Fast Many-to-One Voice Conversion using Autoencoders

Yusuke Sekii[1], Ryohei Orihara[1], Keisuke Kojima[2], Yuichi Sei[1],
Yasuyuki Tahara[1] and Akihiko Ohsuga[1]
[1]*Graduate School of Information Systems, University of Electro-Communications, Chofu-city, Tokyo, Japan*
[2]*Solid Sphere, inc., Tokyo, Japan*

Keywords: Voice Conversion, Autoencoder, Deep Learning, Deep Neural Network, Spectral Envelope.

Abstract: Most of voice conversion (VC) methods were dealing with a one-to-one VC issue and there were few studies that tackled many-to-one / many-to-many cases. It is difficult to prepare the training data for an application with the methods because they require a lot of parallel data. Furthermore, the length of time required to convert a speech by Deep Neural Network (DNN) gets longer than pre-DNN methods because the DNN-based methods use complicated networks. In this study, we propose a VC method using autoencoders in order to reduce the amount of the training data and to shorten the converting time. In the method, higher-order features are extracted from acoustic features of source speakers by an autoencoder trained with source speakers' data. Then they are converted to higher-order features of a target speaker by DNN. The converted higher-order features are restored to the acoustic features by an autoencoder trained with data drawn from the target speaker. In the evaluation experiment, the proposed method outperforms the conventional VC methods that use Gaussian Mixture Models (GMM) and DNNs in both one-to-one conversion and many-to-one conversion with a small training set in terms of the conversion accuracy and the converting time.

1 INTRODUCTION

In recent years, voice conversion (VC), which is a technique used to change timbre features of a source speaker into those of a target speaker, has been actively studied. VC techniques can be applied to alleviate a sense of discomfort due to a change of voice actors or actresses in animated films, to create dubbed voice of movie in voice of the actor or actress themselves, and to assist a call by converting a hard-to-hear voice to an easy-to-hear voice in real time.

VC based on Gaussian Mixture Models (GMM) is a typical conventional VC approach (Stylianou et al., 1998; Toda et al., 2007). However, in recent years, it has been reported that VC approaches employing Deep Neural Networks (DNNs) outperform VC approaches based on GMM (Desai et al., 2009). It can be explained by a fact that the shape of the vocal tract is generally non-linear; VC methods using non-linear operations such as DNNs are more compatible with human speech than methods based on linear operations such as GMM (Nakashika et al., 2015). As the non-linear VC approaches, those employing restricted Boltzmann machines (RBMs) (Chen et al., 2013), deep belief networks (DBNs), which are extended versions of RBMs (Nakashika et al., 2013) and conditional restricted Boltzmann machines (CRBMs) (Wu et al., 2013) are proposed. Furthermore, it has been reported that the conversion accuracy can be improved by pre-training based on RBMs and autoencoders in VC methods using DNNs (Mohammadi and Kain, 2014; Liu et al., 2015).

Although a lot of VC research have used Mel-frequency cepstrum coefficients (MFCC) or Mel-cepstrum (MCEP) as acoustic features, it has been reported that VC methods converting spectral envelope are better than ones converting MFCC (Chen et al., 2013; Nguyen et al., 2016). In VC methods converting MFCC, the similarity in high-frequency range is inferior to ones converting spectral envelope because the information is lost in the former when MFCC is restored to spectral envelope. For this reason, it can be said that we should select the spectral envelope as acoustic features to be used for VC.

From the discussion above, it is best to choose a non-linear operation for the conversion method and spectral envelope as the acoustic feature in order to achieve highly accurate VC. However, the dimensionality of spectral envelope is large compared with MFCC, and it requires a lot of data to create a voice

Sekii Y., Orihara R., Kojima K., Sei Y., Tahara Y. and Ohsuga A.
Fast Many-to-One Voice Conversion using Autoencoders.
DOI: 10.5220/0006193301640174
In *Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017)*, pages 164-174
ISBN: 978-989-758-220-2

converter. In general, because parallel data with the same speech contents of a source speaker and a target speaker is required to create a voice converter, to prepare a lot of ones is expensive.

When the cost to collect data is high, a development of an application becomes difficult. It is necessary to solve this problem in order to develop various applications using VC techniques. Although a VC method using DNN converting spectral envelope is proposed (Xie et al., 2014), due to large dimensionality of log spectral envelope used as input of DNN, the structure of DNN becomes complicated, consequently, there are problems that a lot of training data are required and converting time becomes longer.

Although the aforementioned VC methods are one-to-one VC where a particular source speaker's voice is converted to a particular target speaker's, many-to-one VC methods, conversion from an arbitrary source speaker to a particular target speaker, and many-to-many VC methods, conversion from an arbitrary source speaker to an arbitrary target speaker, have been also proposed (Toda et al., 2006; Liu et al., 2015). Here, an arbitrary speaker is a speaker that has not been used as training data for creating voice converter. If the many-to-one VC becomes possible, the cost of a creating voice converter is reduced because it is not necessary to build a new voice converter for a new source speaker. However, creating a many-to-one voice converter requires more speech data than creating a one-to-one voice converter because a many-to-one voice converter needs to be trained by speech data of multiple speakers. Thus, the voice converter should be built by fewer data in order to realize many-to-one VC easily.

In this study, we aim to reduce the amount of the training data and to shorten the converting time for one-to-one and many-to-one VC by using autoencoders and relatively simple DNN.

In the proposed method, at first, autoencoders, trained with data of the source speaker and the target speaker respectively, are created, and higher-order features from each autoencoder are extracted. Then a DNN which converts the higher-order features of the source speaker into those of the target speaker is trained. The target speaker's higher-order features for a new source speaker's voice data are obtained by inputting higher-order features of the voice data into the DNN. The acoustic features are restored from the converted higher-order features by using the weight of the autoencoder of the target speaker. Finally, the converted voice is obtained from the acoustic features by speech synthesis.

