Noise Modeling to Build Training Sets for Robust Speech Enhancement

Yahui Wang

Beijing University of Posts and Telecommunications

Wenxi Zhang

Chinese Academy of Sciences Aerospace Information Research Institute

Yongbiao Wang

Chinese Academy of Sciences Aerospace Information Research Institute

Xinxin Kong

Chinese Academy of Sciences Aerospace Information Research Institute

Hongxin Zhang (✉ hongxinzhang@bupt.edu.cn)

Beijing university of posts and telecommunications , Beijing, China

Research

Keywords: Noise modeling, Training sets, Generative Adversarial Network, Speech enhancement

Posted Date: November 17th, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-105360/v1>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at Applied Sciences on February 11th, 2022.
See the published version at <https://doi.org/10.3390/app12041905>.

Noise Modeling to Build Training Sets for Robust Speech Enhancement

Author: Yahui Wang^[1,2], Wenxi Zhang^[2], Yongbiao Wang^[2], Xinxin Kong^[2], Hongxin Zhang^[1] (corresponding author)

[1] affiliation: Beijing university of posts and telecommunications, Beijing, China.

[2] affiliation: Key Laboratory of Computational Optical Imaging Technology, Aerospace Information Research Institute, Chinese Academy of Sciences Beijing, China.

Yahui Wang Received the B.S. degrees in Binzhou University in 2013. Then, she received the M.S. degrees in Beijing Institute of Technology in 2016. From 2017 to present, she stays in School of Cyberspace Security, Beijing University of Posts and Telecommunications of China to study information security for her Ph.D. She now with Chinese Academy of Sciences, Beijing, China.

e-mail address: wangyahui@aoc.ac.cn
Address: No.9, Dengzhuang South Road, Haidian District, Beijing

Wenxi Zhang Is the professor and doctoral tutor of University of Chinese Academy of Sciences, China. He is also the senior researcher and the director of the Key Laboratory of Computational Optical Imaging Technology, Chinese Academy of Sciences, Beijing, China. He has long been engaged in signal processing and interferometric research.

e-mail address: zhangwenxi@aoc.ac.cn
Address: No.9, Dengzhuang South Road, Haidian District, Beijing

Yongbiao Wang Received the bachelor's degree from North China Electric Power University, China, in 2017 and the master's degree from University of Chinese Academy of Sciences, China, in 2020. Now, as an assistant research institute, he works at Key Laboratory of Computational Optical Imaging Technology, Chinese Academy of Sciences, Beijing, China. His research interests include speech signal processing, deep learning.

e-mail address: billwangyong@163.com
Address: No.9, Dengzhuang South Road, Haidian District, Beijing

Xinxin Kong Received the bachelor's degree from Tianjin University of Technology, China, in 2010 and the master's degree from University of Chinese Academy of Sciences, China, in 2013. Currently, he is an engineer at Key Laboratory of Computational Optical Imaging Technology, Chinese Academy of Sciences, Beijing, China. He researches in terests include optical interferometry, instrument design and simulation analysis.

e-mail address: xxkong@aoc.ac.cn
Address: No.9, Dengzhuang South Road, Haidian District, Beijing

Hongxin Zhang (corresponding author) Is the professor and doctoral tutor of Electrical Engineering Institute of Beijing university of posts and telecommunications. He is the director of the Broadband Communication and Electromagnetic Engineering, China. He is the review expert of the national natural science fund project. He is also the evaluation expert of Education degree and graduate education development center.

e-mail address: hongxinzhang@bupt.edu.cn
telephone: 86+13716702507

fax numbers: 82178622
Address: No.9, Dengzhuang South Road, Haidian District, Beijing

ABSTRACT

The performance of Deep Neural Network (DNN)-based speech enhancement models degrades significantly in real recordings because the synthetic training sets are mismatched with real test sets. To solve this problem, we propose a new Generative Adversarial Network framework for Noise Modeling (NM-GAN) that can build training sets by imitating real noise distribution. The framework combines a novel U-Net with two bidirectional Long Short-Term Memory (LSTM) layers that act as a generator to construct complex noise. The Gaussian distribution is adapted and used as conditional information to direct the noise generation. A discriminator then learns to determine whether a noise sample is from the model distribution or from a real noise distribution. By adversarial and alternate training, NM-GAN can generate enough recall (diversity) and precision (quality of noise) in its samples for it to look like real noise. Afterwards, realistic-looking paired training sets are composed. Extensive experiments were carried out and qualitative and quantitative evaluation of the generated noise samples and training sets demonstrate that potential of the framework. An Speech enhancement model trained on our synthetic training sets and on real training sets was found to be capable of good noise suppression for real speech-related noise.

KEYWORDS

Noise modeling; Training sets; Generative Adversarial Network; Speech enhancement.

1 Introduction

Speech enhancement ^[1] (SE) refers to the extraction of speech signals while suppressing sources of interference and eliminating noise. SE plays an important role in improving the intelligibility and quality of noisy speech recordings. In recent years, Deep Neural Network (DNN)-based speech enhancement methods have received a lot of attention as part of a broader interest in learning AI (Artificial Intelligence). Recurrent Neural Networks (RNN) ^[2] and Generative Adversarial Nets (GAN) ^[3] have both been explored for their potential for speech enhancement ^[4,5]. Recently, U-Net architectures based on magnitude spectrograms have achieved new-state-of-the-art results in audio source separation ^[6] and speech dereverberation ^[7]. Concurrently, some research has focused on using neural network architectures in the time domain for speech enhancement ^[8].

Problems. Applications need DNN-based SE models that are robust enough to handle real-world noisy speech. Unfortunately, the performance of these models drops dramatically when there is a mismatch between the training sets and real test sets.

In general, training sets for SE models are composed of noisy and clean speech pairs. The noisy speech is usually synthesized by adding noise to clean speech, depending on the Signal-to-Noise Ratio (SNR). Test models developed using these synthetic datasets make it possible to adopt an heuristic approach to model performance. However, it is not guaranteed that these models will perform well when deployed in real-world conditions ^[9]. To ensure that trained models are suitable for real recordings, it is best to collect training sets in real environments. However, there are two challenges confronting the practical realization of this:

- 1) It is difficult to collect a large enough number of real samples of noisy speech;
- 2) It is hard to match and pair the noisy speech with clean speech.

In view of the limited availability of noisy speech with unknown noise information, constructing synthetic paired training sets to develop trained models capable of handling real noise poses a serious problem.

