Learning Structured Sparse Representations for Voice Conversion

Shaojin Ding, Guanlong Zhao, *Member, IEEE*, Christopher Liberatore, *Member, IEEE*, and Ricardo Gutierrez-Osuna, *Senior Member, IEEE*

***Abstract*—Sparse-coding techniques for voice conversion assume that an utterance can be decomposed into a sparse code that only carries linguistic contents, and a dictionary of atoms that captures the speakers' characteristics. However, conventional dictionary-construction and sparse-coding algorithms rarely meet this assumption. The result is that the sparse code is no longer speaker-independent, which leads to lower voice-conversion performance. In this paper, we propose a Cluster-Structured Sparse Representation (CSSR) that improves the speaker independence of the representations. CSSR consists of two complementary components: a Cluster-Structured Dictionary Learning module that groups atoms in the dictionary into clusters, and a Cluster-Selective Objective Function that encourages each speech frame to be represented by atoms from a small number of clusters. We conducted four experiments on the CMU ARCTIC corpus to evaluate the proposed method. In a first ablation study, results show that each of the two CSSR components enhances speaker independence, and that combining both components leads to further improvements. In a second experiment, we find that CSSR uses increasingly larger dictionaries more efficiently than phoneme-based representations by allowing finer-grained decompositions of speech sounds. In a third experiment, results from objective and subjective measurements show that CSSR outperforms prior voice-conversion methods, improving the acoustic quality of the synthesized speech while retaining the target speaker's voice identity. Finally, we show that the CSSR captures latent (i.e., phonetic) information in the speech signal.**

***Index Terms*—Voice conversion, sparse coding, sparse representation, dictionary learning.**

I. Introduction

VOICE conversion (VC) aims to transform the speech of a source speaker to sound as if a target speaker had produced it. VC finds use in a number of applications, such as personalized text-to-speech synthesis [1], pronunciation training [2], and speaker spoofing [3]. Various approaches have been proposed to perform VC. Statistical parametric methods based on Gaussian Mixture Models (GMMs) [4], [5] and Deep Neural Networks (DNNs) [6]–[10] are widely used and can achieve convincing results. A promising alternative to GMMs and DNNs are methods based on sparse representations [11]–[13]. A typical method based on sparse representations consists of a dictionary construction step (to encode the speaker's identity) and a sparse coding step (to encode the content of an utterance). During training, dictionaries consisting of pairs of source and target speech frames are constructed from a parallel training corpus of time-aligned utterances. At runtime, the sparse representation of a source spectrum is computed with respect to the source dictionary, and then the target spectrum is approximated by multiplying the source sparse representation with the target's dictionary. Sparse representation methods have several advantages: they require much smaller training corpora [12] and are more robust to noisy speech than GMMs [11]. As a result, sparse representation methods are particularly appealing in applications where collecting a large corpus is impractical or background noise is inevitable (e.g., pronunciation training [14], [15]).

Sparse representation methods assume that the dictionary captures the speaker identity (i.e., how a speaker produces the various phonetic units), and that the sparse representation is speaker-independent and captures only the linguistic content. In practice, however, satisfying this assumption is difficult. First, the atoms in the dictionary do not fully capture speaker identity, since to do so the dictionary must capture all the phonetic units (e.g., tri-phones), which is not feasible for small corpora. Second, the sparse representation is not speaker-independent – even if the dictionary contains all the phonetic units, since the standard sparse coding objective (i.e., Lasso) ignores the phonetic structure of the dictionary. Namely, the Lasso minimizes the Mean-Square-Error using as few atoms as possible (the effect of the L_1 constraint) regardless of their phonetic content, so the sparse representations of the same utterance from different speakers tend to be different. These two factors are compounded, making the sparse representations less speaker-independent. As a result, the similarity between source and target sparse representations decreases, ultimately degrading the sound quality of the VC syntheses.

To address these problems, we propose a novel Cluster-Structured Sparse Representation (CSSR) for spectral transformation in VC. CSSR consists of two components, a Cluster-Structured Dictionary Learning algorithm (CSDL) and a Cluster-Selective Objective Function (CSOF).[1] The training

Manuscript received March 24, 2019; revised September 26, 2019 and November 14, 2019; accepted November 15, 2019. Date of publication November 22, 2019; date of current version December 24, 2019. This work was supported by NSF Awards 1619212 and 1623750. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Heiga Zen. *(Corresponding author: Shaojin Ding.)*

The authors are with the Department of Computer Science and Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: shjd@tamu.edu; gzhao@tamu.edu; cliberatore@tamu.edu; rgutier@tamu.edu).

Digital Object Identifier 10.1109/TASLP.2019.2955289

[1]Initial findings from this work were presented at the 19th Annual Conference of the International Speech Communication Association (Interspeech 2018) [16], [17]. The earlier conference papers examined the above two sub-problems individually and presented preliminary results, respectively. In this manuscript, we consider the two sub-problems jointly and propose a state-of-the-art sparse representation-based VC framework. We also describe our methods in full detail and significantly expand the validation experiments and analysis of results.

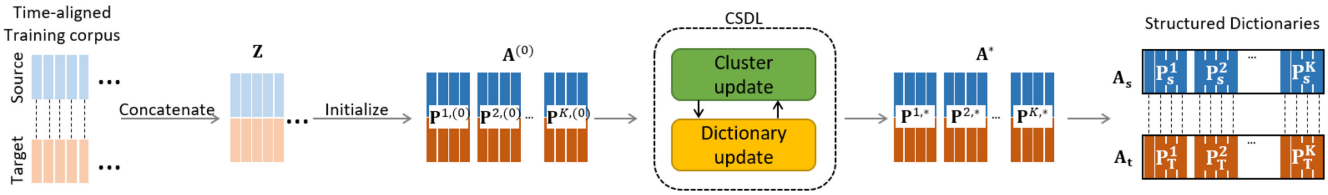


Fig. 1. Training phase of CSSR. Source and target utterances from training corpus are first time-aligned using dynamic time warping. The time-aligned frames are then concatenated, and the structured dictionaries are randomly initialized using the concatenated frames as $\mathbf{A}^{(0)}$. Then, CSDL performs two steps (cluster update and dictionary update) iteratively until convergence. The optimal structured dictionaries, $\mathbf{A}^*$, are then split into a source dictionary $\mathbf{A_s}$ and a target dictionary $\mathbf{A_t}$.

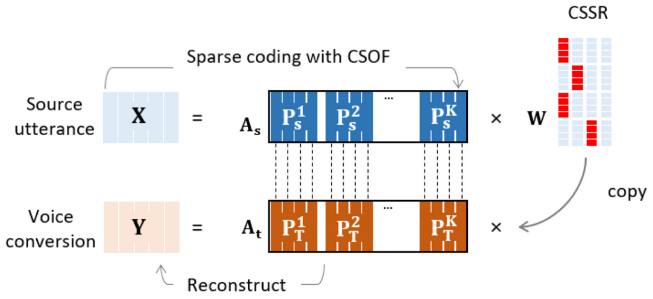


Fig. 2. Testing phase of CSSR. The CSSR $\mathbf{W}$ for the source utterance $\mathbf{X}$ is computed relative to the source structured dictionary $\mathbf{A_s}$. The converted utterance is then generated by multiplying the CSSR $\mathbf{W}$ with the target structured dictionary $\mathbf{A_t}$.

and runtime processes are as shown in Fig. 1 and Fig. 2. During training, and given a time-aligned corpus, CSDL uses a hard-decision Expectation Maximization algorithm to learn a family of "structured" sub-dictionaries, where atoms (i.e., pairs of source-target acoustic frames) within each sub-dictionary (or cluster) are acoustically similar. At runtime, and given the structured source dictionary that was learned, we compute a structured sparse code for the source utterance by optimizing the CSOF, which uses the $L_{2,1}$ norm [18] to promote group sparsity and therefore tends to represent each speech frame using atoms from as few clusters as possible. Finally, we multiply the source's structured sparse code with the target's structured dictionary to generate the voice-converted utterance.

We conducted four experiments on the CMU ARCTIC corpus [19] to evaluate the proposed method: an ablation study to examine the effectiveness of each component in CSSR, an experiment to evaluate the performance of CSSR as a function of the number of atoms in the dictionary, a set of objective and subjective studies to compare the proposed method against baselines from previous studies, and a final set of visualizations and phonetic analyses of CSSR. The results of the ablation study show that both CSDL and CSOF can reduce the difference between source and target sparse representations and improve VC performance, and that combining both components leads to further performance improvements. In addition, results from the second experiment show that CSSR uses increasingly larger dictionaries more efficiently than phoneme-based representations by allowing finer-grained decompositions of speech sounds. Next, results from the objective and subjective studies show that CSSR significantly improves the acoustic quality when compared to the baseline systems. In our final analyses, results show that CSSR is phonetically interpretable.

The rest of the manuscript is organized as follows. Section II reviews mainstream methods for VC, structured dictionary learning and sparse coding, how previous VC methods improve the speaker independence of sparse representations, and the relation of the proposed method to previous work. Section III describes the proposed method, including the overall VC framework, CSDL, and CSOF. Section IV describes the experimental setup, including the corpus and the details in our implementation. Section V shows the results for four experiments. Finally, we conclude the paper with a thorough discussion of the results.