The remainder of the paper is organized as follows: In Section 2, we describe related works which deal with VC methods using RBMs and autoencoders. In Section 3, we explain the overview and important aspects of the proposed method and the benefit of autoencoders. In Section 4, we present the evaluations where the proposed method was compared with the conventional methods in one-to-one VC and many-to-one VC settings. In Section 5, we conclude the paper and discuss future works.

2 RELATED WORKS

There are a lot of studies on VC. We describe VC methods using DNN in this section because they are actively proposed in recent years.

Nakashika et al. (Nakashika et al., 2015) have proposed a VC method using speaker-dependent CRBMs. A CRBM of a source speaker and one of a target speaker are trained, then the higher-order features obtained from the CRBM of the source speaker are converted into the higher-order features obtained from the CRBM of the target speaker by neural network (NN). The converted higher-order features are restored to the acoustic features by the inverse projection of the CRBM of the target speaker, and the speech signal is obtained. In the evaluation experiments, the proposed method outperforms conventional VC methods using GMM, RBM and recurrent neural network (RNN). The voice converter can be created without a large dataset and it is possible to realize VC in shorter time due to the usage of 24-dimensional MFCC as acoustic features.

Nguyen et al. (Nguyen et al., 2016) have proposed a speaker conversion method, which comprehensively converts spectral envelope, fundamental frequency (F0), intensity trajectory and phone duration. In spectral envelope conversion, which corresponds to VC, the method that employs autoencoders with weights using L1 norm constraint in pre-training is proposed. This method outperforms a VC method using DNN with randomly initialized weights. Although this can convert the spectral envelope with high accuracy, a large dataset is required to build a voice converter and its conversion time would be long because the method employs 512-dimensional log spectral envelope and a large NN which has three hidden layers where each hidden layer has 3000 nodes.

Mohammadi et al. (Mohammadi and Kain, 2014) have proposed a VC method using deep autoencoders. In this method, input features are compressed by deep autoencoders of source speaker and target speaker, and higher-order features are obtained. An artificial neural network (ANN) which converts the higher-order features of the source speaker into those of the

target speaker is trained. A DNN is trained by combining the deep autoencoders with the ANN, then the DNN is fine-tuned. This method outperforms conventional VC methods using GMM etc. with a small training set. This method enables a voice converter even with a smaller dataset and it is possible to shorten the converting time due to the usage of 24-dimensional MCEP as acoustic features.

Liu et al. (Liu et al., 2015) have proposed a speaker-independent VC method using a DNN. The spectral features of three concatenated frames are used as input and speech data of multi-source speakers are used in training. The proposed method generates a one-to-one speaker-dependent DNN based on weights initialized by a speaker-independent DNN. It outperforms a VC method using a DNN pre-trained by DBNs. In the evaluation experiment, the proposed method yields as high accuracy as conventional one-to-one VC methods using GMM and DNNs. This method also enables a voice converter even with a smaller dataset and it is possible to shorten the converting time due to the usage of 24-dimensional MCEP as acoustic features.

3 PROPOSED METHOD

3.1 The Overview of Proposed Method

The VC process of the proposed method is described below (Figure 1).

Step 1: Acoustic features, which are spectral envelope in this study, are extracted from a source speech.

Step 2: Higher-order features are extracted by an autoencoder trained by acoustic features of the source speaker.

Step 3: The higher-order features are converted into those of the target speaker by DNN.

Step 4: The converted higher-order features are restored to the target acoustic features by an autoencoder trained by the target speaker's data.

Step 5: The converted voice is created from the acoustic features by speech synthesis.

3.2 Autoencoder

A typical NN is a supervised learning technique, and they require a pair of input and output values. An autoencoder (Hinton and Salakhutdinov, 2006) is an unsupervised learning technique and is a NN equalizes the output values with its input values i.e. we only need the input values.

In a NN which has an input layer, a hidden layer and an output layer (Figure 2), an autoencoder is defined as follows:

$$\mathrm{h} = f(\mathrm{W}_1\mathrm{x} + \mathrm{b}_1), \tag{1}$$

$$\mathrm{y} = g(\mathrm{W}_2\mathrm{h} + \mathrm{b}_2), \tag{2}$$

where x is the input layer, h is the hidden layer, y is the output layer, W_1 and b_1 are the weight and the bias to convert x into h respectively, W_2 and b_2 are the weight and the bias to convert h into x respectively, and f and g are activation functions respectively. By using Equation (1) and (2), the equation converting the input x into the output y is described as follows:

$$\mathrm{y} = g(\mathrm{W}_2 f(\mathrm{W}_1\mathrm{x} + \mathrm{b}_1) + \mathrm{b}_2). \tag{3}$$

An autoencoder decides the weights $(\mathrm{W}_1, \mathrm{W}_2)$ and the biases $(\mathrm{b}_1, \mathrm{b}_2)$ as hyper parameters so that y becomes similar to x, i.e. the hyper parameters are decided to minimize the value of a loss function to measure the distance between y and x. Root mean square error (RMSE) is typically used as a loss function.

$$\mathrm{E} = ||\mathrm{x} - \mathrm{y}||^2 \tag{4}$$

An autoencoder can be used as a pre-training technique like a RBM. NNs, whose weights are initialized with the values obtained through an autoencoder training, yield better results by means of fine-tuning (Mohammadi and Kain, 2014). On the other hand, higher-order features can be seen as compressed ones of the input, if the autoencoder has a hidden layer which is smaller than the input layer. Thus the features of large dimensionality can be represented by those of small dimensionality.

3.3 Converting Acoustic Features

In this study, higher-order features extracted by an autoencoder are used. We aim to reduce the amount of the training data and to shorten the converting time by using higher-order features with smaller dimensionality.

In this study, the proposed structure to convert features is described in Figure 3. At first, autoencoders of each speaker and a DNN converting higher-order features are trained. The acoustic features of a source speaker (x) and those of a target speaker (x') are treated as input, then autoencoders of each speaker are trained. Higher-order features $(\mathrm{h}, \mathrm{h}')$ are extracted by each autoencoder. Then a DNN is created by using the higher-order feature (h) extracted by the autoencoder of the source speaker as input data and the

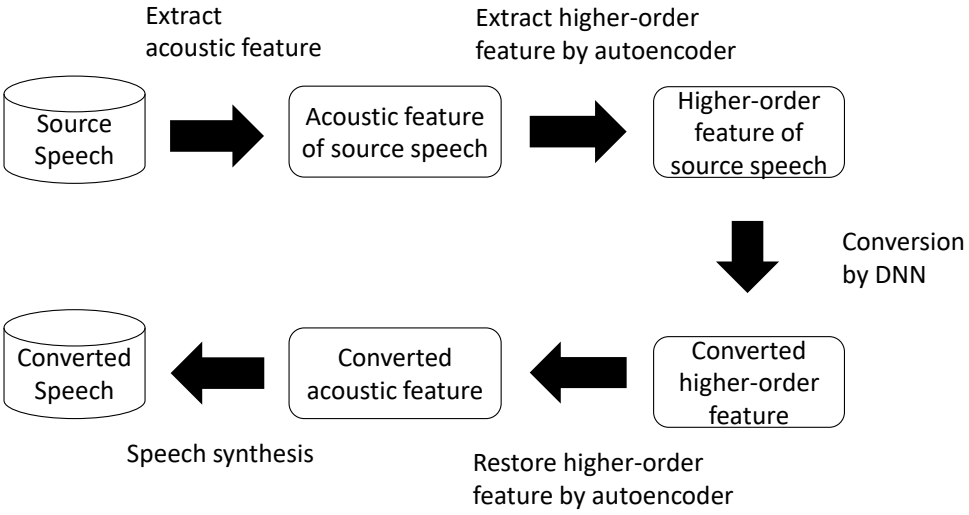

Figure 1: The process of the proposed VC.

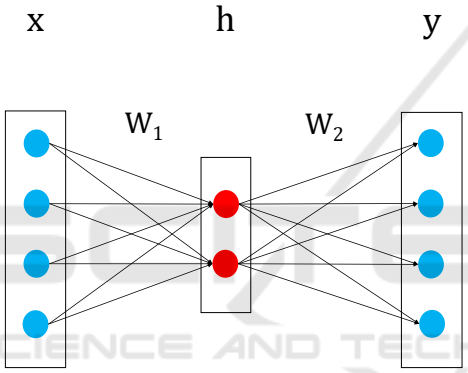

Figure 2: Autoencoder.

higher-order feature (h') extracted by the autoencoder of the target speaker as the ground truth of the conversion. Secondly, a voice converter is created by combining the autoencoders with the DNN (Figure 3 below). The higher-order features are extracted from the input acoustic features by using the weight (W_1) of the encoder part of the autoencoder of the source speaker. The extracted higher-order features are converted into target higher-order features by the DNN. The acoustic features are restored from the converted higher-order features (h'') by using the weight (W'_2) of the decoder part of the autoencoder of the target speaker, then the converted acoustic features (y'') are obtained. Acoustic features can be converted by a network, which consists of an encoder weight of a source speaker's autoencoder, a decoder weight of a target speaker's autoencoder and a higher-order feature conversion DNN.

In order to put many-to-one VC into practice, acoustic features of multiple source speakers as training data are required. An autoencoder is considered to give more generalized higher-order features by using training data consists of multiple source speakers. The DNN that converts the generalized higher-order features into the target higher-order features is expected to be able to convert an arbitrary speech not used as training data of the source speakers with a high degree of accuracy.

4 EXPERIMENTAL EVALUATION

4.1 Preliminary Experiment

In this preliminary experiment, we used a male speaker (YMGT) as a source speaker and a female speaker (RDY) as a target speaker from the speech database created by Solid Sphere, inc.[1].

4.1.1 Determining Appropriate Parameters

We conducted the preliminary experiment in order to identify optimal parameters of the proposed method and methods to be compared with it. In this experiment, we used 450 utterances as a training set and 50 utterances as a testing set. In the proposed method, 100-dimensional higher-order features are converted by various DNNs with different hyper parameters such as the number of hidden layers and the number of hidden nodes. We used 100 epochs for the autoencoder training and 30 epochs for the DNN training. In order to evaluate the quality of spectral conversion,

[1]It's a private speech database. It consists of four male speakers and six female speakers, and 500 utterances are recorded by each speaker.