Contributions. Drawing upon Chen's work on the blind denoising of images^[10], we have succeeded in developing realistic paired training sets that can be used to improve the performance of SE models. In particular, we have made use of the powerful capacity of GANs to learn complex data to design a Noise Modeling GAN (NM-GAN) that can learn the real noise distribution of noisy speech. Paired training sets can then be constructed by using the noise samples generated during the first step. After this, an SE model for denoising can be trained on the synthetic training sets. The layout of the full procedure is shown in Fig. 1.

Figure 1: The NM-GAN for making training sets.

It can be seen from the Fig. 1 that the core of the NM-GAN architecture is an alternating Generative Adversarial Net. A generative model, G , generates noise samples, then a discriminative model, D , estimates the probability that noise samples come from real noise rather than G . The main advantages of this approach are as follows:

- It works end-to-end. Therefore, no features need to be hand-crafted and no explicit assumptions about the raw noise need to be made.
- It operates directly in the time domain, permitting integrated modelling of the phase information and making it possible to take context into account.
- It enables a rapid noise modeling process. A Gaussian distribution is used to provide the conditional information and direct the noise generation.
- It can learn complex noise distributions. The combination of U-Net and LSTM^[11] gives the generator powerful modeling capabilities.

Here, we propose a new GAN for Noise Modeling that can build training sets by imitating real noise distribution. However, evaluation and comparison of the NM-GAN, or, equivalently, the noise samples generated by the NM-GAN, is challenging because of the lack of an explicit likelihood measure. Thus, most previous work in this area has resorted to merely subjective visual evaluations in the case of samples synthesized by GANs. However, it is impossible to judge the quality of samples precisely using subjective evaluation. Recent work has begun to target this challenge by adopting quantitative measures for evaluating GANs^[45,46,47-48]. On the basis of these works, alternative evaluation experiments, "GAN-train" and "GAN-test"^[49], were carried out to compute the distance of the generated samples from the real data manifold in terms of precision and recall. High precision implies that the

generated samples are close to the data manifold and high recall shows that the generator outputs samples that are effectively covering the breadth of the manifold. The GAN-train and GAN-test measures form the basis of four different experiments that were carried out to assess the NM-GAN's precision and recall.

- To verify that the GAN is capable of generating high precision noise that looks like real noise, the first experiment took noise samples from the NOISEX-92 database ^[12] with a known distribution as its input. The Mel-Frequency Cepstral Coefficients (MFCCs) ^[13] were adopted as an indicator of the similarity between the generated samples and real samples. We used MFCCs because these are a standard evaluation metric that have been used by the European Telecommunications Standards Institute for the past 20 years and they are widely used for the similarity evaluation of sound. The goal, here, was to assess the NM-GAN framework's ability to imitate noise. In addition, relevant state-of-the-art methods were used for our baseline comparisons.

- The purpose of the second experiment was to verify that the network can generate "diverse" noise which looks like real noise. This means that the NM-GAN has a high enough recall to generate "different" sounds that don't exist in the training data. Here, we relied on a neural net architecture for speech enhancement for the measurements. This evaluation exercise will be referred to here as the "GAN-train" experiment. In the "GAN-train" experiment, we trained an SE network with training states generated by the NM-GAN, then evaluated its performance on a test set composed of real-world noisy speech samples. The experiment set out to measure the difference between the learned (i.e., generated noise) and the target (i.e., real noise) distributions. We conclude that the generated samples will be similar to real ones if the SE network, which is trained on synthetic training set, can provide good noise reduction for real noisy speech. In other words, the performance of the SE network is itself akin to a recall measure, because a good SE network performance will indicate that the generated samples are sufficiently diverse.

- The third experiment will be referred to here as the "GAN-test" experiment. This measured and compared the accuracy of an SE network trained on real training sets and on generated synthetic samples. This measurement is similar to the precision, in that a high value denotes that the generated samples are a realistic approximation of the (unknown) distribution of natural samples.

- In addition to the above three measurements, we conducted a fourth experiment to study the relative merits of the proposed method by comparing the enhanced speech produced by an SE model trained on the synthetic training set with other typical noise-estimate based SE methods. In this experiment, the quality of the enhanced speech was evaluated according to the Signal-to-Noise Ratio (SNR) ^[14], Perceptual Evaluation of Speech Quality (PESQ) ^[15] and Mean Opinion Score (MOS) ^[16]. These metrics provided objective proof of the effectiveness of the proposed approach.

The rest of the paper is organized as follows: Various approaches to noise modeling are discussed in Section 2. The detailed NM-GAN architecture and the proposed method for building paired training sets is presented in Section 3. Information regarding the noise modeling and evaluation experiments is given in Section 4. Experiments' results together with their discussion are presented in Section 5. Some overall conclusions and potential future directions are presented in Section 6.

2 Related works

Generative adversarial networks have been used in a variety of ways to synthesize speech over recent years, frequently to good effect ^[17-18]. These are, in turn, based upon the use of GANs to synthesize audio waveforms of various kinds ^[19]. There have also been a number of efforts to use DNNs to reduce noise ^[20]. Some effort has also

been made to produce synthetic datasets that can be used to assist with training deep learning networks to deal with noisy speech, specifically in relation to overcoming the Lombard Effect ^[21]. The Lombard Effect relates to shifts in the inflection or pitch of people's speech to overcome noise in the surrounding environment. The latter approach was focused upon generating noisy speech tracks for training based upon a superposition of source signals. However, to the best of our knowledge, our work is the first to specifically focus on using GAN-based noise modeling to build training sets in the field of speech processing. This is significantly more powerful and less labor-intensive than constructing independent superimposed noise datasets. It should be noted that this article does not directly address the use of GANs to support speech enhancement. Instead it concentrates on a preliminary but vital step: the synthesis of noise to train speech enhancement algorithms. Since the effectiveness of the noise modeling directly affects the quality of the subsequent training sets, their performance depends directly upon the design of the noise modeling framework. We briefly examine here three related approaches to DNN-based modeling that differ in relation to how they make use of the time domain and frequency domain.

Traditional GAN. The advantage of a GAN is that its discriminator network can also serve as a loss function for generating data. In other deep learning network structures the loss function is difficult to choose and it is often hard to make it converge. The specific GAN used in our approach was inspired by the work of Chen on blind image denoising ^[10], which used a traditional GAN for image noise modeling. This GAN consisted of a 5-layer generator network and a 5-layer discriminator network. However, this kind of GAN has an unconditioned generative mode and no control over the modes of data being generated. The layers are also not especially deep and some recent deeper architectures have successfully increased the learning efficiency for noise ^[22-23].

