOVA results show that the mean rms log spectral distortion for Session 1 is significantly lower than that of Session 2 ($p = 2.56e - 5$) (see Table 3). Using spectral equalization in Session 3, the rms log spectral distance is significantly reduced as compared to Session 2 ($p = 0.0091$). When we compare Sessions 1 and 3, we observe that the difference is not significant.

Spectral distortion between the transformed and target utterances was 5.84 when both the source and target recording conditions were the same (Con-1-Con-1). This was expected since a good match in the environments for source and target recordings is expected to result in more accurate alignment between source and target utterances. The distortion increases to 6.18 when the recording conditions are different. When spectral equalization is employed (Session 3), the distortion reduces to 5.96. Informal listening tests also showed that both the similarity to target voice and the output quality are enhanced in Session 3. The examples are available at: http://www.sestek.com.tr/voice_conversion/oytun/rvc_demo.html

Fig. 13 shows the long term average spectrum of the target training utterances recorded in Con-1 and the long-term average spectrum of the source training utterances recorded in Con-1 and Con-2. Fig. 13 also shows the frequency response of the spectral equalization filter for the target utterances recorded in Con-1 and source utterances recorded in Con-2. The average of the magnitude of the frame-based DFT spectrum was computed for analysis window duration of 20 ms and a skip rate of 10 ms. Then, the ratio of the target and source average long-term spectra was computed for each frequency bin to obtain $E(w)$.

Fig. 14 shows spectral snapshots for several phonemes from the test utterances. Pre-emphasis with the digital filter $1 - 0.97z^{-1}$ is employed prior to LP analysis. The transformed spectrum

Table 3
Results of pairwise analysis of variance

Session pairs	ANOVA results	Significant?
1–2	$F(1, 3388) = 17.77$, $p = 2.56 \times 10^{-5}$	Yes
2–3	$F(1, 3426) = 6.82$, $p = 0.0091$	Yes
1–3	$F(1, 3378) = 2.62$, $p = 0.1054$	No

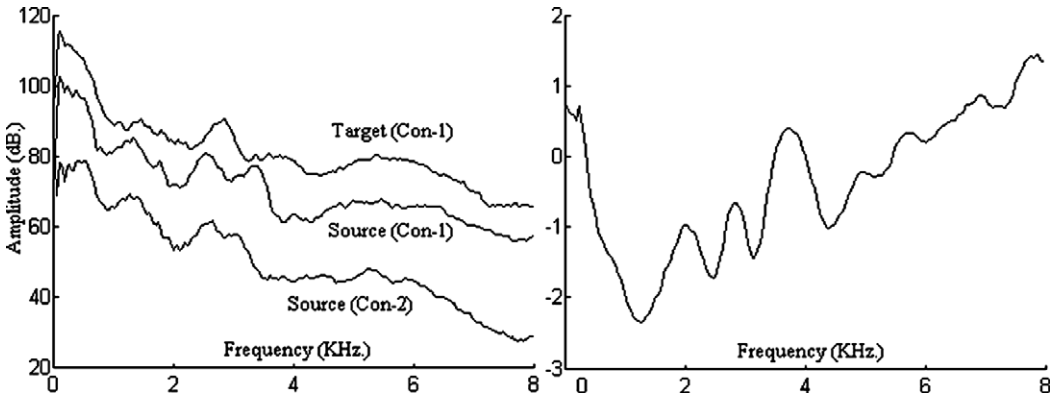

Fig. 13. Long-term average spectrum of the source and target utterances recorded in Con-1 and that of the source utterances recorded in Con-2 (left), frequency response, $E(w)$, of the spectral equalization filter for the target training utterances recorded in Con-1 and source training utterances recorded in Con-2 (right).

