Mapping Frames with DNN-HMM Recognizer for Non-parallel Voice Conversion

Minghui Dong, Chenyu Yang, Yanfeng Lu, Jochen Walter Ehnes,
Dongyan Huang, Huaiping Ming, Rong Tong, Siu Wa Lee, Haizhou Li
Human Language Technology Department, Institute for Infocomm Research, A-Star, Singapore
{mhdong, yangc, luyf, jwehnes, huang, minghp, tongrong, swylee, hli}@i2r.a-star.edu.sg

***Abstract*— To convert one speaker's voice to another's, the mapping of the corresponding speech segments from source speaker to target speaker must be obtained first. In parallel voice conversion, normally dynamic time warping (DTW) method is used to align signals of source and target voices. However, for conversion between non-parallel speech data, the DTW based mapping method does not work. In this paper, we propose to use a DNN-HMM recognizer to recognize each frame for both source and target speech signals. The vector of pseudo likelihood is then used to represent the frame. Similarity between two frames is measured with the distance between the vectors. A clustering method is used to group both source and target frames. Frame mapping from source to target is then established based on the clustering result. The experiments show that the proposed method can generate similar conversion results compared to parallel voice conversion.**

I. INTRODUCTION

Voice conversion is a process to convert a specific source speaker's voice to another specific target voice. The technology has useful applications in generating individuality and diversity of generated voices for applications in education, entertainment, communication, etc.

To make the conversion work, voice samples from both source speaker and target speaker are needed. With the given source and target samples, a conversion function can be trained. Applying the trained conversion function to a new source voice, the expected target voice will be generated. Based on the content of the voice data used for training, voice conversion methods can be classified into parallel voice conversion and non-parallel voice conversion. Parallel voice conversion means the linguistic contents from both source and target voices are the same, while non-parallel voice conversion means the contents from the two speakers are different.

For parallel voice conversion, many methods have been proposed. Early works for voice conversion used vector quantization approaches [1]-[4], where conversion was based on codebooks that represent mapping from source to target voices. Later, conversion methods based on Gaussian Mixture Model (GMM) [5]-[10] were introduced and became popular. In GMM-based methods, training data is modelled by a GMM with multiple Gaussian components. GMM methods use continuous parametric functions which take into account the probabilistic classification and are able to generate good speaker characteristics. There were also attempts in using neural networks [11], [12] and partial least square regressions [13], in which the source spectra are directly mapped to the target spectra. A weighted frequency warping method [14] was also used to modify the source signal so as to match the target voice. Because it only tries to slightly change the source signal, it is relatively easy to keep a good voice quality. Recently, exemplar based methods [15]-[17] have shown promising results in generating high quality voices. Exemplar-based voice conversion directly uses speech exemplars to synthesize the converted speech using a linear combination of a set of exemplars of the target speaker. The exemplar-based methods are able to achieve good quality and speaker identity.

In parallel voice conversion, voice signals between source and target speakers can be matched frame by frame with the Dynamic Time Warping (DTW) method. The mapping between source and target signals is relatively easy to set up. Non-parallel voice conversion deals with more general cases since parallel data is not available in many real scenarios. In case of non-parallel voice conversion, the DTW method does not work anymore. Other solutions need to be used. There are a lot of efforts in non-parallel voice conversion. The popularly used methods can be classified into two categories, adaptation based method and data mapping based method. Both of the methods need to rely on the methods that are already used in parallel voice conversion.

The adaptation based methods normally train a conversion function using some previously recorded parallel data. When non-parallel data are available, the parameters of conversion functions are adapted to the desired target speaker [18]-[22]. In some solutions, a background model was used to support the adaptation of source and target models [23]. Data mapping based methods try to match the data of a non-parallel corpus first, and then use the parallel methods to build the conversion function. There are different ways to set up the mapping. In [24], frame vectors from source and target voice data are clustered separately. Then mapping is set up by finding the closest clusters between source and target data. In [25] a speech recognizer was used to label all the source and target frames. Alignment is done by matching the longest state subsequence. In [26] a unit selection method with dynamic programming was used to create the mapping between the two speakers. The INCA method [27] was proposed to iteratively do signal mapping, conversion and alignment. At each step, the converted signals get closer to the target signals.