STFT-RNN. Other recurrent neural networks (RNNs) apart from GANs have proved to be highly effective at exploiting the temporal context information in embedded signals. Notable here is the work of Glover ^[24], who has proposed a Short-Time Fourier Transform for RNNs (STFT-RNN) that can create musical audio signals. As noise can also be thought of as a special type of speech, this structure can be adapted to generate noise by replacing the inputs with noise samples. The basic idea of the structure is to use STFTs to learn sequences. First, the input signal is extracted from the magnitude and phase components. The magnitudes and frequencies can then be manipulated independently, in this case by passing them through an RNN. Resynthesis forms the last step. Unfortunately, the STFT-RNN structure only modifies the magnitude of the speech spectrum, with the short-time phase being treated as unimportant. This view has been overturned by recent studies, which have shown that using the of short-time phase can lead to significant improvements in speech quality ^[22, 25].

Various end-to-end networks in the time domain have been proposed to solve these problems^[26-27], which allow for the modelling of phase information and avoid fixed spectral transformations.

LSTM-RNN. Another useful technique is using Long Short-term Memory (LSTM) layers in the RNN (LSTM-RNN). This approach was introduced by Ayres to generate babbling speech in 2017^[28]. The structure, here, is three layers with a final fully-connected rectified linear unit layer that can learn data through imitation. The biggest disadvantage of LSTM-based systems is that the calculations are very time-consuming. As a result, they have largely been used for fairly small neural network models, which limits their expressive power.

Our proposed model has deeper layers than the three frameworks discussed above and operates directly in the time domain. The NM-GAN also fully considers the role of phase information and combines the advantages of several of the most recent and successful architectures. By conditioning the model on additional information, our model is able to direct the data generation process. On top of this, a U-Net architecture has been adopted where its encoder is a

stack of convolutional layers and pooling layers. This can extract high-level features from the input noise. The design is further enhanced by the introduction of two bidirectional LSTMs^[29] into the architecture. This makes the system well-suited to one-dimensional sequence signal processing.

3 Noise modeling and building the training sets

The core focus of this paper is upon the development of an advanced GAN for noise modeling and its use to construct paired training sets. In this section, we will briefly present the original principles associated with GANs and the most successful recent architectures. We will then introduce the NM-GAN. The section concludes by examining what is involved in synthesizing the paired training sets.

3.1 GAN: Basic principles and Evolved Architectures

GAN. In its original formulation, a GAN consists of a pair of competing neural networks: a generative model, G , that captures the data distribution for a set of input data; and a discriminative model, D , that estimates the probability that a sample came from a set of training data rather than G . The generator is trained to create samples that will fool the discriminator and an adversarial game is played between the two networks.

The GAN is used for data generation and given a dataset $\chi = \{(x_1, z_1), (x_2, z_2), (x_3, z_3) \dots (x_N, z_N)\}$ consisting of N pairs of signals: a target signal to be generated, x ; and a random signal z . To learn the generator's distribution, P_g , over data x , involves finding a mapping, $f(z): z \rightarrow x$, to map the random input signal, z , to the target distribution, x . Conforming to the principle of a GAN, the adversarial learning process is formulated as a minimax game between G and D , with the following objective:

$$\min_G \max_D V(D, G) = E_{x \sim P_{\text{data}}(x)} [\log D(x)] + E_{z \sim P_z(z)} [\log(1 - D(G(z)))] \quad (1)$$

where, $G(z)$ is a generated sample from the learned generator distribution; and the discriminator, D , receives a pair of signals, $(G(z), x)$ as input. The D is trained to maximize the probability of the correct label being assigned to both the training examples and the samples from G . Simultaneously, G is trained to minimize $\log(1 - D(G(z)))$.

CGAN. In practice, Eq. 1 may not provide a definitive direction for G to learn effectively. To solve this problem, a Conditional Generative Adversarial Network (CGAN)^[30] was proposed. This is an extension of the above that includes a conditional model where both the generator and discriminator are conditioned according to some extra information, x_c . x_c can be any kind of auxiliary information, thereby bypassing the needless generation of data. For a CGAN, the objective function can be expressed as follows:

$$\min_G \max_D V(D, G) = E_{x, x_c \sim P_{\text{data}}(x, x_c)} [\log D(x, x_c)] + E_{z \sim P_z(z)} [\log(1 - D(G(z, x_c), x_c))] \quad (2)$$

LSGAN. There is another problem in the learning process: if G is ineffective, D may reject samples with a high confidence because they are clearly different from the training data. In this case, $\log(1 - D(G(z, x_c), x_c))$ becomes saturated. The least-squares GAN (LSGAN)^[31] uses a least-squares function with binary coding to replace the cross-entropy loss in the original GAN formulation. This helps to stabilize the training and increases the quality of the generated samples in G . Here, the objective function can be expressed as follows:

$$\min_D V(D, G) = \frac{1}{2} E_{x, x_c \sim P_{\text{data}}(x, x_c)} [(D(x, x_c) - 1)^2] + \frac{1}{2} E_{z \sim P_z(z), x_c \sim P_{\text{data}}(x_c)} [D(G(z, x_c), x_c)^2] \quad (3)$$

$$\min_G V(D, G) = \frac{1}{2} E_{z \sim P_z(z), x_c \sim P_{\text{data}}(x_c)} [D(G(z, x_c), x_c - 1)^2] \quad (4)$$

3.2 NM-GAN structure

In this section, we present a structure that can learn to translate a sequence of random data, Z , into a real noise sample. The purpose of the NM-GAN is to generate noise samples that are as close as possible to real-world noise samples. Going from random data Z to real noise data X is a form of end-to-end mapping. The generator uses a symmetrical U-Net ^[32], where the encoder is a stack of convolutional layers and pooling layers. This serves to extract the high-level features from the input noise. The decoder has essentially the same structure as the encoder in reverse. It maps the low-resolution feature maps output from the encoder to full-size input noise feature maps.

As the input and output of the model share the same underlying structure, the G network also has skip connections. These connect each encoding layer to its homologous decoding layer, which is better able to extract the characteristics of the input data based on information regarding its different feature dimensions ^[33]. The G network model is shown in Fig. 2.

Figure 2: G network model.

The G network has an encoder-decoder structure and can repeatedly resample feature maps to compute and combine features at different time scales. The bidirectional LSTM in the middle of the encoder and decoder enables context to be taken into account. G is designed to be convolutional, so only some nodes between any adjacent two layers are connected. During the encoding stage, the input signal is projected and compressed through seven convolutional layers before being passed to a Batch Normalization and Parametric Rectified Linear Unit. This helps to increase the robustness of the model. The dimensions (sampling number $\times$ feature map) of each layer are, respectively, 1024×1 , 512×64 , 256×128 , 128×128 , 64×256 , 16×512 and 8×1024 .

