VOICE MORPHING THAT IMPROVES TTS QUALITY USING AN OPTIMAL DYNAMIC FREQUENCY WARPING-AND-WEIGHTING TRANSFORM

Yannis Agiomyrgiannakis, *Zoi Roupakia**

Google

agios@google.com, zr216@cam.ac.uk

ABSTRACT

Dynamic Frequency Warping (DFW) is widely used to align spectra of different speakers. It has long been argued that frequency warping captures inter-speaker differences but DFW practice always involves a tricky preprocessing part to remove spectral tilt. The DFW residual is successfully used in Voice Morphing to improve the quality and the similarity of synthesized speech but the estimation of the DFW residual remains largely heuristic and sub-optimal. This paper presents a dynamic programming algorithm that simultaneously estimates the Optimal Frequency Warping and Weighting transform (ODFWW) and therefore needs no preprocessing step and fine-tuning while source/target-speaker data are matched using the Matching-Minimization algorithm [1]. The transform is used to morph the output of a state-of-the-art Vocaine-based [2] TTS synthesizer in order to generate different voices in runtime with only +8% computational overhead. Some morphed TTS voices exhibit significantly higher quality than the original one as morphing seems to "correct" the voice characteristics of the TTS voice.

Index Terms— voice-morphing, voice-transformation, DFW, vocaine, matching-minimization

1. INTRODUCTION

Modern TTS systems use extensive single speaker recordings that are costly and laborious. Voice morphing is considered an inexpensive way to create new voices from existing ones but the morphing process introduces degradations to the synthesized speech such as artifacts and muffled speech quality. The latter degrades TTS intelligibility, while the overall quality degradation constrains voice morphing to be a research topic rather than an industry tool. For a practical use of voice morphing for TTS we need to care firstly for intelligibility, secondly for naturalness and finally for similarity since the TTS user is not aware of the original source speaker. In voice morphing for TTS, one can also take advantage of the *data-size asymmetry*: source-speaker data size (i.e. a TTS corpus) is usually much higher than target-speaker data size (i.e. a couple of adaptation utterances).

Voice conversion methods can roughly be divided in statistical ones and Frequency Warping (FW)-based ones. Standard statistical approaches have the form of GMM-based transformations [3, 4]. These methods are able to convert the timbre but reportedly oversmooth the spectral envelope and do not preserve the spectral details during transformation. Over-smoothing can be alleviated via trajectory adaptation techniques such as Global Variance [5] but there is still a degradation in quality.

Statistical speech models [6] can easily be adapted to a target speaker using speaker adaptation approaches developed for speech recognition [7]. Adaptation techniques typically require a lot of target-speaker data but rapid adaptation with a few target-speaker utterances is also possible using factorization techniques like eigenvoices [8]. In eigenvoice-based techniques, the speaker is defined as a "point" in a factorised multi-speaker space by estimating only a few parameters. Non-linear approaches have also been presented in the form of kernel regression [9].

Recent developments in TTS allow us to significantly improve the quality of the synthesized speech, namely by using an LSTM to predict the acoustic parameters from the linguistic ones [10] and Vocaine as the vocoder [2]. Neural-Network-based speech synthesis [11, 12] can also benefit from adaptation [13, 14]. Despite the significant progress in NN-based adaptation, voice morphing still degrades quality [13], which can be attributed to the unconstrained nature of Neural Networks.

Spectrally constrained models like FW allow us to modify speech spectra in a way that is physiologically plausible [15] and have been able to provide high quality speech modifications [16, 17], albeit at the cost of a decreased similarity to the target speaker, which is not a problem in the TTS case since the user has no reference expectations.

FW-based methods warp the frequency axis of the source spectral envelopes to match the target spectral envelopes in order to compensate vocal tract differences. They are a broad generalization of vocal tract length normalization methods [18–20]. The observation that the FW transform cannot cope with voicing or nasality changes led some researchers to introduce a frequency weighting filter [16, 21]. Such methods will henceforth be referred to as Frequency Warping-and-Weighting (FWW) methods. Finding a reasonably good FWW transform is a rather difficult problem because the spectral envelopes have to be pre-processed before applying some Dynamic Frequency Warping (DFW) algorithm to estimate the warping [22]. The frequency weighting filter is computed independently of FW as the mean of the FW residual. The pre-processing step usually rely on heuristics like spectral-tilt removal [21, 22] or estimating spectral peak histograms [16]. Heuristics tend to reduce the robustness of the estimation [23].

