Voice Conversion from Non-parallel Corpora Using Variational Auto-encoder

Chin-Cheng Hsu*, Hsin-Te Hwang*, Yi-Chiao Wu*, Yu Tsao† and Hsin-Min Wang*
* Institute of Information Science, Academia Sinica, Taipei, Taiwan
E-mail: {jeremycchsu, hwanght, tedwu, whm}@iis.sinica.edu.tw
† Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan
E-mail: yu.tsao@citi.sinica.edu.tw

***Abstract*—We propose a flexible framework for spectral conversion (SC) that facilitates training with unaligned corpora. Many SC frameworks require parallel corpora, phonetic alignments, or explicit frame-wise correspondence for learning conversion functions or for synthesizing a target spectrum with the aid of alignments. However, these requirements gravely limit the scope of practical applications of SC due to scarcity or even unavailability of parallel corpora. We propose an SC framework based on variational auto-encoder which enables us to exploit non-parallel corpora. The framework comprises an encoder that learns speaker-independent phonetic representations and a decoder that learns to reconstruct the designated speaker. It removes the requirement of parallel corpora or phonetic alignments to train a spectral conversion system. We report objective and subjective evaluations to validate our proposed method and compare it to SC methods that have access to aligned corpora.**

I. Introduction

Voice conversion is a technique that converts the perceived identity of speaker of a given utterance. A typical case is that, when one wants to convert his or her voices into a celebrity's, it is required that linguistic contents and other speaker-unrelated information remain unchanged after conversion. A complete voice conversion system involves many tasks. In this study, we devote our focus on spectral conversion (SC) and leave inspection on prosody outside the scope of this paper.

A wide variety of techniques have been applied to spectral conversion, including Gaussian mixture models (GMM) [1]–[3], frequency warping [4], [5], deep neural networks (DNN) [6]–[8], and exemplar-based approaches [9], [10]. Most of these methods demand aligned source-target pairs of frames or alignments of phonetic states to train conversion functions or adaptation transformations. The most widely adopted approach is to align source and target frames using dynamic time warping (DTW) technique. However, DTW fails to work if parallel corpora are unavailable.

Many techniques have been conceived to align source and target frames in non-parallel corpora. The most intuitive way is to apply a speech recognizer to the utterances, and proceed with explicit alignment or model adaptation [11], [12]. Applying speech recognizers to each utterance gives every frame a phonetic label (usually of phonetic states). It is particularly suitable for model-based voice conversion techniques because they can readily utilize these labeled frames [13]. The problem with this frame-wise, model-based approach is that it does not apply to cross-lingual conditions, which require a more general form of alignment. To this end, the INCA algorithm and related methods [14]–[16] were proposed to iteratively seek frame-wise correspondence using converted surrogate frames. Another attempt is to separately build frame clusters for the source and the target, and then set up a mapping between them [17].

Let us ponder upon the roles these alignment techniques play in the task of voice conversion. Consider a speech corpus of a source speaker s and a target t. The subset $\boldsymbol{X}_s = \{\boldsymbol{x}_{s,n}\}_{n=1}^{N_s}$ represents all the frames from the source and $\boldsymbol{X}_t = \{\boldsymbol{x}_{t,m}\}_{m=1}^{N_t}$ are those from the target, where N_s and N_t are the total number of frames of the source and the target, respectively.

The first and probably the most general kind of alignment is frame-wise. It seeks index pairs (n, m) such that $\boldsymbol{x}_{s,n}$ and $\boldsymbol{x}_{t,m}$ have similar phonetic contents. For simplicity, we assume that the correspondence is a function (though it is not in most cases). Frame-wise alignment is therefore a function of $\boldsymbol{x}_{s,n}$ that yields a corresponding $\boldsymbol{x}_{t,m}$. Under this circumstance, alignment is not only necessary, but also nearly sufficient for SC because SC also pursues similar functions.

The second kind is frame-to-model alignment attained with the help of speech recognizers. It assumes that every frame corresponds to a (phonetic) model (or, equivalently, a cluster). The alignment is thus (n, k) pairs where k is the model index of a phonetic state. Conversion is then the transformation function that inputs a model from the source, and outputs a model from the target. For the purpose of conversion, alignment is also necessary under this scenario.

In contrast, the factor of speaker plays a rather implicit role in voice conversion. For example, in most pair-wise SC (one source and one target), speaker identity is only responsible for designating a frame to the input (if it is from the source) or to the output (if otherwise). It is curious that we build voice conversion systems without explicitly exploiting speaker-dependent factors considering the purpose.