Thus the signal mapping gets more accurate and the conversion quality improves.

For the adaptation methods, due to their statistical nature, the generated voice quality is limited by over smoothing problems. For the data mapping methods, the existing methods rely on either the comparison of acoustic signals or the results of a speech recognizer signals to find the mapping. The major problem of the acoustic comparison is that it is based on the voice data of the two speakers only. As no prior knowledge of the language is used, it is difficult to achieve good mapping with the voice data only, especially when the data set is small and the two voices are very different. The use of a speech recognizer is a good alternative because a speech recognizer has learned the pronunciations of the language. However, there will be some loss in alignment accuracy when converting the mapping from phone level to frame level. In this work, we propose a method to generate mapping of source and target signals at frame level by using frame level speech recognition.

Considering the remarkable progresses in parallel voice conversion, especially the exemplar based conversion method which has shown good performance in generating voice of good quality, we shall try to adopt the method into non-parallel voice conversion. To use this method, we first need to build a mapping mechanism for the non-parallel data. To set up the mapping of the source and target signals, we propose to use a deep neural network (DNN) based recognition result of frames as a clue. Compared with acoustic features, which are normally used in previous methods for mapping the frames of signals, the recognition result will be more stable as it allows for acoustic variations of phonetic units among different speakers.

In this paper, we will test the proposed method to see what we can achieve on non-parallel data compared with parallel data for the exemplar based voice conversion method.

II. Frame Mapping Based on DNN Recognizer

There has been a lot of research on parallel voice conversion. For example, GMM based methods, weighted frequency warping methods, and exemplar based methods are able to generate speech of relatively high quality. However, they need parallel data to work. To deal with non-parallel data, we need to build up the mapping to match corresponding frames from source speaker to target speaker first. In previous methods, normally acoustic features are used to build the mapping pairs of the frame. Due to the acoustic difference between two speakers, it is difficult to get an accurate mapping directly. The INCA method was introduced to iteratively convert the source voice to a state that is closer to target voice so that it is easier to reach a correct mapping by acoustic comparison. It works relatively well. However, as the mapping between two signals comes from the limited acoustic signals only, it does not make use of any knowledge based on the used language.

In our work, we take an alternative approach, which is to use a DNN hidden Markov model (HMM) hybrid recognizer to recognize the frames of both source and target voices, and set up mapping between the two speakers based on the recognition results. In recent years, DNN based recognizers have been able to perform increasingly well. The advantage of using a recognizer is that it is trained with voice data from hundreds or thousands of speakers in different environments or through various recording channels. Thus the recognition result is quite stable for further processing. If the alignment process is based on a recognizer, the effect of mismatch in environment and channel between the two speakers will not be carried over to the alignment process. We are also able to incorporate whatever improvement there is on speech recognition into the mapping process whenever a better recognizer is built.

The use of speech recognition results has advantages over acoustic features for frame mapping. Fig 1 is an illustration to show the advantage of using a recognizer. The acoustic space of the two speakers can be quite different. The position of the pronunciations of the same phone from two different speakers may not be the same in the acoustic space. However, in voice conversion, since we want to keep the phonetic content of the speech unchanged. So direct mapping of phonetic information is a better option than purely dealing with acoustic signals.

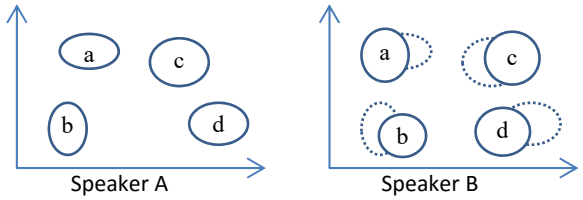

Fig 1: Illustration of phonetic units in acoustic space for different speakers. The speech of same phonetic unit from different speakers may appear at different positions in acoustic space.

In order to measure the similarity of frames from two speakers, we will compare the recognition results of frames. Fig 2 is an illustration showing how the position of a frame in acoustic space is measured. We shall define the position of a frame in acoustic space by calculating its distances to all the phonetic units in the acoustic space. The distance will be represented by measuring the differences between the recognition results, i.e. the values of pseudo likelihood. Similarity between two frames will then be calculated based on the recognition results.