This paper presents a dynamic programming algorithm that simultaneously and optimally estimates the frequency warping and the frequency weighting, hence its name: Optimal Dynamic Frequency Warping-and-Weighting (ODFWW). It is merely an extension of DFW with a bias factor that compensates inter-speaker differences such as voicing and nasality that manifest themselves as spectral tilt and spectral valeys. Section 2 presents how source/target spectra are matched. Section 3 presents the ODFWW algorithm. Section 4 shows how ODFWW is used in a Vocaine-LSTM-based TTS. Finally, extensive experiments are presented in Section 5.

* The second author was an intern at Google, London, UK while pursuing a part of this work.

2. MATCHING SOURCE/TARGET SPEAKER SPECTRA

ODFWW requires the matched pairs of source/target spectral envelopes. For TTS morphing, source-speaker data size (TTS corpus) is much larger than target-speaker data size (adaptation utterances). We use the Matching-Minimization (MM) algorithm [1] that is also presented in this conference to find good source/target-speaker correspondences. The algorithm is used in three phases: the *initial phase*, the *cleaning phase* and the *fine-tuning phase*.

In the initial phase, we set the X-space to be the source speaker (the whole TTS corpus) and the Y-space to be the target speaker (a couple of utterances). MM finds a subset of the X-space that is well matched to the Y-space. In the *cleaning phase* we swap X and Y-space and set X-space to be the target speaker and Y-space to be the source speaker subset. This removes the target-speaker spectra that are not well matched to the source-speaker subset. It is effectively, a mechanism to remove silences and non-speech spectra from the target-speaker utterances. The cleaning phase allows us to use completely unrestricted audio recordings for the target speaker without worrying about silences, external noises, etc. Essentially, we use the clean recording (TTS corpus) to filter the unclean recording. In the *fine-tuning phase* we go back to the original configuration where X-space is assigned to the source-speaker subset and Y-space is assigned to the target-speaker subset. The sequence of these three phases allows us to filter out irrelevant matches between source-speaker and target-speaker spectra.

For clarity of presentation, we will henceforth assume that source and target speaker spectra are matched.

3. OPTIMAL DYNAMIC FREQUENCY WARPING AND WEIGHTING

Let $\vec{s}_n$ denote the n^{th} source-speaker frame

$$\mathcal{S} = \{\vec{s}_n \in \mathbb{R}^P |\, n = 1, \cdots, N\} \tag{1}$$

and $\vec{t}_n$ denote the n^{th} (matched) target-speaker frame

$$\mathcal{T} = \{\vec{t}_n \in \mathbb{R}^P |\, n = 1, \cdots, N\} \tag{2}$$

where P is the dimensionality of the spectral vectors and, N is the number of matched source/target spectra.

The frequency weighting corrective filter is modelled as an additive term in the parametric domain. Therefore, any parameterisation of speech can be used under the constraint that it preserves the homomorphic property of log-spectra/cepstra that a linear filtering operation corresponds to an addition, i.e. cepstrum, Mel-Cepstrum (MCEP), generalised or discrete MCEP [24]. In this paper, source and target speaker spectra are parameterised as MCEP coefficients including power. Source/target speaker spectra are extracted using analysis similar to STRAIGHT [25], excluding unvoiced frames as in [4, 16, 26].

3.1. Joint estimation

Let $S_n(\omega)$ and $T_n(\omega)$ be the log-spectra of $\vec{s}_n$ and $\vec{t}_n$, respectively, sampled at frequency ω:

$$S_n(\omega) = \sum_{p=1}^{P-1} s_n^p \cos(\omega p) + s_n^0, \tag{3}$$

$$T_n(\omega) = \sum_{p=1}^{P-1} t_n^p \cos(\omega p) + t_n^0, \tag{4}$$

where s_n^p and t_n^p are the p-th MCEP parameters of $\vec{s}_n$ and $\vec{t}_n$, respectively. Let $w(\omega)$ be a warping function and $b(\omega)$ be a continuous frequency weighting function. The estimated target-speaker spectral envelope $\hat{T}_n(\omega)$ is obtained by applying the frequency warping and weighting so that

$$\hat{T}_n(\omega) = S_n(w(\omega)) + b(\omega), \tag{5}$$

where $S_n(w(\omega))$ is the warped source-spectral envelope of the n^{th} speech frame

$$S_n(w(\omega)) = \sum_{p=1}^{P-1} s_n^p \cos(w(\omega)p) + s_n^0, \tag{6}$$

where s_n^p is the p-th MCEP parameter of $\vec{s}_n$. The warping function must satisfy the condition that $w(0) = 0$.