We propose a framework that directly exploits speaker identity to build SC systems without explicitly aligning source and target frames. Our proposed formulation decomposes conversion into encoding and decoding stages, and renders conversion a controlled version of self-reconstruction. With this self-reconstruction formulation, aligned frame pairs or

even parallel corpora are no longer necessary for SC tasks. Our experiments showed that its performance is comparable to baseline systems, substantiating this nascent framework for general SC tasks.

The rest of this paper is organized as follows. In Sec. II, we describe the inspiration and the concepts, and elaborate our methods. Experimental settings and results are collected in Sec. III to validate our proposed framework. Finally, we conclude our paper in Sec. IV.

II. The Proposed Method

The proposed method is inspired from an analogous work on generating hand-written digits [18], [19]. The authors of [18] attempted to extract writing style and digit identity from an image of handwriting and to re-synthesize the image with the extractive. Basically what the framework offers is an explanatory model of an observed variable and two causal latent factors: identity and variation. We hypothesize that the explanatory model behind speech frames coincides with that of hand-writing images. For a hand-written digit, the identity is the nominal number and the variation is the hand-writing style. For a speech frame, the identity could be the speaking source and the variation could be the phonetic content.

A. Auto-encoder Reformulation for SC from Unaligned Data

Given spectral frames $\{\boldsymbol{x}_{s,n}\}_{n=1}^{N_s}$ from the source speaker and those $\{\boldsymbol{x}_{t,m}\}_{m=1}^{N_t}$ from the target, conventional SC seeks to estimate conversion functions such that

$$\hat{\boldsymbol{x}}_{t,m} = f(\boldsymbol{x}_{s,n}), \tag{1}$$

where $f(\cdot)$ is a conversion function. In most SC systems, speaker identity (subscripts s and t) are treated implicitly; for example, in (1), the source is always the input while the target is always the desired output.

We explicitly incorporate a speaker representation $\boldsymbol{y}_n$ into the SC formulation. Firstly, the conversion function is reformulated as an auto-encoder. The encoder $f_\phi(\cdot)$ is designed to be speaker-independent; it ignores speaker identity of an incoming frame (so $\boldsymbol{x}_{s,n}$ and $\boldsymbol{x}_{t,m}$ can now be expressed by $\boldsymbol{x}_n$), and converts an observed frame into speaker-independent latent variable:

$$\boldsymbol{z}_n = f_\phi(\boldsymbol{x}_n), \tag{2}$$

where $\boldsymbol{z}_n$ is a latent variable (or *code* in auto-encoder terminology). Presumably, $\boldsymbol{z}_n$ contains information that is irrelevant to speaker, such as phonetic variations. We refer to $\boldsymbol{z}_n$ as phonetic representation later in this paper for convenience (though $\boldsymbol{z}_n$ might cover more than phonetic traits).

Next, we need a decoder $f_\theta(\cdot)$ to reconstruct speaker-dependent frames. For that purpose, we introduce the speaker representation $\boldsymbol{y}_n$ as another latent variable, and concatenate it to $\boldsymbol{z}_n$. The decoder then utilizes the joint vector $(\boldsymbol{y}_n, \boldsymbol{z}_n)$ to reconstruct a speaker-dependent frame $\hat{\boldsymbol{x}}_n$ ($\hat{\boldsymbol{x}}_{s,n}$ or $\hat{\boldsymbol{x}}_{t,m}$, depending on $\boldsymbol{y}_n$):

$$\hat{\boldsymbol{x}}_n = f_\theta(\boldsymbol{z}_n, \boldsymbol{y}_n). \tag{3}$$

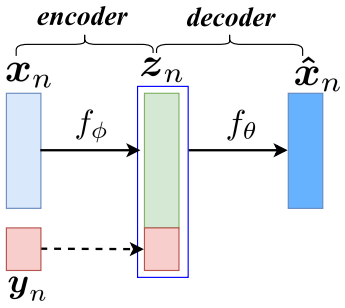

Fig. 1: Illustration of VAE-based non-parallel spectral conversion. The dashed line means copying. The latent variable $\hat{\boldsymbol{z}}_n$ and $\boldsymbol{y}_n$ are concatenated.

To sum up, reformulation is achieved by substituting $f(\cdot)$ in (1) with $f_\theta(\cdot)$ in (3) and then $\boldsymbol{z}_n$ with (2):

$$\hat{\boldsymbol{x}}_n = \hat{f}(\boldsymbol{x}_n, \boldsymbol{y}_n) = f_\theta(\boldsymbol{z}_n, \boldsymbol{y}_n) = f_\theta(f_\phi(\boldsymbol{x}_n), \boldsymbol{y}_n). \tag{4}$$

Alignment plays no roles in this formulation because the encoder-decoder pair accepts a frame $\boldsymbol{x}_n$ and a speaker representation $\boldsymbol{y}_n$ on a frame-wise basis. It then puts together the phonetic representation $\boldsymbol{z}_n$ and the speaker representation $\boldsymbol{y}_n$ to synthesize a frame $\hat{\boldsymbol{x}}_n$. Fig. 1 depicts the structure.

