One-shot Voice Conversion with Disentangled Representations by Leveraging Phonetic Posteriorgrams

Seyed Hamidreza Mohammadi, Taehwan Kim

ObEN, Inc.

hamid@oben.com, taehwan@oben.com

Abstract

We propose voice conversion model from arbitrary source speaker to arbitrary target speaker with disentangled representations. Voice conversion is a task to convert the voice of spoken utterance of source speaker to that of target speaker. Most prior work require to know either source speaker or target speaker or both in training, with either parallel or non-parallel corpus. Instead, we study the problem of voice conversion in nonparallel speech corpora and one-shot learning setting. We convert an arbitrary sentences of an arbitrary source speaker to target speakers given only one or few target speaker training utterances. To achieve this, we propose to use disentangled representations of speaker identity and linguistic context. We use a recurrent neural network (RNN) encoder for speaker embedding and phonetic posteriorgram as linguistic context encoding, along with a RNN decoder to generate converted utterances. Ours is a simpler model without adversarial training or hierarchical model design and thus more efficient. In the subjective tests, our approach achieved significantly better results compared to baseline regarding similarity.

Index Terms: voice conversion, disentangled representations, phonetic posteriorgrams

1. Introduction

Voice Conversion (VC) [1, 2] is a task to convert source speaker's spoken sentences into those of a target speaker's voice. It requires to keep the target speaker's identity while preserving linguistic context spoken by the source speaker. Voice conversion can be applied in many applications such as speaking aids [3, 4], speech enhancement and style conversion [5, 6].

To tackle voice conversion, many approaches have been proposed [7, 8, 9]. However, most prior work require parallel spoken corpus and enough amount of data to learn the target speaker's voice. Recently, there were approaches proposed for voice conversion with non-parallel corpus [10, 11, 12, 13]. But they still require that speaker identity was known *priori*, or included in training data for the model.

Recently, Hsu et al. [14] proposed to use disentangled and interpretable representations to overcome these limitations by exploiting Factorized Hierarchical Variation Autoencoder (FHVAE). They achieved reasonable similarity with just single utterance from a target speaker but it was still not satisfactory. This architecture was also applied to cross-lingual VC [15]. The proposed architecture learns to disentangle speaker and linguistic context embeddings. Thus, the model has to learn both of these aspects. However, training both representations is still a challenging task as unsupervised learning of disentangled representations may be difficult without inductive biases [16]. Therefore, we propose to leverage the acoustic model of a well-trained Automatic Speech Recognition (ASR) model to circumvent learning the context embedding. Phonetic Posteriorgrams (PPGs) have been previously used in VC to provide a speaker-independent representation for mapping purposes [17]. We use the senone posteriorgrams that are the output of the acoustic model as context embedding. The goal of our model is to learn only the speaker embedding in a way to reduce the reconstruction loss when combined with the PPG context embedding. Since the ASR is trained on a large ASR corpus it has seen a good variation of speaker and recording conditions. In addition, the model is trained on a supervised manner and this helps having a more robust context embedding. Therefore, we hypothesize that PPG can be used as context embedding in our model. In our experiments we found that our model was able to have disentangled representations of speaker identity without further assumptions or constraints. Our contributions are:

- We propose a model to learn the disentangled representation of speaker embedding from an acoustic feature sequence. We use a RNN encoder to learn the speaker embedding from acoustic feature sequence. This embedding vector is then fed to a RNN decoder along with PPGs to reconstruct the acoustic features. We train this model using a multi-speaker speech corpus. This model does not require parallel data in training the representation model or during voice conversion.
- We examine the efficacy of the model by performing subjective experiments and find significant improvement, specially with respect to speaker similarity.

2. Related Work

Voice conversion approaches can be divided to two categories: parallel and non-parallel models. For parallel voice conversion, a representative approach is spectral conversion such as Gaussian mixture models (GMMs) [7] and deep neural networks (DNN) [8]. However, these require parallel spoken corpus. Dynamic time warping (DTW) is usually used to align source and target utterances, which can be potentially error-prone. To overcome this limitation, non-parallel voice conversion approaches were proposed, for instance, *eigenvoice* [9], *i-vector* [18], and Variational Autoencoder (VAE) [10, 12, 14] based and adversarial learning based [19, 20, 13] models. However, eigenvoice based approach [9] still requires reference speaker to train the model. In i-vector based approach [18], the i-vectors are converted by replacing the source latent variable by the target latent variable and used for the acoustic feature conversion. However, it was shown that i-vector performed worse than the latent code from the VAE based model [14].

