Speech Time-Scale Modification with GANs

Eyal Cohen, Felix Kreuk, and Joseph Keshet, *Senior Member, IEEE*

***Abstract*—While listening to spoken content, it is often desired to vary the speech rate while preserving the speaker's timbre and pitch. To date, advanced signal processing techniques are used to address this task, but it still remains a challenge to maintain a high speech quality at all time-scales. Inspired by the success of speech generation using Generative Adversarial Networks (GANs), we propose a novel unsupervised learning algorithm for time-scale modification (TSM) of speech, called ScalerGAN. The model is trained using a set of speech utterances, where no time-scales are provided. The ScalerGAN algorithm is composed of a generator that gets as input speech with the desired rate and outputs a time-adjusted speech; a discriminator that works on various spectrum scales; and a decoder that converts the time-adjusted signal back to the original rate to maintain consistency. Using an A/B test and conditional A/B test, human listeners were asked to compare ScalerGAN with other state-of-the-art TSM methods. The results showed that the speech quality of ScalerGAN outperforms all other methods.**

***Index Terms*—time-scale modification, speech synthesis, generative adversarial networks, deep neural networks**

I. Introduction

Time-scale modification (TSM) of speech is defined as speeding up or slowing down a given spoken utterance while maintaining the voice attributes such as speaker identity (including the original pitch), intelligibility, and naturalness. This task can be used to personalize the speaking rate when listening to spoken content such as podcasts or during language learning.

Existing approaches are based on advanced signal processing, which are often based on time-domain [1] or spectral-domain [2], [3] Overlap-Add (OLA). Related works are detailed in Section II. All those methods assume quasi-stationarity of the input speech. Hence they suffer from perceivable artifacts in the generated waveforms. Subjectively, it seems that the quality of modified speech can be improved, especially for extreme slow-down or speed-up.

In this paper, we would like to explore the possibilities of designing and implementing TSM of speech based on deep learning approaches. The main challenge of this task is the lack of supervised data at different time scales. That is, we don't have access to training examples of genuine speech utterances with different speaking rates.

We introduce *ScalerGAN*, a new deep learning algorithm for TSM that can speed up or slow down speech signals at a given rate. The algorithm is trained on a standard corpus of spoken utterances, were we denote the *relative* speaking rate of an utterance in the training corpus by 1. Then, at inference time, the input to the algorithm is an unseen speech signal and the desired rate (can be larger or smaller than 1), and the output is a newly generated adjusted speech, with the same voice properties as the input speech.

Most modern speech synthesis techniques [4], [5] are formulated with two main steps. The first step is to generate time-aligned spectral features from the raw waveform input, such as Mel-spectrogram. The second step applies a *vocoder*, generating a time-domain waveform conditioned on the predicted spectral features. Our algorithm focuses on the former, i.e., generating spectral features corresponding to a time-scaled modified speech by the desired rate. The latter step of converting the spectral features to a waveform is implemented with HiFi-GAN [6], though other vocoders may be considered [7], [8], [9], [10].

The ScalerGAN algorithm is based on Generative Adversarial Networks (GANs). GAN is a class of machine-learning framework that includes a generation network that generates candidates and a discriminative network that evaluates them. The networks are trained simultaneously using a combined loss function [11]. We borrow ideas from CycleGAN [12], StarGAN [13], and InGAN [14]. These algorithms convert images from one domain to another domain. To improve the target image quality, and to preserve consistency, the generated image is converted back to the original domain using an additional generator or the same generator. These algorithms use one or two discriminators to further stir the generator to generate images that cannot be distinguished from "real" images. In ScalerGAN, the speech is time-adjusted to the desired rate using a generator. Then, the quality of the generated speech is improved in two ways: (i) using a discriminator that classifies whether the speech is real or synthetic (fake); and (ii) using a decoder that converts the time-adjusted speech back to the original rate to preserve consistency. The decoder is implemented by using the generator but with an inverse rate.

Our work is related to [14], which tackles the problem of intelligently shrinking and expanding images. Note that this method is designed to work on *a single given image*: it is trained on a single image, and then it can scale this image only. Moreover, the method is not suitable to work with speech signals. The reason is that the axes of images and spectrograms do not carry the same meaning [15]. Our method, in contrast, is trained on a speech corpus and, at inference-time, can adjust the rate of any speech signal.

