MINIMIZING RESIDUALS FOR NATIVE-NONNATIVE VOICE CONVERSION IN A SPARSE, ANCHOR-BASED REPRESENTATION OF SPEECH

Christopher Liberatore, Ricardo Gutierrez-Osuna

Texas A&M University, College Station, Texas, USA

{cliberatore, rgutier}@tamu.edu

ABSTRACT

We present a dictionary-learning algorithm for reducing the sparse coding residual of an exemplar-based method for native-to-nonnative voice conversion (VC). The proposed algorithm iteratively updates the source and target speaker dictionaries to reduce both the residual and voice conversion error, thereby increasing synthesis quality. We evaluate the method on speech from the ARCTIC and L2-ARCTIC corpora and compare it to a baseline exemplar-based VC algorithm. The proposed algorithm significantly improves synthesis quality to more than double that of the baseline system while using two orders of magnitude fewer atoms. Additionally, the proposed algorithm significantly reduces both the VC error and the residual magnitude. We discuss the implications of the algorithm for broad exemplar-based VC systems.

Index Terms--sparse coding, voice conversion, residual, exemplar voice conversion, dictionary learning

1. INTRODUCTION

Speakers who learn a second language (L2) later in life have difficulty acquiring native-like pronunciation [1]. To improve their pronunciation, L2 learners often practice by imitating utterances from a model voice. Several studies [2-5] have shown that pronunciation training can be made more effective by matching the learner with a model voice that resembles the learner's own voice. These findings suggest that the ideal voice for each learner is their own voice, resynthesized to have a native accent [1]. One approach to produce such a voice is to adapt existing voice conversion (VC) techniques by transforming utterances from a native speaker to match the voice quality of the L2 learner. However, VC techniques can require hundreds of training utterances from each learner, which makes them impractical for instructional settings [6]. While exemplar-based VC methods help to circumvent the onerous data requirements of conventional VC methods [7-9], they require time-aligned source and target data, which is affected by the presence of pronunciation errors (e.g., phoneme substitutions, additions, and deletions) [6, 10, 11], ultimately degrading synthesis quality.

In response to this issue, we recently developed a low-resource VC technique based on a sparse, anchor-based representation of speech (SABR) [12-14]. SABR is a compact exemplar-based method that uses a single acoustic "anchor" (i.e., exemplar) per phoneme to model a speaker's voice. Using Lasso regression [15], SABR decomposes a source speaker's spectrum into "weights," (i.e., sparse codes) that encode linguistic content relative to the speaker identity contained in the source anchors. To perform VC, SABR takes the source speaker's weights (i.e., linguistic content) and combines them with anchors from the L2 learner; so long as the phonetic content in the source and target anchor sets is the same, voice conversion can be performed. SABR anchors are learned from phoneme labels gathered via forced alignment, so issues arising from time-aligning source and target speakers (e.g., in the case of native-to-nonnative voice conversion) can be bypassed. However, SABR anchors for the source and target speaker are learned independently and not to optimize the sparse coding. This leads to models which have a large sparse coding residual, lowering synthesis quality.

In this paper, we propose a dictionary learning algorithm called Iterative Retraining (IRT) to reduce the residual of SABR and improve synthesis quality. This algorithm optimizes source and target anchor sets in such a way as to minimize both VC error and the residual. As the name suggests, IRT operates iteratively: it uses the source weights to update the target anchors, and then uses the target weights to update the source anchors, reducing the residual as well as the VC error. In this fashion, the synthesis quality significantly improves for both native-native and native-nonnative conversion.

We evaluate the algorithm using a dataset of speech recordings from native and non-native speakers in the ARCTIC [16] and L2-ARCTIC [17] corpora, respectively, and compared them against an exemplar-based VC baseline [8] that also includes a residual compensation step. We find that IRT is more robust to the effects of native-to-nonnative conversion, with native-to-nonnative quality ratings being closer to their native-to-native counterparts than they are on the baseline system. Further, these improvements in acoustic quality come at no loss in VC performance, measured by the ability of the system to capture the voice quality of the target speaker. These results show that IRT can be an effective tool to improve voice and accent conversion systems in low-resource settings.

