WSRGlow: A Glow-based Waveform Generative Model for Audio Super-Resolution

Kexun Zhang[1], Yi Ren[1], Changliang Xu[2], Zhou Zhao[1]

[1]Zhejiang University, China

[2]State Key Laboratory of Media Convergence Production Technology and Systems, China

{kexunz, rayeren, zhaozhou}@zju.edu.cn, xu@shuwen.com

Abstract

Audio super-resolution is the task of constructing a high-resolution (HR) audio from a low-resolution (LR) audio by adding the missing band. Previous methods based on convolutional neural networks and mean squared error training objective have relatively low performance, while adversarial generative models are difficult to train and tune. Recently, normalizing flow has attracted a lot of attention for its high performance, simple training and fast inference. In this paper, we propose WSRGlow, a Glow-based waveform generative model to perform audio super-resolution. Specifically, 1) we integrate WaveNet and Glow to directly maximize the exact likelihood of the target HR audio conditioned on LR information; and 2) to exploit the audio information from low-resolution audio, we propose an LR audio encoder and an STFT encoder, which encode the LR information from the time domain and frequency domain respectively. The experimental results show that the proposed model is easier to train and outperforms the previous works in terms of both objective and perceptual quality. WSRGlow is also the first model to produce 48kHz waveforms from 12kHz LR audio. Audio samples are available at https://zkx06111.github.io/wsrglow/.

Index Terms: speech super-resolution, deep learning, generative models

1. Introduction

Audio super-resolution is the task of constructing a high-resolution (HR) audio from a low-resolution (LR) audio by adding the missing band in a coherent manner. Formally, given a LR sequence of audio samples $x_l = (x_{1/R_l}, \ldots, x_{R_l T/R_l})$, we wish to synthesize a high-resolution audio signal $x_h = (x_{1/R_h}, \ldots, x_{R_h T/R_h})$, where R_l and R_h are the sampling rates of the low and high-resolution signals. Audio super-resolution can be used to achieve better speech quality with a limited sampling rate in telecommunications. It can also improve the audio quality of ancient recordings and music, and facilitate TTS [1, 2, 3, 4, 5] and singing voice synthesis [6, 7] models to generate higher-quality voices.

Previous approaches to this task include Gaussian mixture models [8, 9], linear predictive coding [10] and neural networks [11]. Learning-based methods perform better in this context because they can capture sophisticated domain-specific information. Convolutional networks [12, 13] and generative adversarial networks (GAN) [14, 15, 16] have shown to greatly improve the quality of the synthesized high-resolution audio. However, GANs are known to be hard to train and produce outputs with significant artifacts. Recently, the success of normalizing flow models has been used on many generative tasks due to its ability to optimize the log-likelihood exactly and parallelizability of both training and synthesis. Generative glow

is capable of efficient realistic-looking synthesis and manipulation of large images [17, 18, 19]. Normalizing flow conditioned on low-resolution images [20] has also improved the performance of image super-resolution, which is similar to speech super-resolution in many ways.

Considering that audio super-resolution is a one-to-many mapping (ill-posed) problem since multiple possible HR audio samples correspond to a given LR one, in this paper, we explore the potential of normalizing flow on modeling the conditional distribution of HR audio samples given an LR one. Specifically, we propose WSRGlow, a novel model for audio super-resolution based on generative flows. WSRGlow generates HR audio by sampling from a simple zero mean spherical Gaussian distribution. The sample taken from the simple distribution is then put through several invertible layers and transformed to the desired output audio. The layers of our model are similar to those in WaveGlow [21] and WaveNet [22]. During training, HR audio is fed to the model as input to obtain the corresponding sample points in the simple distribution, so that we can train our model by directly maximizing the likelihood. During inference, WSRGlow takes a sample from the simple distribution and passes it through several invertible layers to compute the desired output audio.

