applied sciences

MDPI

Article

Noise Modeling to Build Training Sets for Robust Speech Enhancement

Yahui Wang [1,2], Wenxi Zhang [2], Zhou Wu [2], Xinxin Kong [2], Yongbiao Wang [2] and Hongxin Zhang [1,*]

[1] School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China; wangyahui@aoe.ac.cn
[2] Key Laboratory of Computational Optical Imaging Technology, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100081, China; zhangwenxi@aoe.ac.cn (W.Z.); wuzhou@aoe.ac.cn (Z.W.); xxkong@aoe.ac.cn (X.K.); wangyb@aircas.ac.cn (Y.W.)
* Correspondence: hongxinzhang@bupt.edu.cn

Abstract: DNN-based Speech Enhancement (SE) models suffer from significant performance degradation in real recordings due to the mismatch between the synthetic datasets employed for training and real test sets. To solve this problem, we propose a new Generative Adversarial Network framework for Noise Modeling (NM-GAN) that creates realistic paired training sets by imitating real noise distribution. The proposed framework combines a novel 7-layer U-Net with two bidirectional long short-term memory (LSTM) layers that act as a generator to construct complex noise. NM-GAN generates enough recall (diversity) and precision (noise quality) in its samples through adversarial and alternate training, effectively simulating real noise, which is then utilized to compose realistic paired training sets. Extensive experiments employing various qualitative and quantitative evaluation metrics verify the effectiveness of the generated noise samples and training sets, demonstrating our framework's capabilities.

Keywords: Noise Modeling; training set; Generative Adversarial Network; speech enhancement

Citation: Wang, Y.; Zhang, W.; Wu, Z.; Kong, X.; Wang, Y.; Zhang, H. Noise Modeling to Build Training Sets for Robust Speech Enhancement. *Appl. Sci.* **2022**, *12*, 1905. https://doi.org/10.3390/app12041905

Academic Editors: Andrea Prati, Carlos A. Iglesias, Vincent A. Cicirello and Luis Javier García Villalba

Received: 20 January 2022
Accepted: 10 February 2022
Published: 11 February 2022

1. Introduction

Speech enhancement [1] (SE) is 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 SE methods have received significant attention as part of a broader interest in learning-related Artificial Intelligence (AI). Important contributors are the Recurrent Neural Networks (RNNs) [2,3] and Generative Adversarial Nets (GANs) [4,5], along with other DNN-based architectures [6–8] that have already been explored in SE tasks.

1.1. Problem Statement

Most DNN-based SE methods involve many labeled samples to train the SE model parameters. In this case, the trained model learns the characteristics of the labeled samples. However, in some cases, real speech recordings are captured in diverse and noisy conditions, where the same microphone captures simultaneously and under the same acoustic conditions both speech and noise. Therefore, simulating such conditions when using synthetic training data is challenging, as the clean speech and noise signals are captured independently or under different acoustic conditions. As a result, a potential mismatch between the synthetic training sets and real test sets [9] imposes a substantial performance drop.

The optimum way to ensure that the trained SE models are suitable for real noisy speech tasks is to collect the training sets from the same noisy conditions. However, this strategy has the following significant challenges:

(1) When focused upon specific tasks such as speech information acquisition, it is difficult to collect a large enough number of real noisy speech samples.
(2) For most SE tasks, the training set comprises noisy and clean speech pairs. However, the captured noisy speech commonly has no corresponding pure speech usable for real applications.
(3) It is hard to match and pair the noisy speech with clean speech when the Signal-to-Noise Ratio (SNR) of the collected noisy speech is very low or there is continuous interference or interruptions in the recording process.

In practical applications, especially in voice detection related to intelligence acquisition and national security, we can only obtain limited noisy voice samples, as we cannot enter the detection scene in advance, and the time of voice detection is limited. Hence, only limited noisy speech can be collected, and the lack of adequate noisy speech combined with the unavailability of clean speech signals seriously affect current DNN-based SE tasks.

1.2. Contributions

This paper explores ways to learn the common characteristics of specific noise signal classes to address the above problems. Then, we aimed to exploit these characteristics to generate many noise samples for the classes that could form the training set basis. By conducting this investigation and adopting the blind image denoising scheme of [10], we extended GANs to learn complex data and designed a GAN for Noise Modeling (NM-GAN) to learn the real noise distribution of noisy speech samples. Then, we create synthetic paired training sets with a pure speech from the noise samples generated by NM-GAN. Finally, our synthetic paired training sets can train an SE model for denoising. The NM-GAN architecture is illustrated in Figure 1.

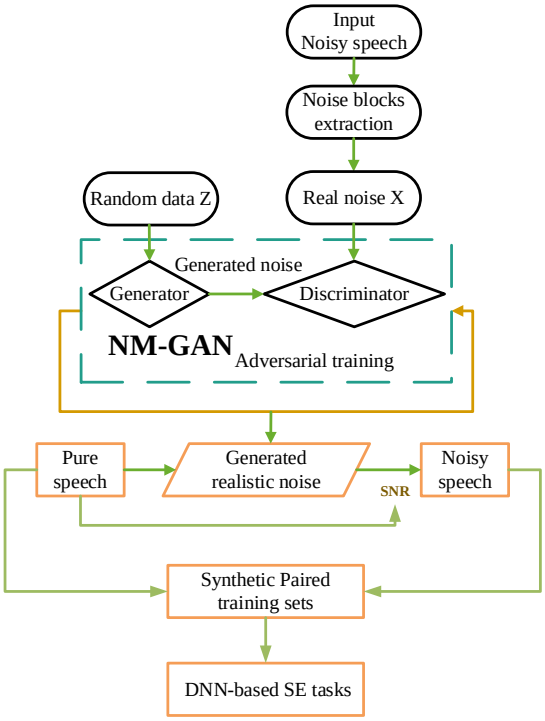

Figure 1. Proposed NM-GAN for creating synthetic training sets.

Figure 1 highlights that the NM-GAN architecture's core process is a Generative Adversarial Net. Specifically, a generative model *G* generates noisy samples, and a discriminative model *D* estimates the probability that any noise sample it receives is real noise rather than noise generated by *G*. NM-GAN aims to make the noise generated by *G* indistinguishable to *D*. Through continuous confrontational learning, the G and D networks ultimately reach a balance. Thus, ultimately *G* can generate many noise samples that are very similar to real

noise samples. Then realistic paired training sets can be composed using these generated noise samples.

However, the following question might arise. why use noise samples generated by NMGAN to make synthetic training sets instead of directly utilizing noise segments from the captured noisy speech? This is because the number of real noise samples is typically small, and therefore we commonly add the real noise samples with the pure speech samples to generate noisy speech signals and create paired training sets. However, such noise samples are quite limited, and the created synthetic noisy speech contains specific noise samples. Hence, an SE model learns these specific samples and fails even if the noisy test speech contains the same type of noise but with a slightly different amplitude or waveform.

NM-GAN can generate enough recall (diversity) and precision (noise quality) in its samples, simulating real noise through adversarial and alternate training than directly using limited real noise synthetic training sets. Then, these samples are employed to create extensive realistic paired training sets, affording to the SE model trained on these training sets to perform better even on real noisy samples under various waveforms. