INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Parallel-data-free Many-to-many Voice Conversion based on DNN Integrated with Eigenspace Using a Non-parallel Speech Corpus

Tetsuya Hashimoto, Hidetsugu Uchida, Daisuke Saito, Nobuaki Minematsu

Graduate School of Engineering, The University of Tokyo, Japan
{hashib,uchida,dsk_saito,mine}@gavo.t.u-tokyo.ac.jp

Abstract

This paper proposes a novel approach to parallel-data-free and many-to-many voice conversion (VC). As 1-to-1 conversion has less flexibility, researchers focus on many-to-many conversion, where speaker identity is often represented using speaker space bases. In this case, utterances of the same sentences have to be collected from many speakers. This study aims at overcoming this constraint to realize a parallel-data-free and many-to-many conversion. This is made possible by integrating deep neural networks (DNNs) with eigenspace using a non-parallel speech corpus. In our previous study, many-to-many conversion was implemented using DNN, whose training was assisted by EVGMM conversion. By realizing the function of EVGMM equivalently by constructing eigenspace with a non-paralell speech corpus, the desired conversion is made possible. A key technique here is to estimate covariance terms without given parallel data between source and target speakers. Experiments show that objective assessment scores are comparable to those of the baseline system trained with parallel data.

Index Terms: parallel-data-free, many-to-many voice conversion, DNN, EVGMM, eigenspace

1. Introduction

Voice conversion (VC) is a technique to modify an input utterance of a speaker so that it sounds as if it is spoken by another speaker while its linguistic content is preserved. This technique has been applied to many applications, such as postprocessing of text-to-speech (TTS) conversion [1], speech enhancement [2], and so on.

In VC studies, statistical approaches have been often used for mapping features of a source speaker to those of a target one. Recently, approaches based on Gaussian mixture models (GMM) or neural networks (NN) have been widely investigated [3, 4]. To construct a conversion model on these approaches, a parallel speech corpus, which consists of the same sentences read by the source and target speakers, is required. However, the constructed model can be used only to modify the speaker identity of the source speaker's arbitrary utterances to that of the target speaker.

As 1-to-1 conversion has less flexibility, researchers focus on many-to-many conversion, where speaker identity is often represented using speaker space bases. For this purpose, approaches which apply prior knowledge from a large amount of pre-stored data, such as eigenvoice conversion (EVC), tensor-based EVC, have been investigated [6, 7, 8, 9]. EVC and tensor-based EVC achieve some improvements of its ability to controll speaker identities by using the pre-stored data. Also in VC based on DNN, using the pre-stored data for training achieves some performance improvements of conversion accuracy and generalization ability [10].

To use the prior knowledge, in our previous study, we implemented an architecture which consists of multiple DNNs to convert input features of a speaker into their eigenspace components [11]. The previous method realized many-to-many VC by an unsupervised speaker adaptation. However, a large amount of pre-stored parallel corpus were used at the stage of constructing EVGMM in our previous study. Then, in this paper, we improve the previous approach to avoid using parallel data in all the processes. A key technique here is to estimate covariance terms without given parallel data between source and target speakers. Finally, the desired conversion is made possible by realizing the function of EVGMM equivalently by constructing eigenspace with a non-parallel speech corpus. As opposed to some approaches that achieve many-to-many conversion using no parallel speech corpus [12, 13], our approach does not require any training data of source speakers.

The remainder of this paper is organized as follows. Section 2 describes EVGMM. Then, section 3 shows our proposed method using multiple DNNs and EVGMM using no parallel speech corpus. In section 4, experimental evaluation about many-to-many VC is described. Finally, section 5 concludes this paper.