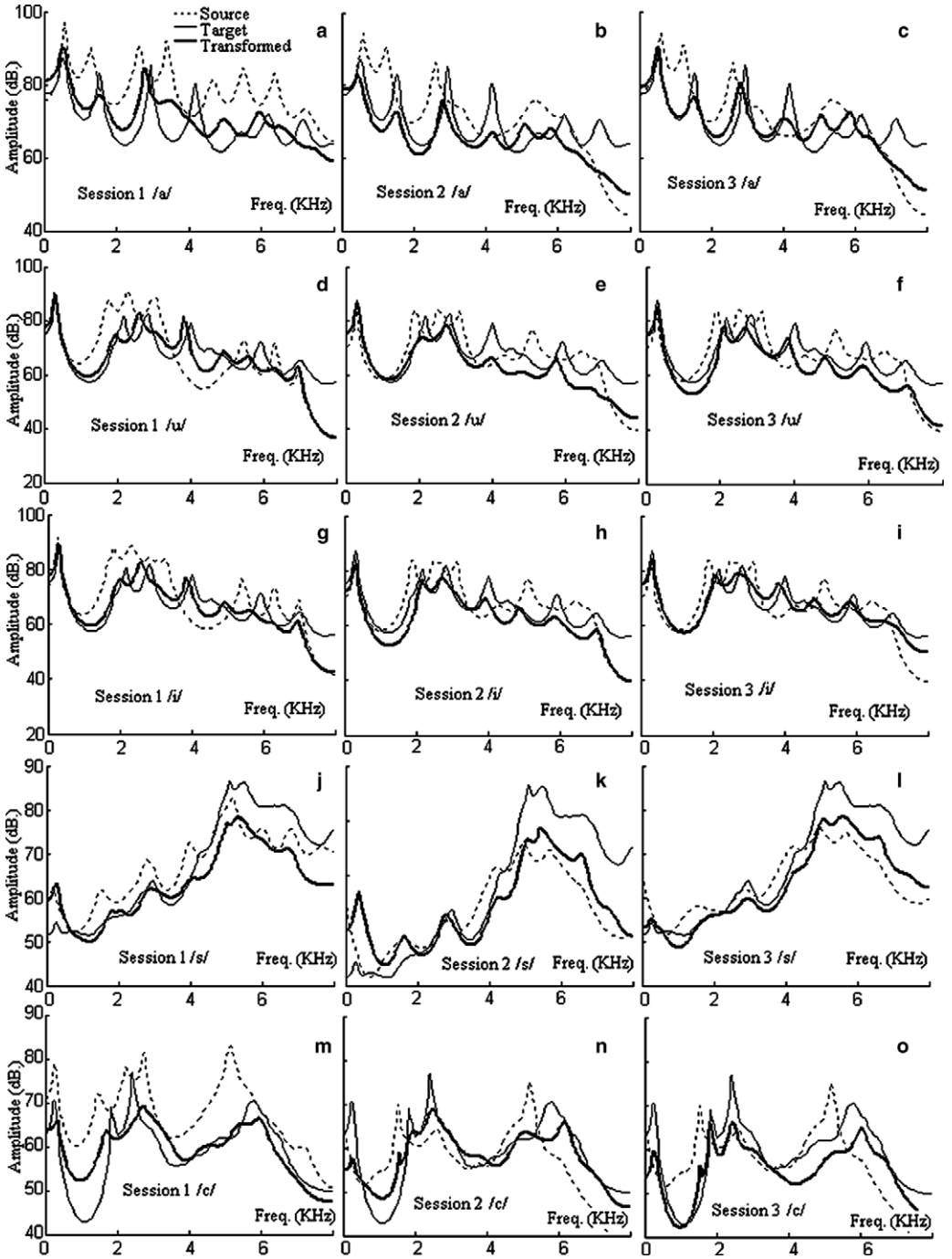

Fig. 14. Spectral equalization examples for different phonemes. (a), (d), (g), (j), and (m) correspond to Session 1 in which the source and target utterances were recorded in the same environment. (b), (e), (h), (k), and (n) are obtained in Session 2 when the recording conditions are different and spectral equalization is not employed. (c), (f), (i), (l), and (o) correspond to Session 3 in which the recording conditions are different and spectral equalization is employed.

