10th ISCA Speech Synthesis Workshop
20-22 September 2019, Vienna, Austria

Voice conversion based on full-covariance mixture density networks for time-variant linear transformations

Gaku Kotani and Daisuke Saito

Graduate School of Engineering, The University of Tokyo, Japan.
kotani@gavo.t.u-tokyo.ac.jp, dsk_saito@gavo.t.u-tokyo.ac.jp

Abstract

This paper integrates a density estimation scheme based on neural networks with voice conversion (VC) under constraints of time-variant linear transformation. In VC, deep neural networks (DNNs) are used as conversion models that represent mapping from source to target features, in which a stack of multiple nonlinear transformations is applied to source ones. In automatic speech recognition and text-to-speech synthesis, direct mapping between source and target features by DNNs works effectively and flexibly since DNNs are suitable for such tasks in which input and output feature domains are heterogeneous, i.e. speech-to-text or text-to-speech. On the other hand, the case of VC is different from them, i.e. input and output features usually exist on the same domain, such as cepstral space. This condition may help more effective and flexible DNN-based VC. From this point of view, VC based on DNNs for time-variant linear transformations has been suggested. The method can utilize the condition, in which a trained model outputs parameters of linear transformations for each time index t: a linear transformation matrix $\boldsymbol{A}_t$ and a bias vector $\boldsymbol{b}_t$. It was observed that the method improved the performance of VC. However, the detailed properties of $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ have still been obscure. In this paper, in order to reveal it, full-covariance mixture density networks are introduced to the VC framework. In the proposed method, joint density of source and target features is directly estimated from the source features by mixture density networks. From the help of tight relationship between Gaussian and linear transformation, the correspondence between the parameters $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$, and density of the feature space become clear. The proposed scheme was investigated by experiments of VC, and the results showed that naturalness improvement was observed compared with naive DNN-based VC and the decided correspondence between $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ was observed.

Index Terms: voice conversion, deep neural networks, mixture density networks, Cholesky decomposition, Gaussian mixture models

1. Introduction

Voice conversion (VC) is a technique to modify the non-linguistic information of an input utterance while maintaining its linguistic information unchanged [1]. Among the various aspects of non-linguistic information, we focus on conversion of speaker identity which is represented in spectral envelopes. VC techniques can be applied to various applications such as modification of non-linguistic information of speech output from a text-to-speech synthesizer [2, 3].

VC techniques are usually implemented by mapping models which map from utterance features of a source speaker to those of a target speaker. For the mapping models, Gaussian mixture models (GMM), non-negative matrix factorization (NMF) and deep neural networks (DNNs) are widely used [2, 4, 5]. DNN-based VC is especially studied because of its remarkable results in automatic speech recognition (ASR) and text-to-speech synthesis (TTS). Generally, DNNs map source and target features directly through a stack of multiple nonlinear transformations. In ASR and TTS, the direct mapping through the transformations can be effective and flexible since their source and target domains are heterogeneous, such as speech and text. These heterogeneous mappings are also intoduced to VC such as posterior to spectral feature mapping [6]. However, VC is essentially a task of homo-domain mapping. In VC, input and output of a system exist on the same feature domain. The traditional GMM-based and NMF-based VC methods utilize the characteristic of VC, in the form of joint density modeling and spectral templates modeling. This indicates that, in VC, the directly mapping architecture of DNNs may be redundant so that input features can be mapped to unrealistic features, especially without a large amount of training data. Hence, it is worthwhile to study architectures exploiting the condition of homo-domain mapping.

Taking the condition into consideration, VC based on DNN for time-variant linear transformations (DNN-TVLT) has been proposed [7]. DNN-TVLT-based VC realizes only linear conversion but in a time-variant way: $\hat{\boldsymbol{y}}_t = \boldsymbol{A}(\boldsymbol{x}_t)\boldsymbol{x}_t + \boldsymbol{b}(\boldsymbol{x}_t)$, which is inspired by joint-GMM-based VC. DNN-TVLT-based VC achieves superior performance of VC than traditional DNN-based VC which maps source and target features directly. However, the relationship between its time-variant parameters $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ are unclear because they are only relevant via the criterion of minimizing mean square error between $\boldsymbol{y}_t$ and $\hat{\boldsymbol{y}}_t$. Our proposal is a novel VC method which makes the correspondence between parameters $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ and density of the feature space clear, so that the trained model is expected to more flexibly utilize the condition of homo-domain mapping. This is VC based on mixture density networks for time-variant linear transformations (MDN-TVLT).

In MDN-TVLT-based VC, an MDN outputs parameters of a *time-variant* joint-GMM which models the joint density of source and target feature vectors, and then the parameters of time-variant linear transformations $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ are constructed from the parameters of the time-variant joint-GMM. The proposed framework is expected to effectively and flexibly utilize the constraints of homo-domain mapping. In our experiments, the proposed method is evaluated in subjective assessments and the results showed that naturalness improvement was observed compared with naive DNN-based one and the decided correspondence of $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ was observed.