ODFWW estimates *jointly* the frequency warping and weighting function, $w(\omega)$ and $b(\omega)$ respectively, by minimising the average (over all frames) log-spectral distortion D between the target spectral envelope $T_n(\omega)$ and the estimated target one $\hat{T}_n(\omega)$:

$$\hat{w}, \hat{b} = \underset{w,b}{\operatorname{argmin}}\, D(w, b), \tag{7}$$

where, using equation (5), the average distortion D is

$$D(w, b) = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\pi} \int_0^\pi \big(T_n(\omega) - S_n(w(\omega)) - b(\omega)\big)^2 \mathrm{d}\omega. \tag{8}$$

In practice, the integral is approximated by a sum, by quantising frequency ω to K equal intervals, i.e., FFT frequency bins. The distortion is, then,

$$D(w, b) \simeq \frac{1}{N} \sum_{n=1}^{N} \frac{1}{K} \sum_{k=0}^{K-1} \big(T_n(\omega_k) - S_n(w(\omega_k)) - b(\omega_k)\big)^2, \tag{9}$$

where ω_k is the frequency at k^{th} frequency interval.

The trick that allows us to solve simultaneously for the warping and the weighting is that at every frequency ω_k, given the frequency warping there is a closed-form solution for the estimation of the frequency weighting. By taking the partial derivative of equation (9) and equating it to zero the optimal frequency weighting is

$$\hat{b}(\omega_k) = \frac{1}{N} \sum_{n=1}^{N} \big(T_n(\omega_k) - S_n(w(\omega_k))\big). \tag{10}$$

The closed form solution of the frequency weighting is equivalent to the post-processing amplitude filtering used in methods presented in [16], [27].

Given the frequency weighting, at every frequency bin the distortion is minimised to estimate pairs of source to target frequencies.

$$\hat{w} : \underset{w}{\operatorname{argmin}}\, D(w, \hat{b}) \tag{11}$$

If the optimal frequency weighting of equation (9) is replaced in equation (10), it is apparent that the distortion criterion depends only on the frequency warping function. Thus, we can optimaly solve the DFWW problem by using a DFW algorithm with the modified distortion $D(w, \hat{b})$.

The distortion criterion is used in a standard dynamic frequency warping algorithm like [28], where the parameters are estimated through a $K \times K$ trellis search. For a pair of frequency bins

$\{\omega_i, \omega j\}$, the distortion is

$$D_{ij}(w, \hat{b}) = \frac{1}{N}\sum_{n=1}^{N}\big(T_n(\omega_i) - S_n(\omega_j) - \hat{b}(\omega_i)\big)^2 \quad (12)$$

As the cost is prohibitive for $K \times K$ frequency bins, the search is limited only inside a Sakoe-Chiba frequency band and only a number of steps are allowed [29].

It should be noted that in standard DFW approaches, it is important to remove the spectral tilt from the spectral envelopes prior to parameter estimation. Since, the computation of spectral tilt is not robust, DFW needs retuning to handle unseen speakers. Joint optimization in ODFWW results in better and more natural speech after the conversion. Overall, the method is efficient even for small matched source/target speaker datasets as very few parameters are estimated.

3.2. Joint Estimation with Regularization

As defined, the frequency weight $b(\omega)$ is prone to discontinuities that may introduce audible artefacts. This is tackled via the introduction of a regularisation term that penalises rapid variations by penalizing the first-order derivatives of $b(\omega)$ or equivalently the first-order differences

$$D'(w, b) = D(w, b) + \lambda(b(\omega_k) - b(\omega_{k-1}))^2, \quad (13)$$

where λ is a regularisation constant and $b(\omega_{-1}) = b(\omega_0)$. The partial derivative of D' with respect to frequency weighting at frequency ω_k is

$$\hat{b}(\omega_k) = \frac{1}{N(\lambda+1)}\sum_{n=1}^{N}\big(T_n(\omega_k) - S_n(w(\omega_k))\big) + \frac{\lambda}{\lambda+1}\hat{b}(\omega_{k-1}). \quad (14)$$

Further penalisation can be made via penalizing second-order derivatives. Note that $b(\omega_k)$ depends on previous frequencies $b(\omega_{k-1})$. Thus, DFWW can still be optimised using dynamic programming. When λ is too big, the frequency weighting function becomes constant.