does not match well with the target spectrum when the recording conditions are different. However, when the source and target utterances are recorded in identical environments or when spectral equalization is employed, a significantly better match between spectra is observed. As an example, one may consider the spectra for the Turkish phoneme /a/, i.e., Figs. 14(a)–(c). In Fig. 14(b), we observe that when the source and target utterances are recorded in different conditions, the first three formants in the transformed spectrum do not match that of the target spectrum. When spectral equalization is employed, a significantly better match is obtained as shown in Fig. 14(c).

3.3. Pre-emphasis

The performance of STASC largely depends on selecting the correct codebook entries during transformation. Identifying the appropriate codebook entries is simply a pattern-matching problem. In any stochastic pattern-matching problem, the feature extraction process should be robust to slight variations in the pattern generation process. In STASC, LSFs are used as the acoustic feature vector. We have observed that when no pre-emphasis was applied, the LSFs were not matching very well with the codebook entries due to small variations in the speech signal. In order to simulate those slight changes, we have shifted a 20 ms speech frame by 1 ms and compared the LSFs of the original and shifted frames. Fig. 14 shows the mean of the differences between LSFs of the original and shifted frames for a window shift of 1 ms. We have also shown the range that covers twice the standard deviation around the mean for the corresponding LSFs. Fig. 15 shows that the variation of high frequency LSFs due to small time shifts of the analysis frame increases at high frequencies. This results in high frequency distortion and formant shifts at the voice conversion output. The LSFs are calculated frame-by-frame for a database of 40 TIMIT utterances from 20 speakers (Garofolo et al., 1990). We have used the algorithm described in (Rothweiler, 1999) for computing the LSFs. The sampling rate was 16 kHz. A Hamming window is used.

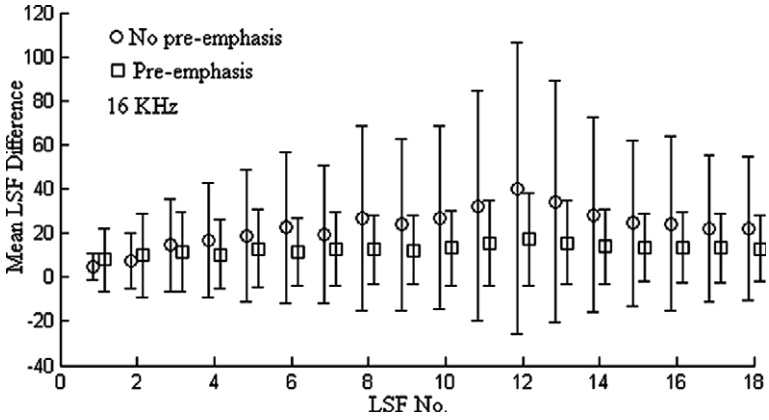

Fig. 15. Mean of the differences between LSFs in Hz. for a window shift of one millisecond at 16 kHz with an LP order of 18. The mean of the differences between LSFs is marked by a circle in the case when pre-emphasis was not employed and with a square when pre-emphasis was employed with $\alpha = 0.97$. Each line segment indicates the range that covers twice the standard deviation of the mean difference of the corresponding LSF.

LP analysis is performed with a prediction order of 18 and the original LSF vector w_1 is computed. The autocorrelation method, which is more robust to window location than the covariance method is employed for LP analysis. Then, the analysis is repeated by shifting the current window by 1 ms to obtain the shifted frame's LSF vector, w_2. The mean and the variance of the absolute difference of the entries of w_1 and w_2 are calculated. Similar steps are then followed by first applying pre-emphasis with $\alpha = 0.97$ in Eq. (23). We observe that pre-emphasis reduces both the mean and the standard deviation of the differences in the LSFs due to slight changes in the speech signal. Similar results were observed at different sampling rates