2. Related works

2.1. Joint-GMM-based VC

In this section, we briefly explain VC based on joint GMM and its interpretation as time-variant linear transformation [2]. Let

10.21437/SSW.2019-14

$\boldsymbol{x}_t$ and $\boldsymbol{y}_t$ be feature vectors of time index t from utterances of source and target speakers, respectively. Note that these utterances are parallel data. In joint-GMM-based VC, joint vector $\boldsymbol{z}_t = \left[\boldsymbol{x}_t^\top, \boldsymbol{y}_t^\top\right]^\top$ is modeled by GMM which has M components as follows

$$P\left(\boldsymbol{z}_t \mid \boldsymbol{\lambda}^{(z)}\right) = \sum_{m=1}^{M} w_m \mathcal{N}\left(\boldsymbol{z}_t; \boldsymbol{\mu}_m^{(z)}, \boldsymbol{\Sigma}_m^{(z)}\right), \quad (1)$$

where w_m, $\boldsymbol{\mu}_m^{(z)}$ and $\boldsymbol{\Sigma}_m^{(z)}$ denote the weight, the mean vector, and the covariance matrix of the m-th Gaussian component, respectively. The mean vector and the covariance matrix can be separately represented by that of source and target features as follows

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

In the conversion phase, a mapping function from source to target features $\mathcal{F}(\cdot)$ is based on the conditional probability density $P(\boldsymbol{y}_t \mid \boldsymbol{x}_t)$. When we use minimizing mean square error for the criterion of the conversion, the mapping function can be written as follows

$$\mathcal{F}(\boldsymbol{x}_t) = \sum_{m=1}^{M} P\left(m \mid \boldsymbol{x}_t, \boldsymbol{\lambda}^{(z)}\right) \boldsymbol{E}_{m,t}^{(y)}, \quad (3)$$

where

$$\boldsymbol{E}_{m,t}^{(y)} = \boldsymbol{\mu}_m^{(y)} + \boldsymbol{\Sigma}_m^{(yx)} \boldsymbol{\Sigma}_m^{(xx)-1} \left(\boldsymbol{x}_t - \boldsymbol{\mu}_m^{(x)}\right). \quad (4)$$

Eq. 3 indicates that the first term $P(m|\boldsymbol{x}_t, \boldsymbol{\lambda}^{(z)})$ plays a role of allocating the source feature at time t to a specific component of GMM, and the second term $\boldsymbol{E}_{m,t}^{(y)}$ plays a role of linear transformation corresponding to the component. To be exact, the conversion in Eq. 3 is carried out by the weighted sum of each component, and then the conversion is not discrete but continuous. From this point of view, the mapping function $\mathcal{F}(\cdot)$ can be represented as a time-variant linear transformation, which is written as follows

$$\hat{\boldsymbol{y}}_t = \mathcal{F}(\boldsymbol{x}_t) = \boldsymbol{A}(\boldsymbol{x}_t)\boldsymbol{x}_t + \boldsymbol{b}(\boldsymbol{x}_t), \quad (5)$$

where

$$\boldsymbol{A}(\boldsymbol{x}_t) = \sum_{m=1}^{M} P\left(m \mid \boldsymbol{x}_t, \boldsymbol{\lambda}^{(z)}\right) \left(\boldsymbol{\Sigma}_m^{(yx)} \boldsymbol{\Sigma}_m^{(xx)-1}\right), \quad (6)$$

$$\boldsymbol{b}(\boldsymbol{x}_t) = \sum_{m=1}^{M} P\left(m \mid \boldsymbol{x}_t, \boldsymbol{\lambda}^{(z)}\right) \left(\boldsymbol{\mu}_m^{(y)} - \boldsymbol{\Sigma}_m^{(yx)} \boldsymbol{\Sigma}_m^{(xx)-1} \boldsymbol{\mu}_m^{(x)}\right). \quad (7)$$

Eq. 6 indicates that GMM-based time-variant linear transformation strongly depends on the properties of GMM, namely that only the weight sum of $\boldsymbol{\Sigma}_m^{(yx)} \boldsymbol{\Sigma}_m^{(xx)-1}$ is permitted as the flexibility of the transformation.