2. Eigenvoice Gaussian mixture models (EVGMM)

In this section, EVGMM with a non-paralell speech corpus is described. Let M and s be the number of mixture components and the index of a speaker, respectively. A distribution of feature vectors of the s-th source speaker $\boldsymbol{X}^{(s)}$ is modeled by the EVGMM in the following formula:

$$P(\boldsymbol{X}^{(s)}|\boldsymbol{\lambda}^{(\mathrm{EV})}, \boldsymbol{w}^{(s)}) = \sum_{m=1}^{M} \alpha_m \mathcal{N}(\boldsymbol{X}_t^{(s)}; \boldsymbol{\mu}_m^{(\mathrm{X})}(\boldsymbol{w}^{(s)}), \boldsymbol{\Sigma}_m^{(\mathrm{X})}) \quad (1)$$

$$\boldsymbol{\mu}_m^{(\mathrm{X})}(\boldsymbol{w}^{(s)}) = \boldsymbol{B}_m \boldsymbol{w}^{(s)} + \boldsymbol{b}_m^{(0)} \quad (2)$$

α_m means the weight for the m-th component. $\boldsymbol{\lambda}^{(\mathrm{EV})}$ denotes the parameters of EVGMM which are independent of the source speakers.

In EVGMM, S pre-stored speakers are used to derive K base speakers ($K < S$). By using their linear combination, the mean of source speaker s can be represented in Equation (2), where K-dimensional weight vector $\boldsymbol{w}^{(s)}$ is estimated for s.

The speaker space is represented by K eigenspace supervectors $\boldsymbol{B} = [\boldsymbol{B}_1^\top, \boldsymbol{B}_2^\top, \ldots, \boldsymbol{B}_M^\top]^\top \in \mathcal{R}^{DM \times K}$ and bias supervector $\boldsymbol{b} = [\boldsymbol{b}_1^{(0)\top}, \boldsymbol{b}_2^{(0)\top}, \ldots, \boldsymbol{b}_M^{(0)\top}]^\top \in \mathcal{R}^{DM \times 1}$. $\boldsymbol{B}$ can be considered as eigenspace bases.

In the training step, first, the speaker-independent (SI) GMM is trained using utterances of all pre-stored speakers. Then, using those of each pre-stored speaker a speaker-dependent (SD) GMM for him/her is trained by adapting the

http://dx.doi.org/10.21437/Interspeech.2017-961

mean vectors of the SI-GMM. By concatenating the resulting mean vectors, that speaker's supervector is formed. After the above process is run for all the S pre-stored speakers, we can obtain S supervectors, on which PCA is applied to derive K base supervectors.

Finally, bias supervector $\boldsymbol{b}$, eigenspace supervectors $\boldsymbol{B}$ and weight for speaker s, $\boldsymbol{w}^{(s)}$ are determined. If one wants to use EVGMM to represent a new speaker, weight vector $\boldsymbol{w}$ has to be estimated adequately for that speaker. This estimation can be done based on the maximum likelihood criterion. This process is carried out in an unsupervised manner, and it can be realized even with a small amount of data.

3. Construction of speaker space based on DNNs for parallel-data-free VC

3.1. Architecture

In EVGMM, as shown in Equation (1), a linear combination of K bases is used to represent the mean vector of the feature distribution of a speaker. Based on this, as is discussed in the following section, the EVGMM-based conversion formula from a source speaker to a target speaker is obtained. Further, by using one-hot coding for weight, it is possible to realize a converter from a source speaker to a specific base speaker. Using this converter, without explicit parallel data given, we can prepare pseudo parallel data from an arbitrary source speaker to each of the base speakers. Using this, DNN-based conversion process to each base speaker can be trained. Once the DNNs for conversion from a speaker to the base speakers are trained, conversion from that speaker to any target speaker can be built. For this, adequate weights for any target speaker are needed and they are estimated using the trained DNNs above, which will be explained shortly and in more detail mathematically.