Generative Adversarial Networks have been successful as a deep generative model. Adversarial learning has been applied for non-parallel voice conversion [19, 20] with the cycle consistency constraint [21], but it still has limitation of requiring to know the target speaker in training time and be trained for each source to target speaker pair. Still a model may be trained

for multiple target speakers but it has the same limitations of having the model trained for known target speakers [13]. VAE is another popular generative model and also applied for voice conversion [10, 12], but speaker identities are not inferred from data and instead required to be known in model training time.

There has been many approaches to exploit disentangled representation of latent code, namely, DC-IGN [22], InfoGAN [23], β-VAE [24], and FHVAE [14]. Also using separate style and content encodings was exploited in computer vision and image synthesis problems [13, 25]. These approaches to uncover disentangled representation may help voice conversion with very limited resource from target speaker, since it might infer speaker identity information from data without supervision, as illustrated in FHVAE [14]. Siamese autoencoders have also been proposed for decomposing speaker identity and linguistic embeddings [26] but this approach requires parallel training data to learn the decomposing architecture. One of the main challenges in training the models with disentangled representation is that it requires several assumptions such as a hierarchical architecture and domain adversarial training to make training such representation feasible, and similarity of converted voices were not good enough. PPGs computed from ASR acoustic models have been previously used in VC to provide a speaker-independent representation for mapping [17] even though their VC is limited to a specific target speaker. We propose to use these PPGs as an already available disentangled representation. Hence we only focus on computing the speaker identity representation which we observe that it helps training of the model.

3. Model

Our proposed model consists of an encoder and a decoder. The encoder is a recurrent neural network (RNN) and takes acoustic features as input and outputs an speaker embedding vector. The decoder is another RNN which takes the generated speaker embedding along with PPGs as input, and generate the acoustic features.

Let acoustic observations from one utterance be $\mathbf{X} = \{x_1, x_2, \ldots, x_{N_x}\} \in \mathbb{R}^{D_x \times N_x}$, corresponding PPG observations $\mathbf{P} = \{p_1, p_2, \ldots, p_{N_p}\} \in \mathbb{R}^{D_p \times N_p}$, and speaker embedding $z \in \mathbb{R}^{D_z}$. For simplicity, we assume $N_x = N_p$ and denote it as N. Then we would like to model $p(z|\mathbf{X})$ and $p(\mathbf{X}|\mathbf{P}, z)$. These are realized by:

$$z = E(\mathbf{X})$$
$$X' = D(\mathbf{P}, z)$$

where E is the encoder, D is the decoder, and $\mathbf{X}' = \{x'_1, x'_2, \ldots, x'_N\} \in \mathbb{R}^{D_x \times N}$ is the reconstructed speech features. The proposed model is illustrated in Figure 1. We train the model by optimizing the training loss:

$$\ell(X, X') = \Sigma_{i=1}^{N} ||x_i - x'_i||_2^2$$

When performing the voice conversion, we estimate the average $\bar{z}^{src}$ and $\bar{z}^{trg}$ from the given utterance(s) of source and target speakers by feeding speech features sub-sequences to the encoder. For a given input utterance, we compute phonetic posteriorgram $\mathbf{P}$ of the input utterance from the source speaker. There are two ways to perform voice conversion. First, we can replace z^{src} values of the source speaker with the average $\bar{z}^{trg}$ from the target speaker. This approach resulted in too muffled generated result. Second, we compute a difference vector between source and target average $\bar{z}^{diff} = \bar{z}^{trg} - \bar{z}^{src}$. This