2.2. DNN-TVLT-based VC

This section explains VC based on time-variant linear transformations of which parameters are estimated by DNN [7]. In traditional DNN-based VC, DNNs are trained to represent mapping directly from source to target spectral features, often characterized as cepstrum, with a stack of multiple nonlinear transformations [5]. In the case that a large amount of training data is available, such DNN-based approaches achieve better performance on the conversion than traditional GMM-based VC. Let $\boldsymbol{h}^{(l)}$ be a feature vector of the l-th layer in a DNN, and then the transformation function between two layers is represented as a combination of linear conversion from the previous layer and an activation function $g(\cdot)$, which is shown as follows

$$\boldsymbol{h}^{(l)} = g\left(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}\right). \quad (8)$$

In traditional DNN-based VC, the DNN is trained to represent a mapping function from source to target features as follows

$$\hat{\boldsymbol{y}}_t = \boldsymbol{G}(\boldsymbol{x}_t), \quad (9)$$

where $\boldsymbol{G}(\cdot)$ is a stack of multiple nonlinear transformations (Eq. 8). In the conversion process, the target feature $\hat{\boldsymbol{y}}_t$ is derived from the given source feature $\boldsymbol{x}_t$ by a stack of multiple nonlinear transformations $G(\cdot)$. The direct mapping can effectively and flexibly connect features in heterogeneous domains, such as text and speech in ASR or TTS. On the other hand, in the case of VC, the direct mapping can be redundant since the task of VC is homo-domain mapping.

In DNN-TVLT-based VC, the homo-domain condition is effectively utilized compared with the traditional direct mapping method. In this study, it is shown that linear conversion can work at least for vocal tract length transformation and if it can be implemented in a time-variant way, the resulting model can convert input speech in a more flexible way. The network architecture of DNN-TVLT-based VC is shown in Fig. 1. The entire network is composed of two sub-networks and their connection. For each input $\boldsymbol{x}_t$, they estimate their corresponding parameters, one for a linear transformation matrix and the other for a bias vector. Using both of them, a source feature vector $\boldsymbol{x}_t$ is mapped into its target feature $\hat{\boldsymbol{y}}_t$, shown as follows,

$$\hat{\boldsymbol{y}}_t = \boldsymbol{A}(\boldsymbol{x}_t)\boldsymbol{x}_t + \boldsymbol{b}(\boldsymbol{x}_t). \quad (10)$$

The adopted training criterion is the same as it often used in conventional VC methods, which is minimization of conversion errors, i.e. cepstrum distortion. The method does not only exploit the constraints more effectively than the traditional DNN-based one, but also can represent the mapping features in a more flexible way than GMM-based one. Since the conversion in DNN-TVLT is constrained to linear conversion, the trained model is expected to avoid mapping to unrealistic speech features. In their experiments, the results suggest that DNN-TVLT-based VC improves the conversion performance in subjective assessments about naturalness.

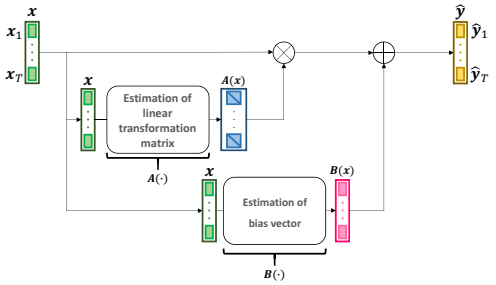


Figure 1: *Framework of DNN-based time-variant linear transformations in [7].*

3. Full-covariance MDN for time-variant linear transformations

3.1. VC based on MDN for time-variant linear transformations

This section explains our proposed method of VC based on time-variant linear transformations of which parameters are fabricated by outputs of an MDN. Its training and conversion schematics are shown in Fig. 2. In DNN-TVLT-based VC, while the constraints of homo-domain mapping is exploited, the relationship between its time-variant parameters $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$, are obscure because they are only relevant via the criterion of minimizing mean square error between $\boldsymbol{y}_t$ and $\hat{\boldsymbol{y}}_t$. In other words, although $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ are expected to correspond to rotation and shift transformations respectively, they do not necessarily correspond to such functions because of the high flexibility of DNNs. Our proposal is that, the parameters $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ are constructed from parameters of *time-variant* joint-GMM which models the joint vector $\boldsymbol{z}_t = \left[\boldsymbol{x}_t^\top, \boldsymbol{y}_t^\top\right]^\top$ given $\boldsymbol{x}_t$.

Our model learns the conditional probability density $P\left(\boldsymbol{z}_t \mid \boldsymbol{x}_t\right)$ which is represented as in the form

$$P\left(\boldsymbol{z}_t \mid \boldsymbol{x}_t, \boldsymbol{\theta}\right) = \sum_{m=1}^{M} w_{t,m}\, \mathcal{N}\left(\boldsymbol{z}_t; \boldsymbol{\mu}_{t,m}^{(z)}, \boldsymbol{\Sigma}_{t,m}^{(z)}\right), \tag{11}$$