3.2. Pseudo parallel data preparation based on EVGMM

In this section, preparation of parallel data using EVGMM is described in detail [11]. In conventional voice conversion methods using EVGMM, the joint EVGMM which models the joint probability density $P(\boldsymbol{Z}_t \mid \boldsymbol{\lambda}^{(\mathrm{EV})}, \boldsymbol{w}^{(s)})$, where $\boldsymbol{Z}_t = [\boldsymbol{X}_t^\top, \boldsymbol{Y}_t^\top]^\top$, is estimated using parallel corpora between source and target features. Then, the conversion from the source speaker $\boldsymbol{X}_t$ to the target speaker $\boldsymbol{Y}_t$ is denoted by Equation (3);

$$\mathrm{F}(\boldsymbol{X}_t) = \sum_{m=1}^{M} \gamma_{m,t}(\boldsymbol{B}_m \boldsymbol{w}^{(\mathrm{Y})} + \boldsymbol{b}_m^{(0)} + \boldsymbol{A}_m(\boldsymbol{X}_t - \boldsymbol{\mu}_m^{(\mathrm{X})})). \quad (3)$$

$\gamma_{m,t}$ is a posterior probability of the m-th component given an input feature, and $\boldsymbol{A}_m$ is calculated from the variance and covariance matrix of $\boldsymbol{Z}_t$.

$$\gamma_{m,t} = P(m \mid \boldsymbol{X}_t, \lambda^{(\mathrm{EV})}), \quad (4)$$

$$\boldsymbol{A}_m = \boldsymbol{\Sigma}_m^{(\mathrm{YX})} \boldsymbol{\Sigma}_m^{(\mathrm{XX})-1}, \quad (5)$$

Equation (3) can be divided into a target-dependent part and a source-dependent part.

$$\mathrm{F}(\boldsymbol{X}_t, \boldsymbol{\lambda}^{(\mathrm{EV})}) = \mathrm{TD}(\boldsymbol{X}_t, \boldsymbol{\lambda}^{(\mathrm{EV})}) + \mathrm{SD}(\boldsymbol{X}_t, \boldsymbol{\lambda}^{(\mathrm{EV})}), \quad (6)$$

$$\mathrm{TD}(\boldsymbol{X}_t, \boldsymbol{\lambda}^{(\mathrm{EV})}) = \sum_{k=1}^{K} \boldsymbol{w}_k^{(\mathrm{Y})} \sum_{m=1}^{M} \gamma_{m,t} \boldsymbol{B}_{m,k},$$

$$\mathrm{SD}(\boldsymbol{X}_t, \boldsymbol{\lambda}^{(\mathrm{EV})}) = \sum_{m=1}^{M} \gamma_{m,t} \{\boldsymbol{b}_m^{(0)} + \boldsymbol{A}_m(\boldsymbol{X}_t - \boldsymbol{\mu}_m^{(\mathrm{X})})\},$$

where $\boldsymbol{w}_k^{(\mathrm{Y})}$ denotes the k-th dimensional element of $\boldsymbol{w}^{(\mathrm{Y})}$ and $\boldsymbol{B}_{m,k}$ denotes the k-th column vector of $\boldsymbol{B}_m$. In EVGMM, $\boldsymbol{b}_m^{(0)}$ means the average vector of all the pre-stored speakers, which corresponds to the m-th component. Hence, $\mathrm{SD}(\boldsymbol{X}_t, \boldsymbol{\lambda}^{(\mathrm{EV})})$ and $\mathrm{TD}(\boldsymbol{X}_t, \boldsymbol{\lambda}^{(\mathrm{EV})})$ are regarded as average term and residual term, respectively as the GMM-based VC formula is often expressed as addition of the two terms.