To better extract the valuable information in LR audio, we propose two encoders: an LR audio encoder and an STFT encoder, which encode LR information from the time and frequency domains, respectively. The LR audio encoder applies μ-law transformation to raw low-resolution waveform [23], quantizes each sample to mu-law code and maps the code sequence to embedding vectors. However, the LR audio encoder only provides a time-domain perspective and it is challenging for LR audio encoder to exploit several characteristics of the signal, including cyclic behavior and long-range dependence, due to the limited receptive field and model capacity. Therefore, we further introduce the frequency domain encoder called STFT encoder, which first applies short-time Fourier transform (STFT) to the waveform and then encodes the phase and magnitude spectrogram, both of which are widely used in modern speech applications and methods [16, 24]. The contributions of this work are summarized as follows:

- To the best of our knowledge, WSRGlow is the first model to use normalizing flow structure for audio super-resolution task and proves the potential of normalizing flow for audio super-resolution.

- We propose an LR audio encoder and an STFT encoder to exploit LR information from both time domain and frequency domain.

- Our experiments on VCTK [25] dataset show that 1) the proposed model is easier to train and outperforms the previous works both objectively and perceptually; and 2) STFT

encoder improves the model performance significantly[1] and makes the convergence much faster.

2. Model Architecture

Audio super-resolution is an ill-posed problem whose solution is not unique. Various possible solutions exist and form a distribution in the solution space. To capture the probabilistic nature of super-resolution audio, we employ normalizing flow to model the conditional distribution of HR audio given the LR audio. WSRGlow captures the bijection between a zero mean spherical Gaussian distribution and the probability space of audio waveform using invertible layers similar to Glow [17]. To generate high-resolution audio, we encode the low-resolution audio using an LR audio encoder and an STFT encoder and then use the encoding as the condition for the glow layers. The architecture of WSRGlow is depicted in Figure 1. We describe the details of each component below.
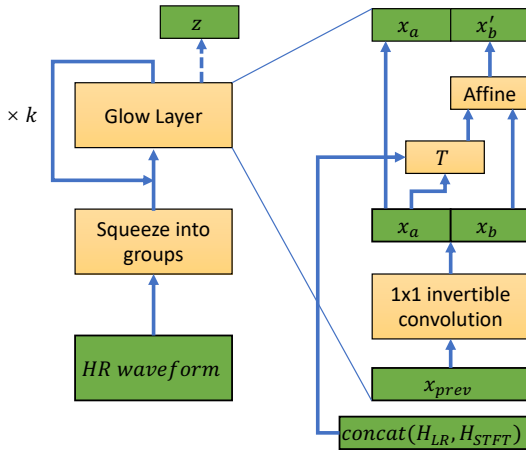

Figure 1: *Overall architecture. During training, the model runs along the direction of the arrows, while the model runs inversely during inference (except the T transformation). The two conditional encoders' outputs are concatenated to be the condition for transformation T.*

2.1. Glow Layers

To generate high-resolution audio, we take a sample ($\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{z}; 0, \boldsymbol{I})$) from a simple Gaussian distribution and passes it through several invertible layers $\boldsymbol{f}_i$ that transforms the simple distribution to the desired distribution $\boldsymbol{x}$ in the audio space:

$$\boldsymbol{x} = \boldsymbol{f}_0 \circ \boldsymbol{f}_1 \circ \cdots \circ \boldsymbol{f}_k(\boldsymbol{z}).$$

During training, high-resolution audio is regarded as a vector and squeezed into groups of 8, as in WaveGlow [21]. This vector is then passed through the invertible layers and transformed into a simple distribution, so that we can optimize the negative loglikelihood directly. The invertibility of the WSRGlow layers ensure that we can calculate the loglikelihood $\log p_\theta(\boldsymbol{x})$ directly using inverted mappings:

$$\log p_\theta(\boldsymbol{x}) = \log p_\theta(\boldsymbol{z}) + \sum_{i=1}^{k} \log |\det(\boldsymbol{J}(\boldsymbol{f}_i^{-1}(x)))|,$$

---

[1]Audio samples are available at https://zkx06111.github.io/wsrglow/. The codes for this paper is available at https://github.com/zkx06111/WSRGlow.

$$\boldsymbol{z} = \boldsymbol{f}_k^{-1} \circ \boldsymbol{f}_{k-1}^{-1} \circ \cdots \circ \boldsymbol{f}_0^{-1}(x),$$

where $\boldsymbol{J}$ is the Jacobian of the inversed transformation $\boldsymbol{f}_i^{-1}$. The invertible layers have a similar structure to those in WaveGlow [21]. Each of the WSRGlow layers consists of an invertible 1x1 convolution $\boldsymbol{f}_{conv}$ and an affine coupling layer $\boldsymbol{f}_{coupling}$. The detailed structure of these layers will be described below.