The framework's viability relies on two assumptions. First, we assume that speaker representation and phonetic representation can be decoupled from a given frame. Second, we assume that the decoder can blend the two factors (phonetic and speaker identity) to synthesize a spectral frame.

B. Architecture

We modify variational auto-encoder (VAE) [18], [19] to tackle the problem of SC from unaligned data. A VAE is a directed probabilistic model realized in the form of neural networks. We choose the variational over vanilla auto-encoder because the former has a more understandable model for the latent space and better a regularization property. We have described the basic concepts and the auto-encoder part in the previous section. Now we shall elaborate some of the details, including the training objective and the inference process.

We regard SC as a generative process of VAE, and therefore try to maximize joint log-probability of individual frames:

$$\log p_\theta(\boldsymbol{X}) = \sum_{n=1}^{N} \log p_\theta(\boldsymbol{x}_n). \tag{5}$$

The individual log-probability of VAE can be re-written as:

$$\begin{aligned}\log p_\theta(\boldsymbol{x}_n) = D_{KL}(q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n)||p(\boldsymbol{z}_n|\boldsymbol{x}_n)) \\ + \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}_n),\end{aligned} \tag{6}$$

where $q_\phi(\cdot)$ is the variational posterior and $p(\cdot)$ is the true posterior. The first right-hand-side (RHS) term $D_{KL}(\cdot||\cdot)$ is the Kullback-Leibler divergence (KLD) of the approximate from the true posterior. The second RHS term is called the

variational lower bound on the marginal probability and can be further rewritten as:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}_n) = -D_{KL}(q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n)||p(\boldsymbol{z}_n)) + \mathbf{E}_{q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n)}[\log p_\theta(\boldsymbol{x}_n|\boldsymbol{z}_n)]. \quad (7)$$

Direct optimization of (6) is usually intractable, so we instead take the variational lower bound (7) as our objective function. The goal is to differentiate and optimize the lower bound w.r.t. the encoder parameters $\boldsymbol{\phi}$ and decoder parameters $\boldsymbol{\theta}$. We will first descibe how to estimate the expectation term which is the cost of induced by latent space modeling. Then, we will derivate the closed-form expression for the KLD term.

1) Estimating the Expectation Term: Sampling methods are frequently adopted to estimate the expectation term in (7):

$$\mathbf{E}_{q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n)}[\log p_\theta(\boldsymbol{x}_n|\boldsymbol{z}_n, \boldsymbol{y}_n)] \approx \sum_{l=1}^{L} \log p_\theta(\boldsymbol{x}_n|\boldsymbol{z}_n, \boldsymbol{y}_n), \quad (8)$$

where L is the number of samples drawn per frame. However, naive sampling is usually problematic, so we resort to the re-parameterization trick [19]. We sample from the distribution of $\boldsymbol{z}_n$ by generating a standard normal random variable and apply a data-driven deterministic function to it:

$$\begin{aligned} \hat{\boldsymbol{z}}_n &\sim \mathcal{N}(\boldsymbol{z}_n; \boldsymbol{\mu}_{\boldsymbol{z}_n}, diag(\boldsymbol{\sigma}^2_{\boldsymbol{z}_n})), \\ \boldsymbol{\epsilon}_n &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \\ \boldsymbol{\mu}_{\boldsymbol{z}_n} &= f_{\boldsymbol{\phi}_1}(\boldsymbol{x}_n) = \boldsymbol{z}_n, \\ \log \boldsymbol{\sigma}_{\boldsymbol{z}_n} &= f_{\boldsymbol{\phi}_2}(\boldsymbol{x}_n), \\ \Rightarrow\ \hat{\boldsymbol{z}}_n &= f_\phi(\boldsymbol{x}_n) = \boldsymbol{\mu}_{\boldsymbol{z}_n} + \boldsymbol{\epsilon}_n \circ \boldsymbol{\sigma}_{\boldsymbol{z}_n}, \end{aligned} \quad (9)$$

where $\circ$ denotes Hadamard (element-wise) product, $f_{\boldsymbol{\phi}_1}$ and $f_{\boldsymbol{\phi}_2}$ are non-linear functions made of feed-forward neural networks, and $\boldsymbol{\phi} = \{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2\}$ is the set of encoder parameters. With re-parameterization, the expectation term in (7) is approximated by:

$$\begin{aligned} &\mathbf{E}_{q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n)}[\log p_\theta(\boldsymbol{x}_n|\boldsymbol{z}_n, \boldsymbol{y}_n)] \\ &= \mathbf{E}_{\mathcal{N}(\boldsymbol{z}_n; \boldsymbol{\mu}_{\boldsymbol{z}_n}, diag(\boldsymbol{\sigma}_{\boldsymbol{z}_n}))}[\log p_\theta(\boldsymbol{x}_n|\boldsymbol{z}_n, \boldsymbol{y}_n)] \\ &= \mathbf{E}_{\mathcal{N}(\boldsymbol{\epsilon}_n; \mathbf{0}, \boldsymbol{I})}[\log p_\theta(\boldsymbol{x}_n|\hat{\boldsymbol{z}}_n, \boldsymbol{y}_n)] \\ &\approx \sum_{l=1}^{L} \log p_\theta(\boldsymbol{x}_n|\hat{\boldsymbol{z}}_n, \boldsymbol{y}_n). \end{aligned} \quad (10)$$

We simplify (10) by setting L to 1, resulting in the final approximated objective function of an individual frame:

$$\hat{\mathcal{L}}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}_n) = -D_{KL}(q_\phi(\hat{\boldsymbol{z}}_n|\boldsymbol{x}_n)||p(\boldsymbol{z}_n)) + \log p_\theta(\boldsymbol{x}_n|\hat{\boldsymbol{z}}_n, \boldsymbol{y}_n). \quad (11)$$

2) Modeling the Latent Space: The prior distribution of latent variable $\boldsymbol{z}_n$ can be thought of as our imagination of the origin of the visible variable $\boldsymbol{x}_n$, and the KLD in (7) can be deemed as a term that regularizes the latent variable not to distribute too differently from the chosen prior of $\boldsymbol{z}_n$. Our choice of $\boldsymbol{z}_n$ is an isotropic standard normal distribution, which concords with [19]. Thanks to the choice of Gaussian latent variable, the KLD term (cost of the latent variable) can be evaluated in closed-form:

$$\begin{aligned} &-D_{KL}(q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n)||p(\boldsymbol{z}_n)) \\ &= -D_{KL}(\mathcal{N}(\hat{\boldsymbol{z}}_n; \boldsymbol{\mu}_{\boldsymbol{z}_n}, diag(\boldsymbol{\sigma}_{\boldsymbol{z}_n}))||\mathcal{N}(\boldsymbol{z}_n; \mathbf{0}, \boldsymbol{I})) \\ &= \frac{1}{2}\sum_{d=1}^{D}(1 + \log \sigma^2_{\boldsymbol{z}_{n,d}} - \mu^2_{\boldsymbol{z}_{n,d}} - \sigma^2_{\boldsymbol{z}_{n,d}}), \end{aligned} \quad (12)$$

where D is the dimension of the latent space.

3) Modeling the Visible Space: We assume that the visible variable of our features (log-spectrum) obeys Gaussian distribution with a diagonal variance matrix:

$$\begin{aligned} \hat{\boldsymbol{x}}_n &\sim \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}_{\boldsymbol{x}_n}, \boldsymbol{\sigma}_{\boldsymbol{x}_n}), \\ \boldsymbol{\mu}_{\boldsymbol{x}_n} &= f_{\boldsymbol{\theta}_1}(\boldsymbol{z}_n, \boldsymbol{y}_n), \\ \log \boldsymbol{\sigma}_{\boldsymbol{x}_n} &= f_{\boldsymbol{\theta}_2}(\boldsymbol{z}_n, \boldsymbol{y}_n), \end{aligned} \quad (13)$$

where $f_{\boldsymbol{\theta}_1}$ and $f_{\boldsymbol{\theta}_2}$ are non-linear functions made of feed-forward neural networks, and $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2\}$ is the set of decoder parameters. The log-probability term in (11) can therefore be expressed in closed-form:

$$\begin{aligned} &\log p_\theta(\boldsymbol{x}_n|\hat{\boldsymbol{z}}_n, \boldsymbol{y}_n) = \log \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}_{\boldsymbol{x}_n}, diag(\boldsymbol{\sigma}_{\boldsymbol{x}_n})) \\ &= -\frac{1}{2}\sum_{d=1}^{D}\Big(\log(2\pi\sigma^2_{\boldsymbol{x}_{n,d}}) + \frac{(x_d - \mu_{\boldsymbol{x}_{n,d}})^2}{\sigma^2_{\boldsymbol{x}_{n,d}}}\Big), \end{aligned} \quad (14)$$

where D is the dimension of the visible (feature) space.