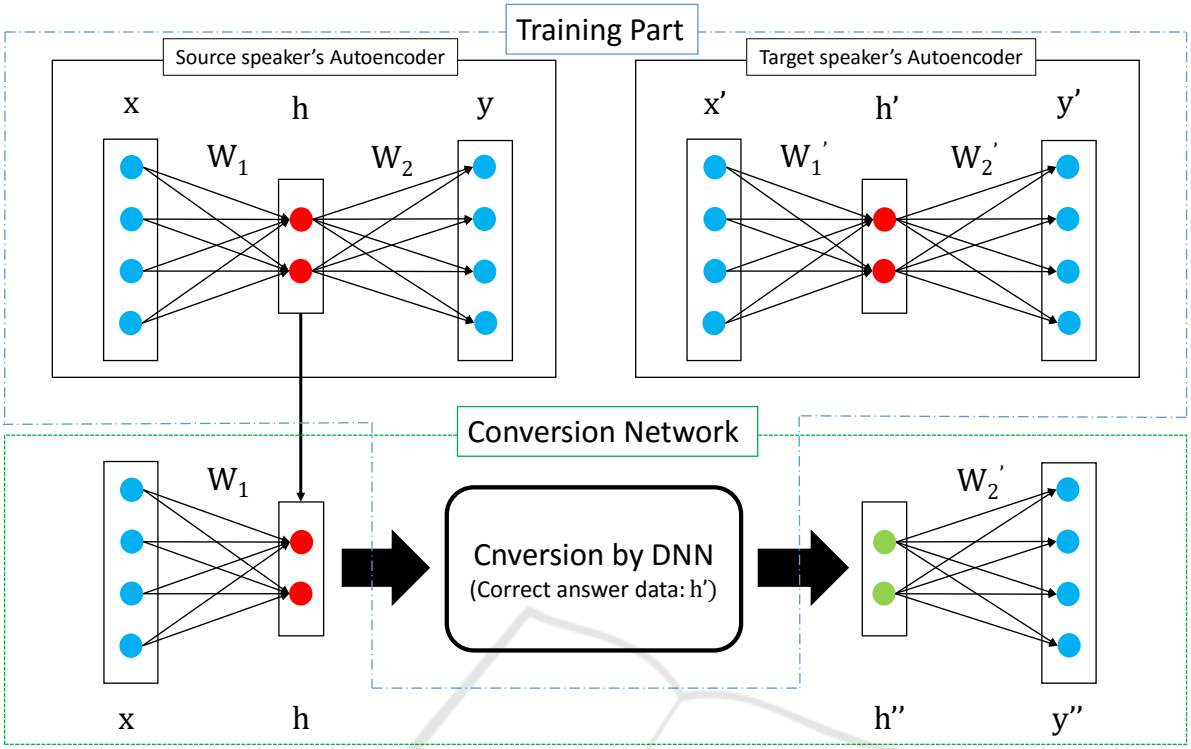

Figure 3: The overview of feature conversion by the proposed method.

log spectral distortion (LSD), which measures how close the converted spectrum becomes to the target spectrum, is employed. LSD is defined as follows:

$$\mathrm{LSD} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(10\log_{10}\frac{x_i}{y_i}\right)^2}, \qquad (5)$$

where x_i is the i-th converted spectrum, y_i is the i-th target spectrum and n is spectral dimensionality (513 in this work). The result of the spectral conversion for combinations of the hyper parameters is shown in Figure 4. From the result, we decided to use a DNN with three hidden layers where each layer has 500 nodes in the actual experiment. We also used a simpler DNN that uses 50-dimensional higher-order features as input in the experiment.

We chose Nguyen et al.'s work (Nguyen et al., 2016), which converts 513-dimensional log spectral envelope by DNN, as a method to be compared with ours, because spectral conversion accuracy of the method is the highest as far as we know. 513-dimensional log spectral envelope is converted by various DNNs with different hyper parameters such as the number of hidden layers and the number of hidden nodes. We used 30 epochs for DNN training. We used LSD in the evaluation of spectral conversion. The result of the spectral conversion for combinations of the hyper parameters is shown in Figure 5. As a result, we decided to use a DNN which has three hidden layers with 3000 nodes in each layer and a DNN which has three hidden layers with 100 nodes in each layer as VC methods to be compared with ours, because they are the most and second most accurate methods in the result respectively.

4.1.2 Effect of the Amount of Data

We studied how the accuracy of the spectral conversion is affected by the amount of training data. In this experiment, we used a proposed method which converts 50-dimensional higher-order features (AE50), a proposed method which converts 100-dimensional higher-order features (AE100) and two VC methods, which convert 513-dimensional spectral envelope by DNNs, to compared with ours (SPEC3000 and SPEC100). DNNs for AE50, AE100, SPEC3000 and SPEC100 have two hidden layers with 200 nodes, three hidden layers with 500 nodes, three hidden layers with 3000 nodes, and three hidden layers with 100 nodes respectively. We used 100 epochs to train autoencoders of AE50 and AE100, and 30 epochs to train DNNs of AE50, AE100, SPEC3000 and SPEC100. We used LSD as a measure to evaluate the accuracy of spectral conversion. The result of the spectral conversion against the amount of training data is shown in Figure 6. AE50 yielded

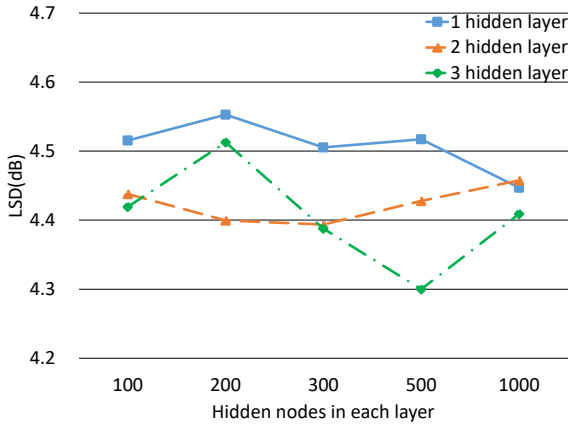

Figure 4: Change of LSD with the proposed method.

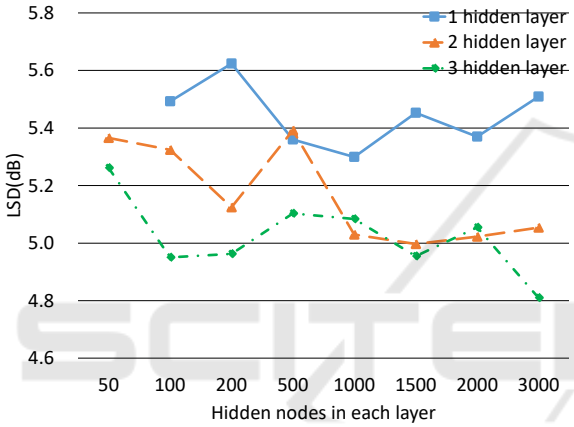

Figure 5: Change of LSD with a method to convert spectral envelope by DNN.

higher accuracy with a small amount of data. AE100 and SPEC3000 yielded higher accuracy with a large amount of data. As a result, one can hypothesize that a method using a simple DNN (such as AE50) yields higher accuracy with a small amount of data, and a method using a complicated DNN (such as AE100 and SPEC3000) yields higher accuracy with a large amount of data. The hypothesis implies that a method using a simple DNN is the best in case that a large training set is unavailable and converting time must be short because the method can generally convert spectra in a shorter time than a method using a complicated DNN. However, the method using a simpler DNN (SPEC100) yielded lower accuracy with a small amount of data and yielded higher accuracy with a large amount of data. In the actual experiment, we verified this hypothesis by using two data sets.

4.2 Experimental Setup

We conducted experiments with proposed methods and other VC methods in one-to-one VC and many-

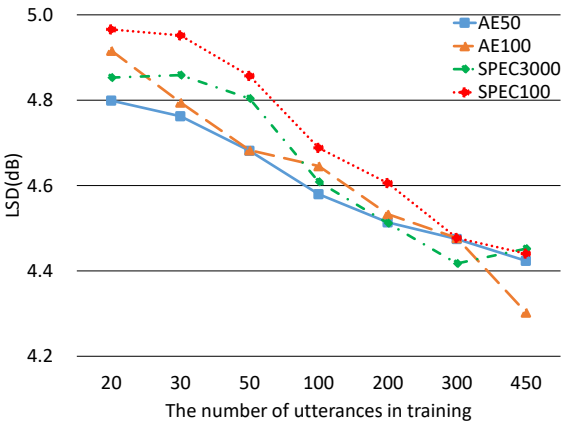

Figure 6: Change of LSD to variation of the number of training speakers.

to-one VC settings. We used the speech database created by Solid Sphere, inc. in the experiment. We prepared four pairs (YMGT to KJM, KJM to HM, TK to YMGT and HM to TK) consist of two male speakers (KJM and YMGT) and two female speakers (HM and TK) for the one-to-one VC. In the many-to-one VC experiment, we created eight voice converters for combinations of two target speakers and four conditions of training speakers, where the number of training speakers is two, four, six and eight respectively. The target speakers are a male speaker (KRT) and a female speaker (HM). In order to evaluate the converters, KRT to HM conversion and HM to KRT conversion were used. Note that none of them are included in the eight training speakers. In both experiments, we used 300 utterances as a large training set (large) and 20 utterances as a small training set (small). The number of training data is constant regardless of the number of training speakers. Furthermore, we used 50 utterances as a testing set. We employed parallel training data, which are created by aligning an utterance of a source speaker with one of a target speaker by dynamic time warping (DTW). The utterances are required to be the same content in both a source speaker and a target speaker.

In this experiment, we compared VC accuracy of two proposed methods based on various parameters with one of four conventional VC methods. We used the methods appeared in 4. 1. 2, namely, AE50, AE100, SPEC3000 and SPEC100, again. Additionally, we used a conventional method which employed a GMM (JDGMM) (Toda et al., 2007) and a method converting MFCC by a DNN (MFCC-DNN) (Desai et al., 2009). We employed 513-dimensional log spectral envelope with TANDEM-STRAIGHT (Kawahara et al., 2008) in AE50, AE100, SPEC3000 and SPEC100. In JDGMM and MFCC-DNN, we

employed 25-dimensional MFCC calculated from the spectral envelope. We used 64 Gaussian components to build the system in JDGMM. DNNs for MFCC-DNN, SPEC3000, SPEC100, AE50 and AE100 have two hidden layers with 50 nodes, three hidden layers with 3000 nodes, three hidden layers with 100 nodes, two hidden layers with 200 nodes, and three hidden layers with 500 nodes respectively. The activation function and the learning optimization algorithm of the autoencoders and the DNNs are ReLU (Nair and Hinton, 2010) and ADAM (Kingma and Ba, 2015) respectively. We used 100 epochs for the autoencoder training and 30 epochs for the DNN training with AE50 and AE100. Moreover, we used 200, 20 and 20 epochs for the DNN training with MFCC-DNN, SPEC3000 and SPEC100 respectively. These parameters were decided by the preliminary experiment and Desai et al.'s study (Desai et al., 2009).