II. Literature Review

A. Voice Conversion Algorithms

Statistical parametric models such as GMMs and DNNs are among the most common algorithms for VC. GMM-based methods [1], [4] learn the joint distribution of source and target short-time spectra and then estimate the target spectral features through least-squares regression. However, the basic GMM-based method suffers from over-smoothing issues [5], [20] on the generated feature sequences. To address this problem, Toda *et al.* [5] proposed to use maximum likelihood parameter generation (MLPG) as a post-processing step for GMM-based methods. Furthermore, global variance (GV) is often combined with MLPG to increase the quality of the synthesized speech [5].

By contrast, DNN-based methods map the source spectral features directly into the target space through various network structures such as restricted Boltzmann machines [6], auto-encoders [7], feed-forward neural networks [9], and recurrent neural networks [10]. More recently, Phonetic Posteriorgrams [21], [22] from acoustic models, generative models including Generative Adversarial Networks [23], [24] and Variational Auto-Encoders [8], [25], [26] have been shown to enhance VC performance. These methods can solve more generalized VC problems such as many-to-many VC and non-parallel VC, but they require relatively large corpora. Other statistical models such as partial least squares [27] and Hidden Markov Models [28] have also shown success in VC tasks.

Methods based on non-parametric sparse representations have received much attention in recent years. Unlike statistical parametric methods, sparse representation methods require much smaller training corpora and are more robust to noisy speech. Takashima *et al.* [11] first applied sparse representations to

perform VC in noisy environments. Following this work, subsequent studies focused on improving either the dictionary construction or the sparse coding process. Wu *et al.* [12] improved the original sparse representation by using both high-resolution and low-resolution features to capture spectral details and enforce temporal continuity. Zhao and Gutierrez-Osuna [29] proposed different strategies to construct more compact and effective dictionaries, while Fu *et al.* [30] used a dictionary learning algorithm to improve the effectiveness of the dictionary. Aihara *et al.* [13], [31], Sisman *et al.* [32], and Liberatore *et al.* [33] incorporated phonetic information in both dictionary construction and sparse coding, which enhanced the speaker independence of the sparse representations. Other innovations have also dramatically improved the quality of sparse representation-based VC. Wu *et al.* [34] and Liberatore *et al.* [35] showed that warping the source residual and adding it to the estimated target spectra can also significantly improve the VC syntheses quality. Wu *et al.* [36] generalized MLPG and GV into sparse representation methods via an approximation algorithm, which also improved the quality of the converted speech.

B. Structured Sparse Coding and Dictionary Learning

Signals such as images and speech are highly correlated and always have internal structures. However, the standard sparse coding objective functions (i.e., Lasso) do not consider any prior information about the internal structure of the data. To take such information into account, various structured-sparse representations have been proposed. Yuan *et al.* [37] first proposed the Group Lasso based on distinct groups (e.g., variables of different categories) and provided two algorithms to solve the Group Lasso. Group-sparse representations have also been generalized to include trees and graph structures [38]–[40]. Accordingly, a number of algorithms have been proposed to learn dictionaries with group structures, such as the Alternating Minimization fashioned algorithm [41], Proximal methods [42], and online dictionary learning algorithms [43]. Given the internal structures of the data, these structured sparse representations are more flexible and accurate than conventional sparse representations. The structured sparse representations have proven to be successful in various computer vision and speech processing tasks such as face recognition [43], [44], image classification [41], [45], speech enhancement [46], [47], speech recognition [48], and source separation [49].

C. Improving the Speaker Independence of the Sparse Representations in Voice Conversion

Several previous studies have proposed solutions to improve the speaker independence of the sparse representation. Aihara *et al.* first examined this problem and provided different solutions [13], [31], [50]. In [13], they used phoneme information to regularize the sparse representation and attempt to make it speaker-independent. Namely, they categorized the atoms into sub-dictionaries according to their phoneme labels and then selected different sub-dictionaries to represent the speech frames. In [50], they proposed an activity-mapping non-negative matrix factorization algorithm to introduce mappings between the source and target sparse representations. To further reduce the computational complexity while enforcing speaker independence, they proposed a parallel dictionary learning algorithm [31] with a graph-embedded discriminative constraint. Sisman *et al.* [32] followed [13] in building phoneme-categorized dictionary but selected sub-dictionaries using phoneme labels at runtime, which also improved the speaker independence of the sparse representations. In related work, Liberatore *et al.* [33] used the centroids of each phoneme as atoms and constructed a more compact dictionary. The more compact dictionary prevented the source and target sparse representations from becoming too different, which implicitly improved the speaker independence.

D. Relation to Prior Work

Our proposed method differs from prior studies in several respects. First, CSDL learns the dictionaries directly from the data, without any supervised information (e.g., phoneme labels [13], [31], [32], etc.) It avoids the use of forced-alignment or automatic speech recognition to generate the labels, thus reducing computation. Second, CSDL is based on "hard-decision Expectation Maximization" algorithms [51]–[55] commonly used for learning models that depend on unobserved latent variables, which is different from previous dictionary learning algorithms [38], [41], [43] and previous dictionary learning VC methods [30], [31]. Finally, we use a CSOF to *implicitly* encourage the sparse coding algorithm to represent a speech frame using a compact set of atoms from a few clusters, rather than using a sub-dictionary selection procedure [13] or phoneme labels [32] at runtime. Additionally, CSDL is seamlessly connected to CSOF. CSDL learns a cluster-structured dictionary, and CSOF enforces the group-sparsity on the structured dictionary. The resulting structured sparse representation captures the internal structure of speech signals, which makes the representation more speaker-independent. As a result, the VC performance is significantly improved.

III. Methods

In the following sub-sections, we first introduce the entire VC framework based on CSSR. Then, we provide a detailed derivation of the two components of CSSR: CSDL and CSOF.

A. Voice Conversion Framework

First, we describe the conventional sparse representation method used in VC. During training, a source dictionary $\mathbf{A_s} \in \mathbb{R}^{D\times N}$ and a target dictionary $\mathbf{A_t} \in \mathbb{R}^{D\times N}$ are learned from time-aligned parallel utterances, where N is the number of atoms in each dictionary, and each atom is a D-dimensional vector. Note that the requirement of parallel utterances can be relaxed by using alignment algorithms such as those in [15], [56]. At runtime, an L-frame source utterance $\mathbf{X} \in \mathbb{R}^{D\times L}$ is represented as,

$$\mathbf{X} \approx \mathbf{A_s}\mathbf{W} \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{N \times L}$ is a sparse non-negative weight matrix (i.e., a sparse representation). Given $\mathbf{X}$ and $\mathbf{A_s}$, $\mathbf{W}$ can be approximated via solving standard sparse coding objective (i.e., Lasso):

$$\mathbf{W} = \underset{\mathbf{W}}{\operatorname{argmin}}\, d(\mathbf{A_s}, \mathbf{W}) + \alpha||\mathbf{W}||_1, \quad \text{s.t.} \quad \mathbf{W} \geq 0 \tag{2}$$

where $d(\cdot)$ is a distance metric, typically the KL-divergence or the Euclidean distance. The L_1 norm term is included to enforce sparsity in $\mathbf{W}$, with α being a sparsity penalty. Given $\mathbf{A_t}$ and $\mathbf{W}$, a target utterance $\hat{\mathbf{Y}} \in \mathbb{R}^{D \times L}$ can be generated as:

$$\hat{\mathbf{Y}} = \mathbf{A_t}\mathbf{W} \tag{3}$$

Voice conversion using CSSR. Compared to the conventional sparse representation used for VC, CSSR further considers that the speech signal has an internal structure (i.e., phonetic). Assume that the spectral frames of a speaker can be divided into K clusters. During training, we use the CSDL algorithm (described in Section III-B) to learn the structured dictionaries $\mathbf{A_s}$ and $\mathbf{A_t}$, each containing K sub-dictionaries:

$$\mathbf{A_s} = [\mathbf{P_s^1}, \mathbf{P_s^2}, \ldots, \mathbf{P_s^K}] \tag{4}$$

$$\mathbf{A_t} = [\mathbf{P_t^1}, \mathbf{P_t^2}, \ldots, \mathbf{P_t^K}] \tag{5}$$

where $\mathbf{P_s^i} \in \mathbb{R}^{D \times M}$ and $\mathbf{P_t^i} \in \mathbb{R}^{D \times M}$ denote the source and the target sub-dictionaries corresponding to the i-th cluster, respectively, and $i \in \{1, 2, \ldots, K\}$. M is the number of atoms in a sub-dictionary.

At runtime, once the structured dictionaries have been learned, we generate the CSSR $\mathbf{W}$ by jointly minimizing the objective function in eq. (2) and CSOF $\mathbf{\Psi}(\mathbf{W})$:

$$\mathbf{W} = \underset{\mathbf{W}}{\operatorname{argmin}}\, d(\mathbf{X}, \mathbf{A_s}\mathbf{W}) + \alpha||\mathbf{W}||_1 + \beta\mathbf{\Psi}(\mathbf{W}), \text{s.t.} \mathbf{W} \geq 0 \tag{6}$$

where β is a penalty term for $\mathbf{\Psi}(\mathbf{W})$. CSOF is based on the $L_{2,1}$ norm; see Section III-C for details. CSOF *implicitly* encourages the sparse coding algorithm to represent a speech frame using atoms from as few clusters as possible, which as we will later show to encode phonetic information (see Section V-D). With the target dictionary $\mathbf{A_t}$ and the computed CSSR $\mathbf{W}$, we then use eq. (3) to estimate the target spectrum.