3.3. Speeding up computation

ODFWW is computationally expensive if the dynamic programming is made using all $K \times K$ combinations of frequency bins, but it can be considerably sped up if the optimization is made on the band-diagonal of the K-by-K trelis. To do so, we have to ensure that the two speakers have the same vocal tract length. Thus, we break the frequency warping function $w(\omega)$ in two parts, a linear frequency warping part α and a residual frequency warping $w'(\omega)$: $w(\omega) = \alpha w'(\omega)$ and then we estimate them independently. The idea is that, the linear frequency warping absorbs most of the inter-speaker mismatch so that the residual frequency warping does not deviate considerably from the main diagonal of the trelis that corresponds to the warping $w'(\omega) = \omega$, allowing us to restrict the computations in a band around that main diagonal.

The linear frequency warping factor can be estimated by scanning a range of linear frequency warping factors for the factor that minimizes the distortion

$$D(\alpha) = \sum_{n=1}^{N}\int_0^{\min(\pi, \alpha\pi)}\big(T_n(\omega) - S_n(\alpha\omega) - \hat{b}(\omega)\big)^2 \mathrm{d}\omega, \quad (15)$$

where

$$\hat{b}(\omega) = \frac{1}{N}\sum_{n=1}^{N}\big(T_n(\omega) - S_n(\alpha\omega)\big). \quad (16)$$

4. RUN-TIME VOICE MORPHING FOR VOCAINE-BASED TTS

The frequency warping and weighting transform is used on the vocoder side of a statistical parametric TTS synthesizer [2] during spectral sampling according to equation (5). In addition, we apply some *power normalization* to preserve the power of the original frame by multiplying each sampled harmonic amplitude $\hat{T}(w(\omega_h))$ by a factor

$$\gamma = \frac{\sum_{h=1}^{H} S(\omega_h)^2}{\sum_{h=1}^{H}\hat{T}(w(\omega_h))^2}, \quad (17)$$

where ω_h is the radial frequency of the h-th sinusoid, the nominator corresponds to the power of the morphed spectra and the denominator corresponds to the power of the unmorphed spectra, according to the Parseval theorem [30]. The power normalization step is important because the transform modifies power and alters transients and the overall power contour. Regarding aperiodicity, we applied the frequency warping transform to the aperiodicity contour of each frame without any frequency weighting.

Our implementation uses a single transform for three reasons: 1) more transforms increase the computational load during runtime, 2) quality degradation is minimized because there is no quality penalty for switching between transforms and 3) similarity is not important for TTS in our use-case. In short, our design choices trade similarity for quality because our use-case is someone listening to a TTS without prior expectations on the identity of the speaker. Under these conditions, we demonstrate that it is possible to even *improve quality* of TTS speech using morphing for some target speakers.

Although this paper focuses on the conversion of spectral characteristics and not on prosody conversion, pitch level is a critical feature that affects the similarity between two voices. As reported in [31], adaptation of $\log(f_0)$ rather than f_0 fits better with the human perception of frequency distances. In this paper we adapt $\log(f_0)$ using a linear transformation based on first and second moments of source and target statistics:

$$\log \hat{f}_0^t = \mu^t_{\log f_0} + \frac{\sigma^t_{\log f_0}}{\sigma^s_{\log f_0}}(\log(f_0^s) - \mu^s_{\log f_0}), \quad (18)$$

where $\hat{f}_0^t$ is the estimated target f_0; $\mu^t_{\log f_0}$, $\mu^s_{\log f_0}$, $\sigma^t_{\log f_0}$, $\sigma^s_{\log f_0}$ are the target and source means and the corresponding variances.