As described in Section 2, speaker individuality is controlled by the K-dimensional weight vector $\boldsymbol{w}^{(s)}$, each dimension of which corresponds to the k-th eigenspace basis (speaker). Thereby using a 1-of-K coded weight vector instead of $\boldsymbol{w}^{(\mathrm{Y})}$, a source speaker's feature can be converted to the feature in its corresponding eigenspace. The k-th eigenbase estimated from $\boldsymbol{X}_t$, $\boldsymbol{E}_t^k$, is represented by

$$\boldsymbol{E}_t^k = \sum_{m=1}^{M} \gamma_{m,t} \boldsymbol{B}_{m,k} \quad (k = 1, 2, ..., K), \quad (7)$$

$$\boldsymbol{E}_t^0 = \sum_{m=1}^{M} \gamma_{m,t} \{\boldsymbol{b}_m^{(0)} + \boldsymbol{A}_m(\boldsymbol{X}_t - \boldsymbol{\mu}_m^{(\mathrm{X})})\}. \quad (8)$$

$\boldsymbol{E}_t^0$ denotes a bias component corresponding to the averaged speaker. The paired data of $[\boldsymbol{X}_t, \boldsymbol{E}_t^k]$ are utilized as parallel data to train a DNN-based converter to the k-th eigenbase.

3.3. VC based on DNNs utilizing the eigenspace

Once the multiple DNNs are trained utilizing the parallel data prepared above, features of the source speaker can be converted to those of an arbitrary target speaker by adapting the weights. By denoting the DNN that is a converter to the k-th base of eigenspace as $\mathrm{DNN}^{(k)}$, the conversion formula of $\boldsymbol{X}_t$ is finally obtained as

$$f(\boldsymbol{X}_t) = \sum_{k=1}^{K} \boldsymbol{w}_k^{(s)} \mathrm{DNN}^{(k)}(\boldsymbol{X}_t) + \mathrm{DNN}^{(0)}(\boldsymbol{X}_t), \quad (9)$$

where $\mathrm{DNN}^{(0)}$ converts the input feature to the bias feature.

Weights for a specific target speaker can be calculated by converting that speaker to the same speaker. The above framework decomposes an input speaker into K base speakers and the input speaker is represented by a weighted sum of the base speakers. Any target speaker is also represented by the weighted sum. If a target speaker is used as input and output simultaneously, like auto-encoder, the weights to generate that target speaker can be estimated.

3.4. Parallel-data-free covariance estimation

In the above section, a parallel corpus among a large number of speakers is needed to train EVGMM, where that corpus is required to estimate the covariance term of $\Sigma_m^{(YX)}$ in Equation (3). In this section, we propose a method to overcome this limitation, in other words, a method to estimate the covariance term of EVGMM without a parallel corpus explicitly given. Even in this condition, by using Equation (3), we can generate automatically a parallel corpus between any pre-stored speaker and the average speaker. By using the resulting parallel corpora, we can obtain the covariance term between pre-stored speakers and the average speaker. This term is substituted for a real one. Mathematical and algorithmic details are explained as follows. The joint probability density of a pre-stored speaker's feature $\boldsymbol{X}_t^{(s)}$ and the averaged speaker's feature $\boldsymbol{M}_t$ is modeled by GMM. The difference between the ordinary GMM-based conversion

and the conversion discussed here, is that the target features, $\boldsymbol{M}_t$, are updated through training GMM. The update algorithm is as follows:

$$P(\boldsymbol{X}_t^{(s)}, \boldsymbol{M}_t \mid \boldsymbol{\lambda}^{(\mathrm{EV})}) = \sum_{m=1}^{M} \alpha_m \mathcal{N}([\boldsymbol{X}_t^{(s)\top}, \boldsymbol{M}_t^{\top}]^{\top}; \mu_{\mathbf{m}}^{(\mathrm{Z}')}(\boldsymbol{w}^{(\mathbf{s})}), \boldsymbol{\Sigma}_{\mathbf{m}}^{(\mathrm{Z}')}) \quad (10)$$

$$\mu_{\mathbf{m}}^{(\mathrm{Z}')}(\boldsymbol{w}^{(\mathbf{s})}) = \begin{bmatrix} \boldsymbol{B}_m \boldsymbol{w}^{(s)} + \boldsymbol{b}_m^{(0)} \\ \boldsymbol{b}_m^{(0)} \end{bmatrix}$$

$$\boldsymbol{\Sigma}_m^{(\mathrm{Z}')} = \begin{bmatrix} \boldsymbol{\Sigma}_m^{(\mathrm{XX})} & \boldsymbol{\Sigma}_m^{(\mathrm{XM})} \\ \boldsymbol{\Sigma}_m^{(\mathrm{MX})} & \boldsymbol{\Sigma}_m^{(\mathrm{MM})} \end{bmatrix}$$