B. Cluster-Structured Dictionary Learning

Let $\mathbf{X} \in \mathbb{R}^{D \times L}$ and $\mathbf{Y} \in \mathbb{R}^{D \times L}$ denote the time-aligned source and target training utterances. Following Fu *et al.* [30], we concatenate the time-aligned source and target training utterances as $\mathbf{Z} = [\mathbf{X^T}, \mathbf{Y^T}]^\mathbf{T}$. Our goal is to learn a concatenated dictionary $\mathbf{A} = [\mathbf{A_s^T}, \mathbf{A_t^T}]^\mathbf{T}$, where $\mathbf{A_s}$ and $\mathbf{A_t}$ consist of sub-dictionaries, as defined in eqs. (4–5). For notational simplicity, we define the concatenated sub-dictionary as $\mathbf{P}^i = [\mathbf{P_s^{iT}}, \mathbf{P_t^{iT}}]^\mathbf{T}$, and $\mathbf{A} = [\mathbf{P}^1, \mathbf{P}^2, \ldots, \mathbf{P}^K]$. We solve this dictionary-learning problem through an iterative algorithm. At each iteration, we perform two steps: a cluster update and a dictionary update. Details of each step are provided in following subsections. The overall algorithm is summarized in Algorithm 1.

1) Cluster Update: We denote the concatenated dictionary and the i-th sub-dictionary at the t-th iteration as $\mathbf{A}^{(t)}$ and $\mathbf{P}^{i,(t)}$.

Algorithm 1: CSDL Algorithm.

Inputs: concatenated training utterances $\mathbf{Z}$, the number of clusters K
Outputs: learned structured dictionary $\mathbf{A}^* = [\mathbf{P}^{1,*}, \mathbf{P}^{2,*}, \ldots, \mathbf{P}^{K,*}]$
Initialization: Randomly assign a cluster label to each training frame and divide the training frames to K clusters according to the cluster labels, as in eq. (10). Then initialize the dictionary $\mathbf{A}^{(\mathbf{0})} = [\mathbf{P}^{1,(\mathbf{0})}, \mathbf{P}^{2,(\mathbf{0})}, \ldots, \mathbf{P}^{K,(\mathbf{0})}]$ by solving eq. (11). Set $t = 0$.

while not converge **do**
  for l in $1, 2, \ldots, L$ **do**
    for i in $1, 2, \ldots, K$ **do**
      $\mathbf{w}_l^{i,(t+1)} = \operatorname{argmin}_\mathbf{w} ||\mathbf{z}_l - \mathbf{P}^{i,(t)}\mathbf{w}||_2^2 + \lambda||\mathbf{w}||_1$
      $r_l^{i,(t+1)} = ||\mathbf{z}_l - \mathbf{P}^{i,(t)}\mathbf{w}_l^{i,(t+1)}||_2^2$
    end for
    $p_l^{(t+1)} = \operatorname{argmin}_i r_l^{i,(t+1)}$
  end for
  for i in $1, 2, \ldots, K$ **do**
    $\mathbf{Z}^{i,(t+1)} = \{\mathbb{I}(p_l^{(t+1)} = i)\mathbf{z}_l\}, \quad l = 1, 2, \ldots, L$
    $\mathbf{P}^{i,(t+1)} = \operatorname{argmin}_{\mathbf{P}^{i,(t)}} ||\mathbf{Z}^{i,(t+1)} - \mathbf{P}^{i,(t)}\mathbf{w}||_2^2 + \lambda||\mathbf{w}||_1$
  end for
  $t = t + 1$
end while
return $\mathbf{A}^* = [\mathbf{P}^{1,*}, \mathbf{P}^{2,*}, \ldots, \mathbf{P}^{K,*}]$

In the cluster update step, all the sub-dictionaries $\mathbf{P}^{i,(t)}$ are fixed. For each frame $\mathbf{z}_l$ in $\mathbf{Z}$, we assign $\mathbf{z}_l$ to the cluster whose sub-dictionary represents $\mathbf{z}_l$ with the lowest residual error. Formally, we denote the residual of $\mathbf{z}_l$ respect to $\mathbf{P}^{i,(t)}$ as,

$$r_l^{i,(t)} = ||\mathbf{z}_l - \mathbf{P}^{i,(t)}\mathbf{w}_l^{i,(t)}||_2^2 \tag{7}$$

where $\mathbf{w}_l^{i,(t)}$ are the coefficients of the sparse representation. We compute $\mathbf{w}_l^{i,(t)}$ as,

$$\mathbf{w}_l^{i,(t)} = \underset{\mathbf{w}}{\operatorname{argmin}} ||\mathbf{z}_l - \mathbf{P}^{i,(t)}\mathbf{w}||_2^2 + \lambda||\mathbf{w}||_1 \tag{8}$$

which we solve using the Least Angle Regression (LARS) [57] algorithm, and λ is the sparsity penalty. Once the residuals are updated, we can assign $\mathbf{z}_l$ a latent cluster label $p_l^{(t)}$ as,

$$p_l^{(t)} = \underset{i}{\operatorname{argmin}}\, r_l^{i,(t)} \tag{9}$$

Then, we divide $\mathbf{Z}$ into K clusters based on their labels $p_l^{(t)}$ as,

$$\mathbf{Z}^{i,(t)} = \{\mathbb{I}(p_l^{(t)} = i)\mathbf{z}_l\}, \quad l = 1, 2, \ldots, L \tag{10}$$

where $\mathbf{Z}^{i,(t)}$ denotes all the speech frames in the i-th cluster, and $\mathbb{I}(\cdot)$ is the indicator function.

2) Dictionary Update: In the dictionary update step, we fix the clusters and update the sub-dictionaries. For each $\mathbf{Z}^{i,(t)}$ (all the speech frames in the i-th cluster), we wish to find a sub-dictionary $\mathbf{P}^{i,(t+1)}$ that can represent it sparsely with minimum

residual. In other words, for each sub-dictionary $\mathbf{P}^{i,(t+1)}$ we solve the problem:

$$\mathbf{P}^{i,(t+1)} = \underset{\mathbf{P}^i}{\operatorname{argmin}} ||\mathbf{Z}^{i,(t)} - \mathbf{P}^i\mathbf{w}||_2^2 + \lambda||\mathbf{w}||_1 \tag{11}$$

which we solve using the online dictionary-learning algorithm proposed by Mairal *et al.* [58].

C. Cluster-Selective Objective Function

The proposed objective function (CSOF) is a generalization of the Phoneme-Selective Objective Function (PSOF) we proposed in previous work [16]. PSOF promotes that each speech frame is represented with atoms from a small number of phonemes, which is achieved by enforcing group sparsity on the groups defined by phoneme labels (one group per phoneme). However, PSOF is limited by the fact that phonemes are often too coarse to capture detailed information in speech (e.g., allophones). To address this issue, CSOF allows the number of groups to increase, which is achieved by enforcing group sparsity on the groups defined in the structured dictionary learned from CSDL. In practice, the most common mathematical tool to enforce group sparsity is the $L_{2,1}$ norm [18]. Therefore, we formulate the CSOF $\mathbf{\Psi}(\mathbf{W})$ as

$$\mathbf{\Psi}(\mathbf{W}) = \sum_{j=1}^{L}\sum_{k=1}^{K}\sqrt{\sum_{i=1,i\in\mathbf{P}_\mathbf{s}^k}^{M} w_{ij}^2} \tag{12}$$

where w_{ij} denotes the (i,j)-th element of the weight matrix $\mathbf{W}$, K denotes the number of sub-dictionaries, $\mathbf{P}_\mathbf{s}^k$ represents the k-th sub-dictionary in the source dictionary, L is the number of frames in the utterance, and M is the number of atoms in a sub-dictionary (see Section III-A). By minimizing CSOF, we force the weights within a sub-dictionary to be activated or suppressed at the same time, and therefore *implicitly* encourage the sparse coding algorithm to represent a spectrum frame with atoms from as few sub-dictionaries as possible.

Since eq. (12) is convex, gradient-based algorithms can still be used to optimize it. The derivative of $\mathbf{\Psi}(\mathbf{W})$ respect to w_{ij} is as,

$$\frac{\partial\mathbf{\Psi}(\mathbf{W})}{\partial w_{ij}} = \frac{w_{ij}}{\sqrt{\sum_{i=1,i\in\mathbf{P}_\mathbf{s}^k}^{M} w_{ij}} + \epsilon} \tag{13}$$

where ϵ is a small positive number that prevents the denominator from becoming zero. In all our experiments, we set $\epsilon = 10^{-6}$. Therefore, each element w_{ij} in $\mathbf{W}$ can be updated as,

$$w_{ij} = w_{ij} - \gamma\frac{\partial\Psi(W)}{\partial w_{ij}} \tag{14}$$

where γ is the step size in each iteration. In practice, sparse coding algorithms such as Non-negative Matrix Factorization (NMF) [59] and Fast Iterative Shrinkage-Thresholding (FISTA) [60] can be extended to solve this group sparse coding problem. In this paper, we adopted FISTA algorithm to solve the group sparse coding problem. The update of $\mathbf{W}$ in NMF and FISTA algorithms can be found in [59], [60].

