Spectral Mapping Using Artificial Neural Networks for Voice Conversion

Srinivas Desai, Alan W. Black, *Member, IEEE*, B. Yegnanarayana, *Senior Member, IEEE*, and Kishore Prahallad, *Member, IEEE*

***Abstract*—In this paper, we use artificial neural networks (ANNs) for voice conversion and exploit the mapping abilities of an ANN model to perform mapping of spectral features of a source speaker to that of a target speaker. A comparative study of voice conversion using an ANN model and the state-of-the-art Gaussian mixture model (GMM) is conducted. The results of voice conversion, evaluated using subjective and objective measures, confirm that an ANN-based VC system performs as good as that of a GMM-based VC system, and the quality of the transformed speech is intelligible and possesses the characteristics of a target speaker. In this paper, we also address the issue of dependency of voice conversion techniques on parallel data between the source and the target speakers. While there have been efforts to use nonparallel data and speaker adaptation techniques, it is important to investigate techniques which capture speaker-specific characteristics of a target speaker, and avoid any need for source speaker's data either for training or for adaptation. In this paper, we propose a voice conversion approach using an ANN model to capture speaker-specific characteristics of a target speaker and demonstrate that such a voice conversion approach can perform monolingual as well as cross-lingual voice conversion of an arbitrary source speaker.**

***Index Terms*—Artificial neural networks (ANNs), cross-lingual, error correction, speaker-specific characteristics, spectral mapping, voice conversion.**

I. Introduction

A voice conversion (VC) system morphs the utterance of a source speaker so that it is perceived as if spoken by a specified target speaker. Such a transformation involves mapping of spectral, excitation and prosodic features including duration and F_0 patterns of a source speaker onto a target speaker's acoustic space [1]–[3]. In the area of spectral mapping, several approaches have been proposed since the first code book-based spectral transformation was developed by Abe *et al.* [4]. These techniques include mapping code books [4], artificial neural networks (ANNs) [5]–[7], dynamic frequency warping [8] and Gaussian mixture model (GMM) [1], [10]–[12]. In the GMM-based approach, the joint distribution of features extracted from the speech signals of a source speaker and a target speaker is modeled. As the number of mixture components increases, the performance of a GMM-based voice conversion improves [13]. The GMM transformation deals with every feature vector independent of its previous and next frames. Thus, it introduces local patterns in converted spectral trajectory which are different than that of the target's natural spectral trajectory. To obtain a better conversion of spectral trajectory, dynamic features are considered in the mapping function and such transformation is referred to as maximum likelihood parameter generation (MLPG) [14].

The relation between the vocal tract shapes of two speakers is typically nonlinear, and hence an ANN-based approach was proposed as ANNs can perform nonlinear mapping [6]. Narendranath *et al.* [6] used ANNs to transform the formants of a source speaker to that of a target speaker. Results were provided showing that the formant contour of a target speaker could be obtained using an ANN model. A formant vocoder was used to synthesize the transformed speech; however, no objective or subjective evaluations were provided to show how good the transformed speech was. The use of radial basis function neural network for voice transformation was proposed in [5]. The work in [2] also uses ANNs for spectral and prosodic mapping, but relies on additional signal processing for automatic extraction of syllable-like regions using pitch synchronous analysis. Our work differs from the earlier approaches using ANNs in the following ways.

- The earlier approaches using an ANN employed used either a carefully prepared training data which involved manual selection of vowels and syllable regions [5], [6] or signal processing algorithms to locate syllable like regions [2]. The proposed approach in this paper needs neither manual efforts nor signal processing algorithms to locate syllable like regions. Our approach makes use of a set of utterances provided from a source and a target speaker and automatically extracts the relevant training data using dynamic programming to train a voice conversion model.
- In previous works, there have been no comparative studies to evaluate how an ANN-based VC system performs in comparison with a widely used GMM-based VC system. In this paper, a comparative study between ANN and GMM-based voice conversion systems is performed and we show that an ANN-based voice conversion performs as good as

Manuscript received October 07, 2009; revised March 24, 2010. Current version published June 16, 2010. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Gokhan Tur.

S. Desai and B. Yegnanarayana are with the International Institute of Information Technology, Hyderabad 500032, India (e-mail: srinivasdesai@research.iiit.ac.in; yegna@iiit.ac.in).

A. W. Black is with Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: awb@cs.cmu.edu).

K. Prahallad is with International Institute of Information Technology, Hyderabad 500032, India, and also with the Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: kishore@iiit.ac.in).

Digital Object Identifier 10.1109/TASL.2010.2047683

that of a GMM-based conversion. Subjective and objective measures are conducted to evaluate the usefulness of ANNs for voice conversion.

- In this paper, we also address the issue of dependency of voice conversion techniques on parallel data between the source and the target speakers [1], [4], [6], [10], [15], [16]. By parallel data we mean a set of utterances recorded by both the source and the target speakers. Availability of such parallel data may not always be feasible. To overcome this limitation, two different methods have been proposed for voice conversion without using parallel data. In the first method, a set of nonparallel utterances from the source and the target speakers are collected and a unit selection approach is employed to find corresponding parallel frames [17]–[19]. In the second method, a voice conversion model is trained on preexisting parallel datasets and speaker adaptation techniques are used to adapt this voice conversion model to a particular pair of source and target speakers for which no parallel data is available [20]. While these methods avoid the need for parallel data, they still require speech data (nonparallel data) from the source speakers *a priori* to build the voice conversion models. This is a limitation to an application where an arbitrary user intends to transform his/her speech to a predefined target speaker without recording anything *a priori*. Thus, it is worthwhile to investigate voice conversion approaches which capture speaker-specific characteristics of a target speaker and avoid the need for speech data from a source speaker to train/adapt a voice conversion model. Such approaches not only allow an arbitrary speaker to transform his/her voice to a predefined target speaker but also find applications in cross-lingual voice conversion. In this paper, we propose a voice conversion approach using an ANN model to capture speaker-specific characteristics of a target speaker and completely avoid the need for speech data from a source speaker to train a voice conversion model. We demonstrate that such an approach can perform monolingual as well as cross-lingual voice conversion.

The organization of this paper is as follows. The first part of this paper (Sections II–IV) provides a comparative study between ANN and GMM-based approaches for voice conversion. In Section II, we describe the framework for voice conversion and provide details of ANN and GMM-based approaches. In Section III, we report the experimental results on ANN and GMM-based VC systems and provide a comparison between these approaches using subjective and objective evaluations. In Section IV, we discuss the enhancements made to improve the performance of an ANN-based voice conversion using delta and contextual features. The second part of this paper (Section V) addresses the issue of building voice conversion models without parallel data, specifically in the direction of using no *a priori* data from a source speaker. In Section V, we describe the method to capture speaker-specific characteristics using an ANN model, where the voice conversion model is built by using a target speaker's data. Such a model avoids the need for speech data from a source speaker and hence could be used to transform an arbitrary speaker including a cross-lingual speaker. A set of transformed utterances corresponding to results discussed in this work is available for listening at http://ravi.iiit.ac.in/~speech/uploads/taslp09_sinivas/.

II. Framework for Voice Conversion

A. Database

Current voice conversion techniques need a parallel database [1], [11], [13] where the source and the target speakers record the same set of utterances. The work presented here is carried out on the CMU ARCTIC database consisting of utterances recorded by seven speakers. Each speaker has recorded a set of 1132 phonetically balanced utterances [21]. The ARCTIC database includes utterances of SLT (U.S. Female), CLB (U.S. Female), BDL (U.S. Male), RMS (U.S. Male), JMK (Canadian Male), AWB (Scottish Male), KSP (Indian Male). It should be noted that about 30–50 parallel utterances are needed to build a voice conversion model [11]. Thus, for each speaker we took around 40 utterances as training data (approximately 2 minutes) and a separate set of 59 utterances (approximately 3 minutes) as testing data.