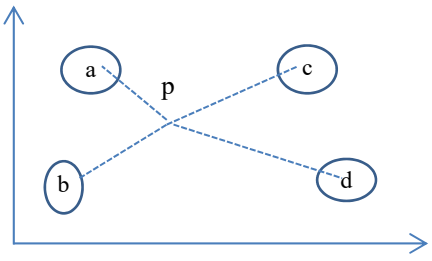

Fig 2: Illustration of the position of frame in acoustic space. The position of a frame can be described by its distances to all the phonetic units in the acoustic space.

A. Description of the Method

We shall describe our proposed mapping method for non-parallel data. Fig 3 shows the flowchart of the process for frame alignment for non-parallel speech data from two speakers.

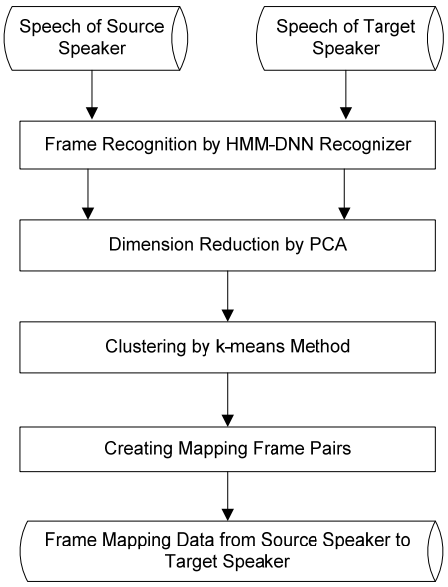

Fig 3: Flowchart of the frame mapping with DNN recognizer for non-parallel voice conversion

For both source and target speech, we use DNN-HMM recognizer to recognize each frame of the voice signals. The result of the frame recognition is a long vector, which represents the pseudo likelihood of the frame belonging to each HMM state.

Due to the large dimension of the recognition result, it is difficult for direct processing for our purpose of frame mapping from source to target speakers. A dimension reduction is done using principal component analysis (PCA) approach.

The dimension reduced vectors for utterances of both speakers are put together. k-mean method is then used to do a clustering. The frames from both speakers belonging to the same cluster are considered matched. If there are frames from both speakers in one cluster, one frame from each speaker is selected to form a frame pair.

B. DNN-HMM Recognizer

In this paper, a DNN-HMM speech recognizer is used. In such a system, a DNN is used to calculate pseudo-likelihoods for the states of HMM. The neural network is used to classify each individual frame.

For an observation o_{ut} at time t in utterance u, the output $y_{ut}(s)$ of the DNN for the HMM state s is calculated using the softmax activation function:

$$\mathrm{y_{ut}(s)} \triangleq \mathrm{P(s|o_{ut})} = \frac{\exp\{\mathrm{a_{ut}(s)}\}}{\sum_{s'} \exp\{\mathrm{a_{ut}(s')}\}} \quad (1)$$

where $a_{ut}(s)$is the activation at the output layer for the HMM state s. The recognizer uses a pseudo log-likelihood of state s given observation o_{ut},

$$\mathrm{q_{ut}(s)} = \log \mathrm{p}(o_{ut}|\mathrm{s}) = \log \mathrm{y_{ut}(s)} - \log \mathrm{P(s)} \quad (2)$$

where *P(s)* is the prior probability of state *s* calculated from the training data.

For each frame at time t in utterance u, a vector of pseudo log-likelihood is obtained by

$$\mathrm{L_u(t)} =< \mathrm{q_{ut}(1), q_{ut}(2), \dots\dots, q_{ut}(N)} > \quad (3)$$

where *N* is the HMM number of states in the recognizer. This vector actually represents how similar the frame is to the phonetic classes in the language.

As the number of HMM states is large, the dimension of the vector of pseudo log-likelihood is very long. To use it effectively in the voice conversion, we use PCA to reduce the dimension of the vector. Thus each frame will be represented by a low dimension vector.

$$\mathrm{R_u(t)} =< \mathrm{r_{ut}(1), r_{ut}(2), \dots\dots, r_{ut}(M)} > \quad (4)$$

where *M* is the reduced number of dimensions after PCA.