The remainder of this paper is organized as follows. Section 2 reviews prior work on Accent Conversion exemplar-based VC. Section 3 describes the VC algorithm used in this paper (SABR) and introduces the proposed optimization approach. Section 4 describes the experimental design and speech corpora used in our study. Section 5 presents objective and subjective experimental results. The article concludes with a summary of our findings and directions for future work.

2. PRIOR WORK

Accent conversion (AC) seeks to synthesize speech with the voice quality of an L2 speaker, but the accent of a native speaker. AC systems operate by building a model of the speaker's voice characteristics and driving it with linguistic gestures from a native speaker. Aryal et al. [6] used a Gaussian Mixture Model (GMM) combined with a special VTLN-based frame-pairing method to perform accent conversion from an L1 source to an L2 target speaker. They found that synthesis quality was degraded as

compared to native-to-native conversion because of the effects of accent. Namely, the method relied on time alignment and was affected by the phoneme inventories of the L1 and L2 speakers. However, this method significantly reduced the perceived nonnative accents while preserving the voice quality of the L2 speaker. Later work by Zhao et al. [18] found that using phonetic posteriorgrams was an even more effective way to align source and target training data for use in GMM-based accent conversion. However, these methods require significant amounts of training data (e.g., hundreds of utterances), making it infeasible for pronunciation training contexts.

Exemplar-based VC was originally developed to address the oversmoothing effects of parametric statistical VC methods [9, 19, 20]. These methods use spectra from time-aligned source and target utterances as dictionaries used in sparse coding. To perform VC, the source dictionary is first used to extract sparse codes from an utterance, and those sparse codes are multiplied by the target dictionary to get an approximation of the target speaker's spectrum for the same utterance. Wu et al. [8] proposed a method of encoding this residual called Exemplar voice conversion with Residual Compensation. Using Partial Least Squares, they computed a transform from the source speaker's residual to the space of the target speaker. This had the net effect of further improving synthesis quality by including spectral details which were discarded at the time of the encoding process. We use this method as the baseline for this study. Zhao and Gutierrez-Osuna [21] explored methods for reducing the number of exemplars required in exemplar dictionaries while still accomplishing high-quality VC. The authors examined sequential forward and sequential backwards selection methods for introducing more exemplars or reducing exemplars in the basis set. They found that the lower VC error could be achieved with a significantly smaller exemplar set.

3. METHODS

3.1. Sparse, Anchor-Based Representation of Speech

The VC algorithm that underlies this work (SABR) represents an utterance as a sparse linear combination of speaker-dependent phonemic anchors [13]. The intuition behind the method is that the sparse code of an utterance relative to these anchors encodes the linguistic content. Given a speech spectrum X_S, SABR decomposes it as:

$$X_S = A_S W_S, \tag{1}$$

where W_S is a sparse set of weights, and A_S is a set of speaker-dependent phoneme "anchors." For an utterance with T frames N acoustic spectral features (e.g., MFCCs), and K anchors, $X_S \in \mathbb{R}^{N\times T}$, $A_S \in \mathbb{R}^{N\times K}$, and $W_S \in \mathbb{R}^{K\times T}$. SABR uses a single anchor per phoneme, which is computed by selecting the centroid of all frames that have the corresponding phoneme label in the speaker's training corpus.

SABR uses the Lasso [15, 22] to estimate the weights (sparse codes) $W_S \in \mathbb{R}^{K\times T}$:

$$\min_{W_S} \|X - A_S W_S\|_2^2 + \|W_S\|_1, \qquad s.t. \|W_S\|_1 \le \lambda \tag{2}$$

where $\|\cdot\|_1$ is the L1 norm and λ is the maximum sum allowed for each W_S, which acts as a regularization term.