$$P(z) = 1 - \alpha z^{-1}. \tag{23}$$

Fig. 16 shows vocal tract spectrum conversion results at 44.1 kHz without and with pre-emphasis. The source speaker is a male adult and the target speaker is a female adult. We observe that when no pre-emphasis is applied the formants are shifted towards higher frequencies in the transformed spectrum resulting in an output voice that sounds like a child rather than a female adult. We also observe that the transformed spectrum does not match the target spectrum in high frequency regions that results in high frequency distortion in the voice conversion output. When pre-emphasis is applied, both problems are solved and the transformed vocal tract spectrum closely matches the target vocal tract spectrum.

Although pre-emphasis is a well-known technique in the field of speech processing, for voice conversion applications its usefulness could have been debatable. This is because pre-emphasis operation removes the spectral tilt to a large extent that may be an important parameter for speaker characterization. In speech coding applications, its use is justifiable since the voice characteristics are not altered significantly before removing pre-emphasis at the end of decoding.

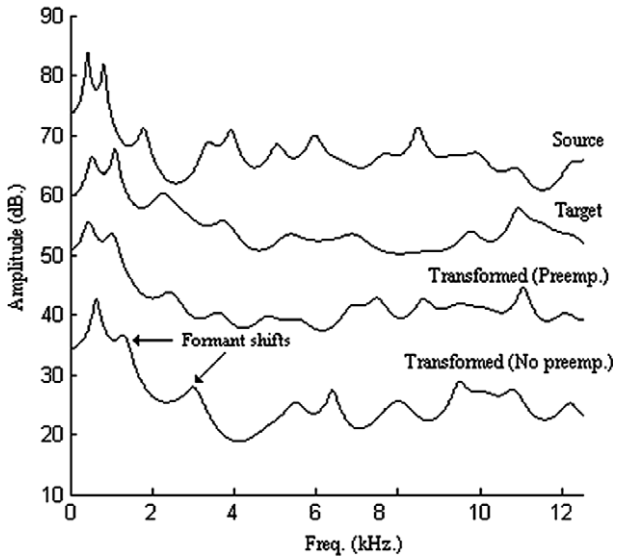

Fig. 16. Transformation at 44.1 kHz with and without pre-emphasis. Only 0–12 kHz range is shown for the phoneme /e/ in the utterance "...from a forg/e/ry". The target, transformed (preemp.) and transformed (no preemp.) spectra are shifted by −15, −30, and −45 dB, respectively, for enhancing readability.

Table 4
t-Test results for the comparison of objective distances between the transformation output and the target voice with and without pre-emphasis

Sampling rate (Hz)	16,000	22,050	44,100
p-Value	0.0607	0.3744	0.0148

p-values correspond to the probability of the mean of objective distances with pre-emphasis being less than the mean of distances without pre-emphasis.

However, in voice conversion the voice is altered significantly and it is not certain that removal of pre-emphasis will recover the spectral tilt of the target speaker. For this reason we performed the analysis in this study. According to informal listening tests, especially at high sampling rates such as 44.1 kHz, pre-emphasis is found to be useful.

We have performed an objective test to evaluate the voice conversion performance at three different sampling rates (16,000, 22,050, and 44,100 Hz) using no pre-emphasis and pre-emphasis with a factor of 0.97. A set of 60 TIMIT utterances from a male and a female American-English speaker recorded at 44,100 Hz is used. The databases at 16,000 and 22,050 Hz are obtained by down sampling the 44,100 Hz database. Codebooks were trained using 50 utterances and 10 utterances were reserved for testing. All utterances were phonetically labeled manually. Then, we have computed the rms log spectral distortion between the corresponding transformed and target frames in the test utterances. Pairwise *t*-tests are employed for comparing the objective distances obtained with and without pre-emphasis. The null hypothesis is that the mean of the distance to the target voice is greater when pre-emphasis is used. Table 4 shows the analysis results. We observe that at 16,000 and 22,050 Hz, the objective distance to the target voice does not change significantly with pre-emphasis. However, at 44,100 Hz pre-emphasis reduces the distance to the target voice. Note that an LP order of 20, 24, and 50 were used for 16,000, 22,050, and 44,100 Hz respectively.