We added the condition $f_{x_c(x_c)}$ to supervise the data generation. According to the central limit theorem, as the number of samples increases, the sample distribution tends towards a normal distribution (a Gaussian distribution with a mean of 0 and a variance of 1). The conditional information $f_{x_c(x_c)} \sim N(0,1)$ is therefore added to accelerate network convergence. This makes the dimensions of the final output of the encoding layer 8×2048 .

The sequence of latent feature vectors in the encoder is then modeled by two bidirectional LSTM layers. The LSTM layers do not change the shape of the data because the output sequence of the LSTM layers is subsequently converted back to the original input shape by the decoder. The encoding process is reversed in the decoding stage by seven deconvolutions, followed again by a Batch Normalization and Parametric Rectified Linear Unit.. The dimensions

(sampling number $\times$ feature map) of each decoding layer are thus 8×2048 , 16×1024 , 2×512 , 4×512 , 128×256 , 256×256 , 512×128 and 1024×1 .

An important feature of G is its end-to-end structure in the time domain, which permits integrated modelling of the phase information. The addition of the bidirectional LSTM improves the ability of the network structure to estimate the noise distribution.

Figure 3: D network model

The function of the D network is to discriminate and classify the real noise and generated noise output by the G network. D has the same one-dimensional convolutional structure as is present in G's encoder stage, so it matches the conventional topology of a convolutional classification network. D in the NM-GAN model has two input channels and both input channels span the one-dimensional convolutional encoding layers to form feature maps. The dimensions (sampling number $\times$ feature map) of each layer are, respectively, 1024×1 , 512×64 , 256×128 , 128×128 , 64×256 , 32×256 , 16×512 and 8×1024 . The structure of D is shown in Fig. 3. Batch Normalization is also added to each layer and all the convolutions apart from the final ones are followed by Leaky-ReLU non-linearities, where $\alpha = 0.2$. Finally, a least squares loss operation is used to distinguish between the real noise and generated noise.

It has been found that adding a secondary component to the loss of G can help minimize the distance between its generations and the real examples. We chose the cross-entropy loss from an Embedding Loss Driven Generative Adversarial Network (EL-GAN^[34]) to replace the discriminator D's cross-entropy loss and assigned a value to L1 of 120. On this basis, the G and D loss functions become:

$$\min_G V(D, G) = \frac{1}{2} E_{z \sim P_{z(z)}, x_c \sim P_{data(x_c)}} [D(G(z, x_c), x_c - 1)^2] + 120 \times \|G(z, x_c) - x\|_1 \quad (6)$$

$$\min_D V(D, G) = \frac{1}{2} E_{x, x_c \sim P_{data(x, x_c)}} [(D(x, x_c) - 1)^2] + \frac{1}{2} E_{z \sim P_{z(z)}, x_c \sim P_{data(x_c)}} [D(G(z, x_c), x_c)^2] \quad (7)$$

where, $f_{x_c(x_c)} \sim N(0,1)$ is the conditional information; z and x is a random input and a target real noise sample, respectively. The inputs of G are the random signal, z , together with the conditional, x_c , and the outputs are imitative noise samples $G(z, x_c)$. The input of D is a combination of the signals $(G(z, x_c), x_c)$ and (x, x_c) . $\|G(z, x_c) - x\|_1$ is the L1 distance between the real noise sample, x , and the generated sample, $G(z, x_c)$. We can

then train the D and the G to minimize and adjust the model parameters so that the random input samples, z , move towards the real noise samples, x .

3.3 Building the training sets

The overall process involved in building training sets from real-world noisy inputs was shown in Fig. 1. The first step is to extract noise samples from noisy speech to establish a noise dataset. The noise samples can, in principal, come from any public dataset containing noisy speech. In our own work, we made use of the NoiseX-92 database [35], which includes 'babble speech' as a category (i.e. multiple speakers all speaking at once). This step is followed by the noise modeling and noisy speech synthesis, which contain the following steps:

Figure 4: Training sets construction process

3.3.1 Noise datasets. Before building the paired training dataset, a set of approximate noise blocks (or patches), which we will call V_N , need to be extracted from given recordings of noisy speech. This way, the noise distribution becomes the principal learning objective for the NM-GAN, making the NM-GAN model more accurate. A Speech Endpoint Detection Method [36] can be used to extract the noise segments from the noisy speech. If the SNR of the noisy speech is extremely low, a viable noise sample cannot be extracted correctly and the sample has to be handled manually later. In addition, long stretches in recordings of real world speech contain no voices. When this part of the process is completed, a body of noise samples is available to use as the training set, V_N .

3.3.2 Noise modeling. The NM-GAN is trained using the method proposed in 3.2, with the real noise samples, V_N . For the training process, Google's TensorFlow framework is adopted and the model is trained using an RMSProp optimizer [37]. The learning rate for G and D was set to 0.0002. Within a minibatch, all the training samples are padded with zeros to have the same number of time steps as the longest sample. The process is stopped after 50 epochs if there is no improvement in the validation set, as measured by the loss. Once this process is complete, the last-obtained model is further fine-tuned, with the batch size being doubled and the learning rate lowered to 0.00001. This is again pursued until there are 50 epochs without any improvement in the validation loss. After this, the model with the best validation loss is selected. The Generator can generate countless noise samples, V_N' , in this way.

3.3.3 Noisy speech synthesis. Working on an assumption that additive noise can be used to create noisy speech, a noisy speech database can be created by adding noise at real Signal to Noise Ratio (SNR) levels to clean speech. In

this way, enough pure-noisy speech pairs $\{D_P \text{ and } D_N\}$ can be synthesized, consisting of pure speech and realistic synthesized noisy speech, to be used as training sets, $\{T_P, T_N\}$.

4. Experiments

To assess the viability of using the proposed NM-GAN model to generate noise that looks like real noise and to make training sets, we carried out four verification and comparison experiments. The first experiment took some noise samples from the NOISEX-92 database ^[35], which has a known distribution, and used them as input to verify the imitation ability of the NM-GAN framework. The second experiment compared the performance of an SE model when it is trained using a synthetic training set and a real test set. The third experiment compared the performance of an SE model when it is trained using a real training set and a synthetic test set. A fourth experiment was then conducted to assess the quality of the speech enhanced by the synthetic training set and to compare it to other typical noise estimation-based SE methods. The detailed experimental setup and experimental results are described below.

5. Results and Discussion

5.1 Noise modeling experiments

The purpose of the experiments in this section was to verify that the network has the ability to imitate complex noise. For a certain noise distribution, the model proposed in this paper can achieve convergence faster and generate samples that have a greater similarity to the target data.