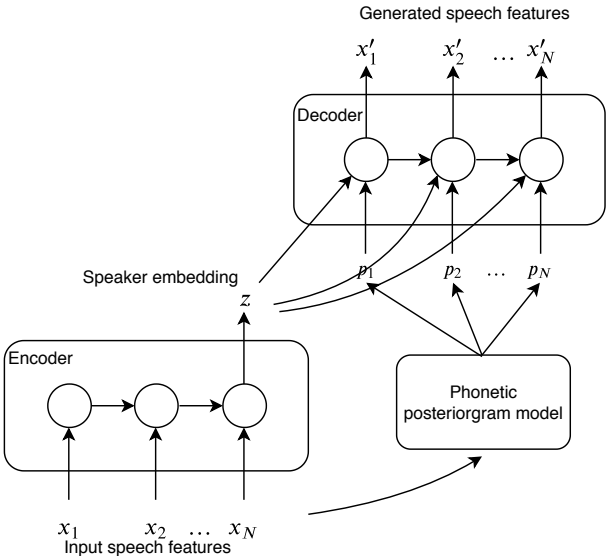

Figure 1: *Our proposed model. Encoder takes speech features* $\mathbf{X}$ *and outputs speaker embedding* z*. Decoder takes estimated speaker embedding* z *and phonetic posteriorgram* $\mathbf{P}$ *and generates speech features* $\mathbf{X}'$*. The detail of our network architecture is in Table 1.*

difference vector is added to z^{src} from the input utterance as $z^{converted} = z^{src} + \bar{z}^{diff}$ and then decoded with P to generate the speech features. In an informal listening test, we decided to do the second approach since it resulted in significantly higher quality generated speech. We suspect that the reason is $\bar{z}^{src}$ might not be completely disentangled so $z^{converted}$ works better than $\bar{z}^{trg}$.

4. Experiments

4.1. Datasets

We used the TIMIT corpus [27] which is a multi-speaker speech corpus as the training data for both our model and the baseline. We used the training speakers as suggested by the corpus to train the models. For test speakers, we select four speakers from TIMIT testing part of the corpus. Finally, for objective testing (which requires availability of parallel data), we utilized four CMU-arctic voices (BDL, SLT, RMS, CLB)[28]. As speech features, we used 40th-order MCEPs (excluding the energy coefficient, dimension D=39), extracted using the World toolkit [29] with a 5ms frame shift. All audio files are transformed to 16kHz and 16 bit before any analysis.

To compute the phonetic posteriorgrams, we use Kaldi ASR toolkit [30]. We use librispeech as speech corpus [31]. We use the nnet2 recipe in Kaldi to build the ASR model. We compute the acoustic model output to obtain the senone posteriorgrams from the audio files.

4.2. Experimental setting

For the encoder in our proposed model, we use Gated Recurrent Units (GRUs) [32] as the first layer followed by a fully-connected layer with Rectified Linear Units (ReLU) activation function on top of the last hidden representation to compute the speaker embedding z. The input to the encoder is the acoustic

feature sequence. For the decoder, it has two inputs: PPGs and speaker embedding z. We feed the PPGs to a fully-connected layer followed by ReLU. Then it was concatenated with the speaker embedding z and we feed it to a GRU. Finally, the output of GRU is fed to a fully-connected layer to get generated output. Detailed description of our model structure is in Table 1. The models were trained with stochastic gradient descent. We use a mini-batch size of 128. The Adam optimizer [33] is used with β_1 = 0.95, β_2 = 0.999, $\epsilon = 10^{-8}$, and initial learning rate of 10^{-3}. The model is trained for 100 epochs and select the model best performing on a development set chosen among training set. For training the FHVAE model, we followed the training configuration in [15].

Encoder	
recurrent layer	GRU-1024, Dropout
output layer	FC-D_z, ReLU

Decoder	
dense block	input PPGs, FC-1024, ReLU, Dropout
combine layer	dense output + speaker embedding z
dense block	FC-1024, ReLU, Dropout
recurrent layer	GRU-1024, Dropout
output layer	FC-D_x

Table 1: *The network architectures of our encoder and decoder models. FC indicates fully-connected layer. ReLU indicates ReLU activation.*

In our experiments, we consider two models: FHVAE [15], and our proposed model. We consider four gender conversions (F: female, M: male): F2F, F2M, M2F, M2M. The voice conversion samples are available at: https://shamidreza.github.io/is19samples. Training of our model takes about 5 hours whereas training of FHVAE takes about two days.

