INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Complex Image Generation SwinTransformer Network for Audio Denoising

Youshan Zhang[1], *Jialu Li*[2]

[1]Yeshiva University, NYC, NY, USA
[2]Cornell University, Ithaca, NY, USA

youshan.zhang@yu.edu, jl4284@cornell.edu

Abstract

Achieving high-performance audio denoising is still a challenging task in real-world applications. Existing time-frequency methods often ignore the quality of generated frequency domain images. This paper converts the audio denoising problem into an image generation task. We first develop a complex image generation SwinTransformer network to capture more information from the complex Fourier domain. We then impose structure similarity and detailed loss functions to generate high-quality images and develop an SDR loss to minimize the difference between denoised and clean audios. Extensive experiments on two benchmark datasets demonstrate that our proposed model is better than state-of-the-art methods.

Index Terms: audio denoising, image generation, complex SwinTransformer

1. Introduction

Audio denoising aims to remove the background noise in the audio to generate better-quality information sources for real-life applications, such as speech enhancement [1], hearing aids [2] and the lung [3] and heart [4] sounds for disease diagnosis. However, due to the degraded quality, unpleasant reverb, and loud background sound, pursuing high-quality denoised audio is still challenging.

Deep learning methods have become prevalent in the audio denoising field, demonstrating a stronger ability to learn data features [5]. In recent years, many time-frequency (T-F) domain deep-learning-based audio denoising approaches [6] have been implemented using short-time Fourier transform (STFT) and applying inverse short-time Fourier transform (ISTFT) to denoise audios [7]. Sonning et al. [8] investigated the performance of a time-domain network for speech denoising. The model was developed to deal with the original inability of STFT/ISTFT-based time-frequency approaches to capture short-time changes and was proved to be useful in a real-time setting. Wang et al. [7] proposed a two-stage transformer neural network for end-to-end audio denoising in the time domain. Their model included an encoder, a two-stage transformer module, a masking module, and a decoder, which outperformed many time- or frequency-domain models with less complex structures.

One problem in DNN-based audio denoising approaches is that they predict a label for each time frame from a small context window around the frame [9]; therefore, it is difficult for models to track a target speaker among multiple interferences, which means that the DNNs are not easy to handle long-term contexts [10]. To cope with this problem, more deep learning approaches are proposed to better capture the audio features, e.g., recurrent neural networks (RNNs). Chen and Wang [11] proposed an RNN-based audio separation model with four hidden long short-term memory (LSTM) layers to deal with speaker generalization. Their model outperformed DNN-based models on unseen speakers and unseen noises regarding objective speech intelligibility. Maas et al. [12] introduced a model using a deep recurrent auto-encoder neural network to denoise input features and capture the temporal nature of speech signals for robust automatic speech recognition [13].

To produce better noise audio processing results, Zhang et al. [14] built a novel deep recurrent convolutional network for acoustic modeling and then applied deep residual learning for audio recognition with faster convergence speed. Tan et al. [9] proposed a recurrent convolutional network that incorporates a convolutional encoder-decoder and long short-term memory into the convolutional recurrent neural network (CRN) architecture to address real-time audio enhancement. Li et al. [10] combined the progressive learning framework with a causal CRN to further mitigate the trainable parameters and improve audio quality and intelligibility. Zhang and Li [15] converted audio denoising into a visual image segmentation problem, and their results demonstrated that a better segmentation result leads to better audio denoising performance.

Audio denoising using waveform domain and transformer has also been explored. Kong et al. [1] proposed an audio enhancement method with pre-trained audio neural networks using weakly labeled data and applied a convolutional U-Net to predict the waveform of individual anchor segments selected by PANNs. Kong et al. [6] proposed CleanUNet, a causal speech denoising model on the raw waveform based on an encoder-decoder architecture combined with several self-attention blocks. Agarwal et al. [16] replaced the Bi-directional LSTM block with a transformer in the open-source Open-Unmix model for audio separation, and the new model trained faster than the unmodified model. However, these transformer methods only focused on denoising audios and usually did not check the quality of generated intermediate matrices.