Experimental Data. Some typical noise samples from the NoiseX-92 database ^[35] were used as input to verify the imitation ability of the NM-GAN framework. The NoiseX-92 database is widely-used for speech enhancement tests and speech recognition tests. It contains 15 types of noises, including, voice babble, factory noise, HF radio channel noise, pink noise, white noise, various military noises and a Volvo 340 ^[38]. Out of these, the most challenging noise conditions relate to multiple-speaker or a voice babble' environments. This kind of noise is uniquely challenging because of its highly time dependent structure and its similarity to the desired target. These difficulties have been well-documented in numerous studies of robustness ^[39]. Apart from babble noise, we took three other types of noise with obvious differences in the time and frequency domain, namely: 'destroyer engine noise'; 'machine-gun noise'; and 'factory floor noise'. We randomly selected three seconds from samples of the four types of noise data to use as specific noise samples. We then separately trained the NM-GAN network to model them by imitation.

Method Comparison. We compared the performance of NM-GAN against three other state-of-the-art DNN-based architectures, as mentioned in section 2, namely, a GAN ^[10], a short-time Fourier transform recurrent neural network (STFT-RNN) ^[24], and an LSTM-RNN ^[28]. All three methods were trained with an initial learning rate of 0.0002 and a batch size of 1. We took the same experimental data from the NoiseX-92 database to compare the effectiveness of the noise mimicry after 500 training epochs.

Results. The waveforms and spectrograms of the generated noise for NM-GAN and the other three methods are shown in Figs. 5, 6, 7 and 8.

Figure 5: Factory floor noise

Figure 6: Destroyer engine noise

Figure 7: Machine-gun noise

Figure 8: Speech babble noise

Looking at the waveforms and spectrograms, we can see that the samples generated by NM-GAN are the most similar to the real noise samples. To assess the similarity objectively, the 12-dimensional mel-frequency cepstrum coefficients ^[33] (MFCCs) taken together with the 1st order differential coefficients were used as evaluation parameters. MFCCs are widely-used in speech similarity evaluation as a way of extracting and selecting parametric representations of acoustic signals ^[40].

We calculated the average Euclidean distance of the MFCCs for noise samples generated by different methods when using real noise. The results are shown in Table 1.

Table 1 Average Euclidean distance of the MFCCs relating to different methods

Method	Average Euclidean distance of the MFCCs			
	Factory floor noise	Destroyer engine noise	Machine-gun noise	Speech babble noise
STFT-RNN	25.32213	27.71124	27.59945	26.15075
LSTM-RNN	21.02848	23.92830	38.82731	28.82629
GAN	24.83476	41.28380	31.33846	37.98472
NM-GAN	13.43358	19.96358	18.57913	19.48051

It can be seen from Table 1 that the average distance of the MFCCs for the real noise samples generated by NM-GAN is smaller than it is for the other models. This confirms that the proposed model can outperform other state-of-the-art DNN-based models and generate more realistic noise samples.

5.2 GAN-train experiment

An important characteristic of the proposed NM-GAN model is that the generated noise samples should not only be realistic, but also recognizable as coming from a given class. This means that a model that perfectly captures the target distribution can generate a new set that is indistinguishable from the original input training set. An SE net trained on either of these should then produce roughly the same validation accuracy. In this experiment, we selected a speech enhancement model to evaluate the ability of NM-GAN to model noise effectively and to verify the overall feasibility of the approach. We reasoned that, if a speech enhancement model trained on the NM-GAN synthetic training sets can enable good noise reduction for real noisy speech, it will prove the diversity of the NM-GAN model.

Experimental setup. We used the Speech Enhancement Generative Adversarial Nets (SEGAN)^[5] model as the SE model for testing. This model was chosen because its code is completely public and widely-cited. The model was trained on synthetic datasets. Then, we compared the enhancement effect of the training model on the synthetic and real test sets respectively. For the dataset, we chose the Voice Bank corpus ^[41] and Demand database ^[42] as the original pure speech and noise datasets. These speech and noise datasets are also completely public and widely-used. We recorded some noisy samples with different SNRs under laboratory conditions (one player played pure speech from the Voice Bank corpus and one played noisy speech from the Demand database). This gave us 150,000 real noisy datasets in total.

Making the training set. In accordance with the process outlined in section 3.3, we randomly selected 8,000 utterances in real noisy speech samples to obtain 10,000 noise blocks, V_N. These noise blocks were adopted as the real noise samples to train the NM-GAN model presented in section 3.3.2. During the training process, the batch size was set to 16 and the epoch was set to 10. After obtaining the noise model, we built a synthetic training set made

up of 150,000 different mixtures {-5 dB, 0 dB, 5 dB} with a total duration of about 100 hours. 140,000 of these were then used to train the SEGAN model.

Test set. The test set consisted of two parts, 10,000 real utterances were randomly selected from real noisy speech datasets as the real test set, and the remaining 10,000 samples from the synthetic noisy speech datasets were used as the synthetic test set.

Evaluation metrics. To compare the quality of the speech enhancement for the synthetic noisy speech and real noisy speech objectively, we used the following objective metrics:

PESQ: Perceptual evaluation of speech quality (from -0.5 to 4.5).

CSIG: MOS prediction of the signal distortion (from 1 to 5).

CBAK: MOS prediction of the intrusiveness of the background noise (from 1 to 5).

COVL: MOS prediction of the overall effect (from 1 to 5).

SNR: Signal-to-Noise Ratio (from 0 to ∞).

Results. The speech enhancement model ^[5] trained on the synthetic training set was evaluated using the real and synthetic test sets.

All of the results were calculated by comparison between the enhanced signal and the pure signal. In order to make the experimental results accurate and reduce any possible errors, all of the metrics were based on the average of 30 speech signals. Table 2 shows the results.

Table 2: Objective SEGAN-enhanced evaluation results comparing the synthetic and real training sets

Metric Model/Test /SNR			PESQ	CSIG	CBAK	COVL	SNR
SE trained on synthetic training set	Synthetic Noise test set	-5dB	1.45	2.81	2.90	2.80	3.67
		0dB	2.52	3.15	3.27	2.93	8.15
		5dB	2.60	3.79	3.72	3.25	7.12
	Real noise test set	-5dB	1.09	2.76	2.72	2.29	3.11
		0dB	2.32	3.15	3.21	2.63	7.94
		5dB	2.45	3.56	3.47	3.16	6.88

It can be seen that, systematically across all of the evaluation metrics, the SE trained on the synthetic training set obtained almost the same scores for the synthetic noise test set as for the real noise test set.