4.3. Visualizing embeddings

In this experiment, we investigate the speaker embeddings z by visualizing them in Figure 2. For visualizing the speaker embeddings, we use 5 sentences from 40 test speakers from TIMIT test set (blue data points represents males and red represents female). We show FHVAE and proposed model computed speaker embeddings in Figures 2-left and 2-right, respectively. In both subplots, the female and male embedding cluster locations are separated. We observe that proposed model's computed speaker embeddings for different speakers fall further apart compared to FHVAE. Also they are more evenly distributed compared to VAE embeddings which tend to be more densely distributed. The gender clusters have a better separation margin. This subjectively depicts a more robust speaker embedding quality. Furthermore, we perform an experiment where we change the dimensionality D_z of the speaker embedding and visualize the embeddings. We use D_z = 8, 16, and 32 shown in Figures 3-top, 3-middle, and 3-bottom, respectively. We observe that the pattern of encodings is similar. For dimension of 8, the female embedding are further apart compared to dimension of 32, however, male embeddings are more dense in the same comparison. Dimension of 16 has a more even distribution for both male and female speakers. For all other experiments, we use $D_z = 16$.

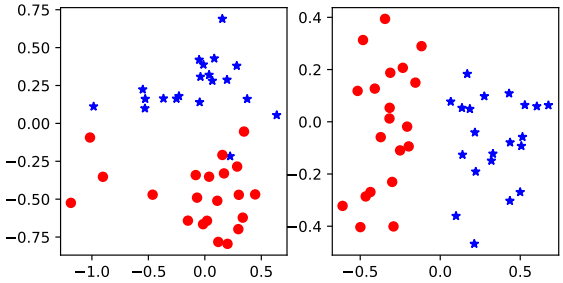

Figure 2: *Visualization of speaker embedding. Each point represents a speaker; blue dots are males and red dots are females. The embeddings are transformed to 2D using PCA. FHVAE (left) vs. Proposed (right).*

4.4. Effect of training data size

We investigate the effect of VC training data size on the performance of the system. In order to be able to do objective test using mel-CD [7], we require parallel data from the speakers. We use 20 parallel CMU-arctic utterance from each speaker for computing the objective score. We vary non-parallel sentence numbers from source and target speaker that is used to compute the speaker embeddings. The results are shown in Figure 4. As can be seen, proposed approach performs better consistently for all sentence sizes.

4.5. Subjective evaluation

To subjectively evaluate voice conversion performance, we performed two perceptual tests. The first test measured speech quality, designed to answer the question "how natural does the converted speech sound?", and the second test measured speaker similarity, designed to answer the question "how accurate does the converted speech mimic the target speaker?". The listening experiments were carried out using Amazon Mechanical Turk, with participants who had approval ratings of at least 90% and were located in North America. Both perceptual tests used three trivial-to-judge trials, added to the experiment to exclude unreliable listeners from statistical analysis. We use one training utterance from target speaker for all results reported here.

4.5.1. Speech quality

To evaluate the speech quality of the converted utterances, we conducted a Comparative Mean Opinion Score (CMOS) test. In this test, listeners heard two stimuli A and B with the same content, generated using the same source speaker, but in two different processing conditions, and were then asked to indicate whether they thought B was better or worse than A, using a five-point scale comprised of +2 (much better), +1 (somewhat better), 0 (same), -1 (somewhat worse), -2 (much worse). We randomized the order of stimulus presentation, both the order of A and B, as well as the order of the comparison pairs. We utilized two processing conditions: FHVAE and proposed. We assessed the VC approach effect by directly comparing Proposed vs. FHVAE utterances. The experiment was administered to 40 listeners with each listener judging 50 sentence pairs. The results are shown in Figure 5. We found statistical significant preference scores for F2M condition achieving $P < 0.05$ in one-sample

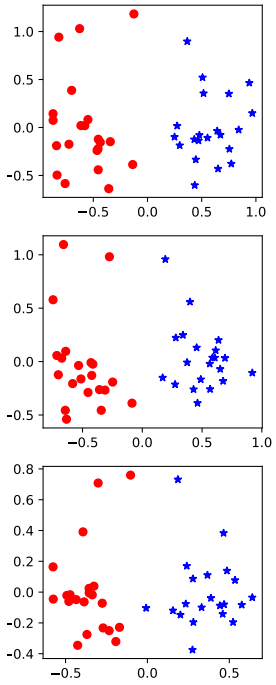

Figure 3: *Visualization of speaker embedding from our model with different dimensions D_z. Each point represents a speaker: blue dots are males and red dots are females. The embeddings are transformed to 2D using PCA. $D_z = 8$ (top), $D_z = 16$ (middle), and $D_z = 32$ (bottom).*