As measures of the objective evaluation, we employed LSD and conversion time of acoustic features. As for the subjective evaluation, we employed mean opinion score (MOS). The MOS is a statistical measurement of voice quality based on human opinion of speech. It is expressed as a numerical value between 1 and 5, where 1 is the lowest voice quality, and 5 is the highest voice quality. Subjects consisting of nine men and women in their twenties listened to the target speech and converted speech, and assessed similarity (how well they can recognize the target speaker from the converted speech) and quality (how clear and natural the converted speech is). We transformed not only the spectral feature but also the fundamental frequency (F0), which is the feature of the voice pitch, for the converted speech. The conversion of F0 is described as follows:

$$\hat{y}_t = \frac{\sigma^{(y)}}{\sigma^{(x)}}(x_t - \mu^{(x)}) + \mu^{(y)}, \quad (6)$$

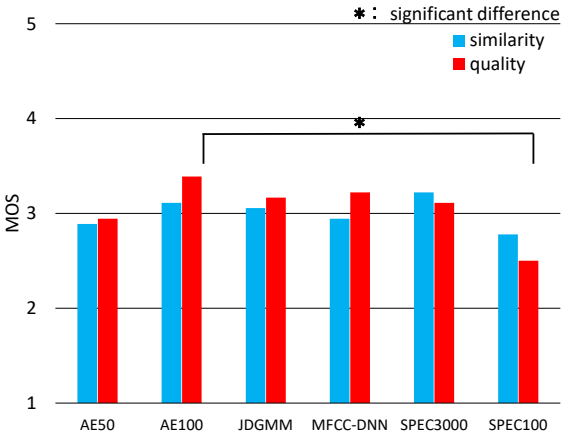

Figure 7: The result of the subjective evaluation in one-to-one VC with small training set.

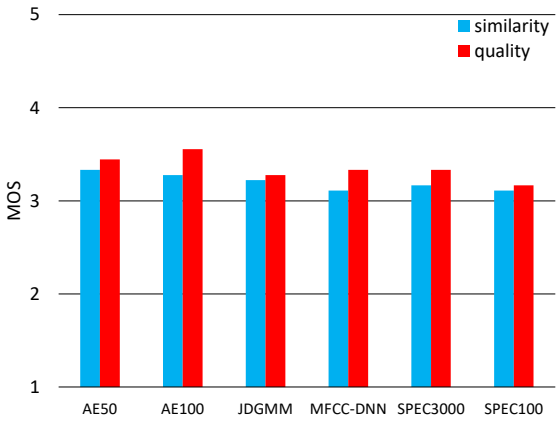

Figure 8: The result of the subjective evaluation in one-to-one VC with large training set.

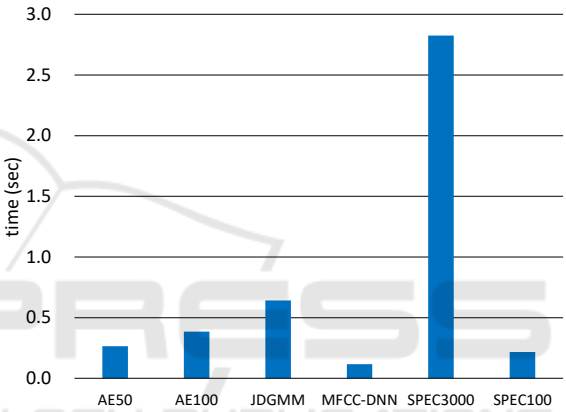

Figure 9: Required time to convert spectra of a speech (sec).

where x_t and $\hat{y}_t$ are a log-scaled F0 of the source speaker and the converted one at frame t respectively, $\mu^{(x)}$ and $\sigma^{(x)}$ are the mean and standard deviation of log-scaled F0 of the source speaker respectively, and $\mu^{(y)}$ and $\sigma^{(y)}$ are those of the target speaker respectively. In the evaluation of the one-to-one VC, two speeches, one from TK to YMGT conversion and the other from HM to TK conversion, both randomly selected, were evaluated and the results were averaged. In the subjective evaluation of the many-to-one VC, a speech randomly chosen from conversions to a target speaker HM was evaluated.

4.3 Results and Discussion

4.3.1 Results of One-to-One Voice Conversion

Table 1 shows the result of LSD evaluation in one-to-one VC with the small training set. The LSD values of AE50, AE100, SPEC3000 and SPEC100 are lower than those of JDGMM and MFCC-DNN i.e. the spectral conversion accuracy by AE50,

Table 1: The result of LSD evaluation in one-to-one VC with small training set.

target	AE50	AE100	JDGMM	MFCC-DNN	SPEC3000	SPEC100
YMGT to KJM	4.53	**4.39**	5.80	5.44	4.44	4.74
KJM to HM	4.71	4.66	6.77	6.31	**4.62**	4.78
TK to YMGT	4.56	**4.33**	5.66	5.30	4.45	4.67
HM to TK	4.18	**4.12**	5.12	4.85	4.14	4.28
average	4.50	**4.38**	5.84	5.48	4.41	4.61

Table 2: The result of LSD evaluation in one-to-one VC with large training set.

target	AE50	AE100	JDGMM	MFCC-DNN	SPEC3000	SPEC100
YMGT to KJM	4.08	**4.04**	5.06	5.19	4.06	4.17
KJM to HM	4.29	**4.20**	4.72	4.96	4.21	4.35
TK to YMGT	4.02	**3.96**	5.04	5.10	**3.96**	4.12
HM to TK	3.88	**3.82**	4.55	4.50	3.88	3.97
average	4.07	**4.01**	4.84	4.94	4.03	4.15

AE100, SPEC3000 and SPEC100 is better than that of JDGMM and MFCC-DNN. This is due to employing MFCC as acoustic features in JDGMM and MFCC-DNN. When converted MFCC is restored to spectral envelope, the high-frequency components are broken. As a result, the similarity of spectral envelope becomes low. However, the difference in the result of the subjective evaluation in Figure 7 is narrower than that in Table 1 since MFCC is a feature taking human speech perception into consideration. Although the difference in spectral conversion accuracy between AE100 and SPEC3000 is small, the accuracy of these methods is higher than that of AE50 and SPEC100. Since the accuracy of AE100 is higher than that of AE50, it is found that the method employing large dimensional higher-order features yields higher accuracy than small ones. In spite that the result shown in Table 2 resembles Table 1, the difference between LSD values of AE50 and SPEC100 and ones of AE100 and SPEC3000 becomes narrow. Since AE50 and SPEC100 employ simpler DNNs than AE100 and SPEC3000, it seems that a method using a simple DNN requires much training data. As a result, the hypothesis set up in the preliminary experiment: "the method using a simple DNN yields higher accuracy with a small amount of data, and the method using a complicated DNN yields higher accuracy with a large amount of data", is rejected.