C. Setup of the Frame Mapping

As the frame features of both source and target speech are available, the next step is to match the corresponding frames from both speakers. We use a clustering approach to group frames of similar phonetic properties into clusters. The process works as follows:

- The frames of all utterances from both source and target speakers are pooled together.
- k-means clustering method is used to cluster the frames into a given number of clusters based on the dimension reduced feature vectors. Each cluster is considered to contain only frames of similar phonetic properties.
- From each cluster, one frame from the source speaker and one frame from the target speaker are extracted to form a frame pair. This frame pair is considered a mapping of a corresponding frame from the source speaker to the target speaker. It is possible that some clusters only contain frames from one speaker. In such cases, the clusters are skipped, and no pairs are drawn from them. In case there are multiple frames from one or both speakers inside a cluster, we select the frame closest to the cluster center from each speaker.

After the mapping of frames has been set up, we are able to use voice conversion approaches for parallel data to train the conversion function.

III. Exemplar-based Voice Conversion with Residual Compensation

In this paper, an exemplar-based voice conversion method with residual compensation is used. The converted spectral envelopes can be generated directly to synthesize the speech for the target speaker in this method. It mainly contains the following two parts, more details can be found in [17].

A. Basic Exemplar-based Sparse Representation

The basic exemplar-based sparse representation is based on the non-negative matrix factorization (NMF) method. The idea of this method is to describe the spectral envelopes as a linear combination of the spectral basis, i.e. exemplar, which can be expressed as:

$$\mathrm{x}^{\mathrm{spec}} \approx \sum_{\mathrm{n}=1}^{\mathrm{N}} \mathrm{a}_{\mathrm{n}}^{\mathrm{spec}} \cdot \mathrm{h}_{\mathrm{n}} = \mathrm{A}^{\mathrm{spec}} \cdot \mathrm{h} \quad (5)$$

where $\mathrm{x}^{\mathrm{spec}} \in \mathcal{R}^{\mathrm{D}\times 1}$ represents the spectral envelope of one speech frame, D is the dimension of the spectral envelope, N is the number of exemplars in the dictionary. a_n^{spec} stands for the fixed exemplars which are selected from the training set. h_n is the non-negative weight, i.e. activation of the n^{th} exemplar.

Each observation is modeled independently, so the spectral envelope sequence of each utterance of the source speaker can then be expressed as:

$$\mathrm{S}^{\mathrm{spec}} \approx \mathrm{A}^{\mathrm{spec}} \cdot \mathrm{H} \quad (6)$$

where $\mathrm{S}^{\mathrm{spec}}$ stands for the spectral envelope sequence for each utterance of source speaker. H represents the activation matrix.

In order to generate the converted spectral envelope, it is assumed that the paired source-target exemplars A^{spec} and B^{spec} can share the same activation matrix H. Here, each row vector in A^{spec} and the corresponding row vector in B^{spec} are the feature pairs obtained from the k-means clustering using the DNN output features. The converted spectral envelopes can then be generated by:

$$\mathrm{T}^{\mathrm{spec}} \approx B^{spec} \cdot H \quad (7)$$

where $\mathrm{T}^{\mathrm{spec}}$ is the converted spectral envelope. H is the shared activation matrix which is estimated by the non-negative matrix factorization technique[17],[28],[29].

B. Compensation for Model Residual

In order to generate the converted spectral envelope more precisely and improve the performance of the conversion, we adopted a residual compensation technique[28].

The residual stands for the modelling errors between the observed spectral envelope $\mathrm{X}^{\mathrm{spec}}$ and the modeled spectral envelope $A^{spec} \cdot H$. This process can be expressed as:

$$\mathrm{R}^{\mathrm{T}} \approx \mathcal{F}(\mathrm{R}^{\mathrm{S}}) \quad (8)$$

where $R^S = \log(\mathrm{S}^{spec}) - \log(A^{spec} \cdot H)$ stands for the source residual and $R^T = \log(\mathrm{T}^{spec}) - \log(B^{spec} \cdot H)$ stands for the target residual. Using the paired source-target residuals, the mapping function $\mathcal{F}$ can then be estimated by kernel partial least squares (KPLS) regression. Finally, the converted spectral envelope with residual compensation can be generated by:

$$\hat{\mathrm{T}}^{\mathrm{spec}} = \exp(\log(\mathrm{B}^{\mathrm{spec}} \cdot \mathrm{H}) + \mathcal{F}(\mathrm{R}^{\mathrm{s}})) \quad (9)$$