Analysis/Discussion. In the case of the PESQ measurements, the trained SE test on the synthetic training set was 0.36 points (23%) different from the SE test on the real noise test set at -5dB. At 0dB it was 0.30 points (13%) and at 5 dB it was 0.15 points (6%). So, overall, for PESQ, the average difference was 0.27 points. For the CSIG measurements, the difference was 0.05 (2%), 0.00 (0%) and 0.23 (6%), for -5dB, 0dB and 5dB, respectively, with an overall difference, again, of 0.09 points. For the CBAK measurements, the difference was 0.18 (7%), 0.06 (2%) and 0.15 (4%), for -5dB, 0dB and 5dB, respectively, with an overall difference of 0.13 points. For the COVL measurements, the difference was 0.51 (22%), 0.30 (11%) and 0.09 (3%), for -5dB, 0dB and 5dB, respectively, with an overall difference of 0.3 points. For the SNR improvements, the difference was 0.56 (18%), 0.21 (3%) and 0.24 (3%), for -5dB, 0dB and 5dB, respectively, with an overall difference of 0.34 points. Across all measurements, the average difference in performance was 0.33 at -5dB, 0.15 at 0dB and 0.14 at 5dB.

Notice that the speech enhancement model trained on the NM-GAN synthetic training sets was able to provide good levels of noise reduction for real noisy speech. As pointed out in the introduction, if an SE network trained on a synthetic training set can provide good noise reduction for real noisy speech, this implies that the generated samples are similar to real ones. As the SE network performed well in this experiment, it proves the diversity of the NM-GAN model.

5.3 GAN-test experiment

The GAN-test experiment was designed to assess the accuracy of an SE net trained on a real training set, but tested on synthetic samples. If the NM-GAN learns well, the noisy speech improvement should be more or less the same because both the sets have the same distribution. Ideally, then, the GAN-test score should be close to the validation score. If it is significantly higher, it means that the NM-GAN is overfitting and simply memorizing the training set. If it is significantly lower, the NM-GAN is not capturing the target distribution well and the modeling quality is poor. The overall setup of the GAN-test experiment was similar to Experiment 2, except that the training set became the real training set.

Experimental setup. We used the same SEGAN model as the SE model for testing. The model was trained on the real dataset. Then, we compared the enhancement effect of the training model on the synthetic and real test sets, respectively.

Results. The speech enhancement model ^[5] trained on the real training set was evaluated using the real and synthetic test sets. All of the results were calculated by comparison between the enhanced signal and the pure signal. In order ensure the accuracy of the experimental results and reduce any possible errors, all of the metrics were based on an average of 30 speech signals. Table 3 shows the results.

Table 3: Objective SEGAN-enhanced evaluation results comparing the synthetic and real training sets

Metric Model/Test /SNR			PESQ	CSIG	CBAK	COVL	SNR
SE trained on real training set	Synthetic noise test set	-5dB	1.01	2.26	2.32	2.19	3.07
		0dB	2.15	2.85	3.18	2.14	7.36
		5dB	2.03	3.12	3.23	3.23	7.82
	Real noise test set	-5dB	1.04	2.32	2.41	2.25	3.18
		0dB	2.21	3.04	3.17	2.21	7.40
		5dB	2.05	3.17	3.29	3.37	8.00

It can be seen that, systematically across all of the evaluation metrics, the SE trained on the real training set obtained almost the same degree of improvement for the synthetic noise test set and real noise test set.

Analysis/Discussion. For the SE trained on the real training set, the enhancement scores for the synthetic noisy speech test set and real noisy speech test set were almost equal. Comparing the PESQ measurements, the differences were 0.03 points, 0.06 points and 0.02 points, for -5dB, 0dB and 5dB, respectively, which amounts to 0.04 points overall. For the CSIG measurements, the differences were 0.06 points, 0.19 points and 0.05 points, for -5dB, 0dB and 5dB, respectively, which is 0.1 points overall. For the CBAK measurements, the differences were 0.09 points, 0.01 points and 0.06 points, for -5dB, 0dB and 5dB, respectively, which is 0.05 points overall. For the COVL

measurements, the differences were 0.06 points, 0.02 points and 0.05 points, for -5dB, 0dB and 5dB, respectively, which is difference of 0.04 points overall. For the SNR improvement, the performance differences were 0.11 points, 0.04 points and 0.18 points, for -5dB, 0dB and 5dB, respectively, which is 0.11 points overall.

These results show that the overall improvement for the synthetic training set with the synthetic test data was only slightly different to when the SE model was trained on the real training set. This indirectly proves that the NM-GAN learns effectively from input noise samples and that the proposed method can synthesize data sets similar to real samples.

5.4 Speech enhancement comparison experiment

Results. To further examine the ability of the model trained on our synthesized training set to enhance real noisy speech, we compared it with some typical noise estimation-based speech enhancement methods, including Martin's minimum tracking algorithm ^[43] and the minima controlled recursive algorithm (MCRA) ^[44]. The comparative results for the different methods are shown in Table 4.

Table 4: Comparative evaluation for SE results using NM-GAN and other methods.

Metric Method/SNR		PESQ	CSIG	CBAK	COVL	SNR
Martin's Algorithm	-5dB	0.71	1.97	1.26	1.38	1.26
	0dB	0.87	2.18	1.84	2.21	5.02
	5dB	1.79	2.38	2.92	2.73	5.81
MCRA	-5dB	0.73	1.27	1.76	1.82	1.04
	0dB	0.96	2.83	1.34	2.38	5.12
	5dB	1.97	3.05	2.31	2.94	5.96
NM-GAN	-5dB	1.09	2.76	2.72	2.29	3.11
	0dB	2.32	3.15	3.21	2.63	7.94
	5dB	2.45	3.56	3.47	3.16	6.88

Analysis/Discussion. It can be seen, here, that NM-GAN produced the best objective evaluation results across all of the different evaluation metrics. To break these down, for the PESQ measurements, the NM-GAN outperformed MCRA by 33%, 58% and 20% for -5db, 0dB and 5dB, respectively. It outperformed Martin's algorithm by 35%, 62% and 27% for -5db, 0dB and 5dB, respectively. For the CSIG measurements, it outperformed MCRA by 54%, 10% and 14% for -5db, 0dB and 5dB, respectively and, for Martin's algorithm by 29%, 31% and 33% for -5db, 0dB and 5dB, respectively. For the CBAK measurements, it outperformed MCRA by 35%, 58% and 33% for -5db, 0dB and 5dB, respectively and, for Martin's algorithm by 54%, 43% and 15% for -5db, 0dB and 5dB, respectively. For the COVL measurements, it outperformed MCRA by 20%, 9% and 7% for -5db, 0dB and 5dB, respectively and, for Martin's algorithm by 40%, 16% and 16% for -5db, 0dB and 5dB, respectively. For the SNR measurements, it outperformed MCRA by 66%, 36% and 13% for -5db, 0dB and 5dB, respectively and, for Martin's algorithm by 59%, 37% and 16% for -5db, 0dB and 5dB, respectively. So, NM-GAN outperformed MCRA by between 7% and 66%, and outperformed Martin's algorithm by between 16% and 62%. These results underscore the advantages offered by using NM-GAN to generate the training sets for DNN-based SE algorithms.