IV. Experimental Setup

A. Corpus

We used four English speakers from the CMU ARCTIC [19] corpus: BDL (male), RMS (male), SLT (female), and CLB (female). For each speaker, we selected three sets of utterances: 20 utterances for training (about 1.5 minutes), 10 utterances for validation, and 50 utterances for testing.[2] Four VC pairs were considered for the experiments: M-M (BDL to RMS), M-F (RMS to SLT), F-F (SLT to CLB), and F-M (CLB to BDL). In what follows, all the results are averaged over these four VC pairs.

B. Implementation Details

We used the WORLD vocoder [61] (D4C edition [62]) to extract a 513-dimensional spectral envelope, fundamental frequency (F_0) and aperiodicity for each utterance with a 5 ms window shift. We computed the 25-dimensional Mel-Frequency Cepstral Coefficient (MFCC) from the WORLD spectral envelope (removing MFCC_0, which is the energy) and used the MFCCs as the acoustic feature in dictionary learning and voice conversion. Source and target utterances were time-aligned using dynamic time warping [63].

In the proposed method, we set the number of atoms in each sub-dictionary M to 100. In the first, the third, and the fourth experiments, we set the number of clusters (sub-dictionaries) K to 40, i.e., the number of phonemes in CMU ARCTIC (except for silence). In the second experiment, we explored different number of clusters, as will be described in Section V-B. For silent frames, we used a voice activity detector to find them and directly copy silent frames from source to target. We used the SPAMS sparse coding toolbox [58], [64] to solve for eqs. (6), (8) and (11). We set α, β, and λ to 0.001, 0.05, and 0.01, respectively, based on preliminary experiments [16], [17]. CSDL will stop when no more than 5% of training frames are re-assigned from one iteration to the next.

Following Toda *et al.* [5], we convert the pitch trajectory (F_0) of the source speech to match the pitch range of the target speaker using log mean variance normalization. We estimate the converted spectral envelope from the converted MFCC, and finally synthesize the converted speech using the WORLD vocoder with the converted spectral envelope, converted F_0 and source aperiodicity.

V. Results

We conducted four experiments to evaluate CSSR. The first experiment was an ablation study that examined the effectiveness of each CSSR component in reducing differences between source and target sparse representations and improving VC performance. In the second experiment we explored the effect of dictionary size and number of clusters in CSSR. In the third experiment, we evaluated the VC performance of CSSR

[2]A small number of training utterances was used to mimic a low-resource setting. Utterances for each set were selected using a maximum entropy criterion to ensure good phonetic balance.

TABLE I
THE FIVE SYSTEM CONFIGURATIONS USED IN THE ABLATION STUDY

		Dictionary construction technique		
		Random	**Phoneme**	**CSDL**
Objective function	**MSE+L_1 (Lasso)**	RDL+Lasso	PSDL+Lasso	CSDL+Lasso
	MSE+L_1+$L_{2,1}$ (CSOF)	N/A[3]	PSDL+CSOF	CSDL+CSOF

and compared it against baselines from previous studies. In a final analysis, we visualized CSSR and provided its phonetic interpretation.

A. Ablation Study

To understand how much each CSSR component contributes to reducing the sparse representations difference and improving VC performance, we conducted an ablation study that evaluates the contribution of each method: the dictionary learning algorithm and the sparse coding cost function. To do so, we compared five different system configurations; also see Table I:

- **Random Dictionary Learning (RDL) + Lasso:** a baseline system following the conventional VC framework based on sparse representations [11], which constructs dictionaries from randomly selected speech frames in training, and optimizes the Lasso (eq. (2)) at runtime.
- **Phoneme Structured Dictionary Learning (PSDL) + Lasso:** a system that constructs the structured dictionary using phoneme labels during training (as in [13], [16], [32]) and optimizes the Lasso at runtime.
- **CSDL + Lasso:** a system that uses the CSDL algorithm to learn a cluster structured dictionary in training, and optimizes the Lasso at runtime.
- **PSDL + CSOF:** a system that constructs the structured dictionaries using phoneme labels during training and optimizes the joint cost function in eq. (6) at runtime.
- **CSDL + CSOF (CSSR):** the *proposed* method: CSDL and CSOF combined.

RDL+Lasso, PSDL+Lasso, and CSDL+Lasso share the same sparse coding cost function (MSE + L_1 norm) but differ in the dictionaries: random vs. derived from phoneme labels vs. learned via CSDL. This allows us to assess the relative merits of each dictionary construction technique. PSDL+Lasso and PSDL+CSOF share the same dictionary but differ in the sparse coding cost functions. This allows us to compare the two cost functions side by side. Finally, by comparing CSSR (i.e., CSDL+CSOF) against CSDL+Lasso and PSDL+CSOF we can evaluate the benefit of combining the two proposed algorithms.

We used two metrics to evaluate the five systems: the distance between the source and target sparse representations, which measures whether the representations are speaker dependent, and the Mel-Cepstral Distortion between the synthesized speech and the ground-truth target speech:

- **Sparse Representation Distance.** As discussed by Aihara *et al.* [13], [31], the loss of speaker independence decreases the similarity between source and target sparse representations. Accordingly, we compute the difference between source and target sparse representations of time-aligned parallel utterances as,

$$D(\mathbf{W_s}, \mathbf{W_t}) = \frac{1}{L}||\mathbf{W_s} - \mathbf{W_t}||_F \tag{15}$$

where $\mathbf{W_s} \in \mathbb{R}^{N\times L}$ and $\mathbf{W_t} \in \mathbb{R}^{N\times L}$ are the source and target sparse representations, L is the number of frames, and $||\cdot||_F$ denotes the Frobenius norm. The lower this distance is, the more similar the source and target sparse representations are, and so the sparse representation tends to be more speaker-independent.

- **Mel-Cepstral Distortion (MCD).** We also measured the MCD of the voice-converted speech and its time-aligned ground-truth target speech to examine the synthesis quality. MCD is the most common objective measurement in VC systems, and is defined as,

$$\text{MCD[dB]} = \frac{10}{ln10}\sqrt{2\sum_{d=1}^{24}(\hat{y}_d - y_d)^2} \tag{16}$$

where $\hat{y}_d$ and y_d are the d-th Mel-Frequency Cepstral Coefficient (MFCC) of the converted speech and the time-aligned ground-truth target speech, respectively. Lower MCD indicates that the converted speech is closer to the ground-truth target speech.

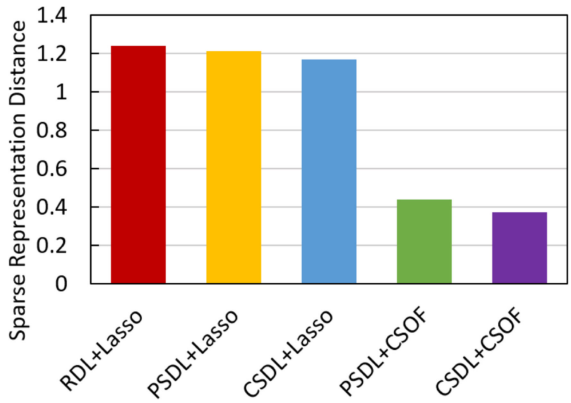


Fig. 3. Sparse representation distance of all the systems in the ablation study. As defined in eq. (15), lower distance means higher similarity between the source and target sparse representations (i.e., improved speaker independence).

Results for the sparse representation distance are shown in Fig. 3. From the results, we found that the sparse representation distance for CSSR (CSDL+CSOF) (0.37) is lower than that of the baselines: CSSR achieves 16.0% relative improvement over PSDL+CSOF (0.44), 68.4% relative improvement over CSDL+Lasso (1.17), 69.4% relative improvement over

[3]We do not consider the combination RDL+CSOF since CSOF requires a structured dictionary, which cannot be randomly selected.

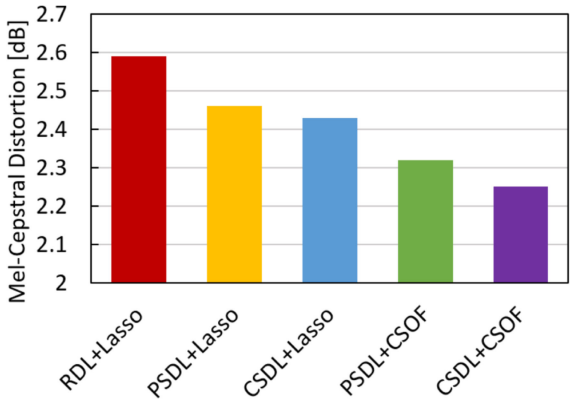


Fig. 4. Average MCD of all the systems in the ablation study. Lower MCD generally leads to better VC performance.