2.1.1. Affine Coupling Layer

We use an affine coupling layer $\boldsymbol{f}_{coupling}$ to make a WSRGlow layer invertible. The channels of the input tensor $\boldsymbol{x}$ is split into two halves $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$. Half of the channels are passed through a transformation T to produce additive term $\boldsymbol{t}$ and multiplicative term $\boldsymbol{s}$ for the other half.

$$
\begin{aligned}
\boldsymbol{x}_a, \boldsymbol{x}_b &= split(x), \\
(\log \boldsymbol{s}, \boldsymbol{t}) &= T(\boldsymbol{x}_a, concat(H_{LR}, H_{STFT})), \\
\boldsymbol{x}_b' &= \boldsymbol{s} \odot \boldsymbol{x}_b + \boldsymbol{t}, \\
\boldsymbol{f}_{coupling}^{-1}(\boldsymbol{x}) &= concat(\boldsymbol{x}_a, \boldsymbol{x}_b'), \\
\log |\det(\boldsymbol{J}(\boldsymbol{f}_{coupling}^{-1}(\boldsymbol{x}))| &= \log |\boldsymbol{s}|.
\end{aligned}
\tag{1}
$$

Here T can be any transformation that does not have to be invertible because the coupling layer keeps the invertibility of the network. When inverting the network, since $\boldsymbol{x}_a$ is passed without changing, we can obtain $\boldsymbol{s}$ and $\boldsymbol{t}$ by recomputing T with $\boldsymbol{x}_a$. We can then solve equation (1) for $\boldsymbol{x}_b$ using $\boldsymbol{s}, \boldsymbol{t}, \boldsymbol{x}_b'$: $\boldsymbol{x}_b = \frac{\boldsymbol{x}_b - \boldsymbol{t}}{\boldsymbol{s}}$.

In our model, the transformation T is similar to the structure of non-causal WaveNet [22] with dilated convolutions, gated-$\tanh$ activations and residual connections. The main difference between T and non-causal WaveNet is that the condition parameter for T is not a mel-spectrogram, but the encoded low-resolution audio signal $concat(H_{LR}, H_{STFT})$.

2.1.2. 1x1 Invertible Convolution

Following Glow [17], we employ a 1x1 invertible convolution layer $\boldsymbol{f}_{conv}^{-1} = \boldsymbol{W}\boldsymbol{x}$ before every affine layer to enable two halves of the channels to modify the one another. The convolution kernel is initialized to be orthonormal to keep its invertibility.

2.2. Conditional Encoder

We propose two encoders to extract information from LR audio for conditioning WSRGlow. One is the LR audio encoder which directly maps raw audio samples to a sequence of embedding vectors. The other is the STFT encoder which encodes LR audio from the frequency domain by encoding its magnitude and phase spectrogram. The size of the group for squeezing LR encoding is chosen to be the same as the time window of STFT, so that the encodings of the two encoders can be concatenated to encode LR audio from both time and frequency domains. The architecture of the two encoders is shown in Figure 2.

2.2.1. LR Audio Encoder

The LR audio encoder Enc_{LR} maps every sample in the low-resolution signal to an embedding space. For a sample x_t, we first apply the μ-law transformation as in WaveNet [22], and then quantize the transformed sample to 256 possible values. We create an embedding vector for each of the 256 possible values. The sequence of embedding vectors is then squeezed into a sequence of groups of vectors H_{LR} to match the dimensionality of T.
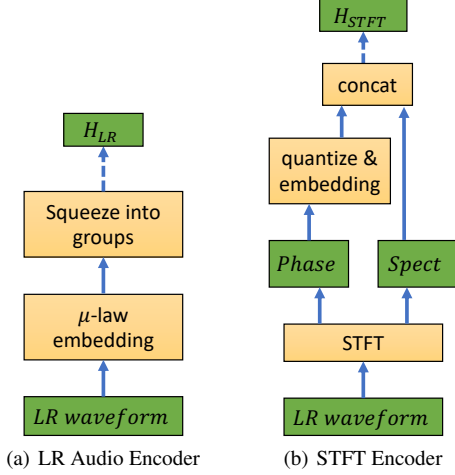
Figure 2: *Architecture of conditional encoders.*