4. Evaluations

In this section, three subjective listening tests are performed in order to compare the voice conversion algorithm that employs all three proposed methods with the baseline method. The first test uses subjective similarity scoring to compare the performance of the proposed and the baseline algorithms. In the second test, ABX comparison method is employed to evaluate the preference of subjects between the two algorithm outputs. Finally, the subjective qualities of the outputs are compared using a mean opinion score (MOS) test.

4.1. Subjective similarity test

We have designed a subjective listening test to evaluate the perceived similarity of the voice conversion outputs to the target speaker's voice. We have used a database of 20 utterances in Turkish from two male speakers. 16 sentences were used in the training and 4 sentences were set aside for testing. The sampling rate was 44.1 kHz. Ten subjects have listened to the original recording and the voice conversion output of each utterance in pairs. The voice conversion output was obtained

Table 5
Subjective test material and average perceived similarity scores

Pair No.	Pair type	Number of utterances	Average perceived similarity score
1	S-S or T-T	2	9.9
2	S-T	2	1.2
3	S-O1	4	3.2
4	S-O2	4	3.4
5	T-O1	4	6.1
6	T-O2	4	7.5

S: Original source speaker recording, T: Original target speaker recording, O1: Voice conversion output using the baseline algorithm, O2: Voice conversion output using the proposed algorithm.

by using either the baseline or the proposed method. The pairs were presented in randomized order. The subjects have scored the similarity of voices of speakers in the pairs by assigning an integer score in the range of 1–10. Note that a similarity score of "1" indicates that the voices of speakers in the pair are dissimilar whereas "10" indicates that they are identical. For calibration, we have also used original recordings of the source or target speakers in 4 pairs (1 source–source, 1 target–target, 2 source–target pairs). There were a total of 20 pairs as shown below.

The subjects have scored 9.9 out of 10 on the average for the similarity of the speakers when the pairs contained different utterances from the same speaker. When the speakers are different, the average similarity score was 1.2. In both the baseline and the proposed methods, similarity of the transformed voice to the source voice is close: 3.2 and 3.4. The similarity of the transformed voice to the target speaker's voice is improved by 23.0% in the proposed method as compared to the baseline method.

We have performed pairwise t-tests to analyze the similarity scores assigned to different pairs. Table 6 shows the t-distribution values for different group of pairs from Table 5. The groups of pairs that are significantly different in the 99% confidence range are typed in boldface characters. We observe that the similarity to target voice is significantly higher using the proposed method (T-O1 vs T-O2, $p = 0.0089$). The similarity of the voice conversion output to the source speakers does not change significantly using the two methods (S-O1 vs S-O2, $p = 0.8192$). The similarity of the voice conversion output to the target voice is significantly higher than the similarity to the source speaker's voice using both algorithms (S-O1 vs T-O1, $p = 0.0015$ and S-O2 vs T-O2, $p = 1.0\text{e} - 5$).

Table 6
Statistical analysis results (p-values) for the subjective similarity scores using pairwise t-tests

Pairs	S-S or T-T	S-T	S-O1	S-O2	T-O1	T-O2
S-S or T-T	×	**0**	**1.2e − 9**	**7.9e − 9**	**2.4e − 7**	**1.2e − 6**
S-T	×	×	**0.0035**	**0.0033**	**1.4e − 8**	**1.3e − 13**
S-O1	×	×	×	0.8192	**0.0015**	**2.3e − 6**
S-O2	×	×	×	×	**0.0043**	**1.0e − 5**
T-O1	×	×	×	×	×	**0.0089**
T-O2	×	×	×	×	×	×

The entries in boldface show the groups that are significantly different for 99% confidence.