To alleviate the aforementioned challenges, our contributions are three-fold:

- We convert the audio denoising into an image generation problem. Our experiment demonstrates that a better-generated complex image will achieve better audio denoising performance.
- We develop a complex image generation SwinTransformer network (CIGSN) model, which is able to generate high-quality complex images in the Fourier domain.
- We also propose image quality check and audio reconstruction modules. We enforce image L1 loss, structural similarity loss, and detailed loss for the image quality check, and employee audio L1 loss and SDR loss to optimize the audio reconstruction module.

10.21437/Interspeech.2023-980

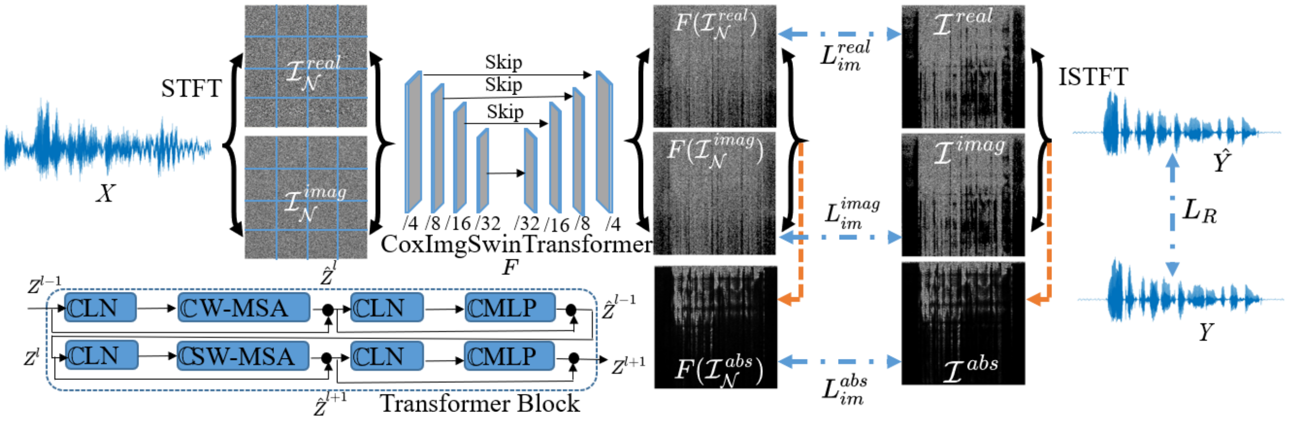

Figure 1: *The schematic diagram of our CIGSN model. Each gray block is a transformer block in the CoxImgSwinTransformer module (F). We first apply STFT to convert audio signals X into complex images (real image $\mathcal{I}_{\mathcal{N}}^{real}$ and imaginary image $\mathcal{I}_{\mathcal{N}}^{imag}$). Then, we feed them into the module (F), and get generated real $F(\mathcal{I}_{\mathcal{N}}^{real})$, imaginary $F(\mathcal{I}_{\mathcal{N}}^{imag})$, and absolute $F(\mathcal{I}_{\mathcal{N}}^{abs})$ images. Finally, we minimize the image quality check loss $L_{im}^{total} = L_{im}^{imag} + L_{im}^{real} + L_{im}^{abs}$, and audio reconstruction loss L_R.*

2. Methodology

In time domain audio denoising, a noisy audio signal x can be typically expressed as:

$$x = y + noise, \qquad (1)$$

where y and $noise$ denote clean audio and additive noise signal, respectively. Given noisy audio signals $X = \{x_i\}_{i=1}^{n}$, we aim to extract the clean audios $Y = \{y_i\}_{i=1}^{n}$ by learning a mapping f, and leverage $f(X) \approx Y$. In the Fourier frequency domain, we convert the audio denoising to an image generation task. Given the noisy audio complex images $\mathcal{I}_{\mathcal{N}} = \{I_{Ni}\}_{i=1}^{n}$ using STFT(X) and clean audio complex images $\mathcal{I} = \{I_i\}_{i=1}^{n}$ using STFT(Y), we also aim to find a function F such that $F(\mathcal{I}_{\mathcal{N}}) \approx \mathcal{I}$, where $F(\mathcal{I}_{\mathcal{N}})$ is the generated complex images.

2.1. Motivation

The existing time-frequency audio denoising methods majorly convert the audio signal to the Fourier domain using STFT and get the reconstructed matrix, and then apply the ISTFT to get the denoised audio. However, the intermediate process of the reconstructed matrix is usually less explored. We aim to pursue a high-quality generated matrix (complex images) and convert it to an image generation problem in the Fourier domain.

2.2. CoxImgSwinTransformer

To generate high-quality real and imaginary images, we develop a CoxImgSwinTransformer model. The details of SwinTransformer can be found in [17]. However, the original SwinTransformer model cannot handle complex image inputs. We hence develop a complex image inputs variant of the SwinTransformer model (CoxImgSwinTransformer). Given an input batch of tensor $T = [N, C, H, W]$, where N is the number of samples in the batch; C is the channel size ($C = 1$ if the audio is a single track, and $C = 2$ if the audio is dual tracks); H and W are the height and width of the image (note that the tensor $T = A + jB \in \mathbb{C}^{N\times C\times H\times W}$ is a complex number, A is the real part, and B is the imaginary part of the complex tensor), we define the basic deep learning operations O as:

$$\mathbb{C}O(T) = O(A) + jO(B), \qquad (2)$$

where j is the square root of -1 and O can be common deep learning layers (Conv2d, MaxPool2d, BatchNorm2d, ReLU, GeLU, Dropout, Interpolate, Sigmoid, LayerNorm, Softmax, Linear, etc.). By applying Eq. (2), we can get the complex version of these layers as (ℂConv2d, ℂMaxPool2d, ℂBatchNorm2d, ℂReLU, ℂGeLU, ℂDropout, ℂInterpolate, ℂSigmoid, ℂLayerNorm, ℂSoftmax, ℂLinear, etc.). With the basis of these layers, we can build the CoxImgSwinTransformer model. Fig. 1 shows the overall architecture of our proposed CoxImgSwinTransformer module, which has three key parts: encoder, decoder, and skip connections.

2.2.1. *Encoder*

The encoder consists of four Swin transformer blocks. Each Swin transformer block is composed of a complex attention layer and a complex feed-forward layer, including a complex LayerNorm (ℂLN) layer, complex multi-head self-attention module, a two-fully connected layers complex MLP (ℂMLP), and a ℂGELU nonlinearity layer. The ℂLN and ℂGELU are computed based on Eq. (2).

The ℂMLP module has five complex layers: ℂLinear, ℂGELU, ℂDropout, ℂLinear and ℂDropout. Between two successive transformer blocks, there is a complex window-based multi-head self-attention (ℂW-MSA) module, and a complex shifted window-based multi-head self-attention (ℂSW-MSA) module. The continuous swin transformer process is represented as:

$$\begin{aligned}
\hat{Z}^{l} &= \mathbb{C}W\text{-}MSA(\mathbb{C}LN(Z^{l-1})) + Z^{l-1} \\
Z^{l} &= \mathbb{C}MLP(\mathbb{C}LN(\hat{Z}^{l})) + \hat{Z}^{l} \\
\hat{Z}^{l+1} &= \mathbb{C}SW\text{-}MSA(\mathbb{C}LN(Z^{l})) + Z^{l} \\
Z^{l+1} &= \mathbb{C}MLP(\mathbb{C}LN(\hat{Z}^{l+1})) + \hat{Z}^{l+1},
\end{aligned} \qquad (3)$$

where $\hat{Z}^{l}$ and Z^{l} represent the outputs of the $\mathbb{C}(S)$W-MSA module and the ℂMLP module of the l^{th} block, respectively. The complex self-attention is computed according to:

$$\mathbb{C}Attention(\mathbb{C}Q, \mathbb{C}K, \mathbb{C}V) = \mathbb{C}SoftMax(\frac{\mathbb{C}Q\mathbb{C}K^{\mathbb{C}T}}{\sqrt{d}} + B)\mathbb{C}V, \qquad (4)$$

where $\mathbb{C}Q, \mathbb{C}K, \mathbb{C}V \in \mathbb{C}^{M^2\times d}$ are the query, key and value matrices; d is the query/key dimension, M^2 is the number of patches in a window and B is taken from bias matrix $\hat{B} \in \mathbb{C}^{(2M-1)\times(2M+1)}$.

In the encoder, the dimensions of features in four transformer blocks are $\frac{H}{4} \times \frac{W}{4} \times C$, $\frac{H}{8} \times \frac{W}{8} \times 2C$, $\frac{H}{16} \times \frac{W}{16} \times 4C$,

$\frac{H}{32} \times \frac{W}{32} \times 8C$, which corresponds to $/4, /8, /16$, and $/32$ of Fig. 1, respectively. For the patch merging layer, we concatenate the input features of each group of 2×2 neighboring patches and use the $\mathbb{C}$linear layers to obtain the specified channel number of output features.

2.2.2. *Decoder*

In the decoder, we also have four symmetric transformer blocks. However, we use the patch expanding layer in the decoder to upsample the extracted deep features. The output dimension of the four blocks are: $\frac{H}{32} \times \frac{W}{32} \times 8C$, $\frac{H}{16} \times \frac{W}{16} \times 4C$, $\frac{H}{8} \times \frac{W}{8} \times 2C$, and $\frac{H}{4} \times \frac{W}{4} \times C$, which corresponds to $/32, /16, /8$, and $/4$ of Fig. 1, respectively.

2.2.3. *Skip connection*

We applied three skip connections to fuse the multi-scale features from the encoder with the decoder. We concatenate the shallow features from the encoder and deep features from the decoder to reduce spatial information loss and form robust features. The final output dimensions of height and width are the same as the input images.

2.3. Image quality check

Given complex input images $\mathcal{I_N}$, our CoxImgSwinTransformer model can output the generated complex images $F(\mathcal{I_N})$. To improve generated image quality, we developed an image quality check module. We first apply L1 loss to minimize the difference between the generated images $F(\mathcal{I_N})$ and the ground truth images $\mathcal{I}$, $L_F = |F(\mathcal{I_N}) - \mathcal{I}|_1$, where F is our proposed CoxImgSwinTransformer, $\mathcal{I_N}$ is noise complex images. Secondly, to further improve the generated images, we impose a structural similarity loss L_S to examine the generated image quality. The L_S is defined as: $1 - SSIM(F(\mathcal{I_N}), \mathcal{I})$, where $SSIM$ is the structural similarity [18]. The range of the L_S is from 0 to 1, where 0 indicates high similarity between images and 1 means they are not similar. Finally, to enhance image details, we add a detailed loss, called L_D and it is given by:

$$L_D = |VGG(F(\mathcal{I_N})) - VGG(\mathcal{I})|_1, \tag{5}$$

where VGG is the activation of the last fully connected layer from the pre-trained VGG19 network. Therefore, we improve the quality of the generated images by minimizing Eq. (6).

$$L_{im} = L_F + L_S + L_D \tag{6}$$

As shown in Fig. 1, we could get three different images given one audio: the real image, the imaginary image, and the absolute image (abs takes the absolute value of a complex tensor from the output of CoxImgSwinTransformer). Eventually, the image quality check module loss consists of these three image minimizations and is defined as:

$$L_{im}^{total} = L_{im}^{imag} + L_{im}^{real} + L_{im}^{abs}, \tag{7}$$

where $L_{im}^{imag} = L_F^{imag} + L_S^{imag} + L_D^{imag}$, and similarly, we could change $imag$ of L_{im}^{imag} into $real$, and abs to get L_{im}^{real} and L_{im}^{abs}, respectively.

2.4. Audio reconstruction

After getting the output from the decoder layers from the CoxImgSwinTransformer model, we could apply ISTFT to get the reconstructed audio as $\hat{Y}$. We first apply L1 loss to minimize the difference between reconstructed audio and the ground truth as $L_A = |\hat{Y} - Y|_1$. We also propose an SDR loss to evaluate the quality of $\hat{Y}$. The SDR is defined as: $SDR(\hat{Y}, Y) = 10 \log_{10} \frac{||Y||^2}{||\hat{Y}-Y||^2}$. We defined the SDR loss as:

$$L_{SDR} = const_{upper} - SDR(\hat{Y}, Y), \tag{8}$$

where $const_{upper}$ is the upper bound constant value, we set it as 20. Therefore, we could ensure that the SDR loss keeps decreasing during the training. The audio reconstruction loss is defined as:

$$L_R = L_A + L_{SDR}. \tag{9}$$

2.5. Objective Function

The architecture of our proposed CoxImgSwinTransformer model is shown in Fig. 1. Considering all loss functions in Sec. 2.3 and Sec. 2.4, our model minimizes the following objective function

$$L = \alpha L_{im}^{total} + (1 - \alpha) L_R, \tag{10}$$

where α is the balance factor between all image loss and audio reconstruction loss. This objective function enables us first to get high-quality generated images and then acquire a better reconstructed denoised audio. Our training procedures are described in Alg.1.

Algorithm 1 Complex Image Generation SwinTransformer Network (CIGSN). $B(\cdot)$ denotes the mini-batch training sets, L is the number of iterations.

```
Input: Noisy audio signals $X = \{x_i\}_{i=1}^n$ and clean audio
    signals $Y = \{y_i\}_{i=1}^n$, where $n$ is the number of audios.
Output: Denoised audio signals
Generate noise audio images $\mathcal{I_N} = \{I_{N i}\}_{i=1}^n$ and clean
    audio images $\mathcal{I} = \{I_i\}_{i=1}^n$ using STFT
for $iter = 1$ to $L$ do
    Derive $B(\mathcal{I_N})$ and $B(\mathcal{I})$ sampled from $\mathcal{I_N}$ and $\mathcal{I}$
    Calculate image quality check loss using Eq. (7)
    Convert complex images to audio signals using ISTFT
        and calculate audio reconstruction loss using Eq. (9)
    Optimize CIGSN model $F$ using Eq. (10)
end for
Output $F(\mathcal{I_N})$, and get denoised audios using ISTFT
```

3. Experiments

3.1. Datasets

We evaluate our model using two benchmark datasets.

VoiceBank-DEMAND [19] is a synthetic dataset created by mixing up clean speech and noise. The training set contains 11,572 utterances (9.4h), and the test set contains 824 utterances (0.6h). The lengths of utterances range from 1.1s to 15.1s, with an average of 2.9s.

BirdSoundsDenoising [15] contains 14,120 audios and is a large-scale dataset of bird sounds collected containing 10000/1400/2720 in training, validation, and testing, respectively. Unlike many audio-denoising datasets, which have manually added artificial noise, these datasets contain many natural noises, including wind, waterfall, rain, etc.

3.2. Implementation details

During the training, we set batch size = 16, training iteration $L = 100$, and learning rate = 0.001, $\alpha = 0.5$ with an Adam optimizer on a 48G RTX A6000 GPU using PyTorch. We applied the STFT to convert audio signals to audio images and utilized 1000-point Hamming as the window function, the size of

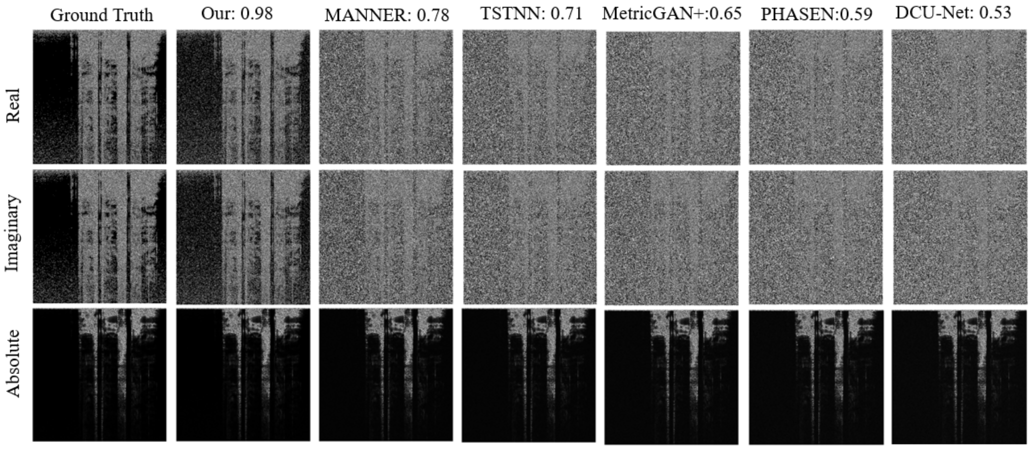

Figure 2: *Comparison of generated images of our CIGSN and the best five baseline methods of real, imaginary, and absolute images.*

Table 1: *Comparison results on the VoiceBank-DEMAND dataset. "$-$" means not applicable.*

Methods	Domain	PESQ	STOI	CSIG	CBAK	COVL	SSIM
CP-GAN [20]	T	2.64	0.942	3.93	3.33	3.28	0.58
PGGAN [21]	T	2.81	0.944	3.99	3.59	3.36	0.56
DCCRGAN [22]	TF	2.82	0.949	4.01	3.48	3.40	0.65
S-DCCRN [23]	TF	2.84	0.940	4.03	2.97	3.43	0.62
DCU-Net [24]	TF	2.93	0.930	4.10	3.77	3.52	0.67
PHASEN [25]	TF	2.99	$-$	4.18	3.45	3.50	0.72
MetricGAN+ [26]	TF	3.15	0.927	4.14	3.12	3.52	0.78
TSTNN [7]	T	2.96	0.950	4.33	3.53	3.67	0.78
MANNER [27]	T	3.21	0.950	4.53	3.65	3.91	0.81
CIGSN	TF	**3.41**	**0.954**	**4.78**	**3.82**	**4.22**	**0.88**

Fourier transform $n\_fft = 1023$. The length of each audio can be different, and we set the distance between neighboring sliding window frames as $hop\_length = int(length(y_t)/512)$, where $length(y_t)$ is the length of each audio. We then resize the input image dimensions as $[512 \times 512 \times 1]$[1].

3.3. Results

Tab. 1 shows the comparison results of the VoiceBank-DEMAND dataset. For nine baseline models, we also get the generated images following the same STFT parameters in Sec. 3.2. We reported the extra structure similarity (SSIM) between generated image and the ground truth (mean SSIM of real, imaginary, and absolute images). Our proposed CIGSN model achieves the highest performance in all six metrics. Particularly, the SSIM metric is much higher than all other methods. To explore the underlying reason, we compare generated complex images with the five best baselines as shown in Fig. 2. We also list their mean SSIM score. The generated real and imaginary images of the CIGSN model are close to ground truth, while the other five models contain many noise areas. Surprisingly, the absolute images of the other five methods are similar to ground truth, which is caused by absolute images taking the absolute values of real and imaginary images so that there are no visible negative values. We find that a higher SSIM score (better-generated images) achieves better audio denoising performance. Tab. 2 presents the results of the BirdSoundsDenoising dataset. Results of F1, IoU, and Dice are omitted since these metrics are used for the audio image segmentation task [15]. Our CIGSN still outperforms all other models in terms of SDR.

Table 2: *Comparison results of the BirdSoundsDenoising dataset ($F1, IoU$, and $Dice$ scores are multiplied by 100.*