The final objective function can be obtained by substituting (14) and (12) into (11). Training is equivalent to iteratively finding the parameters that maximize the variational lower bound:

$$\{\boldsymbol{\theta}^*, \boldsymbol{\phi}^*\} = \underset{\boldsymbol{\theta}, \boldsymbol{\phi}}{\operatorname{argmax}}\ \hat{\mathcal{L}}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{X}). \quad (15)$$

We use stochastic gradient descent (SGD) for optimization in our implementation.

4) Conducting Conversion: Spectral conversion is straightforward since we merely have to specify $\boldsymbol{y}_n$ that corresponds to the desired target. The encoder first transforms the input frame into a latent representation, and next, the decoder transforms ($\boldsymbol{z}_n$, $\boldsymbol{y}_n$) into $\hat{\boldsymbol{x}}_n$. Note that sampling is not needed in the conversion phase.

C. Training Procedures

The training procedures of a VAE-based SC system differ from those of a conventional system. First, a speaker representation $\boldsymbol{y}_n$ has to be introduced to train the decoder. It can be as simple as a one-hot vector, pre-defined for each speaker, or a probability vector. We will describe the speaker representation using the one-hot vector in the following paragraphs. Second, training a VAE involves *sampling* from a probability distribution of a latent variable $\boldsymbol{z}_n$, and this means injection of stochasticity. Third, training the VAE is *point-wise* as opposed to *pair-wise* in conventional systems. That is, $\boldsymbol{x}_{s,n}$ and $\boldsymbol{x}_{t,m}$ are no longer discriminated with the former being input and the latter being output; they are both viewed as $\boldsymbol{x}_n$. The source

and the target sets are deemed as one unified set, and the speaker identity of each frame is added to the training set:

$$(\boldsymbol{X}, \boldsymbol{Y}) = \{(\boldsymbol{x}_n, \boldsymbol{y}_n)\}_{n=1}^{N=N_s+N_t}. \quad (16)$$

Since our proposed method is an auto-encoder that reconstructs the input, training is conducted by feeding a pair of a spectral frame and its corresponding speaker identity $(\boldsymbol{x}_n, \boldsymbol{y}_n)$ into the auto-encoder. Note that the speaker identity of every utterance is always known in most speech corpora for the purpose of voice conversion. Hence, the speaker identity of every frame is also known. Consequently, our proposed framework can explicitly utilize speaker identity as an additional input.

The encoder treats every incoming frame in the same way as if the speaker identity is unknown; It transforms an input frame into a speaker-independent latent phonetic representation. As the encoder receives frames from both the source and the target, it cultivates the ability of speaker-independent encoding.

Subsequently, the decoder reconstructs the input from the latent representations. It first samples from the distribution of the code (latent variable $\boldsymbol{z}_n$), and then reconstructs the input with the aid of speaker representation $\boldsymbol{y}_n$.

Finally, costs defined on the visible and the latent variables are computed and jointly optimized, and the network parameters are updated iteratively. The training procedures would terminate when it reached maximum generation probability.

III. Experiments

A. Experimental Settings

1) The VCC2016 Speech Corpus: The proposed SC system was evaluated on a parallel English corpus from the Voice Conversion Challenge 2016 [20]. There are 5 male and 5 female speakers in this corpus. Each speaker has 150 utterances as the training set and 12 utterances as the evaluation set. The evaluation set was aligned, and the objective evaluations were conducted on this set. Five out of the ten speakers are designated to be the conversion targets (2 female and 3 male speakers) and the other five sources (3 female and 2 male speakers). The testing set comprises 54 utterances per target speaker, and we use this testing set to generate converted voices for subjective evaluation.

We conducted experiments on a subset of the speakers. Two speakers were chosen as sources (SF1 and SM1) and another two as targets (TF2 and TM3). We further divided the training set into disjoint (non-parallel) subsets to train one of the VAE variants in Sec. III-C2. We reported two types of spectral conversion: intra-gender and cross-gender.

2) Feature Sets: We used the STRAIGHT toolkit [21] to exctract speech parameters, including the STRAIGHT spectra (SP for short), aperiodicity (AP), and pitch contours (F0). The FFT length was set to 1024, so the resulting AP and SP were both 513-dimensional. The frame shift was 5 ms and the frame length was 25 ms. We did not incorporate contextual or dynamic features into the feature set. Every input frame of SP was normalized to unit-sum, and the normalizing factor (energy) was taken out as an independent feature and was not modified. The SP was converted using our proposed method or the baseline systems. Note that we further applied logarithm on SP in our proposed method, whereas we used linear (non-negative) SP in the baseline systems. All systems converted F0 using the same linear mean-variance transformations on log-F0 domain. The AP was kept unmodified. After spectral conversion, energy was compensated back to SP, and STRAIGHT took in all the parameters to synthesize utterances.