4.2. ABX comparison test

An ABX listening test was employed in order to compare the proposed voice conversion algorithm with the baseline algorithm in terms of similarity to the target voice. The subjects were presented with a target utterance recording (X), and two voice conversion outputs (A and B) obtained using the baseline method and the proposed method. The subjects have decided whether the speaker in "A" or "B" sounds more like the speaker in "X".

The database that we used in the evaluations consisted of TIMIT utterances (Garofolo et al., 1990) from two male and two female speakers. Twenty-five utterances were used for training and five utterances for the test. Four source–target pairs were selected that cover all source and target gender combinations. Five utterances were transformed for each pair. Roughly 5–10% of the codebook entries were eliminated due to confidence measure constraints. The actual number for a specific speaker pair depends on the automatic alignment accuracy and also the degree of Gaussianity of the actual duration, pitch, energy, and spectrum distributions. Therefore, the test material consisted of 20 ABX triples. We have used 10 subjects in the test. The voice conversion algorithm that uses the proposed methods was preferred over the baseline method by 76.4%.

4.3. MOS test for voice conversion output quality

The two voice conversion algorithms were compared in terms of subjective quality of the output with an MOS listening test. The training database was the same as in the ABX listening test described in the previous subsection. Two TIMIT utterances that were not in the training set were transformed for each source–target speaker pair. The subjects have listened to a voice conversion output obtained by one of the two algorithms and assigned an integer number in the range from 1 to 5 (1 = Bad, 2 = Poor, 3 = Fair, 4 = Good, 5 = Excellent). There were a total of 16 voice conversion outputs in the test, half of which were obtained using the baseline algorithm and the other half using the proposed algorithm. A reference set of six recordings was used before the experiment that contained original recordings and outputs of several standard speech coders. The subjects were presented with the reference set and the corresponding mean opinion scores. The information on the reference set is shown in Table 7.

Ten subjects were used in the test. Table 8 shows the MOS and the standard deviation of the MOS for the baseline and the proposed voice conversion methods. The average MOS values of subjects for the proposed and the baseline methods were also compared using statistical significance testing. The MOS values for the proposed method and the baseline method were found

Table 7
Reference set for the MOS test

Coder or recording format	Bit rate (kbps)	MOS
PCM	64.0	4.4
ADPCM (G.726)	32.0	4.2
LD-CELP (G.728)	16.0	4.2
CSA-CELP (G.729)	8.0	4.2
CELP	4.8	4.0
LPC-10 (FS 1015)	2.4	2.3

Table 8
MOS test results for the comparison of two voice conversion algorithms in terms of subjective quality of the output and the statistical analysis of the results

Output	MOS	SD	Mean of MOS differences	SD of MOS differences	t-Distribution value	Statistically significant? (99% confidence)
Baseline	2.31	0.72	1.08	0.33	10.35	Yes
Proposed	3.39	0.80				

to be significantly different in the 99% confidence interval. Therefore, the proposed methods improve the subjective output quality by 46.8% on the average.

5. Conclusions

Voice conversion algorithms aim to provide high level of similarity to the target voice with an acceptable level of quality. In general, there is a trade-off between the two objectives. In order to keep the output distortion level low, the signal processing algorithms employed for voice conversion should avoid aggressive modifications that result in synthetic output quality. STASC provides a useful framework for determining an operating point where the user can achieve high level of similarity to the target voice with acceptable output quality. As a result, STASC has been successfully used in several movie dubbing and looping applications. Sample outputs from this study can be reached at: http://www.sestek.com.tr/voice_conversion/oytun/rvc_demo.html