6 Key Outcomes and Conclusion

6.1 Key Outcomes

Comparing Tables 2 to 4, we can make the following observations:

- The proposed NM-GAN model has the ability to imitate complex noise.
- Although the enhancement model trained on synthetic training set had a better effect on synthetic noisy speech than real noisy speech, the difference was relatively small. Nonetheless, it was sufficiently large to indicate that the noise samples generated by NM-GAN are sufficiently diverse.
- The enhancement model trained on the real training set had almost the same effect for real noisy speech and synthetic noisy speech, suggesting that the generated samples are a realistic approximation of the unknown distribution of real noise samples.
- The speech enhancement model trained on our synthesized training set improved the quality of noisy speech better than traditional noise estimation-based methods, in many instances to a significant degree, with an improvement of anything up to 62% in relation to Martin's algorithm and up to 66% in relation MCRA.
- The robustness of DNN-based speech enhancement models can be improved by adopting our approach to noise modeling and the construction of synthetic training sets.

6.2 Conclusion

In this paper, we have highlighted the problem of there being a lack of real training sets for DNN-based speech enhancement models, which can limit the effectiveness of their performance and their robustness. In relation to this, we have proposed a method that can generate realistic paired training sets for noise modeling. The method is based on an end-to-end NM-GAN that can model noise in the time domain without any preprocessing or postprocessing. This can then be applied it to the generation of training sets. A U-Net architecture was adopted that can combine high- and low-level features at different time-scales by repeated down-sampling and convolution of the feature maps. The use of a two bidirectional LSTM further improved the NM-GAN's ability to learn noise distributions and generate noise samples.

Four experiments were conducted to assess the viability of the proposed method. In the first of these, the NM-GAN outperformed other state-of-the-art DNN-based noise modeling architectures when they were trained using the same settings and same noise sets. In the second, GAN-train experiment, we verified that a speech enhancement model trained using synthetic training sets generated by the NM-GAN architecture could achieve good noise reduction for real noisy speech. In the third GAN-test experiment, we verified that a speech enhancement model trained using real training sets generated by the NM-GAN architecture could achieve almost same improvement for real noisy speech and synthetic noisy speech. Finally, we established that the quality of the speech enhanced by the NM-GAN trained model was better than that of other noise estimation algorithms.

The method proposed in this article is not only able to increase the robustness of the training for speech enhancement models being applied to actual speech. It can also be used for blind speech denoising when all that is available is limited noisy speech with unknown noise information. Beyond this, the proposed method has been found to be effective for the noise calibration of sound equipment and noise calibration in special environments.

6.3 Limitations and Future work

A limitation of the proposed method is that the noise is assumed to be additive noise with a zero mean. Although this type of noise is common in natural environments and includes a wide range of noise, it has certain limitations. As a next step, we will be examining how to overcome this limitation and intend to investigate the extent to which the model might be capable of performing a spectral analysis. In addition, other methods for the synthesis of noise and pure speech, such as Parallel Model Combination (PMC) ^[33], could be explored to see whether they provide additional advantages.

DECLARATIONS

The submitted manuscript entitled “Noise Modeling to Build Training Sets for Robust Speech Enhancement”, which has no conflict of interest exists. And We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the paper. Besides, the manuscript is approved by all authors for publication and we ensure that the Declarations section contains all of the sub-sections.

Availability of data and materials

The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Funding

The fund support of this paper is National Natural Science Foundation of China(NO:62071057)

Authors' contributions

YahuiWang analyzed and studied the NM-GAN method. Kong Xinxin and Yongbiao Wang collected and produced the Data sets. Hongxin zhang guided the whole research. Wenxi zhang was a major contributor in writing the manuscript. All authors read and approved the final manuscript.

Acknowledgements

Not applicable.

ABBREVIATIONS

Deep Neural Network (DNN)

Generative Adversarial Network framework for Noise Modeling (NM-GAN)

Long Short-Term Memory (LSTM)

Speech enhancement (SE)

AI (Artificial Intelligence)

Recurrent Neural Networks (RNN)

Generative Adversarial Nets (GAN)

Signal-to-Noise Ratio (SNR)

generative model (G)

discriminative model (D)

Mel-Frequency Cepstral Coefficients (MFCCs)

Perceptual Evaluation of Speech Quality (PESQ)

Mean Opinion Score (MOS)

Short-Time Fourier Transform for RNN (STFT-RNN)

Long Short-term Memory (LSTM) in RNN(LSTM-RNN)

Conditional Generative Adversarial Network (CGAN)

The Least-Squares GAN (LSGAN)

Speech Enhancement Generative Adversarial Nets (SEGAN)

MOS prediction of the signal distortion (CSIG)

MOS prediction of the intrusiveness of the background noise (CBAK)

MOS prediction of the overall effect (COVL)

Parallel Model Combination (PMC)