Figure 7 and 8 show the result of the evaluation of the similarity and quality of the conversion based on human auditory perception. MOS values of each method are average score calculated from values of two conversion pairs. In the experiment with the small training set, SPEC3000 results in the highest similarity and AE100 results in the highest quality. Although there was a statistically significant difference between AE100 and SPEC100 in the quality, there were no significant differences in the similarity between the methods. In the experiment with the large training set, the proposed methods (AE50 and AE100) result in the highest similarity and quality. However, there were no significant differences in the similarity and the quality between the methods.

Figure 9 shows required time for spectral conversion by each method. In the methods converting log spectral envelope, the time required for obtaining converted spectral envelope from input log spectral envelope is calculated. On the other hand, in the methods converting MFCC, the time required for obtaining converted MFCC from input MFCC is calculated. Although in Table 1 and 2, the difference between AE100 and SPEC3000 is narrow, in comparison of converting time, the converting time of AE100 was 0.39 seconds, whereas that of SPEC3000 was 2.83 seconds. Namely, the spectral conversion by SPEC3000 takes approximately seven times as long as that by AE100.

Although the upper limit of the conversion time should be decided by a nature of application, let us assume that the target value for realizing a real-time VC is set as 2.5 seconds. In current speech synthesis technologies, the time required to analyze a 2-second speech to get features and to restore to the same speech is approximately 1.9 seconds [2]. Therefore, feature conversion should be carried out in approximately 0.6 seconds, hence the methods to fulfill this are AE50, AE100, MFCC-DNN and SPEC100. In the methods, AE100 is the predominant candidate because of the balance of the conversion accuracy and conversion time.

From the above results, it is seen that the pro-

[2] in case of using TANDEM-STRAIGHT

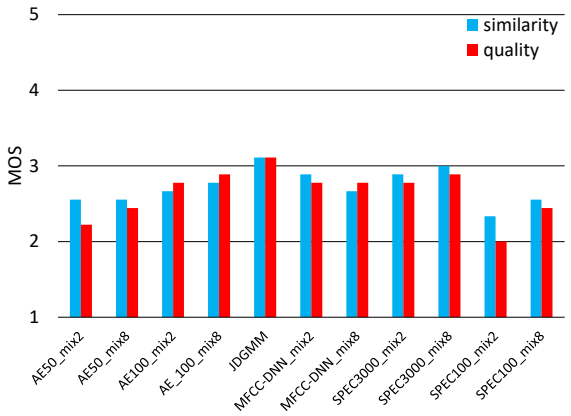

Figure 10: The result of the subjective evaluation in many-to-one VC with small training set.

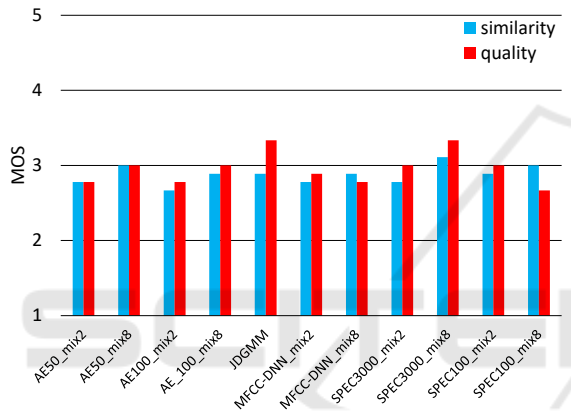

Figure 11: The result of the subjective evaluation in many-to-one VC with large training set.

posed method outperforms the conventional methods in terms of both conversion accuracy and converting time in the one-to-one VC.

4.3.2 Results of Many-to-One Voice Conversion

We carried out spectral conversion of eight patterns by each method, then evaluated the results by LSD. Table 3 and 4 show the results for each number of training speakers; 'mix2' means that the number of training speaker used for building a many-to-one voice converter is two, 'mix4' means four training speakers, and so on. In JDGMM, because it is not capable of many-to-one VC, the results of the one-to-one VC (TK to KRT and KRT to TK) are described. As seen in Table 1 and 2, JDGMM and MFCC-DNN yield lower accuracy than the other methods; we explained the cause of this in 4.3.1. In many-to-one VC experiment, AE100 also yields the highest accuracy with both the small training set and the large training set. Furthermore, AE50 also yields higher accuracy than SPEC3000 in a case of the large training set. Therefore, it seems that the proposed method enables better speaker-independent conversion than a method of converting features directly because generalized higher-order features are obtained from the training data of multiple source speakers by autoencoders. In Table 3 and 4, all the methods yield poor results when the number of training speakers is two. However, it matters little to LSD when it equals 4 or more. In Table 3, however, it was observed that the conversion accuracy of SPEC100 is improved monotonically along the number of training speakers. In this experiment, the effect of the number of the training speakers on LSD when it is more than eight is uncertain. It could be the case that a method of using a lot of training speakers does not necessarily improve accuracy of VC.