We compared ScalerGAN to eleven other methods with six different time-scales using subjective human evaluation. Results suggest that ScalerGAN was preferable by human listeners over other methods at all rates.

This paper is organized as follows. In the next section, we

Manuscript received January 18, 2022; revised March 13, 2022; accepted March 23, 2022. Date of publication; date of current version. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiaodong Cui. (Corresponding author: Eyal Cohen.)
The authors are with the Department of Computer Science, Bar-Ilan University, Ramat-Gan 5290002, Israel (e-mail: eyalcohen308@gmail.com; felixkreuk@gmail.com; jkeshet@cs.biu.ac.il).
Digital Object Identifier 10.1109/LSP.2022.3164361

present previous work. In Section III we present the problem setting. In Section IV we introduce the model and describe its architecture. In Section V we detail the experimental setting and present the results. In Section VI we conclude the paper. Samples and code are publicly available under the following link: https://eyalcohen308.github.io/ScalerGAN.

II. Related work

The TSM task has been traditionally tackled using advanced signal processing techniques. The most popular methods for *time-domain* OLA are synchronized overlap-add (SOLA) [16], and waveform similarity overlap-add (WSOLA) [1]. These techniques were improved by the ESOLA method [17], which proposes epoch-synchronous OLA time/pitch-scaling of speech. The FESOLA algorithm [18] is a modification of ESOLA. It proposes using a cross-correlation function to align time-smeared epochs before overlapping the speech segments. The overlap between frames depends on the time-scaling factor.

In contrast, there are several *spectral domain* OLA methods, where the most prominent method is the Phase-Vocoder (PV) [19]. Identity Phase-Locking Phase Vocoder (IPL) and Scaled Phase Locking (SPL) methods [2] are an improvement of PV. These techniques allow direct manipulation of the signal in the frequency-domain, by pitch-shifting, chorusing, harmonizing, and partial stretching. The PhaVoRIT algorithm [3] uses multi-resolution peak-picking, sinusoidal trajectory heuristics, and silent passage phase reset techniques for improving the audio quality of IPL and SPL. Furthermore, harmonic persuasive separation (HPTSM) [19] suggests to modify the harmonic component of the signal with phase vocoder and the noise-like percussive components with a simple time-domain OLA.

Recently, the μTVS method [20] proposed time-scaling the instantaneous amplitude and the instantaneous phase of a filter bank of time-varying sinusoids.

III. Problem Settings

Given a speech utterance, our goal is to speed up or slow down the speech by a given rate $r \in \mathbb{R}$, while keeping the intelligibility and speaker identity as much as possible. Throughout the paper, we use the term *rate* to denote the desired change of the speaking rate and, later on, use the term *scale* in relation to various scale representations of the signal. The rate r can be higher or lower than 1, which corresponds to slow-down or speeding-up, respectively.

The input speech is represented as a sequence of acoustic features, such as Mel-spectrum or short-time Fourier transform (STFT), denoted by $\bar{\mathbf{x}} = (x_1, \cdots, x_N)$, so every frame $x_i \in \mathcal{X} \subset \mathbb{R}^d$ for $1 \leq i \leq N$ is a d-dimensional vector. We denote the domain of all finite-length sequences by $\mathcal{X}^*$.

The model synthesizes a new time-scaled sequence of acoustic features denoted by $\bar{\mathbf{y}} = (y_1, \cdots, y_M)$, where $y_j \in \mathcal{X}$ for $1 \leq j \leq M$, and $M = \lceil Nr \rceil$ is the target size of the output.

Our goal is to learn a generative function $G : \mathcal{X}^* \times \mathbb{R} \rightarrow \mathcal{X}^*$ that given a finite-length sequence and the desired rate r will generate a finite sequence with a scaled duration according to the given rate r. The most trivial implementation of the function G is by resampling the original signal $\bar{\mathbf{x}}$. However, this implementation does not maintain the speaker identity, pitch, and intelligibility of the spoken content [21].