REFERENCES

- [1] P.C. Loizou, Speech Enhancement: Theory and Practice, 2nd ed. Boca Raton, FL, USA: CRC Press, Inc., 2013.
- [2] Miao Y, Gawayyed M, Metze F. EESEN: End-to-End Speech Recognition using Deep RNN Models and WFST-based Decoding[J]. 2015.
- [3] I. Goodfellow, "Generative adversarial nets," in Advances in Neural Information Processing Systems (NIPS), 2014, pp. 2672-2680.
- [4] Lei Sun, Jun Du, Li-Rong Dai. Multiple-target deep learning for LSTM-RNN based speech enhancement[C]// 2017 Hands-free Speech Communications and Microphone Arrays (HSCMA). IEEE, 2017.
- [5] Santiago P, Antonio B, Joan S. SEGAN: Speech Enhancement Generative Adversarial Network [C]// INTERSPEECH. 2017:3642-3646.
- [6] Andreas Jansson, Eric J. Humphrey, Nicola Montecchio, Rachel M. Bittner, Aparna Kumar, and Tillman Weyde. Singing voice separation with deep u-net convolutional networks. In Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017, pages 745–751, 2017.
- [7] Ori Ernst, Shlomo E. Chazan, Sharon Gannot, and Jacob Goldberger. Speech dereverberation using fully convolutional networks. CoRR, abs/1803.08243, 2018.
- [8] Dario Reithage, Jordi Pons, and Xavier Serra. A Wavenet for Speech Denoising. Technical report.
- [9] Reddy C K A, Beyrami E, Dubey H, et al. The INTERSPEECH 2020 Deep Noise Suppression Challenge: Datasets, Subjective Speech Quality and Testing Framework[J]. 2020.
- [10] Chen J, Chao H, et al. Image Blind Denoising with Generative Adversarial Network Based Noise Modeling[C]// IEEE/CVF Conference on Computer Vision & Pattern Recognition. IEEE, 2018.
- [11] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. Neural Computation, 9(8):1735–1780, 1997.
- [12] The noisex-92 study on the effect of additive noise on automatic speech
- [13] Molau S, Pitz M, Schluter R, et al. Computing Mel-Frequency Cepstral Coefficients on The Power Spectrum[C]// Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on. IEEE, 2001.
- [14] S.R. Quackenbush, T.P. Barnwell, M.A. Clements, Objective Measures of Speech Quality, Prentice-Hall, NJ, 1988
- [15] Voran, S. "Objective estimation of perceived speech quality". IEEE Trans. Speech and Audio Processing, 7 (4), 371–382, July 1999.
- [16] Mahesh Viswanathan, Madhubalan Viswanathan. Measuring speech quality for text-to-speech systems: development and assessment of a modified mean opinion score (MOS) scale[J]. Computer Speech & Language, 19(1):55-83.
- [17] Bińkowski, M., Donahue, J., Dieleman, S., Clark, A., Elsen, E., Casagrande, N., ... & Simonyan, K. (2019). High fidelity speech synthesis with adversarial networks. *arXiv preprint arXiv:1909.11646*.
- [18] Marafioti, A., Perraudin, N., Holighaus, N., & Majdak, P. (2019, May). Adversarial generation of time-frequency features with application in audio synthesis. In *International Conference on Machine Learning* (pp. 4352-4362).
- [19] Donahue, C., McAuley, J., & Puckette, M. (2018). Adversarial audio synthesis. *arXiv preprint arXiv:1802.04208*.

- [20] Choi, I., Kwon, K., Bae, S. H., & Kim, N. S. (2016, September). DNN-based sound event detection with exemplar-based approach for noise reduction. In *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)* (pp. 16-19)
- [21] Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., ... & Ng, A. Y. (2014). Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.
- [22] Chiluveru, S. R., & Tripathy, M. (2019). Low SNR speech enhancement with DNN based phase estimation. *International Journal of Speech Technology*, 22(1), 283-292.
- [23] Shen, Z., He, Z., Cui, W., Yu, J., Zheng, Y., Zhu, C., & Savvides, M. (2019). Adversarial-based knowledge distillation for multi-model ensemble and noisy data refinement. *arXiv preprint arXiv:1908.08520*.
- [24] John Glover, Generating sound with recurrent neural networks. <http://www.johnglover.net/blog/generating-sound-with-rnns.html>
- [25] Wakabayashi, Y. (2019). Speech enhancement using harmonic-structure-based phase reconstruction. *Acoustical Science and Technology*, 40(3), 162-169.
- [26] A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In Proc. ICASSP, 2013.
- [27] A. Graves and J. Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in Neural Information Processing Systems*, volume 21, 2008.
- [28] Phil Ayres. Generating speech from speech with LSTM networks. <http://babble-rnn.consected.com/docs/babble-rnn-generating-speech-from-speech-post.html>
- [29] J. Chen and D. L. Wang, "Long short-term memory for speaker generalization in supervised speech separation," *Proceedings of Interspeech*, pp. 3314–3318, 2016.
- [30] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *ArXiv:1611.07004*, 2016.
- [31] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, and Z. Wang, "Least squares generative adversarial networks," *ArXiv: 1611.04076*, 2016.
- [32] Daniel Stoller, Sebastian Ewert, and Simon Dixon. Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation. 2018.
- [33] Gales, M. J. F., Young, S. J., 1996. Robust continuous speech recognition using parallel model compensation. *IEEE Trans. Acoust. Speech Signal Process.* 4 (5), 352–359.
- [34] Ghafoorian, M., Nugteren, C., Baka, N., Booi, O., & Hofmann, M. (2018). El-gan: Embedding loss driven generative adversarial networks for lane detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*
- [35] Varga, A., & Steeneken, H. J. (1993). Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems. *Speech communication*, 12(3), 247-251
- [36] LIU Qingsheng XU. Research on a Speech Endpoint Detection Method[J]. *Computer Engineering*, 2003.
- [37] T. Tieleman and G. Hinton, "Lecture 6.5-RMSprop: divide the gradient by a running average of its recent magnitude," COURSE: Neural Networks for Machine Learning 4, 2, 2012.
- [38] <http://mi.eng.cam.ac.uk/comp.speech/Section1/Data/noisex.html>
- [39] Krishnamurthy N, Hansen J H L. Speech babble: Analysis and modeling for speech systems[C]// IEEE International Conference on Acoustics. IEEE, 2008.
- [40] ZHAO Gui-lan, SUN Dong. Application of MFCC Based on MATLAB in Evaluation of Speech Similarity[J]. *journal of lanzhou institute of technology*,
- [41] C. Veaux, J. Yamagishi, and S. King, "The voice bank corpus: Design, collection and data analysis of a large regional accent speech database," in *Int. Conf. Oriental COCOSDA, held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*. IEEE, 2013, pp. 1–4.
- [42] J. Thiemann, N. Ito, and E. Vincent, "The diverse environments multi-channel acoustic noise database: A database of multichannel environmental noise recordings," *The Journal of the Acoustical Society of America*, vol. 133, no. 5, pp. 3591–3591, 2013.
- [43] Martin, R. (2001). Noise power spectral density estimation based on optimal smoothing and minimum statistics. *IEEE Transactions on Speech and Audio Processing*, 9(5), 504-512.

- [44] Cohen, I. (2002). Noise estimation by minima controlled recursive averaging for robust speech enhancement. *IEEE Signal Processing Letters*, 9(1), 12-15.
- [45] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In: *NIPS* (2017)
- [46] Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of GANs for improved quality, stability, and variation. In: *ICLR* (2018)
- [47] Lucic, M., Kurach, K., Michalski, M., Gelly, S., Bousquet, O.: Are GANs created equal? A large-scale study. *arXiv preprint arXiv:1711.10337* (2017)
- [48] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training GANs. In: *NIPS* (2016)
- [49] Shmelkov, Konstantin, C. Schmid, and K. Alahari. "How good is my GAN?." (2018).

Figures

Figure 1

The NM-GAN for making training sets.

Figure 2

G network model.

Figure 3

D network model

Real-world noisy speech

Noise blocks extraction

Noise modeling

Generate noise samples

Synthesize noisy speech

Figure 4

Training sets construction process

Figure 5

Factory floor noise

Figure 6

Destroyer engine noise

Figure 7

Machine-gun noise

Figure 8

Speech babble noise