B. Feature Extraction

To extract features from a speech signal, an excitation-filter model of speech is applied. Mel-cepstral coefficients (MCEPs) are extracted as filter parameters and fundamental frequency (F_0) estimates are derived as excitation features for every 5 ms [22]. The number of MCEPs extracted for every 5 ms is 25. Mean and standard deviation statistics of $\log(F_0)$ are calculated and recorded.

C. Alignment of Parallel Utterances

As the durations of the parallel utterances typically differ, dynamic time warping (or dynamic programming) is used to align MCEP vectors of the source and the target speakers [10], [11]. After alignment, let $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$ denote the source and the target feature vectors at frame t, respectively.

D. Process of Training and Testing/Conversion

The training module of a voice conversion system to transform both the excitation and the filter parameters from a source speaker's acoustic space to a target speaker's acoustic space is as shown in Fig. 1. Fig. 2 shows the block diagram of various modules involved in a voice conversion testing process. In testing or conversion, the transformed MCEPs along with F_0 can be used as input to Mel log spectral approximation (MLSA) [22] filter to synthesize the transformed utterance. For all the experiments done in this work, we have used pulse excitation for voiced sounds and random noise excitation for unvoiced sounds.

E. Spectral Mapping Using GMM

In GMM-based mapping [13], [14], the learning procedure aims to fit a GMM model to the augmented source and target

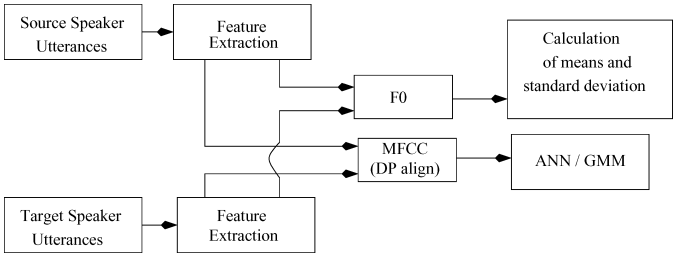


Fig. 1. Training module in voice conversion framework.

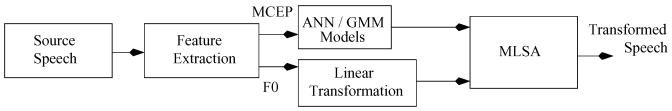


Fig. 2. Testing module in voice conversion framework.

feature vectors. Formally, a GMM allows the probability distribution of a random variable $\boldsymbol{z}$ to be modeled as the sum of M Gaussian components, also referred to as mixtures. Its probability density function $p(\boldsymbol{z}_t)$ can be written as

$$p(\boldsymbol{z}_t) = \sum_{m=1}^{M} \alpha_m \mathcal{N}\left(\boldsymbol{z}_t; \boldsymbol{\mu}_m^{(z)}, \boldsymbol{\Sigma}_m^{(z)}\right), \sum_{m=1}^{M} \alpha_m = 1, \quad \alpha_m \geq 0 \tag{1}$$

where $\boldsymbol{z}_t$ is an augmented feature vector $[\boldsymbol{x}_t^T \ \boldsymbol{y}_t^T]^T$. The notation ^{T} denotes transposition of a vector. $\mathcal{N}(\boldsymbol{z}_t; \boldsymbol{\mu}_m^{(z)}, \boldsymbol{\Sigma}_m^{(z)})$ denotes the parameters of a Gaussian distribution, and α_m denotes the prior probability with which the vector $\boldsymbol{z}_t$ belongs to the mth component. $\boldsymbol{\Sigma}_m^{(z)}$ denotes the covariance matrix and $\boldsymbol{\mu}_m^{(z)}$ denotes the mean vector of the mth component for the joint vectors. These parameters are represented as

$$\boldsymbol{\Sigma}_m^{(z)} = \begin{bmatrix} \boldsymbol{\Sigma}_m^{(xx)} & \boldsymbol{\Sigma}_m^{(xy)} \\ \boldsymbol{\Sigma}_m^{(yx)} & \boldsymbol{\Sigma}_m^{(yy)} \end{bmatrix}, \quad \boldsymbol{\mu}_m^{(z)} = \begin{bmatrix} \boldsymbol{\mu}_m^{(x)} \\ \boldsymbol{\mu}_m^{(y)} \end{bmatrix} \tag{2}$$

where $\boldsymbol{\mu}_m^{(x)}$ and $\boldsymbol{\mu}_m^{(y)}$ are the mean vectors of the mth component for the source and the target feature vectors, respectively. The matrices $\boldsymbol{\Sigma}_m^{(xx)}$ and $\boldsymbol{\Sigma}_m^{(yy)}$ are the covariance matrices, while $\boldsymbol{\Sigma}_m^{(xx)}$ and $\boldsymbol{\Sigma}_m^{(yy)}$ are the cross-covariance matrices, of the mth component for the source and the target feature vectors, respectively. The covariance matrices $\boldsymbol{\Sigma}_m^{(xx)}$, $\boldsymbol{\Sigma}_m^{(yy)}$, $\boldsymbol{\Sigma}_m^{(xx)}$, and $\boldsymbol{\Sigma}_m^{(yy)}$ are assumed to be diagonal in this work. The model parameters $(\alpha_m, \boldsymbol{\mu}_m^{(z)}, \boldsymbol{\Sigma}_m^{(z)})$ are estimated using expectation–maximization (EM) algorithm.

The conversion process (also referred to as testing process) involves regression, i.e., given an input vector $\boldsymbol{x}_t$, we need to predict $\boldsymbol{y}_t$ using GMMs, which is calculated as shown as

$$\begin{aligned} H(\boldsymbol{x}_t) &= E[\boldsymbol{y}_t|\boldsymbol{x}_t] \\ &= \sum_{m=1}^{M} h_m(\boldsymbol{x}_t) \left[\boldsymbol{\mu}_m^{(y)} + \boldsymbol{\Sigma}_m^{(yx)} \left(\boldsymbol{\Sigma}_m^{(xx)}\right)^{-1} \right. \\ &\qquad \left. \times \left(\boldsymbol{x}_t - \boldsymbol{\mu}_m^{(x)}\right)\right] \end{aligned} \tag{3}$$

where

$$h_m(\boldsymbol{x}_t) = \frac{\alpha_m \mathcal{N}\left(\boldsymbol{x}_t; \boldsymbol{\mu}_m^{(x)}, \boldsymbol{\Sigma}_m^{(xx)}\right)}{\sum_{n=1}^{M} \alpha_n \mathcal{N}\left(\boldsymbol{x}_t; \boldsymbol{\mu}_n^{(x)}, \boldsymbol{\Sigma}_n^{(xx)}\right)} \tag{4}$$

is the posterior probability that a given input vector $\boldsymbol{x}_t$ belongs to the mth component.

In this paper, we have conducted GMM-based VC experiments on the voice conversion setup built in FestVox distribution [23]. This voice conversion setup is based on the work done in [14], and supports the conversion considering 1) the correlation between frames (referred to as MLPG) and 2) the global variance (GV) of spectral trajectory.

F. Spectral Mapping Using ANN

ANN models consist of interconnected processing nodes, where each node represents the model of an artificial neuron, and the interconnection between two nodes has a weight associated with it. ANN models with different topologies perform different pattern recognition tasks. For example, a feed-forward neural network can be designed to perform the task of pattern mapping, whereas a feedback network could be designed for the task of pattern association. A multi-layer feed forward neural network is used in this work to obtain the mapping function between the source and the target vectors.