·E step

$$\gamma_{m,t}^{(s)} = \frac{\mathcal{N}(\boldsymbol{Z'}_t^{(s)}; \mu_{\mathbf{m}}^{(\mathrm{Z})}(\boldsymbol{w}^{(\mathbf{s})}), \boldsymbol{\Sigma}_{\mathbf{m}}^{(\mathrm{Z})})}{\sum_{j=1}^{M} \alpha_j \mathcal{N}(\boldsymbol{Z'}_t^{(s)}; \mu_{\mathbf{j}}^{(\mathrm{Z})}(\boldsymbol{w}^{(\mathbf{s})}), \boldsymbol{\Sigma}_{\mathbf{j}}^{(\mathrm{Z})})} \quad (11)$$

·M step

$$\boldsymbol{\Sigma}_m^{(\mathrm{Z})} = \frac{1}{\gamma_m} \sum_{s=1}^{S} \sum_{t=1}^{T(s)} \left(\gamma_{m,t}^{(s)} (\boldsymbol{Z'}_t^{(s)} - \mu_{\mathbf{j}}^{(\mathrm{Z})})(\boldsymbol{Z'}_{\mathbf{t}}^{(\mathbf{s})} - \mu_{\mathbf{j}}^{(\mathrm{Z})})^{\top} \right) \quad (12)$$

$T(s)$ and γ_m mean the number of frames of the pre-stored speaker s and the sum of $\gamma_{m,t}^{(s)}$ in terms of t and s, respectively.

The updated feature $\boldsymbol{M'}_t$ in E step is obtained as

$$\boldsymbol{M'}_t = \sum_{m=1}^{M} P(\boldsymbol{Z}_t^{(s)} \mid \boldsymbol{\lambda}^{(\mathrm{EV})}) E_{m,t}^{(\mathrm{M|X})}. \quad (13)$$

This updated $\boldsymbol{M}_t'$ is used in the next iteration. $E_{m,t}^{(\mathrm{M|X})}$ means the conditional expectation of the feature of the average speaker given $\boldsymbol{X}_t$ as shown below.

$$E_{m,t}^{(\mathrm{M|X})} = \boldsymbol{b}_m^{(0)} + \boldsymbol{\Sigma}_m^{(\mathrm{MX})} \boldsymbol{\Sigma}_m^{(\mathrm{XX})-1} (\boldsymbol{X}_t^{(s)} - \mu_{\mathbf{m}}^{(\mathbf{s})}) \quad (14)$$

The variance matrix and the covariance matrices found in the M step are then used for the next E step, and the iteration process is repeated until convergence.

3.5. Entire process of the proposed VC

Finally, by integrating the estimated covariance matrix and the above framework represented in Section 3.2 and 3.3, we achieve the parallel-data-free and many-to-many VC which does not require any training data of source speakers. The entire process of the proposed VC becomes as follows.

1. Off-line processes
   1.1 Eigenspace construction:
   - Training SI-GMM using features of all pre-stored speakers $\boldsymbol{X}_t^{(s)}$ (these are not parallel data)
   - Training SD-GMM by adapting the mean vectors of the SI-GMM
   - Calculating EVGMM parameters $\boldsymbol{B}$, $\boldsymbol{b}$, $\boldsymbol{\Sigma}^{(\mathrm{X})}$ and speaker-dependent weights $\boldsymbol{w}^{(s)}$
   - Using EVGMM and pre-stored data, estimating covariance matrix $\boldsymbol{\Sigma}^{(\mathrm{Z})}$ without given parallel data as is discussed in the Section 3.4