Networks	Validation				Test			
	$F1$	IoU	$Dice$	SDR	$F1$	IoU	$Dice$	SDR
MTU-NeT [28]	69.1	56.5	69.0	8.17	68.3	55.7	68.3	7.96
Segmenter [29]	72.6	59.6	72.5	9.24	70.8	57.7	70.7	8.52
U-Net [30]	75.7	64.3	75.7	9.44	74.4	62.9	74.4	8.92
SegNet [31]	77.5	66.9	77.5	9.55	76.1	65.3	76.2	9.43
DVAD [32]	82.6	73.5	82.6	10.33	81.6	72.3	81.6	9.96
R-CED [33]	$-$	$-$	$-$	2.38	$-$	$-$	$-$	1.93
Noise2Noise [34]	$-$	$-$	$-$	2.40	$-$	$-$	$-$	1.96
TS-U-Net [35]	$-$	$-$	$-$	2.48	$-$	$-$	$-$	1.98
CIGSN	$-$	$-$	$-$	**10.69**	$-$	$-$	$-$	**10.15**

Table 3: *Ablation study of different modules*

Methods	U+I	U+A	U+I+A	C+I	C+A	C+I+A
SDR	8.54	7.97	8.98	10.0	9.84	10.69

Ablation study To demonstrate the effectiveness of the proposed three modules: CoxImgSwinTransformer, image quality check, and audio reconstruction, we conduct an ablation study using the BirdSoundsDenoising validation dataset in Tab. 3. "U" means the complex U-Net model [24], "I" means image quality check module, "A" means audio reconstruction module and "C" means CoxImgSwinTransformer model. Our CoxImgSwinTransformer model is better than the complex U-Net model. In addition, the image quality check module is more important than the audio reconstruction module.

From the above experiments, we can conclude that our proposed CIGSN model is effective in audio-denoising tasks. There are two compelling reasons. Firstly, our CoxImgSwinTransformer module can distill real and imaginary images, and we can directly visualize the generated complex images. Secondly, the proposed image check and audio reconstruction module is able to minimize the models' prediction and ground truth. One weakness of the model is that it requires high GPU memory to train the CoxImgSwinTransformer module.

4. Conclusions

In this paper, we convert the audio denoising into an image generation problem. We first develop a complex image generation SwinTransformer network to capture more information from the complex images. We then impose structure similarity loss to generate high-quality images and develop an SDR loss to minimize the difference between denoised audio and clean audio. Extensive experiments demonstrate our proposed model outperforms state-of-the-art models.

[1] Source code is available at `https://github.com/YoushanZhang/CoxImgSwinTransformer`.

5. References

[1] Q. Kong, H. Liu, X. Du, L. Chen, R. Xia, and Y. Wang, "Speech enhancement with weakly labelled data from audioset," *arXiv preprint arXiv:2102.09971*, 2021.

[2] M. Aubreville, K. Ehrensperger, A. Maier, T. Rosenkranz, B. Graf, and H. Puder, "Deep denoising for hearing aid applications," in *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2018, pp. 361–365.

[3] M. F. Pouyani, M. Vali, and M. A. Ghasemi, "Lung sound signal denoising using discrete wavelet transform and artificial neural network," *Biomedical Signal Processing and Control*, vol. 72, p. 103329, 2022.

[4] H. Kui, J. Pan, R. Zong, H. Yang, and W. Wang, "Heart sound classification based on log mel-frequency spectral coefficients features and convolutional neural networks," *Biomedical Signal Processing and Control*, vol. 69, p. 102893, 2021.

[5] L. Wang, W. Zheng, X. Ma, and S. Lin, "Denoising speech based on deep learning and wavelet decomposition," *Scientific Programming*, vol. 2021, 2021.

[6] Z. Kong, W. Ping, A. Dantrey, and B. Catanzaro, "Speech denoising in the waveform domain with self-attention," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7867–7871.

[7] K. Wang, B. He, and W.-P. Zhu, "Tstnn: Two-stage transformer based neural network for speech enhancement in the time domain," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 7098–7102.

[8] S. Sonning, C. Schüldt, H. Erdogan, and S. Wisdom, "Performance study of a convolutional time-domain audio separation network for real-time speech denoising," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 831–835.

[9] K. Tan and D. Wang, "A convolutional recurrent neural network for real-time speech enhancement." in *Interspeech*, vol. 2018, 2018, pp. 3229–3233.

[10] A. Li, M. Yuan, C. Zheng, and X. Li, "Speech enhancement using progressive learning-based convolutional recurrent neural network," *Applied Acoustics*, vol. 166, p. 107347, 2020.