B. Baseline Systems

The baseline systems were built on Exemplar-based Non-negative Matrix Factorizations (ENMF) using parallel data. The systems were similar to those described in [9]. The dictionaries were 512 or 3000 randomly selected source-target pair frames and thus the baseline systems were labeled ENMF-512 and ENMF-3000, respectively.

In baseline systems, each parallel training set was aligned using dynamic time warping (DTW) with 24-ordered Mel-cepstral coefficients (MCC) extracted from SP. After alignment, the length of a source utterance remained the same while some frames from the target were duplicated or decimated. Next, energy-based voice activity detection (VAD) was used to exclude the silence segments.

These baseline systems require no training. They convert a spectral frame by optimizing self-reconstruction criterion; in the process, they obtain an activation matrix which is the weights of linear combination of dictionary bases. The activation matrix is then applied to a parallel dictionary of the target to convert into his or her spectral frame. As a result, conversion can be conducted on-line.

C. Variational Auto-Encoders

1) Configurations and Hyper-parameters: The encoder and the decoder were feed-forward neural networks with 2 hidden layers, each with 512 nodes.[1] Rectifier linear units (ReLU) [23] were applied to each layer to provide non-linearity (except for output layers $\boldsymbol{z}_n$ and $\hat{\boldsymbol{x}}_n$, which were linear). The latent (phonetic) space was 64-dimensional. The size of a mini-batch was 128. The optimizer was ADAM [24]. The dimension of speaker representation is identical to the number of speakers in the training subset (2 for one-to-one conversion and 4 for a unified, multiple-speaker conversion).

The visible space of log-spectrum feature was modeled by a Gaussian distribution (as in (13)). We ignored variance modeling and adopted an identity matrix for it because variance did not affect the generative process in our system. The desired prior distribution for the latent variables was an isotropic standard normal distribution (as in (9)).

2) Three Variants: We report SC results of three variants of the proposed framework. The first system, referred to as VAE-pair, was built from a single source and a single target, each with 150 utterances. The second, labeled VAE-multi, was built from the whole training subset of 4 speakers. The last, labeled VAE-disj, was built from non-parallel data. Its

[1] We implemented our systems using Tensorflow's Python API [22].

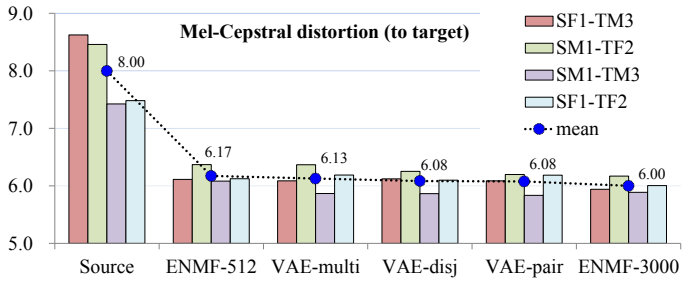

Fig. 2: Mean Mel-cepstral distortion of the proposed method compared to baseline systems that have access to alignment information. The figure is arranged according to mean MCD.

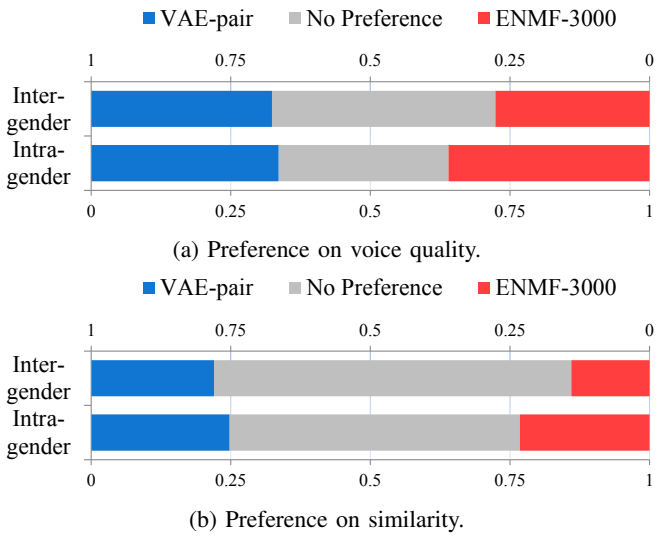

Fig. 3: Preference on voice quality and similarity to the target. The target is TF2 (female) and the source is SF1 and SM1 for intra- and inter-gender conversion, respectively.

training set consisted of the first 75 utterances of the source and the other 75 from the target. We shall clarify three things. First, VAE-pair and VAE-multi were trained using *parallel but unaligned data* while VAE-disj was trained using *non-parallel data*. Second, the size of the training sets of VAE-disj was roughly halved because the set of sentences from the source and that from the target were mutually exclusive.