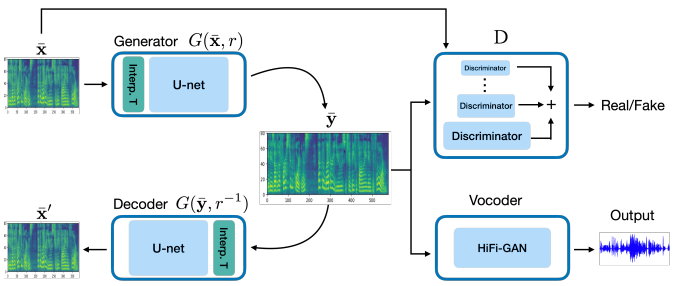

Fig. 1. The ScalerGAN model. The generator G gets a spectrogram $\bar{\mathbf{x}}$ and a rate r as input, and generates a new time-adjusted spectrogram $\bar{\mathbf{y}}$. During training, the resulting spectrogram is further processed by a discriminator D that gets as input either $\bar{\mathbf{x}}$ or $\bar{\mathbf{y}}$ and decides whether the input is real or fake. Parallel to that, a decoder gets $\bar{\mathbf{y}}$ as input and tries to reconstruct $\bar{\mathbf{x}}$ back from it, where the decoder is implemented using the same generator G. At inference time, we use a vocoder to convert the spectrogram into a waveform.

IV. Model

Recall that the main challenge of designing a time-scale system based on machine learning is that we do not have training examples of different rates. In this section, we describe *ScalerGAN* unsupervised network architecture. A generator G gets as input the spectrogram $\bar{\mathbf{x}}$ and the desired rate r and outputs a spectrogram $\bar{\mathbf{y}}$ time-scaled to the desired rate r. During training, the resulting time-scaled spectrogram $\bar{\mathbf{y}}$ is then evaluated using two networks. The first network is a multi-scale discriminator D trained to discriminate between real and time-scaled ("fake") inputs. The second network is a decoder that re-generates the original spectrogram $\bar{\mathbf{x}}$ with the *same* generator G, operated with a rate of r^{-1} instead of r. The reconstructed spectrogram $\bar{\mathbf{x}}'$ is evaluated by comparing it to the original spectrogram $\bar{\mathbf{x}}$. At inference time, the spectrogram $\bar{\mathbf{y}}$ is converted to a waveform using a neural-based vocoder [6]. The proposed architecture is depicted in Figure 1.

In more detail, the generator $G : \mathcal{X}^* \times \mathbb{R} \rightarrow \mathcal{X}^*$ consists of two components: (i) an operator T^r that interpolates the input spectrogram by a factor r on the time domain; and (ii) a deep network in the form of U-Net [22]. The decoupling between the U-Net and the transformation T^r allows us to train G on speech spectrograms regardless of specific rate r while generalizing to any rate at inference time. The U-net learns to transform linearly interpolated spectrograms to spectrograms that represent the original speech. This is achieved by two components: a discriminator and a decoder.

The discriminator D is a function that is trained to discriminate between real input spectrograms and synthetically generated spectrograms. The speaking rate is influenced by the speed of articulation and can influence unevenly on the spectrograms [21]. For example, changes in the rate of utterance would tend to be absorbed more by the vowels than the consonants [23]. To allow the discriminator to work on such variations, it is designed as a set of several sub-discriminators, each of which operates at a different scale of the spectrogram [14]. We denote by $T^{s \times s}$ the operator of the interpolation of

both axes by a factor s. So if we apply $T^{2\times 2}$ on an image of size 8×8, it means we add pixels to generate an image of size 16×16.

The input to each sub-discriminator is a down-scaled version of the spectrogram $\bar{\mathbf{x}}$, where down-scaling is is implemented by $T^{s^{-1}\times s^{-1}}$. Each sub-discriminator $D^s : \mathcal{X}^* \rightarrow [0,1]^*$ is a classifier, composed of 4 convolutional layers, that predicts whether its input is derived from a real signal or a synthetic one ("fake"). The output of the classifier is not a single decision but rather a probability matrix. Each element of this matrix corresponds to a patch in the input, and represents the probability of how realistic the patch is. The probability matrix is up-scaled using $T^{s\times s}$ – so that the outputs of all the sub-discriminator have the same size. The multi-scale discriminator, D, encourages the generator, G, to produce more realistic output in both coarse and fine scales.