where $\boldsymbol{\theta}$ indicates parameters of an MDN, and $w_{t,m}$, $\boldsymbol{\mu}_{t,m}^{(z)}$, $\boldsymbol{\Sigma}_{t,m}^{(z)}$ are outputs of the MDN given the input $\boldsymbol{x}_t$. Although Eq. 11 may look a bit strange because $P\left(\boldsymbol{z}_t \mid \boldsymbol{x}_t, \boldsymbol{\theta}\right)$ includes $P\left(\boldsymbol{x}_t \mid \boldsymbol{x}_t, \boldsymbol{\theta}\right)$, it indicates reconstruction through the network like autoencoders. In other words, the probability density $P\left(\boldsymbol{z}_t \mid \boldsymbol{x}_t\right)$ can be written as

$$\begin{aligned} P\left(\boldsymbol{z}_t \mid \boldsymbol{x}_t\right) &= P\left(\boldsymbol{x}_t, \boldsymbol{y}_t \mid \boldsymbol{x}_t\right) && (12)\\ &= \int P\left(\boldsymbol{x}_t, \boldsymbol{y}_t \mid \boldsymbol{l}_t, \boldsymbol{x}_t\right) P\left(\boldsymbol{l}_t \mid \boldsymbol{x}_t\right) d\boldsymbol{l}_t && (13)\\ &\simeq \int P\left(\boldsymbol{x}_t, \boldsymbol{y}_t \mid \boldsymbol{l}_t\right) P\left(\boldsymbol{l}_t \mid \boldsymbol{x}_t\right) d\boldsymbol{l}_t && (14) \end{aligned}$$

where $\boldsymbol{l}_t$ is a latent representation, which is often regarded as linguistic information in the context of VC, and then the first and second terms of Eq. 14 indicate decoder and encoder, respectively. Note that, in this paper, the latent representation is *not* utilized explicitly and it is one of our future works.

A mapping function $\mathcal{F}\left(\cdot\right)$ to convert the source vector $\boldsymbol{x}_t$ to the target vector $\hat{\boldsymbol{y}}_t$ is derived based on the conditional probability density $P\left(\boldsymbol{z}_t \mid \boldsymbol{x}_t\right)$, in a similar manner to GMM-based VC. This probability density can be represented by parameters of the *time-variant* joint density model, which is the output of the MDN, and then the mapping function $\mathcal{F}\left(\cdot\right)$ can be shown as follows

$$\mathcal{F}\left(\boldsymbol{x}_t\right) = \sum_{m=1}^{M} w_{t,m} \left(\boldsymbol{\mu}_{t,m}^{(y)} + \boldsymbol{\Sigma}_{t,m}^{(yx)} \boldsymbol{\Sigma}_{t,m}^{(xx)-1} \left(\boldsymbol{x}_t - \boldsymbol{\mu}_{t,m}^{(x)}\right)\right). \tag{15}$$

Especially in the case of single component ($M = 1$), it can be represented as follows

$$\mathcal{F}\left(\boldsymbol{x}_t\right) = \boldsymbol{\mu}_t^{(y)} + \boldsymbol{\Sigma}_t^{(yx)} \boldsymbol{\Sigma}_t^{(xx)-1} \left(\boldsymbol{x}_t - \boldsymbol{\mu}_t^{(x)}\right). \tag{16}$$

To compare with the mapping function in DNN-TVLT-based VC (Eq. 10), Eq. 16 can be re-written as follows

$$\mathcal{F}(\boldsymbol{x}_t) = \boldsymbol{A}(\boldsymbol{x}_t)\boldsymbol{x}_t + \boldsymbol{b}(\boldsymbol{x}_t), \tag{17}$$

where

$$\boldsymbol{A}(\boldsymbol{x}_t) = \boldsymbol{\Sigma}_t^{(yx)} \boldsymbol{\Sigma}_t^{(xx)-1}, \tag{18}$$

$$\boldsymbol{b}(\boldsymbol{x}_t) = \boldsymbol{\mu}_t^{(y)} - \boldsymbol{\Sigma}_t^{(yx)} \boldsymbol{\Sigma}_t^{(xx)-1} \boldsymbol{\mu}_t^{(x)}. \tag{19}$$

As shown in Eq. 19, the correspondence between the linear transformation matrix and the bias vector are explicit because they are fabricated by the parameters of the time-variant joint-GMM. The proposed networks estimate all the parameters related to both the source and target features while the conventional ones only capture the relationship between the source and target. That is, it tries to capture a larger amount of information from the source than the conventional network.

In our experiments, the number of mixtures is fixed to one in order to compare the performance of DNN-TVLT and MDN-TVLT. The proposed framework is expected to effectively exploit the constraints of homo-domain mapping and also flexibly learn time-variant properties.

3.2. Full-covariance MDN based on Cholesky decomposition

In this section, we explain our implementation of full-covariance MDN. Modeling the joint density over variables with MDNs is in general much difficult, because it requires outputting a positive definite covariance matrix. In the case of comparing our proposed method with DNN-TVLT-based VC, however, full-covariance matrices can be needed. In this section, the implementation of full-covariance MDNs via the Choleskey decomposition is explained.