Fig. 3 shows the architecture of a four layer ANN used to capture the transformation function for mapping the acoustic features of a source speaker onto the acoustic space of a target speaker. The ANN is trained to map the MCEPs of a source speaker to the MCEPs of a target speaker, i.e., if $G(\boldsymbol{x}_t)$ denotes the ANN mapping of $\boldsymbol{x}_t$, then the error of mapping is given by $\epsilon = \sum_t ||\boldsymbol{y}_t - G(\boldsymbol{x}_t)||^2$. $G(\boldsymbol{x}_t)$ is defined as

$$G(\boldsymbol{x}_t) = \widetilde{g}\left(\boldsymbol{w}^{(3)} g\left(\boldsymbol{w}^{(2)} g\left(\boldsymbol{w}^{(1)} \boldsymbol{x}_t\right)\right)\right) \tag{5}$$

where

$$\widetilde{g}(\vartheta) = \vartheta, g(\vartheta) = a \tanh(b\vartheta). \tag{6}$$

Here $\boldsymbol{w}^{(1)}$, $\boldsymbol{w}^{(2)}$, $\boldsymbol{w}^{(3)}$ represents the weight matrices of first, second, and third hidden layers of the ANN model, respectively. The values of the constants a and b used in tanh function are 1.7159 and 2/3, respectively. A generalized back propagation learning [6] is used to adjust the weights of the neural network so as to minimize ϵ, i.e., the mean squared error between the desired and the actual output values. Selection of initial weights, architecture of ANN, learning rate, momentum, and number of iterations are some of the optimization parameters in training an ANN [24]. Once the training is complete, we get a weight matrix that represents the mapping function between the spectral features of a pair of source and target speakers. Such a weight matrix can be used to transform a feature vector from the source speaker to a feature vector of the target speaker.

G. Mapping of Excitation Features

Our focus in this paper is to get a better transformation of spectral features. Hence, we use the traditional approach of F_0 transformation as used in a GMM-based transformation. A logarithm Gaussian normalized transformation [25] is used to transform the F_0 of a source speaker to the F_0 of a target speaker as indicated as follows:

$$\log(F_{0\,conv}) = \mu_{tgt} + \frac{\sigma_{tgt}}{\sigma_{src}} \left(\log(F_{0\,src}) - \mu_{src}\right) \tag{7}$$

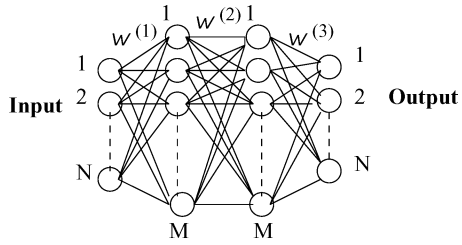


Fig. 3. Architecture of a four-layered ANN with N input and output nodes and M nodes in the hidden layers.

where μ_{src} and σ_{src} are the mean and variance of the fundamental frequency in logarithm domain for the source speaker, μ_{tgt} and σ_{tgt} are the mean and variance of the fundamental frequency in logarithm domain for the target speaker, $F_{0\ src}$ is the pitch of the source speaker, and $F_{0\ conv}$ is the converted pitch frequency.

H. Evaluation Criteria for Voice Conversion

1) Subjective Evaluation: Subjective evaluation is based on collecting human opinions as they are directly related to human perception, which is used to judge the quality of transformed speech. The popular tests are ABX test, MOS test, and similarity test.

- *ABX Test*: For the ABX test, we present the listeners with a GMM transformed utterance and an ANN transformed utterance to be compared against X, which will always be a natural utterance of the target speaker. To ensure that a listener does not become biased, we shuffle the position of ANN/GMM transformed utterances i.e., A and B, with X always constant at the end. The listeners would be asked to select either A or B, i.e., the one which they perceive to be closer to the target utterance.
- *MOS Test*: Mean opinion score (MOS) is another subjective evaluation where listeners evaluate the speech quality of the converted voices using a 5-point scale (5: excellent, 4: good, 3: fair, 2: poor, 1: bad).
- *Similarity Test*: In the similarity test, we present the listeners with a transformed utterance and a corresponding natural utterance of the target speaker. The listeners would be asked to provide a score indicating how similar the two utterances are in terms of speaker characteristics. The range of similarity test is also from 1 to 5, where a score of 5 indicates that both the recordings are from the same speaker and a score of 1 indicates that the two utterances are spoken by two different speakers.

2) Objective Evaluation: Mel cepstral distortion (MCD) is an objective error measure known to have correlation with the subjective test results [13]. Thus, MCD is used to measure the quality of voice transformation [11]. MCD is related to filter characteristics and hence is an important measure to check the performance of mapping obtained by an ANN/GMM model. MCD is computed as follows:

$$MCD = (10/\ln 10) * \sqrt{2 * \sum_{d=1}^{25} \left(mc_d^t - mc_d^e\right)^2} \qquad (8)$$

TABLE I
OBJECTIVE EVALUATION OF A GMM-BASED VC SYSTEM FOR VARIOUS TRAINING PARAMETERS WHERE SET 1: SLT TO BDL TRANSFORMATION; SET 2: BDL TO SLT TRANSFORMATION

No. of mixtures	No. of params.	MCD [dB]					
		Without MLPG		With MLPG (with GV)		With MLPG (Without GV)	
		Set 1	Set 2	Set 1	Set 2	Set 1	Set 2
32	6176	6.367	6.102	6.547	6.072	6.152	5.823
64	12352	6.336	6.107	6.442	6.015	6.057	5.762
128	24704	6.348	6.068	6.389	5.907	6.017	5.682

where mc_d^t and mc_d^e denotes the dth coefficient of the target and the transformed MCEPs, respectively.

III. EXPERIMENTS AND RESULTS

A. Objective Evaluation of a GMM-Based VC System

To build a GMM-based VC system, we have considered two cases: 1) transformation of SLT (U.S. female) to BDL (U.S. male); and 2) transformation of BDL (U.S. male) to SLT (U.S. female). For both the experiments, the number of training utterances is 40 (approximately 2 minutes) and the testing is done on the test set of 59 utterances (approximately 3 minutes). The number of vectors for 40 training utterances in SLT and BDL is 23 679 and 21 820, respectively.

Table I provides the MCD scores computed for SLT-to-BDL and BDL-to-SLT, respectively, for increasing number of Gaussians. It could be observed that the MCD scores decrease with the increase in the number of Gaussians; however, it should be noted that the increase in the number of Gaussians also increases the number of parameters in the GMM. With the use of diagonal covariance matrix, the number of parameters in the GMM with 64 and 128 Gaussian components is 12 352 and 24 704, respectively. We can also observe that the GMM-based conversion with MLPG performs better than that of the GMM-based system without MLPG. However, the GMM-based VC system with MLPG and without GV produced lesser MCD scores than the GMM-based VC system with MLPG and with GV. While GV seemed to improve the quality of transformed speech based on human listening tests, it is not clear from [14] whether it also improves the score according to MCD computation. Considering the number of parameters used in the GMM model, we have used the GMM-based VC system with 64 Gaussian components (with MLPG and without GV) for further comparison with an ANN-based VC system.

B. Objective Evaluation of an ANN-Based VC System

To build an ANN-based VC system, we have considered two cases: 1) SLT-to-BDL; and 2) BDL-to-SLT. For both the experiments, the number of training utterances is 40 (approximately 2 minutes) and the testing is done on the test set of 59 utterances (approximately 3 minutes).