2.2.2. STFT Encoder

The STFT encoder Enc_{STFT} encodes the frequency-domain information in LR audio, including magnitude and phase spectrograms. While magnitude spectrogram is a well-known representation, we also introduce phase spectrogram since it has been proven to be of much help in various speech tasks [26] including speech enhancement, speech recognition and speech synthesis. Besides, it is also shown that the capacity of human perception due to phase is higher than expected [27].

The STFT encoder first applies the short-time Fourier transform (STFT) to the signal. The phases ϕ of each time window is quantized and then embedded to E_ϕ:

$$E_\phi(\boldsymbol{x}) = \text{Embedding}(\text{Quantize}(\text{imag}(\text{STFT}(\boldsymbol{x})))),$$

while the magnitude spectrogram S is directly taken: $S(\boldsymbol{x}) = \text{real}(\text{STFT}(\boldsymbol{x}))$. The two are then concatenated along the channel axis to obtain the STFT encoding H_{STFT}: $H_{STFT} = concat(S(\boldsymbol{x}), E_\phi(\boldsymbol{x}))$.

3. Experiments

3.1. Experimental Settings

3.1.1. Dataset

We evaluate our model on the VCTK corpus [25], which contains 44 hours of data from 108 different speakers. We generate low-resolution audio signal using the *resample* function with default parameters in Librosa [28]. The target sampling rate of HR audio is 48kHz. We randomly split the dataset into training set and test set at utterance level so that the training set also contains the speakers in the test set. The audio file names used in test set are listed in the supplementary materials.

3.1.2. Model Configurations

Four methods for speech super-resolution are implemented for comparison, including two baselines and two proposed WSR-Glow models with different LR encodings.

- AudioUNet: A deep convolutional model for audio super-resolution proposed in [12]. We instantiated the model with 8 down-sampling blocks and 8 up-sampling blocks. We used

the same parameters for each block as [12] and our PyTorch version of AudioUNet is a reimplementation of the Tensor-Flow code provided by the authors[2].

- MUGAN: A GAN-based model for audio super-resolution proposed in [15, 14]. We use a similar structure in AudioUNet for the generator and the discriminator. We use MSE Loss for the generator and cross-entropy loss for the discriminator.

- WSRGlow w/o Enc_{STFT}: WSRGlow with only the encoding of LR audio encoder as the condition. We set the embedding dimension to be 256 and the number of glow layers to be 12. Audio encodings are squeezed into groups of 8. The transformation T in the affine coupling layer has 8 layers of dilated convolutions with 256 channels.

- WSRGlow: WSRGlow with the concatenated encoding of both LR audio encoder and STFT encoder as the condition. WSRGlow follows the setting of WSRGlow w/o Enc_{STFT}. We set the embedding size of the phase spectrogram to be 50. The frame size for STFT is set to 8 so that the time dimensionality of phase embeddings and waveform embeddings can be the same.

3.1.3. Training and Evaluation

For all models, we train them on batches of 12 audio waveforms with the maximum number of HR samples being 8192. We train each model for 100k iterations. For AudioUNet, WSRGlow w/o Enc_{STFT} and WSRGlow, we use the ADAM [29] optimizer with learning rate being 1e-4, $\beta_1 = 0.9$, $\beta_2 = 0.98$. For MUGAN, We set the coefficient for adversarial loss to be 0.001 and start adding the adversarial term to the loss for generator after 40k steps.

Following previous works [12, 14] on speech super-resolution, we evaluate our experiment results using both objective metrics and perceptive surveys.