First, we briefly explain MDN [8]. In MDN, the probability density of the target data $\boldsymbol{y}_t$ is represented as in the form

$$P(\boldsymbol{y}_t|\boldsymbol{x}_t) = \sum_{m=1}^{M} w_{t,m} \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{\mu}_{t,m}^{(y)}, \boldsymbol{\Sigma}_{t,m}^{(y)}). \tag{20}$$

In most studies using MDNs, diagonal-covariance matrix is adopted because its computation and implementation are easy. In the diagonal case, outputs of MDNs are shown as follows

$$w_{t,m} = \frac{\exp(z_{t,m}^{w})}{\sum_{i=1}^{M} \exp(z_{t,i}^{w})}, \tag{21}$$

$$\boldsymbol{\sigma}_{t,m} = \exp(\boldsymbol{z}_{t,m}^{\Sigma}), \tag{22}$$

$$\boldsymbol{\mu}_{t,m} = \boldsymbol{z}_{t,m}^{\mu}, \tag{23}$$

where $z_{t,m}^{w}$, $\boldsymbol{z}_{t,m}^{\Sigma}$ and $\boldsymbol{z}_{t,m}^{\mu}$ are output vectors of the last hidden layer, and $w_{t,m}$, $\boldsymbol{\sigma}_{t,m}$ and $\boldsymbol{\mu}_{t,m}$ indicate the weight, the diagonal element vector of covariance matrix and the mean vector of m-th component of GMM, respectively.

A symmetric positive definite square matrix $\boldsymbol{\Sigma}$, such as a covariance matrix of Gaussian distribution, is decomposed with the Cholesky decomposition as in the form

$$\boldsymbol{\Sigma} = \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{L}\boldsymbol{L}^\top, \tag{24}$$

where $\boldsymbol{U}$ is a unique upper triangular matrix and $\boldsymbol{L}$ is a unique lower triangular matrix. The covariance matrix can be constructed from the lower triangular matrix and its transposed version, and then the full-covariance MDN can be implemented by the MDN outputting the triangular matrix $\boldsymbol{L}_{t,m}$. Fot the diagonal elements of the lower triangular matrix: $\mathrm{diag}\left(\boldsymbol{L}_{t,m}\right) = [l_{t,m,0}, ..., l_{t,m,D-1}]$, it is convenient to represent them in terms

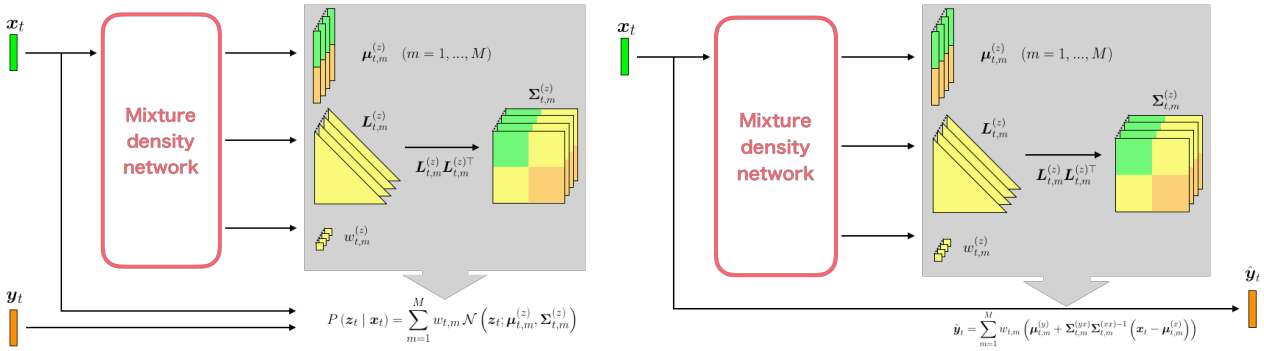


Figure 2: *Framework of time-variant linear conversion based on full-covariance MDNs. At left is a schematic of training of MDN-TVLT-based VC. At right is a schematic of time-variant linear conversion using a trained MDN.*

Table 1: *Experimental conditions of evaluated methods. In DNN-TVLT, the two number of parameters of layers and units indicate the number of parameters of two sub-networks, respectively. The three number of that in MDN-TVLT indicate the number of parameters of a shared sub-network, that of estimating covariance matrix and that of estimating mean vector, respectively.*

Methods	Nb. layers	Nb. Units
Baseline	6	256
DNN-TVLT	6,3	256,256
MDN-TVLT	4,6,4	64,128,128

of exponentials of the corresponding network outputs, which is shown as follows

$$l_{t,m,i} = \exp(z^{\Sigma}_{t,m,i}) \quad (i = 0, 1, ..., D-1), \tag{25}$$

where $l_{t,m,i}$ indicates the i-th element of the diagonal elements. The other lower triangular elements are unrestricted since positive definiteness is guaranteed. This is that, they are represented directly by the network outputs as follows

$$l_{t,m,i} = z^{\Sigma}_{t,m,i} \quad (i = D, ..., D\,(D+1)\,/2 - 1)\,. \tag{26}$$