   1.2 Pseudo parallel data preparation:
   - Calculating the paired data of $[\boldsymbol{X}_t^{(s)}, \boldsymbol{E}_t^{(k,s)}]$ by using Equation (7) and (8)

   1.3 Training DNNs utilizing the eigenspace:
   - Training multiple DNNs as converter to the k-th base of eigenspace by using paired data $[\boldsymbol{X}_t^{(s)}, \boldsymbol{E}_t^{(k,s)}]$

2. Conversion processes:
   - Estimating weights of a target speaker $\boldsymbol{w}^{(tar)}$ by using his features as input and output speakers simultaneously
   - The desired conversion formula $f(\cdot)^{(tar)}$ of new input feature $\boldsymbol{X}_t^{(new)}$ is finally obtained as follows.

$$f(\boldsymbol{X}_t^{(new)})^{(tar)} = \sum_{k=1}^{K} \boldsymbol{w}_k^{(tar)} \mathrm{DNN}^{(k)}(\boldsymbol{X}_t^{(new)}) + \mathrm{DNN}^{(0)}(\boldsymbol{X}_t^{(new)})$$

4. Experiments

4.1. Experimental condition

Experimental evaluations of many-to-many VC were carried out to investigate the effectiveness of the proposed method. Two types of the methods were compared: conventional GMM [14] (trained in a supervised manner) and the proposed parallel-data-free approach. As pre-stored speakers for training the SI-GMM and SD-GMMs to construct eigenspace, we used 96 speakers including 48 male and 48 female speakers from a speech corpus called JNAS (the Japanese Newspaper Article Sentences) [16]. The dataset includes 450 sentences and they are divided into 9 subsets. The utterances of each pre-stored speaker correspond to one of the subsets. For source speakers, a male speaker and a female speaker from ATR Japanese speech database B-set [15] were selected. In adaptation, namely, for target speakers, 4 speakers of 2 males and 2 females were used, and each of them uttered 32 sentences. Then, 21 utterances of the target speakers included in neither training nor adaptation data were used for evaluation. In the proposed method, each DNN which includes 5 layers with 256 units were constructed. Rectified linear units were used as activation functions of DNNs [19], and the DNNs were trained with dropout [18]. The number of eigenspace bases in the proposed method was fixed to 96. This is equivalent to the number of pre-stored speakers.

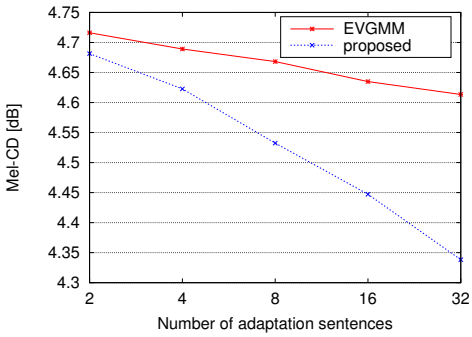

Figure 1: *Results of objective evaluations for 4 target speakers by mel-cepstral distortion (MCD).*

In the conventional GMM method, to achieve the best performance, the number of mixtures was varied from 1 to 128 and the optimal number was selected for each condition of the number of adaptation sentences. Note that the conventional GMM was trained in a supervised manner using parallel data.

We used 24-dimensional mel-cepstrum vectors for spectrum representation (D=24). These were derived by STRAIGHT analysis [17]. Aperiodic components, which are needed to generate mixed excitation in STRAIGHT, were not converted in this study, and they were fixed to -30dB at all the frequencies. The power coefficients and the fundamental frequencies were converted in a simple manner such that only the mean and the standard deviation were considered. The conversion performance was evaluated objectively using mel-cepstral distortion between the converted vectors and the vectors of the targets. Mel-cepstral distortion is denoted as follows,

$$\mathrm{MelCD[dB]} = \frac{10}{\ln 10}\sqrt{2\Sigma_{d=1}^{D}(mc_d - \bar{mc}_d)^2} \qquad (15)$$

where mc_d, $\bar{mc}_d$ are the converted feature vectors and those from the target speaker.