We use the Signal to Noise Ratio (SNR) and Log-spectral distance (LSD) [30] to measure the quality of generated audio samples by comparing them to the actual high-resolution audio. The SNR is defined as $\text{SNR}(x, y) = 10 \log \frac{||y||_2^2}{||x-y||_2^2}$, where x is the reference signal and y is an approximation. The LSD measures the reconstruction quality of individual frequencies, which is defined as $\text{LSD}(x, y) = \frac{1}{L} \sum_{l=1}^{L} \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left(X(l, k) - \hat{X}(l, k) \right)^2}$, where X and $\hat{X}$ are the log-spectral power magnitudes of y and x. They are defined as $X = \log |S|^2$ where S is the short-time Fourier transform of the signal. l is the number of frames and k is the number of frequencies. We use the evaluation setting in [12] where the length of frames is 2048.

Following MUGAN [14], we perform a randomized, single-blinded A/B test with 10 participants to evaluate the performance of WSRGlow with real listeners. The test presents 10 pairs of audio clips generated by WSRGlow with STFT encoding and the best baseline model MUGAN, and asks participants to select a preferred clip, or "Neutral".

3.2. Results and Analyses

3.2.1. Subjective Evaluation

We conduct the preference test to compare the perceptual quality of the audio generated by WSRGlow and one of the base-
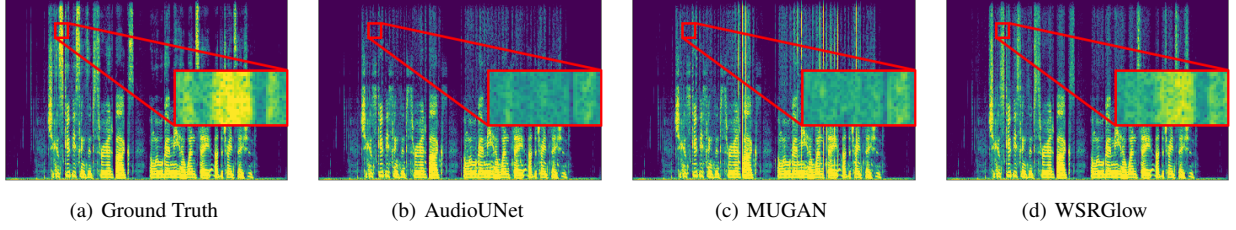
---

[2]https://github.com/kuleshov/audio-super-res

Figure 3: *Spectrogram visualizations of p225_005.wav.*

(a) Ground Truth   (b) AudioUNet   (c) MUGAN   (d) WSRGlow

line methods including AudioUNet and MUGAN. The results are shown in Table 1. From the table, we can see that listeners prefer more audio samples generated by WSRGlow over the baseline methods AudioUNet and MUGAN. We observe that the audio generated by MUGAN contains artifacts that do not sound like audio voiced by human beings. While this kind of artifacts does not exist in audio generated by WSRGlow, indicating that our method can reconstruct the natural and high-quality high-frequency audio.

We also visualize the spectrograms of ground truth audio and synthesized 4x super-resolution samples of AudioUNet, MUGAN and WSRGlow in Figure 3. We observe that the reconstructed part of WSRGlow is much closer to the ground truth than the baseline models, which is consistent with the preference test.

Table 1: *The preference scores of baselines and WSRGlow.*

Baseline	Baseline wins	Neutral	WSRGlow wins
AudioUNet	17%	16%	67%
MUGAN	18%	20%	62%

3.2.2. Objective Evaluation

We also evaluate the objective metrics including SNR and LSD for WSRGlow and baseline methods. The results are shown in Table 2. From row 1, 2 and 4, we can see that WSRGlow outperforms the baselines and WSRGlow w/o Enc_{STFT} significantly, which further demonstrates the superiority of our method. Besides, we remove the STFT encoder Enc_{STFT} in WSRGlow and compare it with the original WSRGlow. The results are shown in row 3 (WSRGlow w/o Enc_{STFT}). We can see that WSRGlow w/o Enc_{STFT} outperforms the baseline models in terms of SNR, but performs poorly in LSD. This indicates that WSRGlow w/o Enc_{STFT} cannot synthesize the high-frequency part of the signal well. Because SNR is calculated using sample-wise distance, while LSD is calculated using the spectrogram. A low LSD score also suggests unsatisfactory perceptual quality since the LSD metric has better correlation with perceptual quality compared to SNR [12, 30]. Besides, we observe that WSRGlow performs much better in LSD score than WSRGlow w/o Enc_{STFT}. This indicates that encoding the LR audio from frequency domain is of great help in synthesizing the high-frequency part of the audio.