The final discriminator is a weighted sum over all outputs,

$$D(\bar{\mathbf{x}}) = \sum_s \alpha_s \, T^{s\times s} \left(D^s(T^{s^{-1}\times s^{-1}}(\bar{\mathbf{x}}) \right),$$

where α_s are weights that are part of the learning parameters. The number of patches s used is described in the next section.

The last component is a decoder. The decoder's goal is to reconstruct the original signal from the synthetic one. It is implemented using the generator G where $\bar{\mathbf{y}}$ and rate of r^{-1} are used as input parameters, namely, $\bar{\mathbf{x}}' = G(\bar{\mathbf{x}}, r^{-1})$.

The system is trained using two loss functions. The first loss is associated with discriminator and controlled by the least-squares (LS) GAN objective. Specifically, the discriminator D is trained to differentiate between a real input $\bar{\mathbf{x}}$ and one generated by the generator, $\bar{\mathbf{y}} = G(\bar{\mathbf{x}}, r)$. Formally,

$$\mathcal{L}_{LS}(G,D) = \mathbb{E}_{\bar{\mathbf{x}}\sim p(\bar{\mathbf{x}})}\left[(D(\bar{\mathbf{x}}) - J)^2\right] + \mathbb{E}_{\bar{\mathbf{x}}\sim p(\bar{\mathbf{x}})}\left[D(G(\bar{\mathbf{x}}, r))^2\right] \ , \quad (1)$$

where J is all-ones matrix of the same size as the output of D. The first term encourages the discriminator D to output matrix of ones for inputs that are real spectrograms, while the second term encourages D to be a matrix of zeros for inputs that are time-scaled synthetic spectrograms. The second part also pushes the generator G to generate more realistic spectrograms that will fool the discriminator D.

The second loss function is associated with the decoder. We would like minimize the L_1 distance between $\bar{\mathbf{x}}' = G(\bar{\mathbf{y}}, r^{-1})$ and $\bar{\mathbf{x}}$ so as to prevent convergence to trivial solutions:

$$\mathcal{L}_R(G) = \|G(G(\bar{\mathbf{x}},\ r),\ r^{-1}) - \bar{\mathbf{x}}\|_1 \ , \quad (2)$$

where $\bar{\mathbf{y}} = G(\bar{\mathbf{x}}; r)$ and $\bar{\mathbf{x}}' = G(\bar{\mathbf{y}}; r^{-1})$. This encourages G to avoid mode-collapse and maintain the same spectral content before and after the model transformation. Overall, we find D that maximizes the loss $\mathcal{L}_{LS}$ and G that minimizes both loss functions. Formally,

$$\min_G \max_D \ \mathcal{L}_{LS}(G,D) + \lambda \mathcal{L}_R(G) \ . \quad (3)$$

This is the loss that was used to train our ScalerGAN.

V. Experiments

A. Datasets

We conducted experiments on the LJSpeech dataset [24], a standard benchmark for speech synthesis models, and the VCTK dataset for testing our method with unseen speakers, similar to [25]. The LJSpeech dataset consists of 13,100 short audio clips of a single female speaker reading passages from 7 non-fiction books with a total length of approximately 24 hours. The audio format is 16-bit PCM with a sampling rate of 22,050Hz. For VCTK, we used a small subset named DR-VCTK. DR-VCTK contains 28 speakers, 14 males and 14 females for training, and one male and one female for testing. For consistency, DR-VCTK files were up-sampled from 16,000 Hz to 22,050 Hz. We extracted Mel-spectrograms for the above data using an FFT window size of 1024, a hop size of 256, and 80 Mel bins.

B. Experimental Setup

The speech examples used to train the models are Mel-spectrograms. During training we randomly sample segments of 256 frames from the original spectrogram (and at inference, the whole spectrogram is used).

The generator is implemented as a U-Net [22] consisting of a bottleneck with 6 residual-blocks [26]. The discriminator is composed of a set of 5 sub-discriminators. Each one is implemented as a ConvNet with 4 layers with kernel sizes of $(3,3,3,1)$ with strides $(1,2,1,1)$. We used the Leaky ReLU activation function between the sub-discriminators layers with a negative slope of 0.2. The input to each sub-discriminator was a down-scaled version of the spectrogram, where the down-scale factors were 1.2^n for $0 \leq n < 5$. We used batch normalization [27] between convolutional layers and spectral normalization [28] in G and D layers.