Figure 10 and 11 show the results of the subjective evaluation in the many-to-one VC. In the same method, a voice converter based on eight training speakers results in higher MOS values than one based on two training speakers in both similarity and quality. As noted here and in Table 1 and 2, a voice converter using more than three training speakers can generate a speech with higher quality than one using two training speakers. However, the results for MFCC-DNN is inconsistent with the trend. Regarding the difference between the methods, SPEC3000 outperforms AE100 in terms of MOS against the results in Table 1 and 2. Although we did one-way variance analysis of both similarity and quality for each method based on eight training speakers in the small training set and large training set respectively, there are no significant differences. Moreover, SPEC3000, which is a many-to-one VC method, outperforms JDGMM, which is a one-to-one VC method, in terms of both similarity and quality in the large training set. As a result, it was found that a many-to-one VC method employing spectral envelope yields higher accuracy than a conventional one-to-one VC method based on GMM. As the reason that the result of SPEC3000 in the subjective experiment is superior, it seems that SPEC3000 can deal with various input due to the complicated DNN structure.

5 CONCLUSION

In this study, we proposed the VC method employing autoencoders in order to reduce the amount of the training data and to shorten the converting time for one-to-one and many-to-one VC. In the evaluation experiment, the proposed method outperforms the conventional voice conversion methods that use GMM and DNN in both one-to-one conversion and many-

Table 3: The result of LSD evaluation in many-to-one VC with small training set.

	AE50	AE100	JDGMM	MFCC-DNN	SPEC3000	SPEC100
mix2	4.60	**4.52**	–	5.33	4.61	4.91
mix4	4.49	**4.44**	–	5.39	4.48	4.81
mix6	4.55	**4.47**	–	5.35	4.51	4.70
mix8	4.55	4.48	–	5.41	**4.47**	4.66
average	4.55	**4.48**	5.65	5.37	4.52	4.77

Table 4: The result of LSD evaluation in many-to-one VC with large training set.

	AE50	AE100	JDGMM	MFCC-DNN	SPEC3000	SPEC100
mix2	4.38	**4.35**	–	5.26	4.40	4.43
mix4	4.32	**4.28**	–	5.11	4.29	4.30
mix6	4.30	**4.26**	–	5.08	4.33	4.35
mix8	4.32	**4.27**	–	5.12	4.36	4.31
average	4.33	**4.29**	4.87	5.14	4.34	4.35

to-one conversion with small training dataset in terms of the conversion accuracy and the converting time. Therefore, the proposed method is superior in cases of developing applications under constraints that converting time must be short and a large training set is unavailable.

In future works, we will improve accuracy by pre-training a DNN to convert the higher-order features and fine-tuning a combined DNN consists of the autoencoders and the feature-converting DNN. Furthermore, we will conduct a many-to-one VC experiment with a lot of training speakers data, then we will specify the appropriate number of training speakers through observing how accuracy changes.

ACKNOWLEDGEMENTS

This work was supported by Solid Sphere, inc. and JSPS KAKENHI Grant Numbers 26330081, 2687020, 16K124111.

REFERENCES

Chen, L. H., Ling, Z. H., Song, Y., and Dai, L. R. (2013). Joint spectral distribution modeling using restricted boltzmann machines for voice conversion. In *Proc. INTERSPEECH*, pages 3052–3056.

Desai, S., Raghavendra, E. V., Yegnanarayana, B., Black, A. W., and Prahallad, K. (2009). Voice conversion using artificial neural networks. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3893–3896.

Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507.

Kawahara, H., Morise, M., Takahashi, T., Nisimura, R., Irino, T., and Banno, H. (2008). Tandem-straight: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, f0, and aperiodicity estimation. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3933–3936.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proc. International Conference for Learning Representations (ICLR)*.

Liu, L. J., Chen, L. H., Ling, Z. H., and Dai, L. R. (2015). Spectral conversion using deep neural networks trained with multi-source speakers. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4849–4853.

Mohammadi, S. H. and Kain, A. (2014). Voice conversion using deep neural networks with speaker-independent pre-training. In *Proc. Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 19–23.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proc. International Conference on Machine Learning*, pages 807–814. Omnipress.

Nakashika, T., Takashima, R., Takiguchi, T., and Ariki, Y. (2013). Voice conversion in high-order eigen space using deep belief nets. In *Proc. INTERSPEECH*, pages 369–372.

Nakashika, T., Takiguchi, T., and Ariki, Y. (2015). Voice conversion using speaker-dependent conditional restricted boltzmann machine. *EURASIP Journal on Audio, Speech, and Music Processing*, 2015(1):1–12.

Nguyen, H. Q., Lee, S. W., Tian, X., Dong, M., and Chng, E. S. (2016). High quality voice conversion using prosodic and high-resolution spectral features. *Multimedia Tools and Applications*, 75(9):5265–5285.

Stylianou, Y., Cappe, O., and Moulines, E. (1998). Continuous probabilistic transform for voice conversion. *IEEE Transactions on Speech and Audio Processing*, 6(2):131–142.

Toda, T., Black, A. W., and Tokuda, K. (2007). Voice

conversion based on maximum-likelihood estimation of spectral parameter trajectory. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(8):2222–2235.

Toda, T., Ohtani, Y., and Shikano, K. (2006). Eigenvoice conversion based on gaussian mixture model. In *Proc. INTERSPEECH 2006 - Ninth International Conference on Spoken Language Processing (ICSLP)*, pages 2446–2249.

Wu, Z., Chng, E. S., and Li, H. (2013). Conditional restricted boltzmann machine for voice conversion. In *Proc. IEEE China Summit International Conference on Signal and Information Processing (ChinaSIP)*, pages 104–108.

Xie, F.-L., Qian, Y., Fan, Y., Soong, F. K., and Li, H. (2014). Sequence error (SE) minimization training of neural network for voice conversion. In *Proc. INTERSPEECH*, pages 2283–2287.