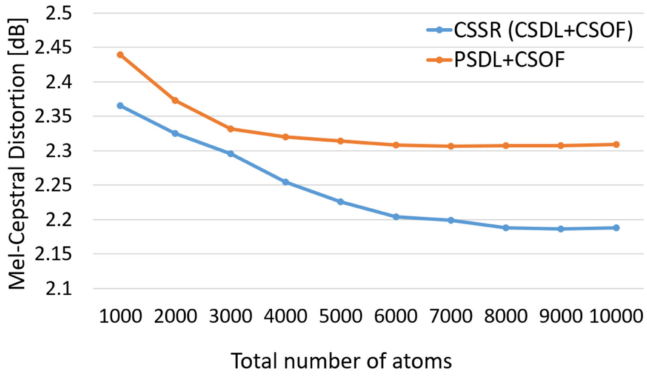


Fig. 5. Average MCD of CSSR and PSDL+CSOF with different number of atoms in total. In CSSR, we fixed the number of atoms in each sub-dictionary to 100, varying the number of sub-dictionaries from 10 to 100. In PSDL+CSOF, we fixed the number of sub-dictionaries to 40 (the number of phonemes in CMU ARCTIC, except for silence), varying the number of atoms in each sub-dictionary from 25 to 250. Lower MCD generally leads to better VC performance.

PSDL+Lasso (1.21), and 70.2% relative improvement over RDL+Lasso (1.24). These results indicate that CSSR systematically increases the similarity between the source and target sparse representations.

Additionally, our results show that the system using CSDL (CSDL+Lasso) and the system using CSOF (PSDL+CSOF) outperform their corresponding baselines (RDL+Lasso and PSDL+Lasso), respectively. These results suggest that both CSDL and CSOF are essential in reducing the representation distance. Moreover, we found that CSSR outperforms both CSDL+Lasso and PSDL+CSOF, which indicates that combining CSDL and CSOF lead to further reductions in representation distance.

Finally, we also found that the sparse coding cost function (CSOF) is more effective than the dictionary construction algorithm (CSDL) in reducing representation distance, and hence in improving speaker independence. A possible explanation for this result is that in CSDL+Lasso, the objective function (Lasso) ignores the phonetic structure of the dictionary and minimizes the Mean-Square-Error using as few atoms as possible regardless of their phonetic content.

Results for the Mel-Cepstral Distortion are shown in Fig. 4. CSSR systematically achieves lower MCD (2.25) than all the baseline systems: a 3.0% relative improvement over PSDL+CSOF (2.32), 7.4% relative improvement over CSDL+Lasso (2.43), 8.5% relative improvement over PSDL+Lasso (2.46), and 13.1% relative improvement over RDL+Lasso (2.59). These results suggest that using CSDL and CSOF individually can improve the voice-conversion syntheses, but that combining the two modules leads to further improvements. Although CSDL only achieves modest reductions in representation distance, it does significantly decrease the MCD. This result shows that the deliberately learned atoms can reduce misalignments and better capture the structure of speech (see section V-D below), which also considerably enhances the voice-conversion syntheses.

B. *Effect of Dictionary Size*

In a second experiment, we characterized the performance of CSSR as a function of the number of atoms in the dictionary. Namely, we fixed the number of atoms in each sub-dictionary (cluster) to $M = 100$ while varying the number of sub-dictionaries $K = \{10, 20, 30, \ldots, 100\}$, so the total number of atoms in the dictionary was $N = \{1000, 2000, 3000, \ldots, 10000\}$. For comparison purposes, we used PSDL+CSOF as a baseline. Because the number of sub-dictionaries in PSDL+CSOF is fixed to 40 (defined by phoneme labels in the CMU ARCTIC, except for silence), we increased the number of atoms in each sub-dictionary so the total number of atoms was equal among the two systems.

Results are shown in Fig. 5 in terms of the average MCD of the two systems as a function of the total number of atoms. In both cases, the MCD decreases with increasing dictionary size. The MCD of the baseline system is systematically higher, and reaches a plateau of 2.31 after 4,000 atoms. In contrast, the MCD of the proposed system continues to decrease past that point, stabilizing at 2.18 with 80 sub-dictionaries (8,000 atoms) or more. These results show that CSSR uses a given dictionary size more effectively by allowing a more fine-grained representation of the data (i.e., more sub-dictionaries) as the number of atoms in the dictionary increase. In other words, for a sufficiently large dictionary size, it is more effective to increase the number of sub-dictionaries (by fixing the number of atoms per cluster) than to increase the number of atoms per sub-dictionary (by fixing the number of sub-dictionaries).

C. *Voice Conversion Performance*

In a third experiment, we evaluated the voice-conversion performance of the CSSR using objective and subjective measures, and compared it against three existing systems:

- **Baseline:** A GMM-based VC method proposed in [5], which models the joint distribution of source and target speech frames.
- **System 1:** The method we proposed in [16], which constructs the structured dictionaries using phoneme labels during training and jointly optimizes the Lasso along with the $L_{2,1}$ norm (eq. (6)) at runtime.

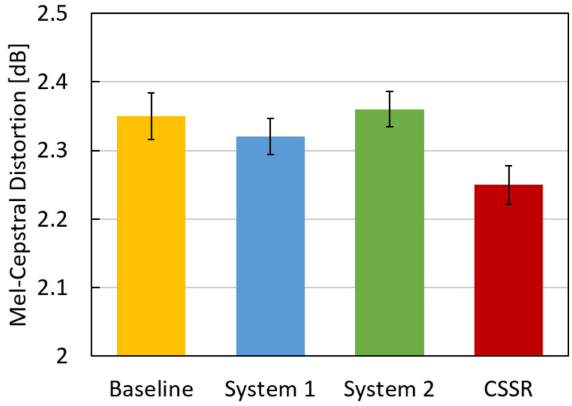


Fig. 6. Average MCD of the proposed method (CSSR) and three existing baselines (Baseline, System 1, and System 2). Lower MCD generally leads to better VC performance. The error bars show 95% confidence intervals.

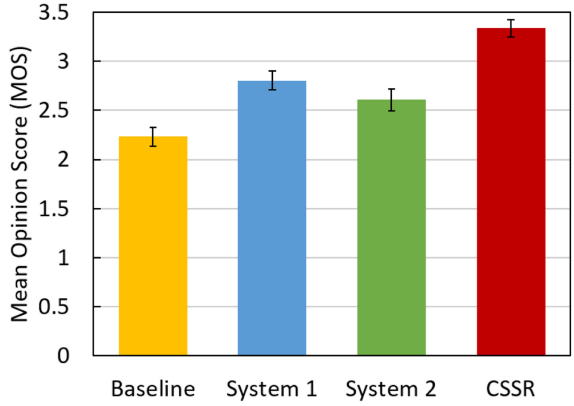


Fig. 7. Mean Opinion Scores (MOS) of the proposed method (CSSR) and three baselines (Baseline, System 1, and System 2). MOS ranges from 1 to 5, with larger MOS indicating higher acoustic quality. The error bars show 95% confidence intervals.

- **System 2:** The method we proposed in [17]. It learns the structured dictionary through CSDL during training, and it selects the most likely sub-dictionary and optimize the Lasso (eq. (2)) within the selected sub-dictionary at runtime as in [13].

By comparing the proposed method (CSSR) against the two previous systems [16], [17], we aim to determine if the two algorithms are complementary. We did not include other sparse representation-based baseline methods (e.g., [12], [13]) in the comparison, since our two previous systems [16], [17] had outperformed them. We also did not include neural network baselines since they require relatively large training corpus (e.g., [10] used 593 utterances, or about 42 mins), whereas our training corpus consists of 20 utterances (or about 1.5 minutes). Instead, we used GMM-based method, which is one of the most common methods in this low-resource setting. For all three sparse representation-based methods (CSSR, System 1 and System 2), we used 40 sub-dictionaries and 100 atoms for each sub-dictionary, following the configurations from [16], [17]. For the GMM, we used 40 mixture components, the same as the number of clusters in the proposed method to ensure a fair comparison. We did not use Maximum Likelihood Parameter Generation (MLPG) and Global Variance (GV) in any system to make the results comparable to those presented in [16], [17], but these techniques can be further incorporated to enhance the voice-conversion synthesis.[4]

1) Objective Evaluation: First, we compared the four systems by computing the MCD between the converted speech and the time-aligned ground-truth target speech with 95% confidence intervals. Fig. 6 summarizes the results. CSSR achieved the lowest MCD (2.25) and outperformed all three existing systems (System 1: 2.32, 3.0% relative improvement, single-tail t-test, $p \ll 0.001$; System 2: 2.36, 4.7% relative improvement, single-tail t-test, $p \ll 0.001$; Baseline: 2.35, 4.3% relative improvement, single-tail t-test, $p \ll 0.001$).

[4]Audio samples are available at [Online]. Available: https://shaojinding.github.io/samples/cssr/cssr_demo

2) Subjective Evaluation: In a final step, we conducted listening tests on Amazon Mechanical Turk to provide a subjective evaluation of the four systems. We measured acoustic quality with a 5-point Mean Opinion Score (MOS) test and speaker identity with a Voice Similarity Score (VSS) test ranging from -7 (definitely different speakers) to $+7$ (definitely the same speaker) [65].