D. Objective Evaluations

We visualize mean Mel-cepstral distortion (MCD) values on the evaluation set in Fig. 2. Our proposed methods trained on unaligned data performed on par with the baselines which utilized aligned frames. The results might imply that all the systems achieved comparable level of performance. As MCD was not a representative indicator for perception, we further conducted subjective evaluations on voice quality and similarity.

E. Subjective Evaluations

As for subjective evaluation, we chose ENMF-3000 as the baseline because it offered higher quality of synthetic voice than ENMF-512. We evaluated our proposed method (VAE-pair) by listening tests. Ten listeners were invited to evaluate the results. We divided our experiments into inter- and intra-gender conversion. Every listener was asked to evaluate a mean opinion score (MOS) on voice quality and ABX tests on voice quality and target similarity. The results are shown in Fig. 3.

The ABX test on target similarity revealed that both systems performed at a comparable level. This result was anticipated, and was consistent with the MCD objective evaluation. As for voice quality, our proposed method also achieved similar level as the ENMF-3000 baseline. VAE-pair achieved 2.76 MOS (with standard deviation 0.44) while ENMF-3000 achieved 2.75 MOS (with standard deviation 0.50). This result was rather encouraging since we initially conjectured that the performance degradation would be somewhat higher because VAE-pair used unaligned training data. Note that the voice quality of ENMF-3000 was rather acceptable (unlike that of ENMF-512, which was at the brink of satisfaction). More subjective evaluations on VAE-multi and VAE-disj will be conducted in our future work.

F. Training from Non-parallel Corpora

We were surprised that the performance of VAE-disj was around the same as VAE-pair in the objective evaluations (cf. Fig. 2), since the training condition of the former was apparently harsher than the latter. While the experiment verified that our framework was applicable to non-parallel corpora, it also pointed out some issues. For example, the capability of the models might not have been fully exploited because the size of the training set of VAE-pair was twice that of VAE-disj. We shall investigate the cause more profoundly in the future.

G. Toward Many-to-Many Voice Conversion

From Fig. 2, we also observed that the performance of VAE-multi was close to that of VAE-pair in the objective evaluations. It is interesting in two aspects. First, the two systems share nearly identical setting of hyper-parameters. The model of VAE-multi had to learn much more complex functions as it consolidated many pair-wise systems into one. Second, VAE-multi is virtually able to convert any of the 12 permutations of the 4 speakers, i.e., VAE-multi consolidates 12 systems into one. Its ability is evocative of many-to-many (M2M) voice conversion.

We conjectured that we could be only one step behind M2M conversion. An M2M conversion system has two requirements. First, it must be capable of convert an arbitrary, even unseen, source to a given target. Second, it must be able to convert a source to a target that never appears in the training phase, but has limited resources during conversion. Conceptually, our framework has the ability to accommodate M2M tasks. This could be achieved by introducing a speaker recognition network (in the form of another encoder) to replace the given speaker representation (one-hot vector in our case). Or, the speaker representation could be in other forms. Once the specker representation of the unknown target speaker

is obtained from the limited speech, it is likely that the decoder can blend speaker and phonetic representations to synthesize a speaker-dependent spectral frame, thus achieving M2M conversion.

IV. Conclusions

In this paper, we have introduced a VAE-based SC framework that is able to utilize unaligned data. It was an attempt toward training without the need of explicit alignment. Objective and subjective evaluations validated its ability to convert spectra, and the performance of the proposed method is comparable to baseline systems that have access to aligned data. We will continue to improve its performance, investigate its ability to accommodate many-to-many voice conversion, and generalize it to more tasks.

References

[1] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, 2007.

[2] S. Takamichi, T. Toda, A. W. Black, and S. Nakamura, "Modulation spectrum-constrained trajectory training algorithm for GMM-based voice conversion," *Proc. ICASSP*, 2015.

[3] H.-T. Hwang, Y. Tsao, H.-M. Wang, Y.-R. Wang, and S.-H. Chen, "Incorporating global variance in the training phase of GMM-based voice conversion," *Proc. APSIPA*, 2013.