5. RESULTS

The ODFWW algorithm is evaluated in the context of a high-quality Vocaine+LSTM-based TTS (VL-TTS) synthesizer [2]. Spectral and $\log(f_0)$ modifications are made during runtime in the vocoder. The overall increase in complexity in the vocoder is minimized to a modest +8% in a benchmark Android/ARM device by computing the spectral warping and weighting and the power normalization together with the spectral sampling because the two processes share a lot of computations. Following [2], the speech parameterization consists of 40-dimensional MCEP (including power), $\log f_0$ and a 7-dimensional band aperiodicity.

The target conversion data are derived from the CSTR VCTK corpus [32]. The corpus contains 108 English speakers with various accents uttered approximately 400 sentences (on average) recorded at 96 kHz sampling rate but downsampled to 22 kHz to match the

Table 1. MOS-Naturalness Scale

Score	Naturalness	Description
5.0	Excellent	Completely natural speech
4.0	Good	Mostly natural speech
3.0	Fair	Equally natural and unnatural speech
2.0	Poor	Mostly unnatural speech
1.0	Bad	Completely unnatural speech

Table 2. AB-7 Preference Scale

Score	Preference
+3.0	A is much better than B
+2.0	A is better than B
+1.0	A is slightly better than B
0.0	A and B are the same
-1.0	B is slightly better than A
-2.0	B is better than A
-3.0	B is much better than A

TTS sampling rate. Only 150 utterances per target speaker are used for training in order to reduce the overall computational cost. The source TTS corpus is an US English voice with 33K utterances recorded in high-quality studio conditions.

Our goal was not to convert our TTS voice to a particular target speaker but to find one or more speakers that are sufficiently distinct and with no quality degradation. Three evaluations were made with that goal in mind; the first evaluation identified a small subset of target speakers from the pool of 108 VCTK speakers, the second evaluation did a more thorough search to the small subset of speakers and the third evaluation was an AB-preference test to derive statistically significant decisions.

All listening tests were conducted by evenly distributing rating tasks to a large pool of listeners that were explicitly told to use headphones while all ratings obtained without headphones were automatically disregarded. Two types of listening tests were used: MOS-Naturalness and AB-preference with the corresponding rating scales displayed in Table 1 and 2, respectively.

The first evaluation was an MOS-naturalness test conducted with the following experimental conditions: 108+1=109 synthesizers, 10 text sentences per synthesizer and 8 ratings per utterance. The results are shown in Figure 1 as distributions due to the large number of measurements. We can observe that a significant portion of morphed TTS voices have higher MOS than the baseline [2] but not with certainty because the confidence intervals are very large.

Table 3. Text-To-Speech Results: MOS + Confidence Interval

Stimuli	MOS (US-EN)
USEL 22.05 kHz	3.798 ± 0.132
VL-TTS $\rightarrow$ p362	3.794 ± 0.097
USEL 16 kHz	3.776 ± 0.117
VL-TTS $\rightarrow$ p269	3.757 ± 0.099
VL-TTS	3.737 ± 0.091
VL-TTS $\rightarrow$ p330	3.723 ± 0.115
VL-TTS $\rightarrow$ p244	3.693 ± 0.088
VL-TTS $\rightarrow$ p233	3.682 ± 0.097
VL-TTS $\rightarrow$ p351	3.677 ± 0.094
VL-TTS $\rightarrow$ p253	3.669 ± 0.103
VL-TTS $\rightarrow$ p265	3.659 ± 0.095
VL-TTS $\rightarrow$ p306	3.619 ± 0.099
VL-TTS $\rightarrow$ p248	3.618 ± 0.119
VL-TTS $\rightarrow$ p238	3.617 ± 0.097
VL-TTS $\rightarrow$ p286	3.605 ± 0.090
VL-TTS $\rightarrow$ p277	3.580 ± 0.085
VL-TTS $\rightarrow$ p294	3.395 ± 0.104

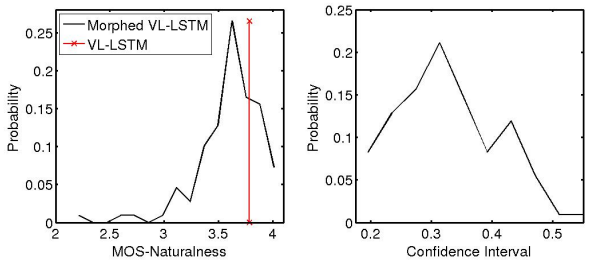

Fig. 1. MOS-Naturalness distributions

Table 4. Text-To-Speech Results: AB preference.