Mean Opinion Score. Thirty-one participants rated 92 utterances from the four systems: 20 utterances per system, 5 utterances per speaker pair plus 12 calibration utterances to detect if participants were cheating and remove them if they did [66]. We excluded ratings of the calibration utterances from the data analysis. Fig. 7 shows the Mean Opinion Scores of the four methods with 95% confidence intervals. The proposed method (CSSR) obtains a 3.34 MOS, which is higher than that of the other three systems with statistical significance: System 1 (2.80 MOS; 19.3% relative improvement; single-tail t-test, $p \ll 0.001$), System 2 (2.61 MOS; 28.0% relative improvement; single-tail t-test, $p \ll 0.001$), and GMM (2.23 MOS; 49.8% relative improvement; single-tail t-test, $p \ll 0.001$). These results show that combining the proposed dictionary construction algorithm (CSDL) and the proposed sparse coding cost function (CSOF) improves acoustic quality more than applying each technique individually. Additionally, System 1 and System 2 achieve statistically significant improvement over the Baseline (System 1: 25.6% relative improvement, single-tail t-test, $p \ll 0.001$; System 2: 17.0% relative improvement, single-tail t-test, $p \ll 0.001$), which corresponds to the results in [16], [17].

Voice Similarity Score. Twenty-eight participants rated 140 utterance pairs: 32 pairs (16 VC-SRC and 16 VC-TGT pairs) for each system and 8 pairs (4 VC-SRC and 4 VC-TGT pairs) for each speaker pair; 12 calibration utterances. A VC-SRC pair consists of a voice-converted (VC) utterance and an utterance randomly selected from the source speaker (SRC), and a VC-TGT pair consists of a voice-converted (VC) utterance and an utterance randomly selected from the target speaker (TGT). We used the utterance that is randomly selected from the source/target speaker to avoid the interference of linguistic

TABLE II
VOICE IDENTITY RESULTS OF THE PROPOSED METHOD (CSSR) AND THE THREE REFERENCE SYSTEMS (BASELINE, SYSTEM 1, AND SYSTEM 2).VOICE SIMILARITY SCORE RANGES FROM -7 (DEFINITELY DIFFERENT SPEAKERS) TO +7 (DEFINITELY THE SAME SPEAKER). VC-SRC: VSS BETWEEN VC AND THE SOURCE SPEAKER; VC-TGT: VSS BETWEEN VC AND THE TARGET SPEAKER. ALL THE RESULTS ARE SHOWN AS AVERAGE ±95% CONFIDENCE INTERVALS

System	All pairs		Intra-gender		Inter-gender	
	VC-SRC	VC-TGT	VC-SRC	VC-TGT	VC-SRC	VC-TGT
Baseline	-5.89 ± 0.08	3.92 ± 0.18	-4.93 ± 0.14	4.68 ± 0.14	-6.85 ± 0.04	3.16 ± 0.26
System 1	-6.11 ± 0.07	4.07 ± 0.17	-5.43 ± 0.12	4.95 ± 0.16	-6.78 ± 0.04	3.18 ± 0.23
System 2	-5.80 ± 0.09	4.00 ± 0.18	-4.84 ± 0.16	4.72 ± 0.17	-6.76 ± 0.06	3.24 ± 0.22
CSSR	-5.90 ± 0.09	4.44 ± 0.17	-5.04 ± 0.15	5.32 ± 0.16	-6.76 ± 0.07	3.56 ± 0.24

content and prosody. For each utterance pair, participants were required to decide whether the two utterances were from the same speaker and then rate their confidence in the decision on a 7-point scale. Following [65], VSS is computed by collapsing the above two fields into a 14-point scale.

As shown in Table II, participants were "quite confident" that (1) CSSR utterances and source (SRC) utterances were produced by different speakers (VSS: -5.90); and that (2) CSSR utterances and target (TGT) utterances were produced by the same speaker (VSS: 4.44). When analyzing VC-SRC pairs, we found no statistically significant differences in VSS between CSSR and the other three systems (System 1: $p = 0.22$; System 2: $p = 0.36$; Baseline: $p = 0.48$; two-tail t-test). When comparing VC-TGT pairs, we found no statistically significant differences in VSS between CSSR and the other three systems (System 1: $p = 0.27$; System 2: $p = 0.22$; Baseline: $p = 0.20$; two-tail t-test). Thus, these results indicate that the four methods can produce speech that is different from the source speaker and the same as the target speaker equally well. In Table II, we also presented the VSS of the intra-gender pairs (M-M and F-F) and that of the inter-gender pairs (M-F and F-M). In all cases, we found no statistically significant difference between CSSR and the other three systems. Additionally, we found that the VC-SRC VSS of intra-gender pairs are slightly lower than that of inter-gender pairs. A possible reason is that the pitch ranges of the speakers in intra-gender pairs are closer to each other than those in inter-gender pairs. For inter-gender pairs, pitch (F0) conversion makes the voice-conversion more distinguishable from the source utterances. Moreover, we found that the VC-TGT VSS of inter-gender pairs are lower than that of intra-gender pairs, due to the fact that inter-gender voice conversion is more challenging than intra-gender voice conversion.

D. Phonetic Interpretation of CSSR

In a final analysis, we seek to provide a phonetic interpretation of CSSR. We first analyze the learned cluster-structured dictionaries by exploring the relationship between ground-truth phoneme labels and the learned clusters. In "hard-decision" algorithms, clusters commonly represent latent variables; phonetic information of speech frames can be thought of as latent variables in CSDL. Accordingly, we assigned each speech frame in the training set to the cluster that minimized its residual (eq. (7)). In parallel, we used forced-alignment to assign a phoneme label to each frame, and computed how each phoneme was distributed among the clusters. Then, we matched each phoneme to the cluster that most frequently represented it.

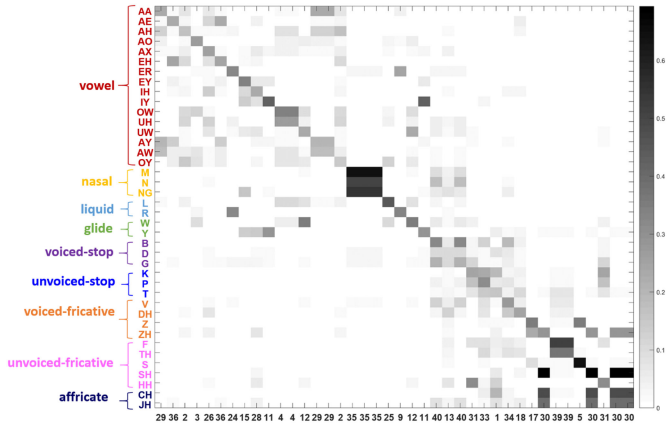


Fig. 8. Confusion matrix between forced-aligned phoneme labels and the matched clusters. Y-axis values are phonemes (sorted by the manner of articulation), and X-axis values are the cluster IDs.

The confusion matrix of ground-truth phonemes[5] vs. matched clusters is shown in Fig. 8. The dark diagonal elements indicate that each cluster is preferentially associated with a single phoneme label. Confusions do occur but are usually restricted to be within the same manner of articulation. For example, the sub-dictionary for cluster "35" represents all the nasals. Likewise, clusters "40", "31", "33", "1", "34", "18" are all used for stops. Both "15", "28" and "11" can represent /EY/, /IH/, /IY/ well, which are all front vowels. In addition, confusions also appear on phonemes that often co-occur, which can be caused by inaccurate forced alignments. For example, "9" is good at representing /ER/ and /R/, which usually co-occur in words ending with er. These results indicate that the proposed algorithm can learn the latent (i.e., phonetic) structure of speech and does it so without supervision. The learned latent structures are not restricted to phonemes but emerge directly from the data. Such structures can more accurately capture variability in pronunciations, which can further improve the similarity between source and target sparse representations than methods based on phoneme labels [13], [16], [32].

Next, we visualized the CSSR of an utterance to show that it is also phonetically meaningful. We used a similar approach as above to associate the learned sub-dictionaries (clusters) with phoneme labels, except each cluster was matched to the phoneme whose frames occurred most frequently in that cluster; this ensured that each cluster was matched with at least one

[5]We used Arpabet to represent the phonemes.