The methods developed in this study help to improve voice conversion output quality and similarity to the target voice. The first method employs confidence measures in the training stage to eliminate alignment mismatches and to improve continuity and naturalness in the voice conversion output. The confidence measures are based on the spectral distance, f_0 distance, energy distance, and duration differences between the pairs of source and the target speaker states that are matched by the Sentence-HMM method. The second method makes use of the importance of pre-emphasis on LSF based vocal tract transformation. Employing pre-emphasis reduces the errors in the codebook matching stage in transformation. The last method focuses on the case when the source and target recording conditions are significantly different. Spectral equalization is employed in order to convert the long-term average power spectrum of the source speaker recordings to that of the target speaker recordings. The voice conversion algorithm that employs the new techniques is compared with the baseline algorithm, STASC, in subjective listening tests. The proposed algorithm improves similarity to target voice by 23%. It was also preferred over the baseline algorithm by 76.4% in an ABX test for similarity of the voice conversion output to the target voice. In the third subjective test, MOS results indicate that the new techniques improve output quality by 46.8% on the average.

References

Abe, M., Nakamura, S., Shikano, K., Kuwabara, H., 1988. Voice conversion through vector quantization. Proc. IEEE ICASSP, 565–568.

Acero, A., 1993. Acoustical and Environmental Robustness in Automatic Speech Recognition. Kluwer Academic Publishers, Dordrecht.

Arslan, L.M., Talkin, D., 1997. Voice conversion by segmental codebook mapping of line spectral frequencies and excitation spectrum. In: EUROSPEECH Proceedings, vol. 3, Rhodes Greece, pp. 1347–1350.

Arslan, L.M., 1999. Speaker transformation algorithm using segmental codebooks. Speech Commun. 28, 211–226.

Childers, D.G., 1995. Glottal source modeling for voice conversion. Speech Commun. 16 (2), 127–138.

Childers, D.G., Lee, C.-K., 1991. Vocal quality factors: analysis, synthesis, and perception. J. Acoust. Soc. Am. 90, 2394–2410.

Crosmer, J.R., 1985. Very low bit rate speech coding using the line spectrum pair transformation of the LPC coefficients. Ph.D. Thesis, Elec. Eng., Georgia Inst. Technology.

Davis, S., Mermelstein, P., 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoust. Speech Signal Process. 28, 357–366.

Drioli, C., 1999. Radial basis function networks for conversion of sound spectra. In: Proceedings of the 2nd COST G-6 Workshop on Digital Audio Effects (DAFx99), NTNU, Trondheim.

Dutoit, T., 1997. High-quality text-to-speech synthesis: an overview. J. Electrical Electronics Eng., Australia: Special Issue on Speech Recognition and Synthesis 17 (1), 25–37.

Fant, G., Liljencrants, J., Lin, Q. 1985. A four-parameter model of the glottal flow. Speech Transmission Laboratory Quarterly Progress and Status Reports, No. 4, Royal Institute of Technology, Stockholm, Sweden, pp. 1–13.

Flanagan, J.L., Golden, R.M., 1966. Phase vocoder. Bell Syst. Tech. J. 45, 1493–1500.

Furui, S., 1986. Research on individuality features in speech waves and automatic speaker recognition techniques. Speech Commun. 5 (2), 183–197.

Holmes, W., Holmes, J., Judd, M., 1990. Extension of the bandwith of the JSRU parallel-formant synthesizer for high quality synthesis of male and female speech. Proc. IEEE ICASSP 90 (1), 313–316.

Hunt, A., Black, A.W., 1996. Unit selection in a concatenative speech synthesis system using a large speech database. Proc. IEEE ICASSP 96, 373–376.

Garofolo, J.S., Lamel, L.F., Fisher, W.M., Fiscus, J.G., Pallett, D.S., Dahlgren, N.L., 1990. DARPA-TIMIT acoustic–phonetic continuous speech corpus (CDROM).

Itakura, F., 1975a. Line spectrum representation of linear predictor coefficients of speech signals. J. Acoust. Soc. Am. 57 (A), S35.

Itakura, F., 1975b. Minimum prediction residual principle applied to speech recognition. IEEE Trans. Acoust. Speech Signal Process. ASSP-23 (1), 67–72.

Itoh, K., Saito, S., 1982. Effects of acoustical feature parameters of speech on perceptual identification of speaker. IECE Trans. J65-A, 101–108.