IV. Experiments

We propose a frame mapping method to deal with non-parallel voice conversion problem by using the existing exemplar-based conversion method. For non-parallel data, it normally needs more data to find sufficient number of frame mapping pairs. In the experiment, we will try to understand whether this proposed method works well on non-parallel data as compared to using parallel data when used in exemplar-based voice conversion method [17].

A. Experiment Settings

In this work, a deep neural network (DNN) based acoustic model is trained with data from several sources: King-ASR-136 [30] (about 43.8 hours), King-ASR-139 [31] (51.3 hours), Hub4 [32] (138.1 hours) and our in-house Singapore English data (about 104.5 hours).

The feature vector of the ASR system consists of 13 dimensional MFCC features in conjunction with 1 dimension of F0, and their derived deltas, acceleration and third-order deltas. The dimension of the feature vector is 56. The ASR system is trained using the Kaldi toolkit [33] first; a baseline acoustic model is trained with Maximum Mutual Information (MMI) criterion. Then DNN training is performed using the state level alignment obtained from the MMI model. There are 5 hidden layers in the DNN models. There are 156 context dependent phones and 5723 tied states. The frame interval is set to be 0.01 second.

After feeding the speech data to the DNN recognizer, we obtained a vector with a dimension size 5723. Then a dimension reduction process was done to reduce the dimension to 200, which covers 95% of the variance.

The CMU ARCTIC databases [34] were used in our experiments to evaluate the performance of the proposed method. The conversions of four different source-target pairs were conducted using two male speakers (M1: bdl, M2: rms) and two female speakers (F1: clb, F2: slt). The four source-target pairs are male to male (M1 to M2), male to female (M1 to F2), female to female (F1 to F2), and female to male (F1 to M2). For each conversion with different source-target pair, 100 utterances were selected for the model training and 20 utterances were selected as the test set for the objective evaluation. For the subjective evaluation, 20 utterances were selected from the test sets of the different source-target pairs.

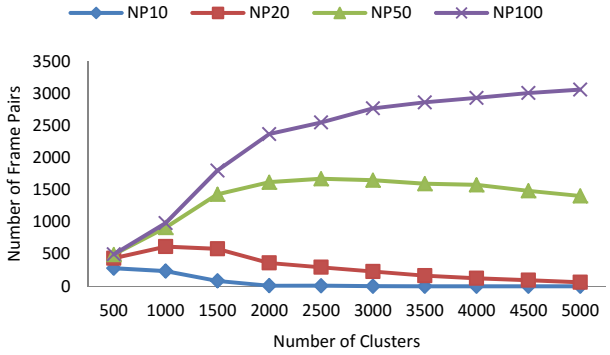

Fig 4: Number of matched frames vs the number of clusters defined for cases of 10, 20, 50 and 100 non-parallel sentences from source speaker M1 and target speaker M2.

B. *Clustering of Frames*

To set up the mapping from frames of source speaker to the target speaker, we need to group similar frames from both source and target speakers. It would be easy to get enough matched pairs from the non-parallel data from two speakers if we have very large speech data set from both speakers. However, in real applications, it is better to have less data from each speaker so that the technology can be used for more general cases. Therefore, we first test the clustering process to decide how many sentences from each speaker are enough for our further experiments.

We selected 10, 20, 50 and 100 utterances from both source speaker M1 and the same number of utterances from target speaker M2. We changed the number of clusters from 500 to 5000 with an interval of 500. For each setting, we generated the matched frame pairs. Fig. 4 shows the number of matched frame pairs under different cluster settings for the four cases. For the case of 10 and 20 sentences, we noticed that the numbers of matched pairs are less than 1000 even if we increase the number of clusters. This is due to the limited number of frames from both speakers belonging to the same phonetic units available in the data. For the case of 50 sentences, we can achieve about 1500 pairs. For the case of 100 sentences, we can achieve more than 2500 pairs. Based on previous experience in parallel voice conversion, the exemplar-based methods need at least 2500 pairs to achieve good results. So we decide to use 100 sentences from both source and target speakers in the following experiments.