A	B	Score+CI
VL-TTS	USEL 22.05 kHz	-0.611 ± 0.170
VL-TTS	USEL 16 kHz	-0.351 ± 0.170
VL-TTS $\rightarrow$ p269	VL-TTS	0.096 ± 0.060
VL-TTS $\rightarrow$ p362	VL-TTS	0.107 ± 0.101
VL-TTS $\rightarrow$ p362	USEL 16 kHz	-0.191 ± 0.164
VL-TTS $\rightarrow$ p362	USEL 22.05 kHz	-0.346 ± 0.173

The second evaluation was used to clarify the results of the first evaluation. We hand-picked a number of TTS synthesizers that yielded higher than baseline MOS in the first evaluation and a few that yielded somewhat lower and we conducted a second experiment similar to the first one but with 100 utterances per synthesizer in order to reduce the confidence interval. In this evaluation we also included a state-of-the-art Unit-Selection TTS (USEL) synthesizer as a baseline, with two sampling rates: 16 kHz and 22.05 kHz. The results of the MOS-naturalness test are presented in Table 3. The morphed TTS synthesizers are indicated with an arrow pointing to the target speaker codename, e.g. VL-TTS $\rightarrow$ p362 corresponds to a VL-TTS synthesizer that is morphed to the VCTK speaker p362. We can observe that many morphed voices have quality that is comparable to the baseline and that two voices actually beat the baseline. Furthermore, the score of the best morphed voice matches the score of the 22.05 kHz USEL system.

The third evaluation was an AB preference test between the best TTS systems of the second evaluation and is depicted in Table 4. All results are statistically significant. We can observe that both evaluated morphed TTS outperformed the baseline and narrowed the gap between the USEL systems and VL-TTS. Qualitatively, these voices sound like different speakers. The key observation, however, is that quality improved.

6. ACKNOWLEDGMENTS

We would like to thank Hanna Silen for her tireless help conducting these experiments and her comments in various stages of the process.

7. CONCLUSION

This paper presented a novel spectral transformation algorithm that simultaneously recovers the optimal frequency warping and weighting between spectra of different speakers. The algorithm is used in a Vocaine+LSTM-based TTS synthesizer where it is shown to improve the quality of synthesized speech. We hypothesize that this is because morphing improves vocal characteristics that listeners find less pleasant.

8. REFERENCES

[1] Yannis Agiomyrgiannakis, "The Matching-Minimization algorithm, the INCA algorithm and a mathematical framework for Voice Conversion with unaligned corpora.," in *ICASSP*, 2016.

[2] Yannis Agiomyrgiannakis, "Vocaine the vocoder and applications in speech synthesis," in *ICASSP*, 2015.

[3] Alexander Kain and Michael W Macon, "Spectral voice conversion for text-to-speech synthesis," in *ICASSP*, 1998, pp. 285–288.

[4] Yannis Stylianou, Olivier Cappe, and Eric Moulines, "Continuous probabilistic transform for voice conversion," *Speech and Audio Processing, IEEE Transactions on*, vol. 6, no. 2, pp. 131–142, Mar 1998.

[5] Tomoki Toda, Alan W. Black, and Keiichi Tokuda, "Spectral conversion based on maximum likelihood estimation considering global variance of converted parameter," in *ICASSP*, March 2005, vol. 1, pp. 9–12.

[6] Heiga Zen, Keiichi Tokuda, and Alan W. Black, "Review: Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, Nov. 2009.

[7] Junichi Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of speaker adaptation algorithms for hmm-based speech synthesis and a constrained smaplr adaptation algorithm," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 17, no. 1, pp. 66–83, Jan 2009.

[8] Tomoki Toda, Y. Ohtani, and K. Shikano, "One-to-many and many-to-one voice conversion based on eigenvoices," in *ICASSP*, April 2007, vol. 4, pp. IV–1249–IV–1252.

[9] Hanna Silen, Jani Nurminen, Elina Helander, and Moncef Gabbouj, "Voice conversion for non-parallel datasets using dynamic kernel partial least squares regression.," in *Interspeech*. 2013, pp. 373–377, ISCA.

[10] Heiga Zen and Hasim Sak, "Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis," in *ICASSP*, 2015, pp. 4470–4474.

[11] Orhan Karaali, Gerald Corrigan, and Ira A. Gerson, "Speech synthesis with neural networks," *CoRR*, vol. cs.NE/9811031, 1998.