Table 2: *Objective comparison with other methods.*

	2x SR		4x SR	
	SNR↑	LSD↓	SNR↑	LSD↓
AudioUNet	22.68	1.01	17.15	2.24
MUGAN	24.81	0.90	16.87	2.12
WSRGlow w/o Enc_{STFT}	25.91	1.41	17.45	2.76
WSRGlow	**30.84**	**0.68**	**18.38**	**1.64**

3.2.3. Ablation Study

To verify the effectiveness of our proposed components and analyze our methods, we conduct ablation study in three aspects:

Convergence speed: We train WSRGlow on 4x super-resolution task for only 50k iterations (which is half of the iterations run for the baselines) to show that WSRGlow is easy to train. After 50k iterations, WSRGlow achieves $SNR = 17.53, LSD = 1.80$ on the test set, which is already better than the baseline models. This proves that WSRGlow converges at high speed.

Necessity of phase and magnitude inputs: The comparison between WSRGlow and WSRGlow w/o Enc_{STFT} in Table 2 already shows that STFT information (phase and magnitude) is crucial in WSRGlow. To examine whether one of the two can be omitted, we train two partial WSRGlow models for 60k iterations. We observe that WSRGlow without phase embeddings results in $SNR = 12.91, LSD = 2.35$ while WSRGlow without magnitude embeddings results in $SNR = 5.68, LSD = 2.32$. Both results are poor compared to even the baseline models. This proves that both Enc_{LR} and Enc_{STFT} are effective and necessary.

Temperature in sampling: We sample the high-frequency audio results using the same WSRGlow model with different temperatures (variance of Gaussian) $T = 0.5, 0.666, 0.8, 1.0, 1.2, 1.5$ to explore the effect on audio quality. We observe that as the temperature becomes lower, SNR score tends to increase while LSD score tends to decrease. Perceptually, we find that the speech quality is the best when $T = 1.0$.[3]

4. Conclusion

In this paper, we proposed WSRGlow, a glow-based generative model for audio super-resolution: 1) we use normalizing flow to address the one-to-many problem by modeling the conditional distribution of HR audio; and 2) we proposed an LR encoder and an STFT encoder to exploit the information in LR audio from both time domain and frequency domain, significantly improving model's ability to synthesize the high-frequency part. Our experiments demonstrated that WSRGlow outperforms previous models in both objective and subjective metrics and converges quickly. The components were verified to be necessary and effective through ablation study.

In the future, we may develop our model to address more generalized audio super-resolution tasks, such as speech in heterogeneous environments and music recordings. We may also model the conditional distribution of HR spectrogram instead of HR waveform.

---

[3]Due to the limitation of space, we put audio samples and objective evaluation for different temperatures in the supplementary materials.

5. References

[1] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," *arXiv preprint arXiv:1905.09263*, 2019.

[2] Y. Ren, C. Hu, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text-to-speech," *arXiv preprint arXiv:2006.04558*, 2020.

[3] Y. Ren, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Almost unsupervised text to speech and automatic speech recognition," in *ICML*. PMLR, 2019, pp. 5410–5419.

[4] J. Xu, X. Tan, Y. Ren, T. Qin, J. Li, S. Zhao, and T.-Y. Liu, "Lrspeech: Extremely low-resource speech synthesis and recognition," in *26th SIGKDD*, 2020, pp. 2802–2812.

[5] M. Chen, X. Tan, Y. Ren, J. Xu, H. Sun, S. Zhao, T. Qin, and T.-Y. Liu, "Multispeech: Multi-speaker text to speech with transformer," *arXiv preprint arXiv:2006.04664*, 2020.

[6] Y. Ren, X. Tan, T. Qin, J. Luan, Z. Zhao, and T.-Y. Liu, "Deepsinger: Singing voice synthesis with data mined from the web," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 1979–1989.