Table II provide MCD scores for SLT-to-BDL and BDL-to-SLT, respectively, for different architectures of ANN. In this paper, we have experimented with 3-layer, 4-layer, and 5-layer ANNs. The architectures are provided with the number of nodes

TABLE II
MCD OBTAINED ON THE TEST SET FOR DIFFERENT ARCHITECTURES OF AN ANN MODEL. (NO. OF ITERATIONS: 200, LEARNING RATE: 0.01, MOMENTUM: 0.3) SET 1: SLT TO BDL; SET 2: BDL TO SLT

S.No	ANN architecture	No. of params.	MCD [dB]	
			Set 1	Set 2
1	25L 75N 25L	3850	6.147	5.652
2	**25L 50N 50N 25L**	**5125**	**6.048**	**5.504**
3	25L 75N 75N 25L	9550	6.147	5.571
4	25L 75N 4L 75N 25L	4529	6.238	5.658
5	25L 75N 10L 75N 25L	5435	6.154	5.527
6	25L 75N 20L 75N 25L	6945	6.151	5.517

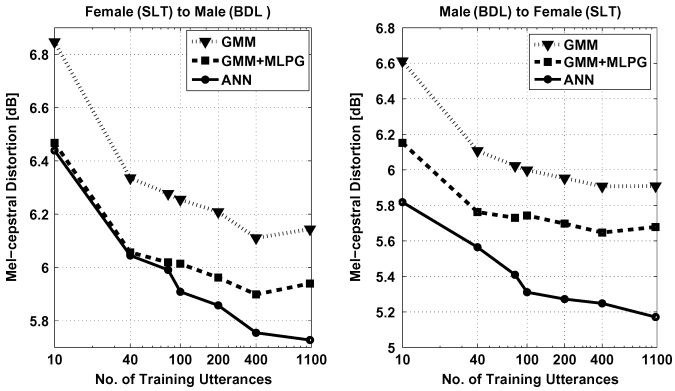


Fig. 4. MCD scores for ANN, GMM + MLPG, and GMM (without MLPG)-based VC systems computed as a function of number of utterances used for training. The results for GMM-based VC systems are obtained using 64 mixture components.

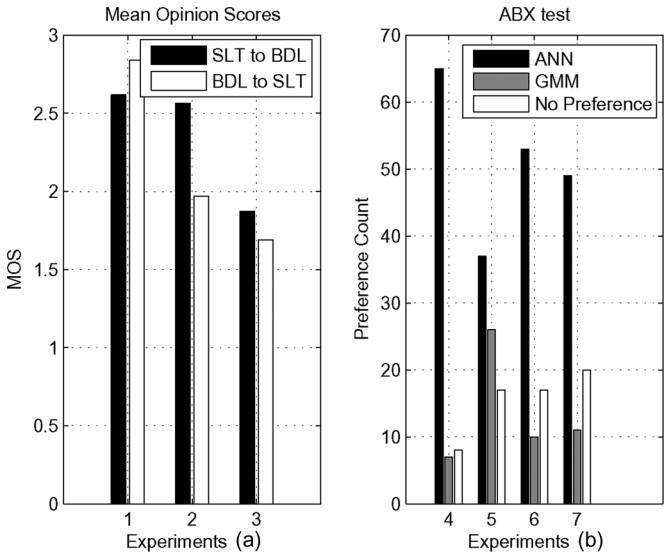


Fig. 5. (a) MOS scores for 1: ANN, 2: GMM + MLPG, 3: GMM. (b) ABX results for 4: ANN, GMM+MLPG(M− > F), 5: ANN, GMM+MLPG(F− > M), 6: ANN, GMM(M− > F), 7: ANN, GMM(F− > M).

TABLE III
AVERAGE SIMILARITY SCORES BETWEEN TRANSFORMED UTTERANCES AND THE NATURAL UTTERANCES OF THE TARGET SPEAKERS

Transformation Method	Avg. Similarity Score	
	SLT to BDL	BDL to SLT
ANN	2.93	3.02
GMM + MLPG	1.99	2.56

in each layer and the activation function used for that layer. For example, an architecture of 25L 75N 25L means that it is a 3-layer network with 25 input and output nodes and with 75 nodes in the hidden layer. Here, L represents "linear" activation function and N represents "tangential $(\tanh(.))$" activation function. From Table II, we see that the four-layered architecture 25L 50N 50N 25L (with 5125 parameters) provides better results when compared with other architectures. Hence, for all the remaining experiments reported in this section, the four layer architecture is used.

In order to determine the effect of number of parallel utterances used for training the voice conversion models, we performed experiments by varying the training data from 10 to 1073 parallel utterances. Please note that the number of test utterances was always 59. Fig. 4 shows the MCD scores for ANN, GMM + MLPG and GMM-based (without MLPG) VC systems computed as a function of number of utterances used for training. From Fig. 4, we could observe that as the number of training utterances increase, the MCD values obtained by both GMM and ANN models decrease.

C. Subjective Evaluation of GMM and ANN-Based VC Systems

In this section, we provide subjective evaluations for ANN and GMM-based voice conversion systems. For these tests, we have made use of voice conversion models built from 40 parallel utterances, as it was shown that this modest set produces good enough transformation quality in terms of objective measure. We conducted MOS, ABX, and similarity tests to evaluate the performance of the ANN-based transformation against the GMM-based transformation. It has to be noted that all experiments with GMM use static and delta features but the experiments with ANN use only the static features.

A total of 32 subjects were asked to participate in the four experiments listed below. Each subject was asked to listen to ten utterances corresponding to one of the experiments. Fig. 5(a) provides the MOS scores for 1) ANN, 2) GMM + MLPG, and 3) GMM-based (without MLPG) VC systems. Fig. 5(b) provides the results of ABX test for the following cases:

4) BDL to SLT using ANN + (GMM + MLPG);
5) SLT to BDL using ANN + (GMM + MLPG);
6) BDL to SLT using ANN + GMM;
7) SLT to BDL using ANN + GMM.

The MOS scores and ABX tests indicate that the ANN-based VC system performs as good as that of the GMM-based VC system. The MOS scores also indicate that the transformed output from the GMM-based VC system with MLPG was perceived to be better than that of the GMM-based VC system without MLPG.

A similarity test is also performed between the output of the ANN/GMM-based VC system and the target speaker's natural utterances. The results of this similarity test are provided in Table III, which indicate that the ANN-based VC system seems to perform better or as good as that of the GMM-based VC system. The significance of difference between the ANN and the GMM + MLPG-based VC systems for MOS and similarity scores was tested using hypothesis testing based on Student t-test, and the level of confidence indicating the difference was found to be greater than 95%.

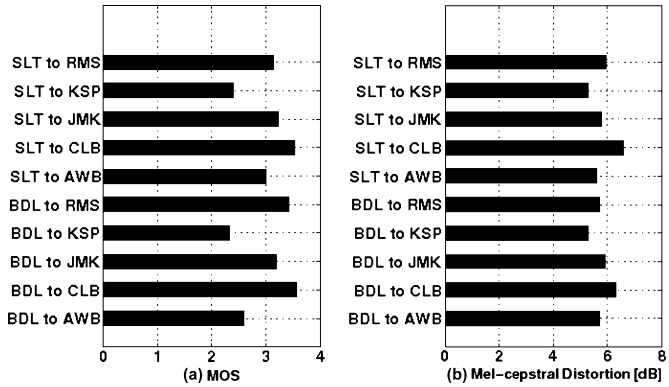


Fig. 6. (a) MOS and (b) MCD scores for ANN-based VC systems on ten different pairs of speakers.

D. Experiment on Multiple Speaker Pairs

In order to show that the ANN-based transformation can be generalized over different databases, we have provided MOS and MCD scores for voice conversion performed for ten different pairs of speakers as shown in Fig. 6. While MCD values were obtained over the test set of 59 utterances, the MOS scores were obtained from 16 subjects, each performing the listening tests on ten utterances. An analysis drawn from these results show that inter-gender voice transformation (ex: male to female) has an average MCD and a MOS score of 5.79 and 3.06, respectively, while the intra-gender (ex: male to male) voice transformation has an average MCD and a MOS score of 5.86 and 3.0, respectively. Another result drawn from the above experiments indicates that the transformation performance between two speakers with the same accent is better than that when compared with performance on speakers with different accents. For example, the voice transformation from SLT (US accent) to BDL (US accent) obtained an MCD value of 5.59 and a MOS score of 3.17, while the voice transformation from BDL (US accent) to AWB (Scottish accent) obtained an MCD value of 6.04 and a MOS score of 2.8.