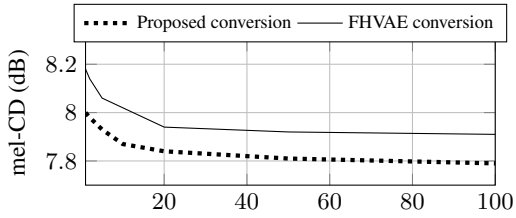

Figure 4: *Effects of varying number of training sentences from 1 to 100*

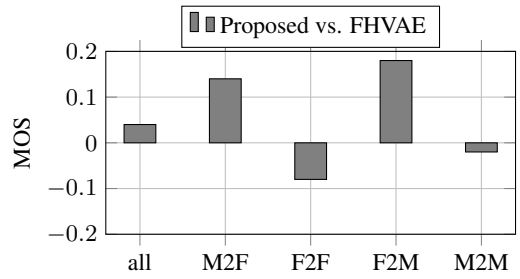

Figure 5: *Speech Quality average score with gender breakdown. Positive scores favor proposed model. F2M preference score is statistically significant.*

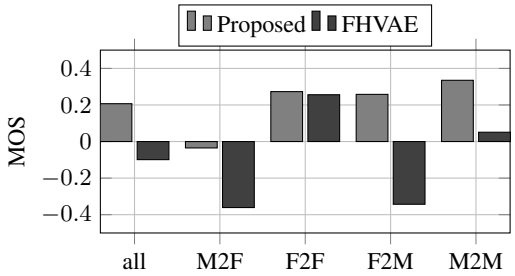

Figure 6: *Speaker Similarity average score with gender breakdown. Positive scores are desirable. (confidence intervals for all is close to 0.11, and all score-pairs are statistically significant)*

t-test compared to chance. We speculate this is due to F2M conversions having higher similarity as shown in next subsection, resulting in better acoustics and F0 matching together to sound more natural to human subjects. We did not find statistically significant difference for other comparisons, although M2F was close to significance.

4.5.2. Speaker similarity

To evaluate the speaker similarity of the converted utterances, we conducted a same-different speaker similarity test [34]. In this test, listeners heard two stimuli A and B with different content, and were then asked to indicate whether they thought that A and B were spoken by the same, or by two different speakers, using a five-point scale comprised of +2 (definitely same), +1 (probably same), 0 (unsure), -1 (probably different), and -2 (definitely different). One of the stimuli in each pair was created by one of the two conversion methods, and the other stimulus was a purely MCEP-vocoded condition, used as the reference speaker. The listeners were explicitly instructed to disregard the content of the stimuli and merely judge based on the fact whether they think the utterances are from the same speaker regardless of the content. Half of all pairs were created with the reference speaker identical to the target speaker of the conversion (expecting listeners to reply "same", ideally); the other half were created with the reference speaker being the same gender, but not identical to the target speaker of the conversion (expecting listeners to reply different). We only report "same" scores. The experiment was administered to 40 listeners, with each listener judging 50 sentence pairs. The results are shown in Figure 6. The results show proposed model and FHVAE achieving 0.20±0.11 and -0.10±0.12, respectively. We found that the proposed model performs statistically significantly better than FHVAE in all comparison pairs, all achieving $P < 0.05$ in two-sample t-test.

5. Conclusion

We proposed voice conversion model from arbitrary source speaker to target speaker with disentangled representations. We learn disentangled representation of speaker identity by providing linguistic context encoding, which makes the model simpler and thus more efficient to train. We use a RNN encoder for speaker embedding and phonetic posteriorgram computed from an ASR acoustic model as linguistic context encoding, followed with a RNN decoder. We performed objective and subjective experiments, and in the subjective tests, we showed that our approach achieved statistically significantly better similarity compared to the baseline.

6. References

[1] Y. Stylianou, "Voice transformation: a survey," in *Proceedings of ICASSP*, 2009.