To obtain an estimate of the target speaker's spectrum, a target anchor set A_T is built in the same manner as A_S: one spectral anchor per phoneme label, corresponding to the centroid of all target speaker training data with that label. An estimate of the target

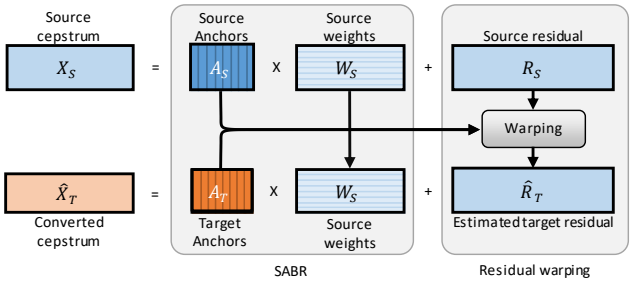


Figure 1: illustration of the overall SABR algorithm. The proposed optimization method operates on the anchors A_S and A_T.

speaker's spectral envelope $\hat{X}_T$ is obtained as the product of the source weights and target anchors:

$$\hat{X}_T = A_T W_S. \tag{3}$$

This estimated spectrum lacks spectral detail, since the residual from eq. (2) is discarded. To address this issue, in prior work [14] we proposed a method for transforming the source residual to the target speaker's space using frequency warping. The overall SABR approach is illustrated in Figure 1. Let R_S denote the residual error in eq. (1) , i.e., the portions of the spectrum that the sparse representation does not capture:

$$X_S = A_S W_S + R_S. \tag{4}$$

Because this residual contains source-specific information, we transform it to the space of the target speaker prior to adding it to the estimated target spectrum via a function F_R:

$$\hat{X}_T = A_T W_S + F_R(R_S). \tag{5}$$

Here, $F_R(\cdot)$ is a linear combination of frequency-warping based cepstral transforms, learned from the source and target anchors A_S and A_T.

For each pair of source-target anchors A_S^k and A_T^k, we find the frequency warp T_k that minimizes the sum-squared error of the transformed source and target anchors:

$$T_k = \underset{T(\Theta)}{\operatorname{argmin}} \sum \left(T(\Theta)A_S^k - A_T^k\right)^2, \tag{6}$$

where T is a piecewise linear warping function and Θ are the parameters to the function (following [23]). For the t^{th} source frame X_S^t , weight vector W_S, and the frame residual R_S^t, we estimate the target speaker's spectrum $\hat{X}_T^t$ as:

$$\hat{X}_T^t = A_T W_S + \left(\sum_{k=1}^{K} W_S^k T_k\right) R_S^t. \tag{7}$$

Because this method uses the source residual to improve synthesis quality, minimizing the magnitude of the residual will have a strong impact on synthesis quality.

3.2. The Iterative Retraining (IRT) algorithm

In this section, we discuss the IRT algorithm we propose for modifying the source and target speaker anchor sets so they minimize the residuals. IRT is based on the Method of Optimal Directions (MOD) [24], a dictionary learning method. MOD learns a dictionary by updating an initial dictionary using the sparse codes and residuals of training data.

Given an utterance X, a source dictionary A, and an activation matrix W, MOD computes an update ΔA to the dictionary as:

$$\Delta A = (X - AW)(W^+), \tag{8}$$

where $(\cdot^{+})$ is the Moore-Penrose Pseudoinverse. Optimizing the source and target anchors *independently* will reduce their respective residuals. However, using eq. (8) to update the source and target anchor sets independently may cause these sets to diverge from each other in terms of phonetic content; if this occurs, using the anchor sets in VC would result in significant distortions and reduced intelligibility. To alleviate this, we add a second term to eq. (8) that updates the anchor sets in the direction that reduces the VC error. This serves as a proxy for ensuring parallel phonetic content. Including this term ensures that the updates to the anchors are conditioned to both minimize residuals and perform voice conversion between the two speakers.