IV. Enhancements to Voice Conversion Using ANN

In order to enhance the performance of spectral mapping by ANNs, we investigated two different methods. All the experiments in this section are designed based on the use of parallel training data. The results of these experiments are provided on the test set of 59 utterances.

A. Appending Deltas

The GMM-based approach explained in Section II-E appends dynamic features to the static features [13], [15], [16]. In Section III-B, we have compared the GMM-based system with deltas with the ANN-based system without deltas and hence we wanted to find out whether the use of deltas would further improves the performance of the ANN-based system.

In this context, we performed an experiment on SLT (female) to BDL (male) transformation, where the model is trained with deltas and on varying number of parallel training utterances. A set of three experiments were conducted, and the architectures of ANN used in these experiments are as follows:

TABLE IV
Results of Appending Deltas and Delta-Deltas of MCEPs for SLT (Female) to BDL (Male) Transformation

No. of training utterances	static features MCD [dB]	deltas MCD [dB]	delta-delta MCD [dB]
40	6.118	6.117	6.088
100	6.018	5.995	5.905
200	5.858	5.854	5.836
400	5.755	5.750	5.695

1) 25L 50N 50N 25L: static features;
2) 50L 100N 100N 50L: static and delta features;
3) 75L 150N 150N 75L: static, delta and acceleration/delta-delta features.

The MCD scores obtained for these three experiments are provided in Table IV and indicate that the ANN transformation with deltas is better than the ANN-based transformation without deltas. The results using delta-delta features are also provided in Table IV. It could be observed that the use of delta-delta features further reduces the MCD score for the ANN-based spectral mapping. The set of 40 training utterances used in this experiment is different than the one used in Section III-B and hence we find minor differences in the MCD score for static features.

B. Transformation With Use of Contextual Features

The use of deltas and delta-delta coefficients are computed over a context of three frames, and provide slope and acceleration coefficients of MCEPs [26]. Instead of computing slope and acceleration coefficients, we wanted to investigate the effect of augmented feature vectors, i.e., append MCEPs from previous and next frames to the MCEPs of a current frame, and provide these augmented features as input to train the ANN model.

In this context, we performed an experiment on SLT (female) to BDL (male) transformation, where the model is trained on varying context size and varying number of parallel training utterances. A context size of one indicates that the MCEPs from one left and one right frame are appended to MCEPs of the current frame. The results of SLT to BDL transformation are provided in Fig. 7, where a plot showing the MCD score with increasing number of training utterances and context size is provided.

Fig. 7 shows that the MCD score decreases with the increase in context size from 0 to 3 (i.e., 3 left and 3 right frames). The MCD score at the context size of 0 in Fig. 7 indicates the base line performance as explained in Section III-B. The ANN architectures used for context size of 1, 2, and 3 are 75L 225N 225N 75L, 125L 375N 375N 125L, and 175L 525N 525N 175L, respectively. From Fig. 7, it could also be observed that an increase in the number of training utterances from 40 to 200 leads to a decrease in MCD scores and thus improves the performance of the ANN-based spectral mapping.

From the experiments conducted in Sections IV-A and IV-B, we could observe that the use of deltas, acceleration coefficients, and contextual features improves the performance of an ANN-based VC system. However, an increase in the dimensionality of feature vectors also increases the size of an ANN architecture and the computational load in training and testing of a VC system.

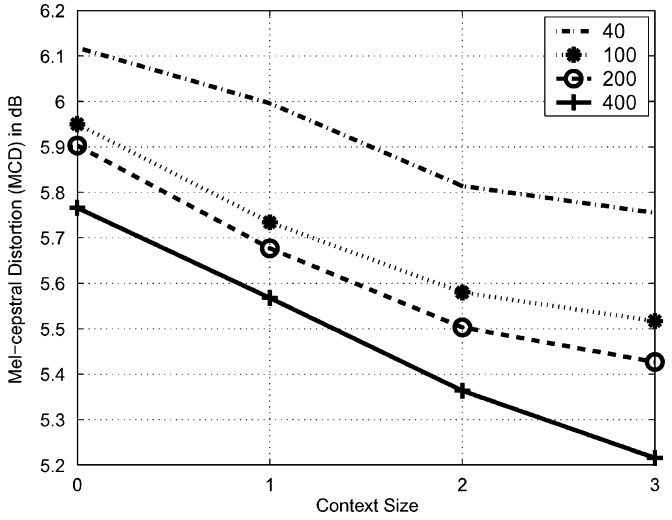


Fig. 7. Graph of MCD as a function of context size for varying number of training utterances for SLT (female) to BDL (male) transformation. Context 0 indicates the baseline performance.

V. Models Capturing Speaker-Specific Characteristics

So far we have discussed VC approaches which rely on existence of parallel data from the source and the target speakers. There have been approaches proposed in [17]–[20], which avoid the need for parallel data, however still require speech data (though nonparallel) from source speakers *a priori* to build a voice conversion model. Such approaches cannot be applied in situations, where an arbitrary user intends to have his/her voice transformed to a predefined target speaker, without recording anything *a priori*. In this section, we propose a voice conversion approach using an ANN model which captures speaker-specific characteristics of a target speaker. Such an approach does not require speech data from a source speaker and hence could be used to transform an arbitrary speaker including a cross-lingual speaker.

The idea behind capturing the speaker-specific characteristics using an ANN model is as follows. Let l_q and s_q be two different representations of a speech signal from a target speaker q. A mapping function $\Omega(l_q)$ could be built to transform l_q to s_q. Such a function would be specific to the speaker q and could be considered as capturing the essential speaker-specific characteristics. The choice of representation of l_q and s_q plays an important role in building such mapping networks and in their interpretation. If we assume that l_q represents linguistic information, and s_q represents linguistic and speaker information, then a mapping function from l_q to s_q should capture speaker-specific information in the process. The interpretation of order of linear prediction (LP) could be applied in deriving l_q and s_q. A lower order ($\leq$6) LP spectrum captures the first few formants and mostly characterizes the message (or linguistic) part of the signal, while a higher order ($\geq$12) LP spectrum captures more details in the spectrum and hence captures message and speaker characteristics [27]. Thus, l_q represented by a lower order LP spectrum or first few formants could be interpreted as speaker independent representation of the speech signal, and s_q represented by the MCEPs could be interpreted as carrying message and speaker information. An ANN model could be trained to minimize the error $||s'_q - s_q||$, where $s'_q = \Omega(l_q)$.

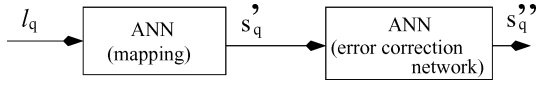


Fig. 8. Block diagram of an error correction network.

In this paper, l_q is represented by six formants, their bandwidths, and delta features. The formants, bandwidths, F_0, and probability of voicing are extracted using the ESPS toolkit [28]. The formants also undergo a normalization technique such as vocal tract length normalization as explained in Section V-A. s_q is represented by traditional MCEP features as it would allow us to synthesize using the MLSA synthesis technique. The MLSA synthesis technique generates a speech waveform from the transformed MCEPs and F0 values using pulse excitation or random noise excitation [22].

An ANN model is trained to map l_q to s_q using the backpropagation learning algorithm. Once the model is trained, it could be used to convert l_r to s'_q, where l_r could be from any arbitrary speaker r.

A. Vocal Tract Length Normalization (VTLN)

VTLN is a speaker normalization technique that tries to compensate for the effect of speaker-dependent vocal tract lengths by warping the frequency axis of the magnitude spectrum. Apart from use in speech recognition, VTLN has also been used in voice conversion [17]–[19].