[2] S. H. Mohammadi and A. Kain, "An overview of voice conversion systems," *Speech Communication*, vol. 88, pp. 65–82, 2017.

[3] A. B. Kain, J.-P. Hosom, X. Niu, J. P. Van Santen, M. Fried-Oken, and J. Staehely, "Improving the intelligibility of dysarthric speech," *Speech communication*, vol. 49, no. 9, pp. 743–759, 2007.

[4] K. Nakamura, T. Toda, H. Saruwatari, and K. Shikano, "Speaking-aid systems using gmm-based voice conversion for electrolaryngeal speech," *Speech Communication*, vol. 54, no. 1, pp. 134–146, 2012.

[5] Z. Inanoglu and S. Young, "Data-driven emotion conversion in spoken english," *Speech Communication*, vol. 51, no. 3, pp. 268–283, 2009.

[6] T. Toda, M. Nakagiri, and K. Shikano, "Statistical voice conversion techniques for body-conducted unvoiced speech enhancement," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 9, pp. 2505–2517, 2012.

[7] 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*, vol. 15, no. 8, pp. 2222–2235, 2007.

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

[9] Z. Wu, T. Kinnunen, E. S. Chng, and H. Li, "Mixture of factor analyzers using priors from non-parallel speech for voice conversion," *IEEE Signal Processing Letters*, 2012.

[10] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," in *Proceedings of ICASSP APSIPA*, 2016.

[11] P. Song, W. Zheng, and L. Zhao, "Non-parallel training for voice conversion based on adaptation method," in *Proceedings of ICASSP*, 2013.

[12] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks," *arXiv preprint arXiv:1704.00849*, 2017.

[13] J.-c. Chou, C.-c. Yeh, H.-y. Lee, and L.-s. Lee, "Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations," *arXiv preprint arXiv:1804.02812*, 2018.

[14] W.-N. Hsu, Y. Zhang, and J. Glass, "Unsupervised learning of disentangled and interpretable representations from sequential data," in *Advances in neural information processing systems*, 2017, pp. 1876–1887.

[15] S. H. Mohammadi and T. Kim, "Investigation of using disentangled and interpretable representations for one-shot cross-lingual voice conversion," in *Proceedings of Interspeech*, 2018.

[16] F. Locatello, S. Bauer, M. Lucic, S. Gelly, B. Schölkopf, and O. Bachem, "Challenging common assumptions in the unsupervised learning of disentangled representations," *arXiv preprint arXiv:1811.12359*, 2018.

[17] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *Proceedings of ICME*, 2016.

[18] T. Kinnunen, L. Juvela, P. Alku, and J. Yamagishi, "Non-parallel voice conversion using i-vector plda: Towards unifying speaker verification and transformation," in *Proceedings of ICASSP*, 2017.

[19] T. Kaneko and H. Kameoka, "Parallel-data-free voice conversion using cycle-consistent adversarial networks," *arXiv preprint arXiv:1711.11293*, 2017.

[20] F. Fang, J. Yamagishi, I. Echizen, and J. Lorenzo-Trueba, "High-quality nonparallel voice conversion based on cycle-consistent adversarial network," in *Proceedings of ICASSP*, 2018.

[21] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *arXiv preprint arXiv:1703.10593*, 2017.

[22] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum, "Deep convolutional inverse graphics network," in *Advances in Neural Information Processing Systems*, 2015, pp. 2539–2547.

[23] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2016, pp. 2172–2180.

[24] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," 2016.

[25] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, "Toward multimodal image-to-image translation," in *Advances in Neural Information Processing Systems*, 2017, pp. 465–476.

[26] S. H. Mohammadi and A. Kain, "Siamese autoencoders for speech style extraction and switching applied to voice identification and conversion," *Proceedings of Interspeech*, pp. 1293–1297, 2017.

[27] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "Darpa timit acoustic-phonetic continous speech corpus cd-rom. nist speech disc 1-1.1," *NASA STI/Recon technical report n*, vol. 93, 1993.

[28] J. Kominek and A. W. Black, "The cmu arctic speech databases," in *Fifth ISCA Workshop on Speech Synthesis*, 2004.

[29] 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.

[30] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," IEEE Signal Processing Society, Tech. Rep., 2011.

[31] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[32] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[34] A. B. Kain, "High resolution voice transformation," 2001.