Hence, the outputs of networks can construct lower triangular matrix $\boldsymbol{L}_{t,m}$, and then full-covariance matrix $\boldsymbol{\Sigma}_{t,m}$ is constructed by the lower triangular matrix and its transposed version. MDNs which output the lower triangular entries in the Cholesky decomposition of full-covariance matrix have been also mentioned in some previous studies [9, 10].

4. Experiments

4.1. Experimental conditions

To evaluate our proposed method, subjective evaluations were carried out. The ATR Japanese speech dataset B-set were used as a source and target male-to-male pair (MHT to MMY) [11]. We compared three methods in the evaluations, which are the conventional DNN-based VC (Baseline), the VC based on DNN for time-variant linear transformations (DNN-TVLT) and the proposed method (MDN-TVLT).

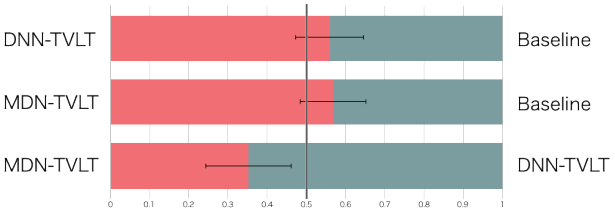
From the dataset, subset A, B, I and J of phoneme-balanced sentences were used, which had about 50 sentences for each subset. The first two subsets, the third and the forth subsets were used for training, validation and testing, respectively. Speech signals were sampled at 20 kHz. Feature vectors were extracted with a 5-ms shift and the feature vector consisted of the 0-th through 24-th mel-cepstrums, which were derived from WORLD analysis [12] (D4C edition [13]). As preprocessing for the cepstral features, trajectory smoothing with a cutoff modulation frequency of 50 Hz was used [14]. Alignment of parallel data was performed by Affine-DTW which iteratively performs dynamic time warping and global, time-invariant, affine transformation as a coarse voice conversion to avoid that the difference of speaker identities affects alignment of parallel data [15]. For the generation of speech waveform, voice conversion with direct waveform modification based on spectrum differential was applied [16]. In addition, F_0 transformation was performed by linear transformation.

For the input and output features of all the DNNs, 24-dimensional mel-cepstrums were used. The activation functions were ReLU and a linear mapping in hidden and output layers, respectively. Dropout with probability of 0.5 was applied to each layer. As an optimization method, AMSGrad with a learning rate of 0.001 was used [17]. The batch size was 1024 and the number of epochs was fixed to 5. The other hyper-parameters of the methods are shown in Tab. 1, which were determined in validation loss by preliminary experiments.

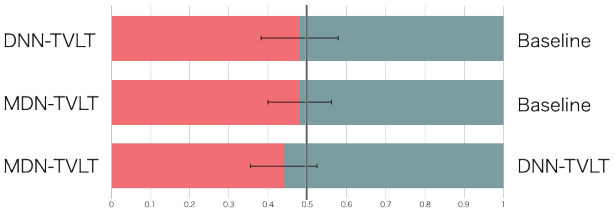
Subjective evaluations were performed to evaluate naturalness of converted speech and similarity between the converted and target speech. The evaluations were conducted with 25 subjects for each test. To evaluate the naturalness, AB test was carried out, where pairs of two converted samples were presented to subjects, and then each subject judged which sample sounded more natural. To evaluate the similarity, ABX test was performed, where pairs of two samples were presented after presenting the reference sample of the target speech, and then each subject judged which sample sounded more similar to the reference one in terms of speaker identity. The test sentences were randomly selected from test set and the number of sample pairs evaluated by each subject was 15, which means 5 sample pairs for each method pair, in each test.

4.2. Experimental results

Fig. 3 shows the results of naturalness and similarity tests. In the naturalness test, the MDN-TVLT and the DNN-TVLT outperformed the Baseline. This would be because time-variant linear transformations of both MDN-TVLT and DNN-TVLT worked effectively to model the conversion. However, the proposed MDN-TVLT did not outperform the DNN-TVLT. In all the cases, the proposed method achieved the reasonable perfor-