Following the work in [29], we estimate the warp factors using pitch information and modify both formants and bandwidths. A piece-wise linear warping function as described in [29] is used in this work. The features representing l undergo a VTLN, to normalize the speaker effect on the message (or linguistic) part of the speech signal.

B. Error Correction Network

We introduce a concept of an error correction network which is essentially an additional ANN network, used to map the predicted MCEPs to the target MCEPs so that the final output obtained features represent the target speaker in a better way. The block diagram for an error correction network is shown in Fig. 8. Once s'_q are obtained, they are given as input to the second ANN model and it is trained to reduced the error $||s'_q - s_q||$. Such a network acts as an error correction mechanism to correct any errors made by the first ANN model. Let s''_q denote the output from the error correction network. It is observed that while the MCD values of s'_q and s''_q do not vary much, the speech synthesized from s''_q was found to be smoother than that of speech synthesized from s'_q. To train the error correction network, we use 2-D features i.e., feature vectors from three left frames, and three right frames are added as context to the current frame. Thus, the ANN model is trained with 175 dimensional vector (25 dimension MCEPs $*$ $(3 + 1 + 3)$). The architecture of this error correction network is 175L 525N 525N 175L.

C. Experiments With Parallel Data

As an initial experiment, we used parallel data of BDL and SLT. Features representing l_r were extracted from the BDL

TABLE V
RESULTS OF SOURCE SPEAKER (SLT-FEMALE) TO TARGET SPEAKER (BDL-MALE) TRANSFORMATION WITH TRAINING ON 40 UTTERANCES OF SOURCE FORMANTS TO TARGET MCEPS ON A PARALLEL DATABASE. HERE **F** REPRESENTS FORMANTS, **B** REPRESENTS BANDWIDTHS, Δ AND $\Delta\Delta$ REPRESENTS DELTA AND DELTA-DELTA FEATURES COMPUTED ON **ESPS** FEATURES, RESPECTIVELY. **UVN** REPRESENTS UNIT VARIANCE NORMALIZATION

S.No	Features	ANN architecture	MCD [dB]
1	4 F	4L 50N 12L 50N 25L	9.786
2	4 F + 4 B	8L 16N 4L 16N 25L	9.557
3	4 F + 4 B + UVN	8L 16N 4L 16N 25L	6.639
4	4 F + 4 B + Δ + $\Delta\Delta$ + UVN	24L 50N 50N 25L	6.352
5	F_0 + 4 F + 4 B + UVN	9L 18N 3L 18N 25L	6.713
6	F_0 + 4 F + 4 B + Δ + $\Delta\Delta$ + UVN	27L 50N 50N 25L	6.375
7	F_0 + Prob. of Voicing + 4 F + 4 B + Δ + $\Delta\Delta$ + UVN	30L 50N 50N 25L	6.105
8	**F_0 + Prob. of voicing + 6 F + 6 B + Δ + $\Delta\Delta$ + UVN**	**42L 75N 75N 25L**	**5.992**
9	**(F_0 + Prob. of voicing + 6 F + 6 B + Δ + $\Delta\Delta$ + UVN) + (3L3R MCEP to MCEP error correction)**	**(42L 75N 75N 25L) + (175L 525N 525N 175L)**	**5.615**

speaker and were mapped onto the s_q of SLT. This experimentation was done to obtain a benchmark performance for the experiments which map l_q to s_q (as explained in Section V-D).

The features representing l undergo a VTLN (as discussed in Section V-A), to normalize the speaker effect on the message (or linguistic) part of the signal. However, in this experiment, the mapping is done between BDLs l_r to SLTs s_q. The process of training such a voice conversion model is similar to the process explained in Section III-B. In Section III-B, the features of BDL speaker were represented by MCEPs, where as in this experiment, the formants and bandwidths are used. The results obtained in this section could also be compared with the results obtained in Section III-B. Hence, VTLN was not performed on the features representing l_r in this experiment.

Training was done to map BDL formants to SLT MCEPs with only 40 utterances. Testing was done on a set of 59 utterances. Table V shows the different representations of l_r and their effect on MCD values. These different representations include combination of different number of formants and their bandwidths, delta and acceleration coefficients of formants and bandwidths, pitch and probability of voicing. From the results provided in Table V, we can observe that experiment 9 (which uses six formants, six bandwidths, probability of voicing, pitch along with their delta and acceleration coefficients) employing an error correction network provided better results in terms of MCD values. These results are comparable with the results of voice conversion with BDL MCEPs to SLT MCEPs mapping as found in Section III-B.

D. Experiments Using Target Speaker's Data

In this experiment, we built an ANN model which maps l_q features of SLT onto s_q features of SLT. Here, l_q extracted from SLT utterances is represented by six formants, six bandwidths, F_0, probability of voicing and their delta and acceleration coefficients as shown in feature set for experiment 9 in Table V. The formants and bandwidths representing l_q undergo VTLN to normalize the speaker effects. s_q is represented by MCEPs extracted from SLT utterances. We use the concept of error correction network to improve the smoothness of the converted voice.

Fig. 9 provides the results for mapping l_r (where r = BDL, RMS, CLB, JMK voices) onto the acoustic space of SLT. To perform this mapping, the voice conversion model is built to map l_q to s_q (where $q = \mathrm{SLT}$) is used. To perform VTLN, we have used the mean pitch value of SLT. Hence, all the formants of source speaker are normalized with VTLN using mean of SLT F_0 and then are given to ANN to predict the 25-dimensional MCEPS. Similar results where the voice conversion model is built by capturing BDL speaker-specific features are also provided in Fig. 9.

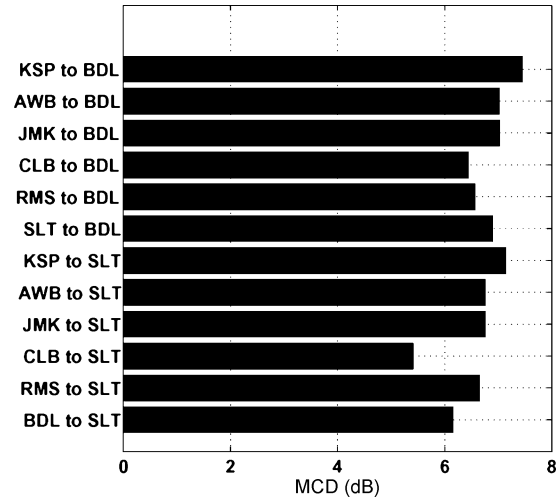


Fig. 9. Plot of MCD scores obtained between multiple speaker pairs with SLT or BDL as the target speaker. The models are built from a training data of 24 minutes and tested on 59 utterances (approximately 3 min).

TABLE VI
SUBJECTIVE EVALUATION OF VOICE CONVERSION MODELS BUILT BY CAPTURING SPEAKER-SPECIFIC CHARACTERISTICS

Target Speaker	MOS	Similarity tests
BDL	2.926	2.715
SLT	2.731	2.47

We have also performed listening tests whose results are provided in Table VI for MOS scores and similarity tests. For the listening tests, we chose three utterances randomly from each of the transformation pairs. Table VI provides a combined output of all speakers transformed to the target speaker (SLT or BDL). There were ten listeners who participated in the evaluations tests. The MOS scores and similarity test results are averaged over ten listeners.

The results shown in Fig. 9 and Table VI indicate that voice conversion models built by capturing speaker-specific characteristics using ANN models are useful. As this approach does not need any utterances from a source speaker to train a voice conversion model, we can use this type of model to perform

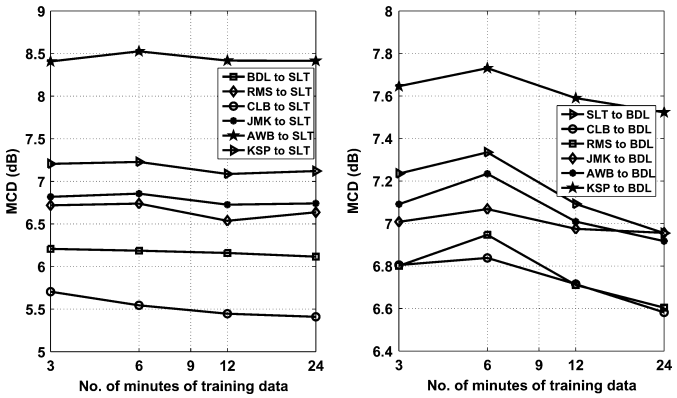


Fig. 10. Plot of MCD versus data size for different speaker pairs, with SLT or BDL as the target speaker.