Both generator and discriminator were trained with a batch size of 24 and for 500 epochs. The learning rate for both model was 5e-5 using ADAM optimizer [29] with $\beta_1 = 0.5, \beta_2 = 0.999$, and the trade-off between losses was $\lambda = 0.1$. At each batch, the desired output size, determined by r, was chosen randomly in the range $[0.3, 1.8]$. We used curriculum learning for sampling r: the initial rate r was sampled from $[1.0, 1.0]$ and gradually transformed to be sampled from $[0.3, 1.8]$ as the training progressed. After 200 epochs, we trained the decoder twice per epoch. The generator and discriminator were updated at every iteration. All the interpolations were bilinear [30].

C. Results

We compared our method against 11 state-of-the-art methods: PhaseVocoder [2], ESOLA [17], FESOLA [18], WSOLA [1], IPL and SPL [2], PhaVoRIT IPL and PhaVoRIT SPL [3], Élastique, HPTSM [19], and μTVS [20][1]. We evaluated our approach with qualitative experiments with human listeners using a crowd-sourcing platform. Our evaluation assessed the

[1]All the implementations were done using the TSM Toolbox [19], except Élastique, which is a commercial state-of-the-art TSM algorithm by zPlane, where we used a wrapper by AudioLabs. See https://www.audiolabs-erlangen.de/resources/MIR/TSMtoolbox/

TABLE I

MEAN SCALERGAN SUCCESS RATE IN TWO EXPERIMENTS: A/B TEST ('AB') AND CONDITIONAL A/B TEST ('CAB'). THE VALUES INDICATE HOW MANY TIMES THE SUBJECTIVE SOUND QUALITY OF THE SPEECH GENERATED BY SCALERGAN WAS BETTER THAN THE SPEECH GENERATED BY METHOD X, FOR THE DIFFERENT RATES. VALUES ARE GIVEN AS PERCENTAGE AND A VALUE HIGHER THAN 50% MEANS SCALERGAN IS BETTER THAN METHOD X. THE METHODS FROM LEFT TO RIGHT ARE: PHASEVOCODER [2], ESOLA [17], FESOLA [18], WSOLA [1], IPL [2], PHAVORIT IPL [3], SPL [2], PHAVORIT SPL [3], ÉLASTIQUE[1], HPTSM [19], AND μTVS [20].

	PV		**ES**		**FES**		**WS**		**IPL**		**P_IPL**		**SPL**		**P_SPL**		**EL**		**HP**		**μTVS**	
Rate	AB	CAB	AB	CAB	AB	CAB	AB	CAB	AB	CAB	AB	CAB	AB	CAB	AB	CAB	AB	CAB	AB	CAB	AB	CAB
0.5	95.00	78.33	65.00	66.77	70.56	61.11	66.67	66.67	73.89	61.11	73.33	76.67	78.89	66.11	92.78	76.67	55.00	55.00	73.89	68.33	75.56	68.89
0.7	95.00	85.00	73.33	68.89	63.33	62.22	86.11	75.00	78.33	71.67	68.89	78.33	82.22	64.44	90.56	78.33	58.89	56.11	71.11	65.00	66.67	66.11
0.9	97.78	90.00	70.56	60.0	47.22	36.11	84.44	69.44	82.22	66.67	71.11	69.44	76.11	68.89	86.67	69.44	66.67	48.33	80.00	70.00	65.56	57.78
1.1	98.89	85.56	70.00	64.44	75.00	67.78	81.67	73.33	83.89	76.67	77.22	81.11	86.67	79.44	85.56	81.11	64.44	53.33	82.22	70.56	77.22	67.22
1.3	97.78	90.56	85.00	75.56	91.11	69.44	90.56	79.44	83.89	77.78	86.67	83.33	89.44	79.44	90.56	83.33	67.22	59.44	88.33	76.67	81.67	55.56
1.5	96.11	86.67	94.44	77.22	88.33	80.56	93.89	82.22	84.44	81.67	88.89	83.33	93.89	81.11	90.00	83.33	58.89	55.56	88.33	77.22	81.67	71.67
Overall	**96.76**	**86.02**	**76.39**	**68.80**	**72.59**	**62.87**	**83.89**	**74.35**	**81.11**	**72.59**	**77.69**	**69.44**	**84.54**	**73.24**	**89.35**	**78.70**	**61.85**	**54.63**	**80.65**	**71.30**	**74.42**	**64.54**