(a) Naturalness



(b) Similarity

Figure 3: *Results of subjective evaluations.*

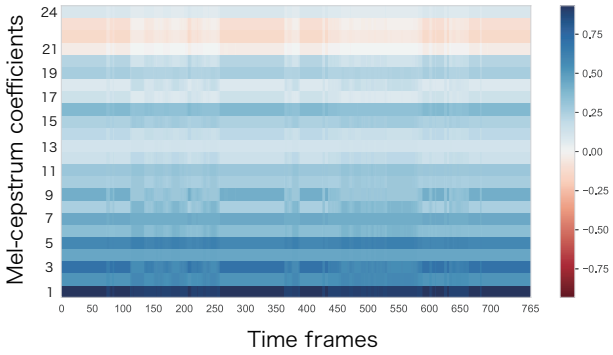
mance in similarity. Unlike the DNN-TVLT, the MDN-TVLT tried more challenging task which is modeling the probability density over all the parameters related to both the source and target features. As a result, the MDN-TVLT was inferior to the DNN-TVLT in our experimental condition. Further experiments with a large amount of training data and cross-gender speaker pair are our future works.

To investigate the correspondence between $\boldsymbol{A}(\boldsymbol{x}_t)$ and $\boldsymbol{b}(\boldsymbol{x}_t)$, the parameters from an utterance in test set were visualized in Fig. 4, 5. In Fig. 4, the time correspondence between $\boldsymbol{A}(\boldsymbol{x}_t)$ and $\boldsymbol{b}(\boldsymbol{x}_t)$ which are fabricated by outputs of the trained MDN-TVLT can be clearly observed, while it can not in Fig. 5. From this point of view, the proposed MDN-TVLT worked well.

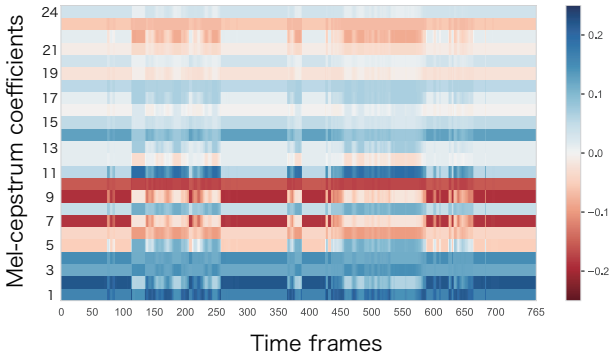
The proposed scheme based on MDNs has possibilities that it can be utilized various kinds of applications in addition to one-to-one VC. The proposed networks estimate all the parameters related to both the source and target features while the conventional ones only capture the relationship between the source and target. That is, it tries to capture a larger amount of information from the source than the conventional network. Although the proposed networks are not easy to be trained in a limited task such as one-to-one VC, they are expected to be utilized for a larger task such as arbitrary speaker conversion, because the task requires not only the relationship between input and output, but also the whole properties of the feature space.

5. Conclusions

This paper has integrated a density estimation scheme based on neural networks with VC under constraints of time-variant linear transformations. Our proposal is a novel VC method based on MDN-TVLT which makes the correspondence between the parameters $\boldsymbol{A}_t$, $\boldsymbol{b}_t$ clear so that the trained model is expected to more flexibly utilize the condition of homo-domain mapping. In MDN-TVLT-based VC, an MDN outputs parameters of *time-variant* joint-GMM which models the joint density of source and target features, and then the parameters $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ are fabricated by parameters of the *time-variant* joint-GMM. In our experiments, the proposed method was evaluated in subjective assessments and the results showed that naturalness improvement was observed compared with traditional the DNN-based one but it was not with the DNN-TVLT-based one. The MDN-TVLT



(a) The diagonal elements of linear transformation matrix $\boldsymbol{A}(\boldsymbol{x}_t)$



(b) The bias vector $\boldsymbol{b}(\boldsymbol{x}_t)$

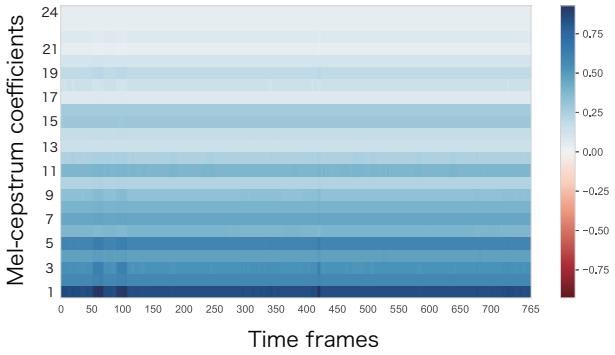
Figure 4: *Visualizations of parameters of time-variant linear transformation $\boldsymbol{A}(\boldsymbol{x}_t)$ and $\boldsymbol{b}(\boldsymbol{x}_t)$, which are fabricated by outputs of a trained MDN-TVLT. In the figures, the correspondence of the parameters can be observed.*

has tried more challenging task than the DNN-TVLT, which is modeling the probability density over all the parameters related to both the source and target features. Our experimental condition could be disadvantageous to the MDN-TVLT so further experiments are needed. In the experimental results, however, the correspondence between the parameters $\boldsymbol{A}_t$ and $\boldsymbol{b}_t$ can be observed in the MDN-TVLT while it can not in the DNN-TVLT.