[12] Heiga Zen, Andrew Senior, and Mike Schuster, "Statistical parametric speech synthesis using deep neural networks," in *ICASSP*. IEEE, 2013, pp. 7962–7966.

[13] Zhizheng Wu, Pawel Swietojanski, Christophe Veaux, Stephen Renals, and Simon King, *A study of speaker adaptation for DNN-based speech synthesis*, International Speech Communication Association, 2015, Date of Acceptance: 01/06/2015.

[14] Toru Nakashika, Ryoichi Takashima, Tetsuya Takiguchi, and Yasuo Ariki, "Voice conversion in high-order eigen space using deep belief nets.," in *Interspeech*. 2013, pp. 369–372, ISCA.

[15] Daniel Erro, Agustín Alonso, Luis Serrano, Eva Navas, and Inma Hernáez, "Towards physically interpretable parametric voice conversion functions," in *Advances in Nonlinear Speech Processing*, Thomas Drugman and Thierry Dutoit, Eds., vol. 7911 of *Lecture Notes in Computer Science*, pp. 75–82. Springer Berlin Heidelberg, 2013.

[16] Elizabeth Godoy, Olivier Rosec, and Thierry Chonavel, "Voice conversion using dynamic frequency warping with amplitude scaling, for parallel or nonparallel corpora," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 4, pp. 1313–1323, May 2012.

[17] Daniel Erro, Asunción Moreno, and A. Bonafonte, "Voice conversion based on weighted frequency warping," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 5, pp. 922–931, 2010.

[18] Lakshmi Saheer, Hui Liang, John Dines, and Philip N. Garner, "Vtln-based rapid cross-lingual adaptation for statistical parametric speech synthesis," Idiap-RR Idiap-RR-12-2012, Idiap, 4 2012.

[19] Lakshmi Saheer, P.N. Garner, John Dines, and Hui Liang, "Vtln adaptation for statistical speech synthesis," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, March 2010, pp. 4838–4841.

[20] Lakshmi Saheer, John Dines, Philip N. Garner, and Hui Liang, "Implementation of vtln for statistical speech synthesis," in *Proceedings of ISCA Speech Synthesis Workshop*, 9 2010.

[21] Daniel Erro and Asunción Moreno, "Weighted frequency warping for voice conversion," in *Proc. InterSpeech*, 2007.

[22] H Valbret, Eric Moulines, and Jean-Pierre Tubach, "Voice transformation using PSOLA technique," *Speech Communication*, vol. 11, no. 2, pp. 175–187, 1992.

[23] Tudor-Cătălin Zorilă, Daniel Erro, Yannis Stylianou, and Inma Hernáez, "Towards a robust dynamic frequency warping text-independent voice conversion system," in *IberSpeech*, 2012.

[24] Keiichi Tokuda, Takao Kobayashi, Takashi Masuko, and Satoshi Imai, "Mel-generalized cepstral analysis-a unified approach to speech spectral estimation.," in *ICSLP*, 1994.

[25] Hideki 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.

[26] Hui Ye and Steven Young, "High quality voice morphing," in *ICASSP*. May 2004, vol. 1, pp. I–9–12 vol.1, IEEE.

[27] Tudor-Catalin Zorila, Daniel Erro, and Inma Hernáez, "Improving the quality of standard gmm-based voice conversion systems by considering physically motivated linear transformations.," in *IberSPEECH*. 2012, vol. 328, pp. 30–39, Springer.

[28] H. Valbret, Eric Moulines, and J.P. Tubach, "Voice transformation using psola technique," in *ICASSP*, Mar 1992, vol. 1, pp. 145–148 vol.1.

[29] Hiroaki Sakoe, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, pp. 43–49, 1978.

[30] Thomas Quatieri, *Discrete-Time Speech Signal Processing: Principles and Practice*, Prentice Hall Press, 2001.

[31] Daniel Erro, Asunción Moreno, and A. Bonafonte, "Voice conversion based on weighted frequency warping," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 5, pp. 922–931, July 2010.

[32] Christophe Veaux, Junichi Yamagishi, and Kirsten MacDonald, "English multi-speaker corpus for CSTR voice cloning toolkit," http://homepages.inf.ed.ac.uk/jyamagis/page3/page58/page58.html.