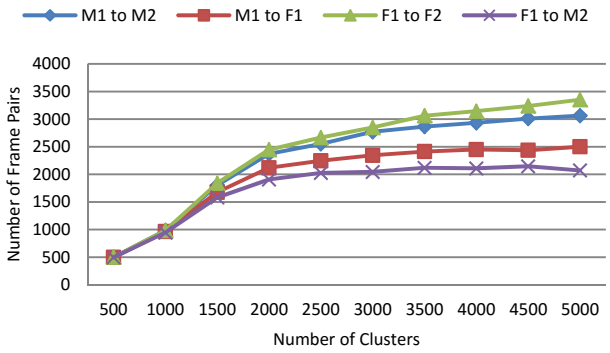

Fig 5: Number of matched frames vs the number of clusters defined for cases of using 100 non-parallel sentences as training data for conversion of speaker pairs M1 to M2, M1 to F1, F2 to F2, and F1 to M2.

Based on 100 utterances from both source and target speakers, we worked on 4 speaker pairs, M1 to M2, M1 to F1, F1 to F2, and F1 to M2. We tested the setting of the number of clusters from 500 to 5000 with an interval of 500. The number of matched pairs vs the number of clusters for the four cases is shown in Fig. 5. We can see from the figure, if the cluster number increased to 2500, the number of matched pairs exceeds 2000 for all the four cases. It increases slowly when the number of cluster is more than 3000. As we are targeting to have about 2500 pairs in our further testing, we decided to use the setting of 3500 clusters. Under this setting, the numbers of matched pairs are 2865, 2414, 3063, and 2119 for the four cases respectively. We have conducted our conversion experiment based on these settings.

C. *Conversion Experiments*

Using the same exemplar-based method, we shall compare the conversion results on two frame mapping methods: (1) The proposed method using non-parallel data with DNN-HMM recognizer mapping method. (2) baseline method using same amount of parallel data with DTW alignment mapping method.

In the proposed method, 100 utterances from source speaker and another 100 utterances with different content from target speaker were used for conversion training. The proposed DNN-HMM recognizer and clustering approach were used to obtain the mapping frame pairs. For the four speaker pairs in the conversions, the numbers of matched pairs were from 2000 to 3000.

In the baseline method, 100 pairs of parallel speech utterances from source and target speakers were used for training. The parallel speaker utterances were aligned with DTW method. Kindly take note that, although there were more matching pairs available in parallel data, only 3000 frame pairs were randomly selected from the matched pairs for the training due to the computation limit of the exemplar-based method.

Table I. Comparison of Log Spectral Distortion (LSD) Ratio of Baseline and Proposed Methods

LSD ratio	Parallel with DTW method (Baseline)	Non-parallel with DNN recognizer method (Proposed)
M1 to M2	0.5325	0.5360
M1 to F1	0.5311	0.5616
F1 to F2	0.6031	0.5830
F1 to M2	0.5511	0.6124
Average	0.5545	0.5733

D. *Objective Evaluation*

In this paper, the log spectral distortion between two spectral envelopes was used in the objective evaluation. The distortion of the t th frame can be calculate as:

$$\mathrm{d}\left(S_x(t), \mathrm{S_y}(t)\right) = \frac{1}{M}\sum_{m=1}^{M}(10\,log_{10}\,x_m(t) - 10log_{10}\;y_m(t))^2 \qquad (10)$$

where $\{x_m(t)\}$, $\{y_m(t)\}$ are the amplitudes sampled from the spectral envelope S_x, $\mathrm{S_y}$ at M uniformly-spaced frequency bins. A distortion ratio between converted-to-target distortion and the source-to-target distortion could be defined as:

$$\text{LSD ratio} = \frac{\sum_{t=1}^{T} d(S_x(t),S_y(t))}{\sum_{t=1}^{T} d(S_{\hat{y}}(t),S_y(t))} \times 100\% \qquad (11)$$

where $\{S_x(t)\}$, $\{S_y(t)\}$ and $\{S_{\hat{y}}(t)\}$ denote the source spectral envelope, target spectral envelope and the converted spectral envelope for frame t respectively. A smaller LSD ratio indicates better performance of conversion.