4.2. Objective evaluations of the proposed method about many-to-many VC

In this evaluation, the conversion performance for the 2 source speakers and the 4 target speakers was evaluated. Fig. 1 shows the results of the two methods in terms of average mel-cepstral distortion for the test data as a function of the number of adaptation, or training sentences (the conversion to 4 target speakers). Objective assessment scores of the proposed methods are comparable to those of the "GMM" trained with parallel data when using a small amount of adaptation data. This means that prior knowledge underlying the pre-stored data set is effectively utilized for improvement of the performance.

On the other hand, compared with "proposed" and "GMM", "GMM" outperforms "proposed" when using adaptation data more than or equal to 4. It might be due to the low complexity of the proposed method. While our proposed method updates only weights, the conventional GMM updates other parameters such as mean vectors and covariance matrices. Hence, to improve the conversion performance, it might be effective that target-dependent scalar weights are extended to weight vectors when using a large amount of adaptation data.

4.3. Subjective evaluations of the proposed method about many-to-many VC

A listening test was carried out to evaluate the naturalness and the speaker individuality of converted speech. The test was

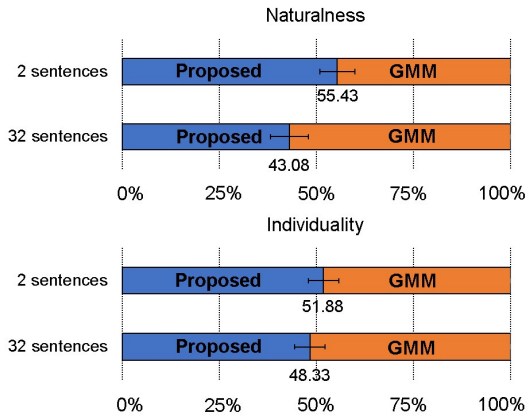

Figure 2: *Results of subjective evaluation. The blue area indicates the ratio of proposed method evaluated as more natural (or more similar to the reference speech) and the orange area indicates that of GMM. (Error bars represent the 95% confidence intervals.)*

conducted with 5 subjects who are native Japanese with normal hearing. To evaluate the naturalness, a paired comparison was carried out. In this test, pairs of two different types of the converted samples were presented to the subjects, and then they choosed which sample sounded more natural as native spoken Japanese with a confidence score.

To evaluate speaker individuality, an RAB preference test was performed. In this test, pairs of two different types of the samples were presented after presenting the reference sample of the target speech, and then they evaluated which sample sounded more similar to the reference speech as native spoken Japanese with confidence score. In the tests, for 2 source speakers and 4 target, 3 sentences per a pair of speakers were used. Then, the number of sample pairs evaluated by each subject was 48 in each test.

Fig. 2 shows the results of the two methods evaluated about the naturalness and individuality. In the tests, the number of adaptation, or training sentences were fixed to 2 or 32. When using 2 sentences, the "Proposed" slightly outperforms "GMM" in both naturalness and speaker individuality. When using 32 sentences, the "GMM" slightly outperforms "Proposed" in both naturalness and speaker individuality. Meanwhile, compared with "Naturalness" with "Individuality" scores, the difference of "Individuality" scores between "GMM" and "Proposed" is very small. This result shows that the performance of the proposed method is comparable to that of the "GMM" except in naturalness when using a large amount of adaptation data.

5. Conclusion

In this paper, we have proposed a new architecture for the parallel-data-free many-to-many voice conversion based on DNN with eigenspace using a non-parallel speech corpus. In proposed method, input features are decomposed into eigenspace base speakers and the target speaker feastures are represented as a weighted sum of the base speakers. Experiments show that objective assessment scores are comparable or slightly worse to those of the baseline system trained with parallel data. However, in individuality, subjective assessment scores are comparable to those of the baseline system trained with parallel data. For further improvements of the conversion performance, integration of our method with other effective methods to improve naturalness, such as post-filter should be verified.