[7] J. Liu, C. Li, Y. Ren, F. Chen, P. Liu, and Z. Zhao, "Diffsinger: Diffusion acoustic model for singing voice synthesis," *arXiv preprint arXiv:2105.02446*, 2021.

[8] Yan Ming Cheng, D. O'Shaughnessy, and P. Mermelstein, "Statistical recovery of wideband speech from narrowband speech," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 4, pp. 544–548, 1994.

[9] H. Pulakka, U. Remes, K. Palomäki, M. Kurimo, and P. Alku, "Speech bandwidth extension using gaussian mixture model-based estimation of the highband mel spectrum," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, Jul. pp. 5100–5103.

[10] J. Bradbury, "Linear predictive coding," *Mc G. Hill*, 2000.

[11] K. Li, Z. Huang, Y. Xu, and C. Lee, "Dnn-based speech bandwidth expansion and its application to adding high-frequency missing features for automatic speech recognition of narrowband speech," in *INTERSPEECH*, 2015.

[12] V. Kuleshov, S. Z. Enam, and S. Ermon, "Audio super resolution using neural networks," *CoRR*, vol. abs/1708.00853, 2017. [Online]. Available: http://arxiv.org/abs/1708.00853

[13] T.-Y. Lim, R. A. Yeh, Y. Xu, M. Do, and M. Hasegawa-Johnson, "Time-frequency networks for audio super-resolution," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 646–650, 2018.

[14] S. Kim and V. Sathe, "Adversarial audio super-resolution with unsupervised feature losses," 2018.

[15] S. E. Eskimez and K. Koishida, "Speech super resolution generative adversarial network," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3717–3721.

[16] S. Hu, B. Zhang, B. Liang, E. Zhao, and S. Lui, "Phase-Aware Music Super-Resolution Using Generative Adversarial Networks," in *Proc. Interspeech 2020*, 2020, pp. 4074–4078. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-2605

[17] D. P. Kingma and P. Dhariwal, "Glow: generative flow with invertible 1×1 convolutions," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 10 236–10 245.

[18] A. Pumarola, S. Popov, F. Moreno-Noguer, and V. Ferrari, "C-flow: Conditional generative flow models for images and 3d point clouds," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 7946–7955.

[19] R. Kondo, K. Kawano, S. Koide, and T. Kutsuna, "Flow-based image-to-image translation with feature disentanglement," in *NeurIPS*, 2019.

[20] A. Lugmayr, M. Danelljan, L. Van Gool, and R. Timofte, "Srflow: Learning the super-resolution space with normalizing flow," in *European Conference on Computer Vision*. Springer, 2020, pp. 715–732.

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

[22] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," 2016.

[23] ITU-T, "Pulse code modulation (pcm) of voice frequencies," International Telecommunication Union, Recommendation G.711, 1988.

[24] M. Wang, Z. Wu, S. Kang, X. Wu, J. Jia, D. Su, D. Yu, and H. Meng, "Speech super-resolution using parallel wavenet," in *2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, 2018, pp. 260–264.

[25] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2016.

[26] K. Paliwal, K. Wójcicki, and B. J. Shannon, "The importance of phase in speech enhancement," *Speech Commun.*, vol. 53, pp. 465–494, 2011.

[27] P. M. B. Mahale, R. Saeidi, and Y. Stylianou, "Phase importance in speech processing applications," in *INTERSPEECH*, 2014.

[28] B. McFee, V. Lostanlen, A. Metsai, M. McVicar, S. Balke, C. Thomé, C. Raffel, F. Zalkow, A. Malek, Dana, K. Lee, O. Nieto, J. Mason, D. Ellis, E. Battenberg, S. Seyfarth, R. Yamamoto, K. Choi, viktorandreevichmorozov, J. Moore, R. Bittner, S. Hidaka, Z. Wei, nullmightybofo, D. Hereñú, F.-R. Stöter, P. Friesch, A. Weiss, M. Vollrath, and T. Kim, "librosa/librosa: 0.8.0," Jul. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3955228

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR (Poster)*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[30] A. Gray and J. Markel, "Distance measures for speech processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 5, pp. 380–391, 1976.