Table I presents the LSD ratio of the parallel method and the proposed non-parallel method using exemplar-based voice conversion on voiced frames. It can be found that the average LSD ratio of the proposed non-parallel method increased only a little compared to the parallel method. This shows that the proposed method using non-parallel data can achieve similar quality based on the objective measure

E. Subjective Evaluation

A subjective evaluation was conducted to assess both of the speech quality and speaker similarity of the proposed method. The parallel method of the exemplar-based voice conversion was used here as the baseline. 10 listeners took part in this evaluation. 20 pairs of converted utterances (5 from each case) were played to the listeners. Each pair consists of one utterance generated by the baseline and proposed method respectively. The order of appearance was random. The mean opinion score (MOS) was used here as the measurement (5=excellent, 4=good, 3=fair, 2=poor, 1=bad).

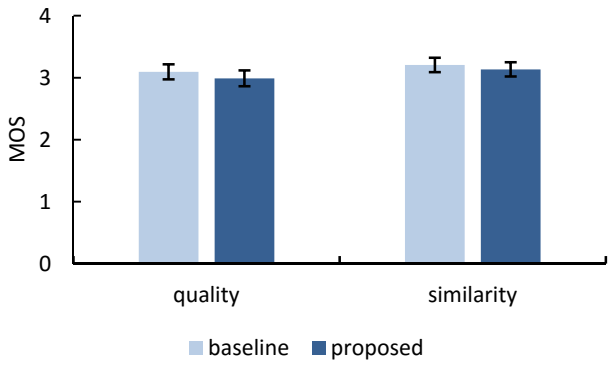

Fig 6: Mean opinion score (MOS) of speech quality and speaker similarity with 95% confidence intervals.

Fig. 6 shows you the MOS of the speech quality and the speaker similarity. It can also be found that the proposed method is comparable with the baseline in both quality and similarity.

Based on the results from both objective and subjective evaluations, we can see that the proposed frame mapping method for non-parallel data, when sufficient data are available, can generate voice with similar quality as DTW mapping method with parallel data.

V. CONCLUSIONS

In this paper, we proposed to use a DNN-HMM speech recognizer to recognize the frames of speech utterances. Based on the pseudo likelihood vector of DNN output, frames from both source and target speakers are clustered with the k-mean method. Frame mapping from source to target is then established based on the clustering result. Experiments show that the proposed method can generate similar conversion results as parallel voice conversion if there is sufficient non-parallel data available from both speakers.

REFERENCES

[1] H. Kuwabara and Y. Sagisaka, "Acoustic characteristics of speaker individuality: Control and conversion," *Speech Communication*, vol. 16, no. 2, pp. 165–173, 1995.

[2] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP),* New York, Apr. 1988, pp. 655–658.

[3] L. M. Arslan and D. Talkin, "Speaker transformation using sentence HMM based alignments and detailed prosody modification," in *Proc. IEEE Int. Conf Acoustics, Speech and Signal Processing (ICASSP),* Seattle, WA, May 1998, pp. 289–292.

[4] L. M. Arslan, "Speaker transformation algorithm using segmental codebooks (STASC)," *Speech Commun.,* vol. 28, pp. 211–226, 1999.

[5] G. Baudoin and Y. Stylianou, "On the transformation of the speech spectrum for voice conversion," in *IEEE Proc. Int. Conf. Spoken Language Processing (ICSLP)*, Philadephia, PA, Oct. 1996, pp. 1405–1408.

[6] 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, Mar. 1998.

[7] A. Kain and M.W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, Seattle, WA, May 1998, pp. 285–289.

[8] A. Kain and M. W. Macon, "Design and evaluation of a voice conversion algorithm based on spectral envelope mapping and residual prediction," in *Proc. ICASSP, IEEE*, 2001, vol. 2, pp. 813–816.

[9] T. Toda, H. Saruwatari, and K. Shikano, "Voice conversion algorithm based on gaussian mixture model with dynamic frequency warping of straight spectrum," in *Proc.IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2001, vol. 2, pp. 841–844.