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

[5] E. Godoy, O. Rosec, and T. Chonavel, "Voice conversion using dynamic frequency warping with amplitude scaling, for parallel or nonparallel corpora," *IEEE Trans. Audio, Speech, Lang., Process*, vol. 20, no. 4, pp. 1313–1323, May. 2012.

[6] S. Desai, A. W. Black, B. Yegnanarayana, and K. Prahallad, "Spectral mapping using artificial neural networks for voice conversion," *IEEE Trans. Audio, Speech, Lang., Process.*, vol. 18, no. 5, pp. 954–964, 2010.

[7] T. Nakashika, R. Takashima, T. Takiguchi, and Y. Ariki, "Voice conversion in high-order eigen space using deep belief nets," *Proc. INTERSPEEH*, 2013.

[8] H.-T. Hwang, Y. Tsao, H.-M. Wang, Y.-R. Wang, and S.-H. Chen, "A probabilistic interpretation for artificial neural network-based voice conversion," *Proc. APSIPA*, 2015.

[9] Z. Wu, T. Virtanen, E. S. Chng, and H. Li, "Exemplar-based sparse representation with residual compensation for voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 1506–1521, 2014.

[10] Y.-C. Wu, H.-T. Hwang, C.-C. Hsu, Y. Tsao, and H.-M. Wang, "Locally linear embedding for exemplar-based spectral conversion," *Proc. INTERSPEECH*, in press.

[11] M. Dong, C. Yang, Y. Lu, J. W. Ehnes, D. Huang, H. Ming, R. Tong, S. W. Lee, and H. Li, "Mapping frames with DNN-HMM recognizer for non-parallel voice conversion," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2015, Hong Kong, December 16-19, 2015*. IEEE, 2015, pp. 488–494. [Online]. Available: http://dx.doi.org/10.1109/APSIPA.2015.7415320

[12] M. Zhang, J. Tao, J. Tian, and X. Wang, "Text-independent voice conversion based on state mapped codebook," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2008, March 30 - April 4, 2008, Caesars Palace, Las Vegas, Nevada, USA*. IEEE, 2008, pp. 4605–4608. [Online]. Available: http://dx.doi.org/10.1109/ICASSP.2008.4518682

[13] P. Song, W. Zheng, and L. Zhao, "Non-parallel training for voice conversion based on adaptation method," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013*. IEEE, 2013, pp. 6905–6909. [Online]. Available: http://dx.doi.org/10.1109/ICASSP.2013.6639000

[14] D. Erro, A. Moreno, and A. Bonafonte, "INCA algorithm for training voice conversion systems from nonparallel corpora," *IEEE Trans. Audio, Speech & Language Processing*, vol. 18, no. 5, pp. 944–953, 2010. [Online]. Available: http://dx.doi.org/10.1109/TASL.2009.2038669

[15] H. Benisty, D. Malah, and K. Crammer, "Non-parallel voice conversion using joint optimization off alignment by temporal context and spectral distortion," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4-9, 2014*. IEEE, 2014, pp. 7909–7913. [Online]. Available: http://dx.doi.org/10.1109/ICASSP.2014.6855140

[16] Y. Agiomyrgiannakis, "The matching-minimization algorithm, the INCA algorithm and a mathematical framework for voice conversion with unaligned corpora," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016, Shanghai, China, March 20-25, 2016*. IEEE, 2016, pp. 5645–5649. [Online]. Available: http://dx.doi.org/10.1109/ICASSP.2016.7472758

[17] H. Ney, D. Sündermann, A. Bonafonte, and H. Höge, "A first step towards text-independent voice conversion," in *INTERSPEECH 2004 - ICSLP, 8th International Conference on Spoken Language Processing, Jeju Island, Korea, October 4-8, 2004*. ISCA, 2004.

[18] D. P. Kingma, D. J. Rezende, S. Mohamed, and M. Welling, "Semi-supervised learning with deep generative models," *CoRR*, vol. abs/1406.5298, 2014. [Online]. Available: http://arxiv.org/abs/1406.5298

[19] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *ICLR*, 2014. [Online]. Available: http://arxiv.org/abs/1406.5298

[20] T. Toda, L.-H. Chen, D. Saito, F. Villavicencio, M. Wester, Z. Wu, and J. Yamagishi, "The voice conversion challenge 2016," *Proc. INTERSPEECH*, in press.

[21] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive timefrequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Commun.*, no. 3-4, pp. 187–207, 1999.

[22] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[23] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," *Proc. ICML*, p. 807–814, 2010.

[24] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *Proc. ICLR*, 2015.