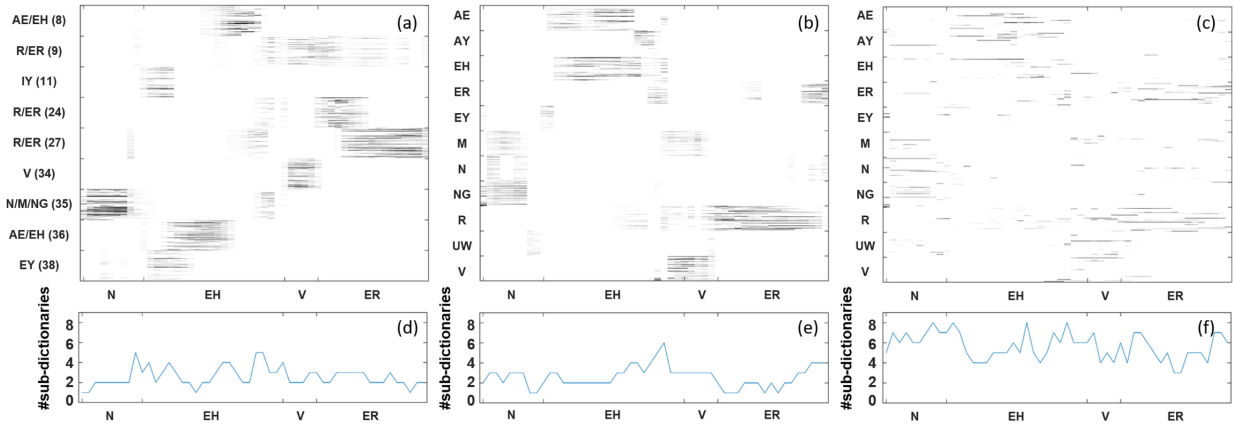


Fig. 9. Visualization of sparse representations for the word 'never'. (a) CSSR, (b) PSDL+CSOF, (c) PSDL+Lasso. The x-axis denotes the transcription of the word, and the y-axis shows the cluster labels (denoted by numbers) of the sub-dictionaries and the associated phoneme labels. (d), (e), and (f): number of sub-dictionaries that were used in the sparse representations.

phoneme. Fig. 9a shows the CSSR of the word never from speaker BDL; for clarity we only show the sub-dictionaries that were activated. As Fig. 9a shows, the associated phoneme labels of the activated sub-dictionaries correspond to the ground-truth phoneme labels of the word, indicating that CSSR is phonetically meaningful. Mismatches occur but are mostly in transitions and are restricted to adjacent clusters. For example, in the transition between /EH/ and /V/, atoms from clusters "9", "24", "27", and "35" are activated; the speech frames of /ER/ are represented by atoms from clusters "9", "24", and "27", whose associated phoneme labels are all /ER/ and /R/.

Lastly, we compared the representation that emerges from CSSR (Fig. 9a) with those of PSDL+CSOF (Fig. 9b) and PSDL+Lasso (Fig. 9c). To ensure a fair comparison, we set the sparsity penalty of the three systems to 0.05. As shown in Fig. 9c, when using the Lasso cost function, a speech frame is represented by atoms from arbitrary phoneme labels, and this reduces the interpretability of the representation. Compare this to Fig. 9a–b, where activation tends to occur on a few clusters/phonemes, as a result of adding the CSOF term to the Lasso. Fig. 9d–f offers a complementary view of by showing the number of sub-dictionaries activated at each frame of the utterance. CSSR and PSD+CSOF usually activate fewer sub-dictionaries ($\sim$2) than PSDL+Lasso ($\sim$6 sub-dictionaries).

VI. Discussion

In previous work [16], [17], we showed that CSDL and CSOF alone could improve voice-conversion performance relative to other sparse representation methods in the literature. This paper corroborates our earlier results and, more importantly, shows that jointly combining CSDL and CSOF can provide further improvements in voice-conversion performance.

In a first ablation study, we evaluated each CSSR component (CSDL and CSOF) by its ability to increase the speaker independence of the representation and reduce the MCD between the synthesized speech and the ground-truth target speech. Our results showed that both CSDL and CSOF are essential in reducing the sparse representation distance and the MCD, corresponding to the results in our previous work. Moreover, we found that combining both (CSSR) leads to further reductions in sparse representation distance and MCD.

In a second experiment, we compared the performance of CSSR against that of our previous system [16] as the number of atoms in the dictionary increases. CSSR increases the number of clusters (sub-dictionaries) in the representation (while keeping the number of atoms per cluster constant) whereas our previous system increases the number of atoms in each cluster (by maintaining the number of clusters constant). Our results show that CSSR is the more effective of the two approaches, as measured by the MCD between the converted speech and the ground truth. Thus, CSSR improves upon our previous work [16] by allowing more fine-grained speech information than phonemes.

In our study, we also evaluated the voice-conversion performance of CSSR through both objective and subjective measurements. We compared CSSR against the two systems from our previous work [16], [17], and against a GMM [5] baseline. In the objective evaluation, results showed that CSSR significantly improved the MCD over the three reference systems. In the subjective evaluation, CSSR was rated to have the highest acoustic quality (in agreement with results from the objective evaluation) and was rated to have the same similarity to the voice identity of the target speaker as the other systems. Additionally, we found that the comparisons between System 1 and Baseline as well as that between System 2 and Baseline are corresponding to the results presented in [16], [17].

In a final analysis, we provided a phonetic interpretation for CSSR. First, we visualized the confusion matrix of ground-truth phonemes vs. matched clusters for the cluster-structured dictionary. The results showed that CSDL can learn the phonetic structure of speech without supervision. Then, we visualized the CSSR representation and found that it is phonetically interpretable. Additionally, when comparing it with PSDL+CSOF and PSDL+Lasso, CSSR and PSDL+CSOF usually activate fewer sub-dictionaries than PSDL+Lasso, which demonstrated the effectiveness of CSOF.

VII. Conclusion

In this paper, we proposed CSSR for spectral transformation in voice conversion. CSSR consists of two inter-connected components: CSDL and CSOF. CSDL learns a structured dictionary from training utterances, and CSOF produces a structured sparse code at runtime. We conducted four experiments to evaluate CSSR. We first conducted an ablation study to examine the effectiveness of each component in CSSR. Then, we conducted an experiment to evaluate the performance of CSSR as a function of the number of atoms in the dictionary. Next, we conducted both objective and subjective experiments to evaluate the performance of CSSR and compared it with previous methods. Lastly, we provided visualizations and phonetic analyses of CSSR. The ablation study showed that both CSDL and CSOF promote the sparse representation to be speaker-independent and improve VC performance, and that combining the two components leads to further performance improvements. In addition, results from the second experiment show that CSSR uses increasingly larger dictionaries more efficiently than phoneme-based representations by allowing finer-grained decompositions of speech sounds. Next, results of objective and subjective studies show that CSSR significantly improves both acoustic quality and voice identity over the previous two systems. Finally, the visualization results show that CSSR is phonetically interpretable.

Acknowledgment

The authors are grateful for the feedback from the reviewers, which helped improve the quality of this manuscript. The authors would like to thank for the support from S. Li.

References

[1] A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1998, vol. 1, pp. 285–288.

[2] D. Felps, H. Bortfeld, and R. Gutierrez-Osuna, "Foreign accent conversion in computer assisted pronunciation training," *Speech Commun.*, vol. 51, no. 10, pp. 920–932, 2009.

[3] Z. Wu and H. Li, "Voice conversion and spoofing attack on speaker verification systems," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2013, pp. 1–9.

[4] Y. Stylianou, O. Capp, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 2, pp. 131–142, Mar. 1998.

[5] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 8, pp. 2222–2235, Nov. 2007.

[6] L.-H. Chen, Z.-H. Ling, L.-J. Liu, and L.-R. Dai, "Voice conversion using deep neural networks with layer-wise generative training," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1859–1872, Dec. 2014.

[7] S. H. Mohammadi and A. Kain, "A voice conversion mapping function based on a stacked joint-autoencoder," in *Proc. INTERSPEECH*, 2016, pp. 1647–1651.

[8] 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 *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2016, pp. 1–6.

[9] 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, Jul. 2010.

[10] L. Sun, S. Kang, K. Li, and H. M. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 4869–4873.

[11] R. Takashima, T. Takiguchi, and Y. Ariki, "Exemplar-based voice conversion in noisy environment," in *Proc. IEEE Spoken Lang. Technol. Workshop.*, 2012, pp. 313–317.

[12] Z. Wu, E. S. Chng, and H. Li, "Exemplar-based voice conversion using joint nonnegative matrix factorization," *Multimedia Tools Appl.*, vol. 74, no. 22, pp. 9943–9958, 2015.

[13] R. Aihara, T. Nakashika, T. Takiguchi, and Y. Ariki, "Voice conversion based on non-negative matrix factorization using phoneme-categorized dictionary," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 7894–7898.

[14] S. Aryal, D. Felps, and R. Gutierrez-Osuna, "Foreign accent conversion through voice morphing," in *Proc. INTERSPEECH*, 2013, pp. 3077–3081.

[15] S. Aryal and R. Gutierrez-Osuna, "Can voice conversion be used to reduce non-native accents?," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 7879–7883.

[16] S. Ding, G. Zhao, C. Liberatore, and R. Gutierrez-Osuna, "Improving sparse representations in exemplar-based voice conversion with a phoneme-selective objective function.," in *Proc. INTERSPEECH*, 2018, pp. 476–480.

[17] S. Ding, C. Liberatore, and R. Gutierrez-Osuna, "Learning structured dictionaries for exemplar-based voice conversion," in *Proc. INTERSPEECH*, 2018, pp. 481–485.

[18] C. H. Q. Ding, D. Zhou, X. He, and H. Zha, "R1-PCA: Rotational invariant L1-norm principal component analysis for robust subspace factorization," in *Proc. Int. Conf. Mach. Learn.*, 2006, pp. 281–288.

[19] J. Kominek and A. W. Black, "The CMU Arctic speech databases," *Proc. 5th ISCA Speech Synthesis Workshop*, 2004, pp. 223–224.

[20] Y. Chen, M. Chu, E. Chang, J. Liu, and R. Liu, "Voice conversion with smoothed GMM and MAP adaptation," in *Proc. INTERSPEECH*, 2003.

[21] L. Sun, K. Li, H. Wang, S. Kang, and H. M. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2016, pp. 1–6.