Let $R_{S|T}$ represent the residual error when representing a source utterance X_S using parallel target weights W_T:

$$R_{S|T} = X_S - A_S W_T, \tag{9}$$

where W_T is computed from and A_T and X_T using eq. (2). Following eq. (8), we update A_S in a way that accounts for the source residual in eq. (4) and the VC error term in eq. (9):

$$\Delta A'_S = \left(\alpha R_s + (1-\alpha) R_{S|T}\right) W_S^{+}, \tag{10}$$

where α is a parameter that balances the two update terms. The first term of eq. (10) functions as the first term of eq. (8), but here we account for both the residual and VC error when updating the anchor set A_S. Source anchors on iteration t are then updated as:

$$A_S^{t+1} = A_S^{t} + \Delta A'_S. \tag{11}$$

The target anchors are then updated in a similar fashion, replacing source terms with target terms and vice-versa. To prevent IRT from overfitting, we split the training data into two disjoint subsets: one subset used to update the source anchors, and the other subset used to update the target anchors. In our experiments, each subset was half of the training corpus.

The algorithm then iterates, updating the source anchors based on the target weights (using the first subset), and then updating the target anchors using the source weights (using the second training subset), following eqs. (9)-(11). IRT can iterate indefinitely, but in practice we find that the VC error converges within 20 iterations.

4. EXPERIMENTS

4.1. Data and implementation details

We evaluated the proposed algorithm using the CMU ARCTIC speech corpus [19] and the L2-ARCTIC speech corpus (v 1.0) [20]. L2-ARCTIC is a corpus based on the prompts of the ARCTIC database, but with L2 speakers of English from six first languages: Mandarin, Hindi, Arabic, Spanish, Korean, and Vietnamese.

We conducted both subjective and objective evaluations in both native-to-native (ARCTIC to ARCTIC, *A2A*) and native-to-nonnative (ARCTIC to L2-ARCTIC, *A2L2*) contexts. For objective experiments, we evaluated all possible speaker pairs in A2A and A2L2 conditions, but for perceptual experiments, we only evaluated the speaker pairs in Table 1 (A2A) and Table 2 (A2L2). In a prior study [12], we observed that SABR was very effective in reducing accentedness when converting from a native speaker to a nonnative speaker; because of this, we did not perform accentedness tests in this study. To illustrate the time-alignment difference between native and nonnative speaker pairs, we examined the average difference between the computed DTW trajectories for A2A and A2L2 speaker pairs (see Table 3). These results highlight the challenges of using conventional exemplar-based VC methods, which require accurate alignment, when the target speakers are non-native.

Table 1: A2A speaker pairs for perceptual experiments.

Source speaker	Target speaker
BDL (M)	RMS (M)
SLT (F)	CLB (F)
RMS (M)	SLT (F)
CLB (F)	BDL (M)

Table 2: Speaker pairs for the A2L2 perceptual experiments.

Source speaker	Target speaker	First language
BDL (M)	HKK (M)	Korean
SLT (F)	SKA (F)	Arabic
RMS (M)	YDCK (F)	Mandarin
CLB (F)	EVBS (M)	Spanish

Table 3: Average time alignment differences.

Corpus	Average alignment error
L2-ARCTIC	221 ms ± 36 ms
ARCTIC	124 ms ± 15 ms

We trained models using 20 parallel, time-aligned utterances to obtain the SABR anchors and as input to the IRT algorithm. We performed time alignment using MFCC features and dynamic time warping (DTW) [25]. We used STRAIGHT [26] with 1ms frame steps and 80ms window size to extract aperiodicity, fundamental frequency, and spectral envelope from each utterance. We then computed a 25-dimension MFCC vector. SABR models ignored $MFCC_0$, as it contains energy. For synthesis, the energy from the source speaker was copied to the estimated target speaker's spectrum. We converted the pitch of the source utterance to match the pitch range of the target speaker using log mean-variance scaling [27]. To solve for the SABR weights, we used the LARS solver from the SPAMS toolbox [28].