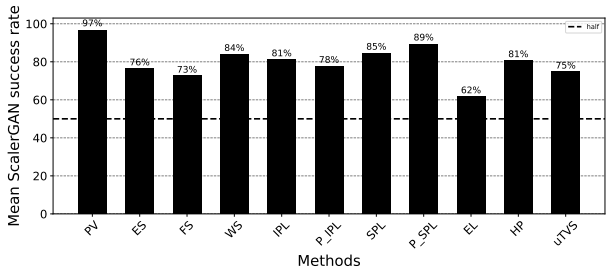

Fig. 2. A/B test comparison between ScalerGAN and other methods. The graphs shows the mean of ScalerGan's success rates versus each of the methods.

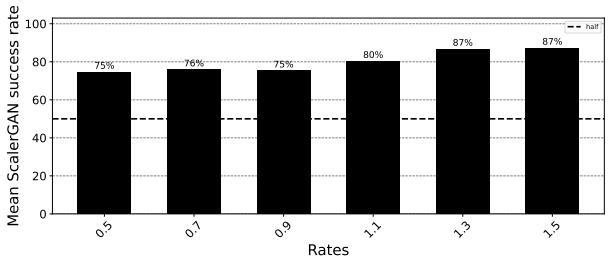

Fig. 3. A/B test comparison between ScalerGAN model and other methods. The graphs shows the mean of ScalerGan's success rates grouped by each tested rate.

perceived quality of the generated audio with different time scales. To that end, we randomly selected 20 utterances from the LJSpeech dataset and 25 utterances from the DR-VCTK test-set (13 males, 12 females). We used them as references for both experiments. We generated six different time-scaled versions for each utterance $r \in \{0.5, 0.7, 0.9, 1.1, 1.3, 1.5\}$ using the ScalerGAN and all other methods.

Mean Opinion Score (MOS) is a numerical measure of subjective human evaluation, often used to measure synthesized speech quality. However, MOS is often very subjective and hard to reproduce [31]. In contrast to standard MOS evaluations, we used a pairwise comparison that allowed us to verify the statistical significance of our results. We used a crowd-source platform (Amazon mTurk) with native American English raters. Each rater was presented with 60 audio samples during the evaluation and asked to select the most natural-sounding one. We recorded the percent of raters that preferred ScalerGAN over the competing method for each experiment.

We used to type of testing methods:

a) A/B Test: The first experiment was an *A/B Test*: raters listened to two speech utterances, one generated by ScalerGAN and the competing method. The listener was tasked to select the recording that sounded most natural and of high perceived quality.

b) Conditional A/B Test: The second experiment was a *Conditional A/B Test*: First, raters listened to the origin speech utterance and then to the two generated speech utterances (ScalerGAN and a competing method). Then, based on the origin utterance, the raters were asked to choose which generated speech utterances sounded the best.

Table I summarizes the ScalerGAN results versus each method for the above two experiments. Each value in the table is the mean aggregated ScalerGAN's success rate versus a given method. Results suggest that ScalerGAN outperforms all methods in both experiments. We will further present them in two different views to better analyze the result.

We computed the statistical significance of our result. Based on a binomial test ($N = 1080$), participants were significantly more likely to prefer ScalerGAN than any other methods (the worst p value was $3.2e-15$ for the A/B test and 0.0013 for conditional A/B test).

In Figure 2, we show the percentage of ScalerGan compared to other methods. We aggregated the success rate overall rates values. It is evident that ScalerGAN outperforms all other methods using the qualitative assessment we presented. A closer look at the numbers shows that the tightest gap was with the commercial system Élastique[1] (denoted as ES).

In Figure 3, ScalerGan's success rate was aggregated and grouped by each rate. Results suggest that while ScalerGAN is preferred at all rates, its relative improvement is even more apparent at higher rates. At such rates, new audio samples need to be generated in a way that maintains the same perceived audio quality. ScalerGAN is explicitly trained to generate new speech samples indistinguishable from real data, and therefore can operate at high scales while maintaining high audio quality.