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

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

[12] S. Desai, E. Raghavendra, B. Yegnanarayana, A. Black, and K. Prahallad, "Voice conversion using artificial neural networks," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2009.

[13] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, "Voice conversion using partial least squares regression," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 912–921, 2010.

[14] D. Erro, A. Moreno, and A. Bonafonte, "Voice conversion based on weighted frequency warping," *IEEE Trans. Audio, Speech and Language Processing*, vol. 18, no. 5, pp. 922–931, 2010.

[15] R. Takashima, T. Takiguchi, and Y. Ariki, "Exemplar based voice conversion in noisy environment," in *Spoken Language Technology workshop (SLT)*, 2012, pp. 313–317.

[16] Z. Wu, T. Virtanen, T. Kinnunen, E. S. Chng, and H. Li, "Exemplar-based voice conversion using non-negative spectrogram deconvolution," in *8th ISCA Speech Synthesis Workshop*, 2013.
[17] Z. Wu, T. Virtanen, E. S. Chng, and H. Li, "Exemplar based sparse representation with residual compensation for voice conversion," *IEEE Transactions on Speech and Audio Processing*, vol. 22, no. 10, pp. 1506–1521, 2014.
[18] A. Mouchtaris, J. Van der Spiegel, and P. Mueller, "Non-parallel training for voice conversion by maximum likelihood constrained adaptation," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2004, vol. 1, pp. I–1.
[19] H. Ye and S. Young, "Voice conversion for unknown speakers," in *Proc. Int. Conf. Spoken Lang. Process.*, 2004, pp. 1161–1164.
[20] T. Toda, Y. Ohtani, and K. Shikano, "Eigenvoice conversion based on Gaussian mixture model," in *Proc. ICSLP* 2006, pp. 2446–2449.
[21] A. Mouchtaris, J. Van der Spiegel, and P. Mueller, "Non-parallel training for voice conversion based on a parameter adaptation approach," *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 14, no. 3, pp. 952–963, 2006.
[22] C. H. Lee and C. H. Wu, "MAP-based adaptation for speech conversion using adaptation data selection and non-parallel training," in *Proc. Int. Conf. Spoken Lang. Process.*, 2006, pp. 2446–2449.
[23] P. Song, W. Zheng, and L. Zhao, "Non-parallel training for voice conversion based on adaptation method," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing(ICASSP)*, 2013.
[24] D. Sündermann, A. Bonafonte, H. Ney, and H. Höge, "A first step towards text-independent voice conversion," in *Proc. Int. Conf. Spoken Lang. Process.*, 2004, pp. 1173–1176.
[25] M. Zhang, J. Tao, J. Tian, and X. Wang, "Text-independent voice conversion based on state mapped codebook," in Proc. ICASSP, IEEE, 2008, pp. 4605–4608.
[26] D. Sündermann, H. Höge, A. Bonafonte, H. Ney, A. W. Black, and S. Narayanan, "Text-independent voice conversion based on unit selection," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2006, vol. 1, pp. 81–84.
[27] D. Erro, A. Moreno, and A. Bonafonte, "INCA algorithm for training voice conversion systems from non-parallel corpora," *IEEE Trans. Audio, Speech and Language Processing*, vol. 18, no. 5, pp. 944–953, 2010.
[28] D. D. Lee, H. S. Seung. Algorithms for non-negative matrix factorization[C], *Advances in neural information processing systems*. 2001: 556-562.
[29] H. Ming, D. Huang, L. Xie and S. Zhang, M. Dong and H. Li, "Fundamental Frequency Modeling Using Wavelets for Emotional Voice Conversion", *the workshop on Affective Social Multimedia Computing 2015*, Sep 2015.
[30] http://www.speechocean.com/en-ASR-Corpora/598.html
[31] http://www.speechocean.com/en-ASR-Corpora/607.html
[32] https://catalog.ldc.upenn.edu/LDC2002S11
[33] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, et al. 2011. The kaldi speech recognition toolkit. In *Proc. ASRU*, pages 1–4.
[34] John Kominek, Alan W Black, The CMU Arctic Speech Databases, *5th ISCA Speech Synthesis Workshop*, 2006