4.2. Comparison systems

We evaluated the proposed algorithm against the original SABR model and a baseline VC system[1]:

- *IRT*: source and target anchors optimized by the IRT algorithm (Section 3.2).
- *SABR*: the default SABR anchors—one anchor per phoneme, selected by computing the centroid of all frames with that phoneme label (Section 3.1).
- *Baseline:* A time-aligned exemplar-based VC approach, including residual compensation [8].

The same utterances were used to train the SABR and IRT models and the baseline system. We used 20 parallel utterances to train the three systems. The training utterances were selected in such a way as to maximize phoneme variability. Note that the dictionaries in the baseline model were substantially larger than that of the SABR and IRT models (4720 atoms on average for the baseline system, 39 for both SABR and IRT). For all systems, we measured the Mel-Cepstral Distortion (MCD) [27] of both the voice conversion error (eq. (9)) and the residual magnitude (eq. (4)) on a

[1] Synthesis samples from the above systems can be found at https://cliberatore.github.io/samples/sabr-irt.html

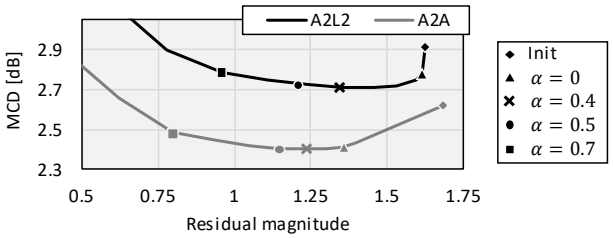
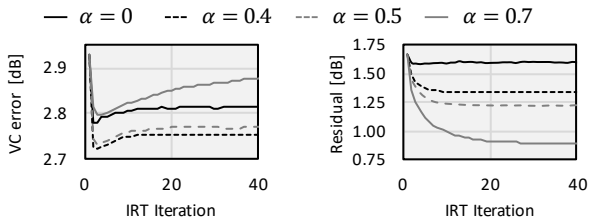


Figure 2: VC error vs residual error for different values of parameter α on the cross-validation set. "Init" refers to initial SABR model



(a) (b)

Figure 3: IRT performance by alpha value.

Table 4: Voice conversion MCD results. Standard error

	VC error		Residual	
Method	A2A	A2L2	A2A	A2L2
Baseline	2.46 ± 0.13	2.74 ± 0.14	0.92 ± 0.04	0.96 ± 0.04
SABR	2.59 ± 0.13	2.91 ± 0.12	1.68 ± 0.13	1.67 ± 0.11
IRT	2.42 ± 0.12	2.72 ± 0.12	1.05 ± 0.16	1.24 ± 0.08

test set of 200 utterances selected from the ARCTIC "*A*" set of utterances.

5. RESULTS

5.1. Objective experiments

In a first experiment, we performed a 4-fold cross-validation experiment to determine the optimal α value in eq. (11) and the optimal number of iterations to run the IRT algorithm. The tradeoff between optimizing the residual and VC error (eq. (10)) is shown in Figure 2, and the IRT iterative results are shown in Figure 3. For both A2A and A2L2 speaker pairs, $\alpha = 0.4$ achieved the lowest VC error; however, the MCD at $\alpha = 0.5$ was negligibly higher (0.01 dB) with a significant decrease in residual. For $\alpha \geq 0.7$, the VC error diverged as the IRT algorithm was biased towards minimizing the source and target residuals independent of minimizing the VC error. Based on these results, in what follows we set $\alpha = 0.5$ and perform IRT until the residual converges (in practice, this occurs at around 20 iterations).

Using these parameters, we evaluated the IRT algorithm on the test set of A2A and A2L2 speakers, measuring the VC error and residual and comparing it with the baseline method. The results are shown in Table 4. For both A2A and A2L2 pairs, the proposed optimization methods had significantly lower VC error than the original SABR model (A2A and A2L2, $p \ll 0.001$, paired t-test). Notably, there was no significant difference in the VC error of IRT and the baseline model (A2A, $p = 0.07$; A2L2, $p = 0.35$, paired t-test), a positive result given that the baseline model had dictionaries more than two orders of magnitude larger.