6. References

[1] A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," ICASSP, vol. 1, pp. 285–288, 1998.

[2] L. Deng, A. Acero, L. Jiang, J. Droppo, and X. Huang, "High-performance robust speech recognition using stereo training data," ICASSP, pp. 301–304, 2001.

[3] D. Saito, H. Doi, N. Minematsu, and K. Hirose, "Application of matrix variate Gaussian mixuture model to statistical voice conversion," INTERSPEECH, pp. 2504–2508, 2014.

[4] S. Desai, E.V. Raghavendra, B. Yegnanarayana, A.W. Black, and K. Prahallad,"Voice conversion using artificial neural networks," ICASSP, pp. 3893–3896, 2009.

[5] C.H. Lee, and C.H. Wu, "Map-Based Adaptation for Speech Conversion Using Adaptation Data Selection and Non-Parallel Training," INTERSPEECH, pp. 2254–2257, 2006.

[6] T. Toda, Y. Ohtani, and K. Shikano, "EigenVoice Conversion Based on Gaussian Mixture Model," INTERSPEECH, pp. 2446–2449, 2006.

[7] T. Toda, Y. Ohtani, H. Saruwatari, and K. Shikano, "Adaptive training for voice conversion based on eigenvoices," IEICE TRANS. INF. &SYST., VOL.E93-D, NO.6, pp. 1589–1598, 2010.

[8] Y. Ohtani, T. Toda, H. Saruwatari, and K. Shikano, "Non-parallel training for many-to-many eigenvoice conversion," ICASSP, pp. 4822–4825, 2010.

[9] D. Saito, K. Yamamoto, N. Minematsu, and K. Hirose, "One-to-Many Voice Conversion Based on Tensor Representation of Speaker Space," INTERSPEECH, pp. 653–656, 2011.

[10] L.J. Liu, L.H. Chen, Z.H. Ling, and L.R. Dai, "Spectral Conversion Using Deep Neural Netwotks Trained With Multi-Source Speakers," ICASSP, pp. 4849–4853, 2015.

[11] T. Hashimoto, D. Saito, and N. Minematsu, "Arbitrary speaker conversion based on speaker space bases constructed by deep neural networks," APSIPA, 2016.

[12] T. Nakashika, T. Takiguchi and Y. Ariki, "PARALLEL-DATA-FREE, MANY-TO-MANY VOICE CONVERSION USING AN ADAPTIVE RESTRICTED BOLTZMANN MACHINE," MLSLP, 2015.

[13] Wu. J, Wu. Z, and Xie. L, "On the Use of I-vectors and Average Voice Model for Voice Conversion without Parallel Data," APSIPA, 2016.

[14] Y. Stylianou, O. Cappe and E. Moulines, "Continuous probabilistic transform for voice conversion", IEEE Trans. on SAP, 6, 2, pp. 131–142, 1998.

[15] A. Kurematsu, K. Takeda, Y. Sagisaka, S. Katagiri, H. Kuwabara, and K. Shikano, "ATR Japanese speechdatabase as a tool of speech recognition and synthesis," Speech Communication, vol. 9, pp. 357–363, 1990.

[16] "Jnas: Japanese newspaper article sentences," http://www.milab.is.tsukuba.ac.jp/jnas/instruct.html

[17] H. Kawahara, I. Masuda-Katsuse, and A.de Cheveigne, "Restructuring speech representations using a pitch-adaptive time frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," Speech Communication, vol. 27, pp. 187–207, 1999.

[18] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," arXiv preprint arXiv:1207.0580, 2012.

[19] Vinod Nair, and Geoffrey E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," Proceedings of the 27th International Conference on Machine Learning (ICML-10), 2010.