TABLE VII
PERFORMANCE OF VOICE CONVERSION MODELS BUILT BY CAPTURING SPEAKER-SPECIFIC FEATURES ARE PROVIDED WITH MCD SCORES. ENTRIES IN THE FIRST COLUMN REPRESENT SOURCE SPEAKERS AND THE ENTRIES IN THE FIRST ROW REPRESENT TARGET SPEAKERS. ALL THE EXPERIMENTS ARE TRAINED ON 6 MINUTES OF SPEECH AND TESTED ON 59 UTTERANCES OR APPROXIMATELY 3 MINUTES OF SPEECH

Source \ Target	RMS	CLB	AWB	KSP
BDL	6.260	6.137	6.558	6.820
SLT	7.430	5.791	6.354	7.278
CLB	7.066	NA	6.297	7.166
JMK	6.617	6.616	6.224	6.878
RMS	NA	6.716	6.251	6.891
AWB	6.847	6.517	NA	6.769
KSP	7.392	7.239	6.517	NA

cross-lingual voice conversion. Fig. 10 shows the effect of amount of training data in building the ANN models capturing speaker-specific characteristics. It could be observed that the MCD scores tend to decrease with the increase in the amount of training data.

E. Experiments on Multiple Speakers Database

To test the validity of the proposed method, we conducted experiments on other databases from the ARCTIC set, such as RMS, CLB, JMK, AWB, and KSP. The training for all these experiments was conducted on 6 minutes of speech data. However, the testing was done on the standard set of 59 utterances. The MCD scores provided in Table VII indicate that the methodology of training an ANN model to capture speaker-specific characteristics for voice conversion could be generalized over different datasets.

F. Application to Cross-Lingual Voice Conversion

Cross-lingual voice conversion is a task where the language of the source and the target speakers is different. In the case of a speech-to-speech translation system, a source speaker may not know the target language. Hence, to convey information in his/her voice in the target language, cross-lingual voice conversion assumes importance. The availability of parallel data is difficult for cross-lingual voice conversion. One solution is to perform a unit selection approach [17]–[19] to find units in the utterances of the target speaker that are close to the source speaker or use utterances recorded by a bilingual speaker [20]. Our solution to cross-lingual voice conversion is to employ the ANN model which captures speaker-specific characteristics.

TABLE VIII
SUBJECTIVE RESULTS OF CROSS-LINGUAL TRANSFORMATION DONE USING CONVERSION MODEL BUILT BY CAPTURING SPEAKER-SPECIFIC CHARACTERISTICS. TEN UTTERANCES FROM EACH OF TELUGU (NK), HINDI (PRA), AND KANNADA (LV) SPEAKERS ARE TRANSFORMED INTO BDL MALE SPEAKER'S VOICE

Source Speaker	Target Speaker	MOS	Similarity tests
NK (Telugu)	BDL (English)	2.88	2.77
PRA (Hindi)	BDL (English)	2.62	2.15
LV (Kannada)	BDL English	2.77	2.22

In this context, we performed an experiment to transform three female speakers (NK, PRA, LV) speaking Telugu, Hindi, and Kannada, respectively, into a male voice speaking English (U.S. male-BDL). Our goal here is to transform NK, PRA, and LV voices to BDL voice and hence the output will be as if BDL were speaking in Telugu, Hindi, and Kannada, respectively. We make use of BDL models built in Section V-D to capture speaker-specific characteristics. Ten utterances from NK, PRA, LV voices were transformed into BDL voice and we performed MOS test and similarity test to evaluate the performance of this transformation. Table VIII provides the MOS and similarity test results averaged over all listeners. There were ten native listeners of Telugu, Hindi, and Kannada who participated in the evaluations tests. The MOS scores in Table VIII indicate that the transformed voice was intelligible. The similarity tests indicate that cross-lingual transformation could be achieved using ANN models, and the output is intelligible and possesses the characteristics of BDL voice.

VI. CONCLUSION

In this paper, we have exploited the mapping abilities of ANN and have shown that ANN can be used for spectral transformation in the voice conversion framework on a continuous speech signal. The usefulness of ANN has been demonstrated on different pairs of speakers. Comparison between ANN and GMM-based transformations has shown that the ANN-based spectral transformation yields results which are as good as that of a GMM-based transformation. The use of contextual features was shown to improve the performance of an ANN-based transformation. We have also shown that it is possible to build a voice conversion model by capturing speaker-specific characteristics of a speaker. We have used an ANN model to capture the speaker-specific characteristics. Such a model does not require any speech data from source speakers and hence could be considered as independent of a source speaker. We have also shown that the ANN model capturing speaker-specific characteristics could be applied for monolingual as well as for cross-lingual voice conversion.

ACKNOWLEDGMENT

The authors would like to thank Dr. T. Toda for useful hints on GMM-based voice conversion. They would also like to thank E. V. Raghavendra, K. Venkatesh, Lakshmikanth, and S. Joshi

of Speech Lab, IIIT-Hyderabad, for feedback and useful discussions on this work. Finally, they would also like to thank all the people who participated in various listening tests carried out for this work.

REFERENCES

[1] A. Kain and M. W. Macon, "Design and evaluation of a voice conversion algorithm based on spectral envelop mapping and residual prediction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Salt Lake City, UT, May 2001, vol. 2, pp. 813–816.

[2] K. S. Rao, "Voice conversion by mapping the speaker-specific features using pitch synchronous approach," *Comput. Speech Lang.*, vol. 23, no. 2, pp. 240–256, 2009.

[3] A. R. Toth and A. W. Black, "Incorporating durational modification in voice transformation," in *Proc. Interspeech*, Brisbane, Australia, Sep. 2008, pp. 1088–1091.

[4] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, New York, 1988, vol. 1, pp. 655–658.

[5] T. Watanabe, T. Murakami, M. Namba, T. Hoya, and Y. Ishida, "Transformation of spectral envelope for voice conversion based on radial basis function networks," in *Proc. Int. Conf. Spoken Lang. Process.*, Denver, CO, Sep. 2002, pp. 285–288.

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

[7] D. Srinivas, E. V. Raghavendra, B. Yegnanarayana, A. W. Black, and K. Prahallad, "Voice conversion using artificial neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Taipei, Taiwan, Apr. 2009, pp. 3893–3896.

[8] H. Valbret, E. Moulines, and J. P. Tubach, "Voice transformation using PSOLA technique," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, San Francisco, CA, Mar. 1992, pp. 145–148.

[9] Y. Stylianou, O. Cappe, and E. Moulines, "Statistical methods for voice quality transformation," in *Proc. European Conf. Speech Commun. (Eurospeech)*, Madrid, Spain, Sep. 1995, pp. 447–450.

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

[11] A. R. Toth and A. W. Black, "Using articulatory position data in voice transformation," in *Proc. 6th ISCA Workshop Speech Synth. (SSW6)*, Bonn, Germany, Aug. 2007, pp. 182–187.

[12] T. Toda, A. W. Black, and K. Tokuda, "Acoustic-to-articulatory inversion mapping with Gaussian mixture model," in *Proc. Int. Conf. Spoken Lang. Process.*, Jeju, Korea, Oct. 2004, pp. 1129–1132.