Table 5: Subjective experiment results. Average values and standard errors shown.

	MOS		Identity	
Method	A2A	A2L2	A2A	A2L2
Baseline	2.20 ± 0.08	1.27 ± 0.04	90% ± 5.5%	91% ± 7.1%
SABR	3.11 ± 0.09	2.52 ± 0.08	90% ± 6.4%	87% ± 8.9%
IRT	3.45 ± 0.09	3.05 ± 0.09	87% ± 7.5%	85% ± 8.2%

5.2. Subjective

5.2.1. Mean Opinion Score

We performed listening tests on Amazon Mechanical Turk to measure the synthesis quality of the three systems on a 5-point Mean Opinion Scores (MOS) scale (1 = "low quality"; 5 = "high quality"). For each system, participants ($n = 20$) rated 40 utterances (5 per speaker pair). Following [29], we included unmodified references to ensure participants were not randomly guessing. Results are shown in Table 5. A2L2 ratings for the IRT algorithm were approximately twice as high as those of the baseline system[1] ($p \ll 0.01$, single-tailed t-test), a remarkable result given that they include far fewer anchors in their dictionaries (39 vs. 4720). This pattern continued in the A2A ratings, where the optimization methods were also significantly higher than the baseline (2.20; $p < 0.01$, single-tailed t-test). IRT significantly improved on the initial SABR synthesis quality for both A2A and A2L2 speaker pairs ($p < 0.01$, single-tailed t-test).

5.2.2. Speaker identity test

To determine if the IRT algorithm affected the identity of VC utterances, we performed an XAB speaker identity test comparing synthesis from the three systems. Participants ($n = 20$) were presented with three utterances: a VC utterance (X), and utterances from the source or target speaker (A, B). The order of A and B was counter balanced. Following [6], utterances were played in reverse to mask the effects of accent, and allow participants to focus on the identity of the speaker. For each method, we performed 32 evaluations (4 evaluations per speaker pair). We included unmodified references to ensure participants were not randomly guessing. Results are shown in Table 5. There was no statistically significant difference between the baseline and IRT in either A2A or A2L2 conversions ($p > 0.05$, two-tailed t-test), *even though the optimization methods used two orders of magnitude fewer anchors than the baseline system*.

6. CONCLUSION

We have proposed IRT, a dictionary learning method to optimize dictionaries in a low-resource exemplar-based VC method (SABR). IRT reduced both the VC error and residual magnitude of the SABR models, in both native and nonnative VC cases. Additionally, IRT significantly increased the synthesis quality of SABR while maintaining the speaker identity of the VC utterances. In future work, applying IRT to expanded anchor sets (i.e. including subphoneme states) or larger exemplar dictionaries would provide an avenue for further improving exemplar-based VC methods.

Acknowledgements: This work was partially supported by NSF Grants 1619212 and 1623750.

[1] This MOS rating is significantly lower than those reported by the authors of the baseline system. We believe this is due to the difficulty in time-aligning native to non-native utterances, which is critical for the baseline system. In contrast, SABR is less affected by these issues as it does not require time-alignment, and the IRT algorithm have built-in mechanisms to address misalignments.

7. REFERENCES

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

[2] M. P. Bissiri, H. R. Pfitzinger, and H. G. Tillmann, "Lexical stress training of German compounds for Italian speakers by means of resynthesis and emphasis," in *Australian Int. Conf. on Speech Science & Technology*, 2006, pp. 24-29.

[3] K. Nagano and K. Ozawa, "English speech training using voice conversion," in *Proc. of the First Int. Conf. on Spoken Language Processing (ICSLP)*, Kobe, Japan, 1990.

[4] M. Peabody and S. Seneff, "Towards automatic tone correction in non-native mandarin," in *International Symposium on Chinese Spoken Language Processing*, 2006, pp. 602-613: Springer.