VI. CONCLUSIONS

In this work, we introduced ScalerGAN, an algorithm for time-scale modification of speech signals. Results suggest that our proposed model has a significant advantage in subjective listening tests. Compared to the signal-processing-based model, one major drawback of our model is its processing time. Our future work will focus on improving this issue and will furthermore focus on expanding the method to music.

REFERENCES

[1] W. Verhelst and M. Roelands, "An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, 1993, pp. 554–557.

[2] J. Laroche and M. Dolson, "New phase-vocoder techniques for pitch-shifting, harmonizing and other exotic effects," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 1999, pp. 91–94.

[3] T. Karrer, E. Lee, and J. O. Borchers, "PhaVoRIT: A phase vocoder for real-time interactive time-stretching," in *International Computer Music Conference (ICMC)*, 2006, pp. 708–715.

[4] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. J. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning Wavenet on Mel spectrogram predictions," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4779–4783, 2018.

[5] W. Ping, K. Peng, A. Gibiansky, S. Ö. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep Voice 3: Scaling text-to-speech with convolutional sequence learning." in *International Conference on Learning Representation*, 2018.

[6] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[7] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3617–3621.

[8] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *SSW*, p. 2, 2016.

[9] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6199–6203.

[10] M. Binkowski, J. Donahue, S. Dieleman, A. Clark, E. Elsen, N. Casagrande, L. C. Cobo, and K. Simonyan, "High fidelity speech synthesis with adversarial networks," in *International Conference on Learning Representation*, 2020.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, vol. 27, 2014, pp. 2672–2680.

[12] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE international conference on computer vision (ICCV)*, 2017, pp. 2223–2232.

[13] Y. Choi, M.-J. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *CVPR*, 2018.

[14] A. Shocher, S. Bagon, P. Isola, and M. Irani, "Ingan: Capturing and retargeting the" dna" of a natural image," in *IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4492–4501.

[15] L. Wyse, "Audio spectrogram representations for processing with convolutional neural networks," in *Proceedings of the First International Conference on Deep Learning and Music*, 2017, pp. 37–41.

[16] S. Roucos and A. Wilgus, "High quality time-scale modification for speech," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 10, 1985, pp. 493–496.

[17] S. Rudresh, A. Vasisht, K. Vijayan, and C. S. Seelamantula, "Epoch-synchronous overlap-add (ESOLA) for time-and pitch-scale modification of speech signals," *arXiv preprint arXiv:1801.06492*, 2018.

[18] T. Roberts and K. K. Paliwal, "Time-scale modification using fuzzy epoch-synchronous overlap-add (FESOLA)," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 31–34, 2019.

[19] J. Driedger, M. Müller, and S. Ewert, "Improving time-scale modification of music signals using harmonic-percussive separation," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 105–109, 2013.

[20] N. Sharma, S. Potadar, S. R. Chetupalli, and T. Sreenivas, "Mel-scale sub-band modelling for perceptually improved time-scale modification of speech and audio signals," in *2017 Twenty-third National Conference on Communications (NCC)*, 2017, pp. 1–5.

[21] B. Sylvestre and P. Kabal, "Time-scale modification of speech using an incremental time-frequency approach with waveform structure compensation," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, 1992, pp. 81–84 vol.1.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[23] J. M. Pickett, *The sounds of speech communication*. University Park Press, 1980.

[24] K. Ito and L. Johnson, "The LJ Speech Dataset," https://keithito.com/LJ-Speech-Dataset/, 2017.

[25] F. Fang, J. Yamagishi, I. Echizen, M. Sahidullah, and T. Kinnunen, "Transforming acoustic characteristics to deceive playback spoofing countermeasures of speaker verification systems," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2018, pp. 1–9.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[27] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. PMLR, 2015, pp. 448–456.

[28] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," in *International Conference on Learning Representations*, 2018.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representation*, 2015.

[30] A. Prajapati, S. Naik, and S. Mehta, "Evaluation of different image interpolation algorithms," *International Journal of Computer Applications*, vol. 58, no. 12, pp. 6–12, 2012.

[31] R. C. Streijl, S. Winkler, and D. S. Hands, "Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives," *Multimedia Systems*, vol. 22, pp. 213–227, 2014.