[11] J. Chen and D. Wang, "Long short-term memory for speaker generalization in supervised speech separation," *The Journal of the Acoustical Society of America*, vol. 141, no. 6, pp. 4705–4714, 2017.

[12] A. Maas, Q. V. Le, T. M. O'neil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent neural networks for noise reduction in robust asr," 2012.

[13] H. Zhao, S. Zarar, I. Tashev, and C.-H. Lee, "Convolutional-recurrent neural networks for speech enhancement," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2401–2405.

[14] Z. Zhang, Z. Sun, J. Liu, J. Chen, Z. Huo, and X. Zhang, "Deep recurrent convolutional neural network: Improving performance for speech recognition," *arXiv preprint arXiv:1611.07174*, 2016.

[15] Y. Zhang and J. Li, "Birdsoundsdenoising: Deep visual audio denoising for bird sounds," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 2248–2257.

[16] A. Agarwal, B. Li, V. Menon, N. Peddinti, Y. Qian, D. Torres, and S. Dasgupta, "Implementing transformer architectures for audio source separation," in *2022 IEEE MIT Undergraduate Research Technology Conference (URTC)*. IEEE, 2022, pp. 1–5.

[17] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[18] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[19] C. Valentini-Botinhao *et al.*, "Noisy speech database for training speech enhancement algorithms and tts models," *University of Edinburgh. School of Informatics. Centre for Speech Technology Research (CSTR)*, 2017.

[20] G. Liu, K. Gong, X. Liang, and Z. Chen, "Cp-gan: Context pyramid generative adversarial network for speech enhancement," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6624–6628.

[21] Y. Li, M. Sun, and X. Zhang, "Perception-guided generative adversarial network for end-to-end speech enhancement," *Applied Soft Computing*, vol. 128, p. 109446, 2022.

[22] H. Huang, R. Wu, J. Huang, J. Lin, and J. Yin, "Dccrgan: Deep complex convolution recurrent generator adversarial network for speech enhancement," in *2022 International Symposium on Electrical, Electronics and Information Engineering (ISEEIE)*. IEEE, 2022, pp. 30–35.

[23] S. Lv, Y. Fu, M. Xing, J. Sun, L. Xie, J. Huang, Y. Wang, and T. Yu, "S-dccrn: Super wide band dccrn with learnable complex feature for speech enhancement," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7767–7771.

[24] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, "Phase-aware speech enhancement with deep complex u-net," in *International Conference on Learning Representations*, 2019.

[25] D. Yin, C. Luo, Z. Xiong, and W. Zeng, "Phasen: A phase-and-harmonics-aware speech enhancement network," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 9458–9465.

[26] S.-W. Fu, C. Yu, T.-A. Hsieh, P. Plantinga, M. Ravanelli, X. Lu, and Y. Tsao, "Metricgan+: An improved version of metricgan for speech enhancement," *arXiv preprint arXiv:2104.03538*, 2021.

[27] H. J. Park, B. H. Kang, W. Shin, J. S. Kim, and S. W. Han, "Manner: Multi-view attention network for noise erasure," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7842–7846.

[28] H. Wang, S. Xie, L. Lin, Y. Iwamoto, X.-H. Han, Y.-W. Chen, and R. Tong, "Mixed transformer u-net for medical image segmentation," in *ICASSP 2022*. IEEE, 2022, pp. 2390–2394.

[29] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7262–7272.

[30] 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.

[31] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[32] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.

[33] S. R. Park and J. W. Lee, "A fully convolutional neural network for speech enhancement," *Proc. Interspeech 2017*, pp. 1993–1997, 2017.

[34] M. M. Kashyap, A. Tambwekar, K. Manohara, and S. Natarajan, "Speech denoising without clean training data: A noise2noise approach," *Proc. Interspeech 2021*, pp. 2716–2720, 2021.

[35] E. Moliner and V. Välimäki, "A two-stage u-net for high-fidelity denoising of historical recordings," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 841–845.