[13] T. Toda, A. W. Black, and K. Tokuda, "Mapping from articulatory movements to vocal tract spectrum with Gaussian mixture model for articulatory speech synthesis," in *Proc. 5th ISCA Speech Synth. Workshop (SSW5)*, Pittsburgh, PA, Jun. 2004, pp. 31–36.

[14] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum likelihood estimation of spectral parameter trajectory," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 15, no. 8, pp. 2222–2235, Nov. 2007.

[15] T. Toda, Y. Ohtani, and K. Shikano, "One-to-many and many-to-one voice conversion based on eigenvoices," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2007, vol. 4, pp. 1249–1252.

[16] T. Toda, Y. Ohtani, and K. Shikano, "Eigenvoice conversion based on Gaussian mixture model," in *Proc. Interspeech*, Pittsburgh, PA, Sep. 2006, pp. 2446–2449.

[17] D. Sundermann, A. Bonafonte, H. Hoge, and H. Ney, "Voice conversion using exclusively unaligned training data," in *Proc. ACL/SEPLN 2004, 42nd Annu. Meeting Assoc. for Comput. Linguistics/XX Congreso de la Sociedad Espanola para el Procesamiento del Lenguaje Natural*, Barcelona, Spain, Jul. 2004.

[18] D. Sundermann, H. Ney, and H. Hoge, "VTLN based cross-language voice conversion," in *Proc. 8th IEEE Autom. Speech Recognition and Understanding Workshop (ASRU)*, Virgin Islands, Dec. 2003.

[19] D. Sundermann, H. Hoge, A. Bonafonte, H. Ney, and S. Narayanan, "Text-independent voice conversion based on unit selection," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Toulouse, France, May 2006, vol. 1, pp. 81–84.

[20] A. Mouchtaris, J. V. Spiegal, and P. Mueller, "Nonparallel training for voice conversion based on a parameter adaptation approach," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 14, no. 3, pp. 952–963, May 2006.

[21] J. Kominek and A. W. Black, "The CMU ARCTIC speech databases," in *Proc. 5th ISCA Speech Synthesis Workshop (SSW5)*, Pittsburgh, PA, Jun. 2004, pp. 223–224.

[22] S. Imai, "Cepstral analysis synthesis on the Mel frequency scale," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Boston, MA, Apr. 1983, pp. 93–96.

[23] A. W. Black and K. Lenzo, "Building Voices in the Festival Speech Synthesis System," [Online]. Available: http://festvox.org/bsv/ 2000

[24] B. Yegnanarayana, *Artificial Neural Networks*. New Delhi, India: Prentice-Hall, 1999.

[25] K. Liu, J. Zhang, and Y. Yan, "High quality voice conversion through phoneme based linear mapping functions with STRAIGHT for mandarin," in *Proc. 4th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD 2007)*, 2007, vol. 4, pp. 410–414.

[26] S. Furui, "Cepstral analysis technique for automatic speaker verification," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Georgia, GA, Mar. 1981, vol. 29, no. 2, pp. 254–272.

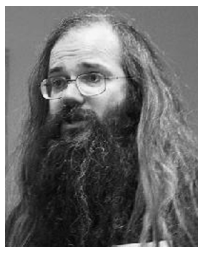
[27] H. Misra, S. Ikbal, and B. Yegnanarayana, "Speaker-specific mapping for text-independent speaker recognition," *Speech Commun.*, vol. 39, no. 3–4, pp. 301–310, 2003.

[28] ESPS, ESPS Source Code From the ESPS/Waves+ Package, 2009 [Online]. Available: http://www.speech.kth.se/software/

[29] A. Faria, "Pitch based vocal tract length normalization," Univ. of California, Berkeley, ICSI Tech. Rep. TR-03-001, Nov. 2003.

Srinivas Desai received the B.Tech. degree in electronics and communications engineering from Visvesvaraya Technological University (VTU), Belgaum, India, in 2005. He is currently pursuing the M.S. (by Research) degree at the International Institute of Information Technology (IIIT) Hyderabad, India.

His research interests include voice conversion, speech synthesis, and speaker recognition.



Alan W. Black (M'03) received the B.Sc. (Hons) degree in computer science from Coventry University, Coventry, U.K., in 1984 and the M.Sc. degree in knowledge-based systems and the Ph.D. degree in computational linguistics from Edinburgh University, Edinburgh, U.K., in 1986 and 1993, respectively.

He is an Associate Professor in the Language Technologies Institute, Carnegie Mellon University (CMU), Pittsburgh, PA. He previously was with the Centre for Speech Technology Research, University of Edinburgh, and before that at ATR in Japan. He is one of the principal authors of the Festival Speech Synthesis System, the FestVox voice building tools, and CMU Flite, a small footprint speech synthesis engine. Although his recent work primarily focused on speech synthesis, he also works on small footprint speech-to-speech translation systems (Arabic, Farsi, Dari, and Pashto), telephone-based spoken dialog systems, and speech for robots. In 2004, with Prof. K. Tokuda, he initiated the now annual Blizzard Challenge, the largest multi-site evaluation of corpus-based speech synthesis techniques.

Prof. Black was an elected member of the IEEE Speech Technical Committee 2004–2007 and is an elected member of the ISCA Board. He was program chair of the ISCA Speech Synthesis Workshop 2004, and was general co-chair of Interspeech 2006—ICSLP.

B. Yegnanarayana (M'78–SM'84) received the B.Sc. degree from Andhra University, Waltair, India, in 1961, and the B.E., M.E., and Ph.D. degrees in electrical communication engineering from the Indian Institute of Science (IISc) Bangalore, India, in 1964, 1966, and 1974, respectively.

He is a Professor and Microsoft Chair at the International Institute of Information Technology (IIIT), Hyderabad, India. Prior to joining IIIT, he was a Professor in the Department of Computer Science and Engineering, Indian Institute of Technology (IIT), Madras, India, from 1980 to 2006. He was the Chairman of the Department from 1985 to 1989. He was a Visiting Associate Professor of computer science at Carnegie Mellon University, Pittsburgh, PA, from 1977 to 1980. He was a member of the faculty at the IISc from 1966 to 1978. He has supervised 32 M.S. theses and 24 Ph.D. dissertations. His research interests are in signal processing, speech, image processing, and neural networks. He has published over 300 papers in these areas in IEEE journals and other international journals, and in the proceedings of national and international conferences. He is also the author of the book *Artificial Neural Networks* (Prentice-Hall of India, 1999).

Dr. Yegnanarayana was an Associate Editor for the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING from 2003 to 2006. He is a Fellow of the Indian National Academy of Engineering, a Fellow of the Indian National Science Academy, and a Fellow of the Indian Academy of Sciences. He was the recipient of the Third IETE Prof. S. V. C. Aiya Memorial Award in 1996. He received the Prof. S. N. Mitra Memorial Award for the year 2006 from the Indian National Academy of Engineering.

Kishore Prahallad (M'07) received the B.E. degree from Deccan College of Engineering and Technology, Osmania University, Hyderabad, India, in 1998 and the M.S. (by Research) degree from the Indian Institute of Technology (IIT) Madras, in 2000. He is currently pursuing the Ph.D. degree from the Language Technologies Institute, Carnegie Mellon University (CMU), Pittsburgh, PA.

He is a Senior Research Scientist at the International Institute of Information Technology, Hyderabad (IIIT-H). He has been associated with IIIT-H since March 2001, and started the speech activities in Language Technologies Research Center at IIIT-H. He has also been a visiting scholar to CMU since 2004. His research interests are in speech and language processing.

Mr. Prahallad has been serving as the chair of ISCA Special Interest Group on Indian Language Speech Processing (SIG-ILSP) since 2009, and is on the organizing committee of the Workshop on Image and Speech Processing (WISP) and the Winter School on Speech and Audio Processing (WiSSAP), held annually in India.