Kain, A., Macon, M., 1998. Personalizing a speech synthesizer by voice adaptation. In: Proceedings of the Third ESCA/COCOSDA International Speech Synthesis Workshop, pp. 225–230.

Knohl, L., Rinscheid, A., 1993. Speaker normalization with self-organizing feature maps. In: Proceedings of the IJNN-93-Nagoya, International Joint Conference on Neural Networks, pp. 243–246.

Kuwabara, H., Sagisaka, Y., 1995. Acoustic characteristics of speaker individuality: control and conversion. Speech Commun. 16, 165–173.

Laroche, J., Stylianou, Y., Moulines, E., 1993. HNS: Speech modification based on a harmonic + noise model. In: Proceedings of the IEEE ICASSP-93, Minneapolis.

Makhoul, J., 1975. Linear prediction: a tutorial review. Proc. IEEE 63, 561–580.

Matsumoto, H., Hiki, S., Sone, T., Nimura, T., 1973. Multidimensional representation of personal quality of vowels and its acoustical correlates. IEEE Trans. AU AU-21, 428–436.

McAulay, R.J., Quatieri, T.F., 1995. Sinusoidal coding. In: Kleijn, W.B., Paliwal, K.K. (Eds.), Speech Coding and Synthesis. Elsevier, Netherlands, pp. 121–173.

Mizuno, H., Abe, M., 1995. Voice conversion algorithm based on piecewise linear conversion rules of formant frequency and spectrum tilt. Speech Commun. 16, 153–164.

Moulines, E., Charpentier, F., 1990. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. Speech Commun. 9, 453–467.

Moulines, E., Sagisaka, Y. (Eds.), 1995. Voice conversion: state of the art and perspectives. Speech Commun. 16 (2), 125,126.

Moulines, E., Verhelst, W., 1995. Time-domain and frequency-domain techniques for prosodic modification of speech. In: Kleijn, W.B., Paliwal, K.K. (Eds.), Speech Coding and Synthesis. Elsevier, Netherlands, pp. 519–555.

Narendranath, M., Murthy, H.M., Rajendran, S., Yegnanarayana, B., 1995. Transformation of formants for voice conversion using artificial neural networks. Speech Commun. 16, 207–216.

Necioglu, B.F., Clements, M.A., Barnwell III, T.P., Schmidt-Nielsen, A., 1998. Perceptual relevance of objectively measured descriptors for speaker characterization. Proc. IEEE ICASSP, 869–872.

Quatieri, T.F., McAulay, R.J., 1992. Shape invariant time-scale and pitch modification of speech. IEEE Trans. Signal Process. 40 (3), 497–510.

Rabiner, L.R., Schafer, R.W., 1978. Digital Processing of Speech Signals. Prentice-Hall, Englewood Cliffs, NJ.

Rabiner, L.R., 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE 77 (2), 257–286.

Rothweiler, J., 1999. A root-finding algorithm for line spectral frequencies. Proc. IEEE ICASSP 99, 661–664.

Stylianou, Y., Cappe, O., Moulines, E., 1998. Continuous probabilistic transform for voice conversion. IEEE Trans. Speech Audio Proc. 6 (2), 131–142.

Talkin, D., 1995. A robust algorithm for pitch tracking (RAPT). In: Kleijn, W.B., Paliwal, K.K. (Eds.), Speech Coding and Synthesis. Elsevier, Netherlands, pp. 121–173.

Turk, O., Arslan, L.M., 2002. Subband based voice conversion. Proc. ICSLP2002 1, 289–292.

Turk, O., Arslan, L.M., 2003. Voice conversion methods for vocal tract and pitch contour modification. Proc. Eurospeech (Interspeech), 2845–2848.

Zhang, W., Shen, L.Q., Tang, D., 2001. Voice conversion based on acoustic feature transformation. In: Proceedings of the 6th National Conference on Man–Machine Speech Communications.