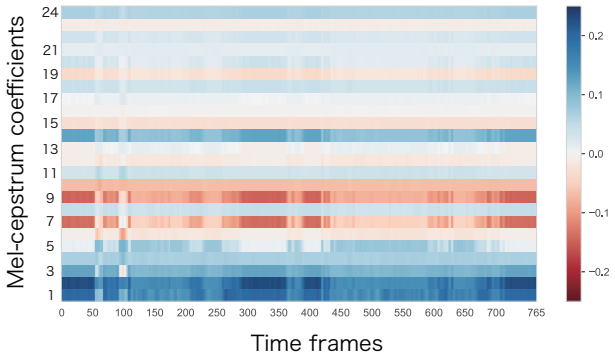
The proposed framework does not only effectively exploit the constraints that the input and output exist on the same domain, but also has possibilities that it can be utilized various kinds of applications in addition to one-to-one VC. In other words, the proposed networks estimate all the parameters related to both the source and target features while the conventional ones only capture the relationship between the source and target. The flexibility of our proposal for one-to-many or many-to-many voice conversion will be investigated. Applying our proposal to parallel data free voice conversion via utilizing latent representation is also one of our future works.

6. Acknowledgements

This research and development work was supported by the Ministry of Internal Affairs and Communications.



(a) The diagonal elements of linear transformation matrix $\boldsymbol{A}(\boldsymbol{x}_t)$



(b) The bias vector $\boldsymbol{b}(\boldsymbol{x}_t)$

Figure 5: *Visualizations of parameters of time-variant linear transformation $\boldsymbol{A}(\boldsymbol{x}_t)$ and $\boldsymbol{b}(\boldsymbol{x}_t)$, which are outputted by a trained DNN-TVLT.*

7. References

[1] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 1988, pp. 655–658.

[2] A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 1998, pp. 285–288.

[3] H. Kawanami, Y. Iwami, T. Toda, H. Hiroshi, and K. Shikano, "GMM-based voice conversion applied to emotional speech synthesis," in *Proceedings of European Conference on Speech Communication and Technology*, 2003, pp. 1–4.

[4] Z. Wu, T. Virtanen, T. Kinnunen, E. S. Chng, and H. Li, "Exemplar-based voice conversion using non-negative spectrogram deconvolution," in *Proceedings of ISCA Workshop on Speech Synthesis*, 2013, pp. 201–206.

[5] S. Desai, E. V. Raghavendra, B. Yegnanarayana, A. W. Black, and K. Prahallad, "Voice conversion using artificial neural networks," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 3893–3896.

[6] L. J. Liu, Z. H. Ling, Y. Jiang, M. Zhou, and L. R. Dai, "Wavenet vocoder with limited training data for voice conversion," in *Proceedings of INTERSPEECH*, 2018, pp. 1983–1987.

[7] G. Kotani, D. Saito, and N. Minematsu, "Voice conversion based on deep neural networks for time-variant linear transformations," in *Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Dec 2017, pp. 1259–1262.

[8] C. M. Bishop, "Mixture density networks," Aston University, Tech. Rep., 1994.

[9] H. F. Lopes, R. E. McCulloch, and R. S. Tsay, "Cholesky stochastic volatility," University of Chicago, Tech. Rep., 2011.

[10] W. Tansey, K. Pichotta, and J. G. Scott, "Better conditional density estimation for neural networks," *arXiv:1606.02321 [stat.ML]*, 2016.

[11] A. Kurematsu, K. Takeda, Y. Sagisaka, S. Katagiri, H. Kuwabara, and K. Shikano, "ATR Japanese speech database as a tool of speech recognition and synthesis," *Speech Communication*, vol. 9, no. 4, pp. 357 – 363, 1990.

[12] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Transactions on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.

[13] M. Morise, "D4C, a band-aperiodicity estimator for high-quality speech synthesis," *Speech Communication*, vol. 84, pp. 57 – 65, 2016.

[14] S. Takamichi, T. Toda, A. W. Black, G. Neubig, S. Sakti, and S. Nakamura, "Postfilters to modify the modulation spectrum for statistical parametric speech synthesis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 755–767, 2016.

[15] H. Suda, G. Kotani, S. Takamichi, and D. Saito, "A revisit to feature handling for high-quality voice conversion based on Gaussian mixture model," in *Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 2018, pp. 816–822.

[16] K. Kobayashi, S. Takamichi, S. Nakamura, and T. Toda, "The NU-NAIST voice conversion system for the voice conversion challenge 2016," in *Proceedings of INTERSPEECH*, 2016, pp. 1667–1671.

[17] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of Adam and beyond," in *Proceedings of International Conference on Learning Representations*, 2018.