[5] K. Probst, Y. Ke, and M. Eskenazi, "Enhancing foreign language tutors–-in search of the golden speaker," *Speech Communication,* vol. 37, no. 3, pp. 161-173, 2002.

[6] S. Aryal and R. Gutierrez-Osuna, "Reduction of non-native accents through statistical parametric articulatory synthesis," *Journal of the Acoustical Society of America,* vol. 137, no. 1, pp. 433-446, 2015.

[7] R. Aihara, T. Nakashika, T. Takiguchi, and Y. Ariki, "Voice conversion based on non-negative matrix factorization using phoneme-categorized dictionary," in *ICASSP*, 2014, pp. 7894-7898.

[8] Z. Wu, T. Virtanen, E. S. Chng, and H. Li, "Exemplar-based sparse representation with residual compensation for voice conversion," *IEEE/ACM Transactions on Audio, Speech and Language Processing,* vol. 22, no. 10, pp. 1506-1521, 2014.

[9] Z. Wu, T. Virtanen, T. Kinnunen, E. S. Chng, and H. Li, "Exemplar-based voice conversion using non-negative spectrogram deconvolution," in *ISCA Workshop on Speech Synthesis (SSW)*, 2013.

[10] G. Zhao and R. Gutierrez-Osuna, "Using phonetic posteriorgram based frame pairing for segmental accent conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing,* vol. 27, no. 10, pp. 1649-1660, 2019.

[11] S. Aryal and R. Gutierrez-Osuna, "Accent conversion through cross-speaker articulatory synthesis," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7694-7698: IEEE.

[12] S. Ding *et al.*, "Golden Speaker Builder - An interactive tool for pronunciation training," *Speech Communication,* vol. 115, pp. 51-66, 2019.

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

[14] C. Liberatore, Native-Nonnative Voice Conversion by Residual Warping in a Sparse, Anchor-Based Representation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29, pp. 3040-3051, 2021.

[15] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society,* vol. Series B, pp. 267-288, 1996.

[16] J. Kominek and A. W. Black, "The CMU ARCTIC speech databases," in *ISCA Workshop on Speech Synthesis (SSW)*, 2004, pp. 223-224.

[17] G. Zhao *et al.*, "L2-ARCTIC: A Non-Native English Speech Corpus," in *INTERSPEECH*, 2018.

[18] G. Zhao, S. Sonsaat, J. Levis, E. Chukharev-Hudilainen, and R. Gutierrez-Osuna, "Accent Conversion Using Phonetic Posteriorgrams," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5314-5318: IEEE.

[19] R. Aihara, T. Nakashika, T. Takiguchi, and Y. Ariki, "Voice conversion based on non-negative matrix factorization using phoneme-categorized dictionary," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7894-7898: IEEE.

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

[21] G. Zhao and R. Gutierrez-Osuna, "Exemplar selection methods in voice conversion," in *ICASSP*, 2017, pp. 5525-5529.

[22] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics,* vol. 32, no. 2, pp. 407-499, 2004.

[23] M. Pitz and H. Ney, "Vocal tract normalization equals linear transformation in cepstral space," *IEEE Trans. on Speech and Audio Processing,* vol. 13, no. 5, pp. 930-944, 2005.

[24] K. Engan, S. O. Aase, and J. H. Husoy, "Method of optimal directions for frame design," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 1999, vol. 5.

[25] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *KDD workshop*, 1994, vol. 10, no. 16, pp. 359-370: Seattle, WA.

[26] H. Kawahara, "STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds," *Acoustical Science and Technology,* vol. 27, no. 6, pp. 349-353, 2006.

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

[28] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research,* vol. 11, no. Jan, pp. 19-60, 2010.

[29] S. Buchholz and J. Latorre, "Crowdsourcing Preference Tests, and How to Detect Cheating," in *INTERSPEECH*, 2011, pp. 3053-3056.