[22] S. Liu, J. Zhong, L. Sun, X. Wu, X. Liu, and H. Meng, "Voice conversion across arbitrary speakers based on a single target-speaker utterance," in *Proc. INTERSPEECH*, 2018, pp. 496–500.

[23] F. Fang, J. Yamagishi, I. Echizen, and J. Lorenzo-Trueba, "High-quality nonparallel voice conversion based on cycle-consistent adversarial network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 5279–5283.

[24] T. Kaneko, H. Kameoka, K. Hiramatsu, and K. Kashino, "Sequence-to-sequence voice conversion with similarity metric learned using generative adversarial networks," in *Proc. INTERSPEECH*, 2017, pp. 1283–1287.

[25] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," *Adv. Neural Inf. Process. Syst.*, pp. 6306–6315, 2017.

[26] W.-N. Hsu, Y. Zhang, and J. R. Glass, "Unsupervised learning of disentangled and interpretable representations from sequential data," *Adv. Neural Inf. Process. Syst.*, pp. 1878–1889, 2017.

[27] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, "Voice conversion using partial least squares regression," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 5, pp. 912–921, Jul. 2010.

[28] T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, "Speech synthesis using HMMs with dynamic features," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1996, vol. 1, pp. 389–392.

[29] G. Zhao and R. Gutierrez-Osuna, "Exemplar selection methods in voice conversion," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 5525–5529.

[30] S.-W. Fu, P.-C. Li, Y.-H. Lai, C.-C. Yang, L.-C. Hsieh, and Y. Tsao, "Joint dictionary learning-based non-negative matrix factorization for voice conversion to improve speech intelligibility after oral surgery," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 11, pp. 2584–2594, Nov. 2017.

[31] R. Aihara, T. Takiguchi, and Y. Ariki, "Parallel dictionary learning for voice conversion using discriminative graph-embedded non-negative matrix factorization," in *Proc. INTERSPEECH*, 2016, pp. 292–296.

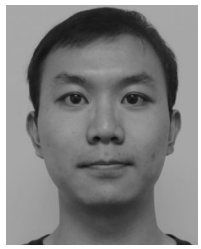
[32] B. Cicman, H. Li, and K. C. Tan, "Sparse representation of phonetic features for voice conversion with and without parallel data," in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop*, 2017, pp. 677–684.

[33] C. Liberatore, S. Aryal, Z. Wang, S. Polsley, and R. Gutierrez-Osuna, "SABR: sparse, anchor-based representation of the speech signal," in *Proc. INTERSPEECH*, 2015, pp. 608–612.

[34] Z. Wu, T. Virtanen, E. S. Chng, and H. Li, "Exemplar-based sparse representation with residual compensation for voice conversion," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 10, pp. 1506–1521, Oct. 2014.

[35] C. Liberatore, G. Zhao, and R. Gutierrez-Osuna, "Voice conversion through residual warping in a sparse, anchor-based representation of speech," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 5284–5288.
[36] Y.-C. Wu, H.-T. Hwang, C.-C. Hsu, Y. Tsao, and H.-M. Wang, "Locally linear embedding for exemplar-based spectral conversion," in *Proc. INTERSPEECH*, 2016, pp. 1652–1656.
[37] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Roy. Statist. Soc. Ser. B-Statist. Methodol.*, vol. 68, no. 1, pp. 49–67, 2006.
[38] L. Jacob, G. Obozinski, and J.-P. Vert, "Group lasso with overlap and graph lasso," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 433–440.
[39] S. Kim and E. P. Xing, "Tree-guided group lasso for multi-task regression with structured sparsity," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 543–550.
[40] P. Zhao, G. Rocha, and B. Yu, "The composite absolute penalties family for grouped and hierarchical variable selection," *Ann. Statist.*, vol. 37, pp. 3468–3497, 2009.
[41] S. Bengio, F. C. N. Pereira, Y. Singer, and D. Strelow, "Group sparse coding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 82–89.
[42] R. Jenatton, J. Mairal, F. R. Bach, and G. R. Obozinski, "Proximal methods for sparse hierarchical dictionary learning," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 487–494.
[43] Z. Szabo, B. Poczos, and A. Lorincz, "Online group-structured dictionary learning," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, 2011, pp. 2865–2872.
[44] Y. Sun, Y. Quan, and J. Fu, "Sparse coding and dictionary learning with class-specific group sparsity," *Neural Comput. Appl.*, vol. 30, no. 4, pp. 1265–1275, 2018.
[45] Z. He, L. Liu, R. Deng, and Y. Shen, "Low-rank group inspired dictionary learning for hyperspectral image classification," *Signal Process.*, vol. 120, pp. 209–221, 2016.
[46] H.-T. T. Duong, Q.-C. Nguyen, C.-P. Nguyen, T.-H. Tran, and N. Q. K. Duong, "Speech enhancement based on nonnegative matrix factorization with mixed group sparsity constraint," in *Proc. Int. Symp. Inf. Commun. Technol.*, 2015, pp. 247–251.
[47] A. Jukic, T. Van Waterschoot, T. Gerkmann, and S. Doclo, "Group sparsity for MIMO speech dereverberation," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2015, pp. 1–5.
[48] A. Hurmalainen, R. Saeidi, and T. Virtanen, "Group sparsity for speaker identity discrimination in factorisation-based speech recognition," in *Proc. INTERSPEECH*, 2012, pp. 2138–2141.
[49] D. L. Sun and G. J. Mysore, "Universal speech models for speaker independent single channel source separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 141–145.
[50] R. Aihara, T. Takiguchi, and Y. Ariki, "Activity-mapping non-negative matrix factorization for exemplar-based voice conversion," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 4899–4903.
[51] S. B. Cohen and N. A. Smith, "Viterbi training for PCFGs: Hardness results and competitiveness of uniform initialization," in *Proc. Meeting Assoc. Comput. Linguist.*, 2010, pp. 1502–1511.
[52] R. Samdani, M. W. Chang, and D. Roth, "Unified expectation maximization," in *Proc. Conf. North Am. Chapter Assoc. Comput. Linguist.: Human Lang. Technol.*, 2012, pp. 688–698.
[53] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[54] J. Macqueen, "Some methods for classification and analysis of multivariate observations," in *Proc. Berkeley Symp. Math. Statist. Probability, Vol. 1, Statist.*, 1967, vol. 1, pp. 281–297.
[55] Y.-C. Chen, V. M. Patel, J. K. Pillai, R. Chellappa, and P. J. Phillips, "Dictionary learning from ambiguously labeled data," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 353–360.
[56] G. Zhao, S. Sonsaat, J. Levis, E. Chukharev-Hudilainen, and R. Gutierrez-Osuna, "Accent conversion using phonetic posteriorgrams," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 5314–5318.
[57] B. Efron *et al.*, "Least angle regression," *Ann. Statist.*, vol. 32, no. 2, pp. 407–499, 2004.
[58] J. Mairal, F. R. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, no. 1, pp. 19–60, 2010.
[59] S. Sra and I. S. Dhillon, "Generalized nonnegative matrix approximations with Bregman divergences," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 283–290.
[60] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
[61] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Trans. Inf. Syst.*, vol. 99, no. 7, pp. 1877–1884, 2016.
[62] M. Morise, "D4C, a band-aperiodicity estimator for high-quality speech synthesis," *Speech Commun.*, vol. 84, pp. 57–65, 2016.
[63] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 1994, pp. 359–370.
[64] J. Mairal, F. R. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 689–696.
[65] D. Felps and R. Gutierrez-Osuna, "Developing objective measures of foreign-accent conversion," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 5, pp. 1030–1040, Jul. 2010.
[66] S. Buchholz and J. Latorre, "Crowdsourcing preference tests, and how to detect cheating," in *Proc. INTERSPEECH*, 2011, pp. 3053–3056.

Shaojin Ding received the B.S. degree in automation from Xian Jiaotong University, Xian, China, in 2015. He is currently working toward the Ph.D. degree in computer science at Texas A&M University, College Station, TX, USA. His research interests include speech synthesis, voice conversion, and speaker recognition.



Guanlong Zhao (M'17) received the B.S. degree in applied physics from the University of Science and Technology of China, Hefei, China, in 2015. He is currently working toward the Ph.D. degree in computer science at Texas A&M University, College Station, TX, USA. His research interests include speech synthesis, voice conversion, and accent conversion.

Christopher Liberatore (M'15) received the B.S. degree in computer science from California State University, Sacramento, CA, USA, in 2013. He is currently working toward the Ph.D. degree in computer science with the Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA. He is currently working on his dissertation, centered on using a sparse, anchor based representation to perform voice conversion and accent conversion. His current research interests include sparse coding, speech processing, voice conversion, speech synthesis, and accent conversion.

Ricardo Gutierrez-Osuna (M'00–SM'08) received the B.S. degree in electrical engineering from the Polytechnic University of Madrid, Madrid, Spain, in 1992 and the M.S. and Ph.D. degrees in computer engineering from North Carolina State University, Raleigh, NC, USA, in 1995 and 1998, respectively. He is a Professor with the Department of Computer Science and Engineering, Texas A&M University, College Station. His current research interests include voice and accent conversion, speech and face perception, wearable physiological sensors, and active sensing.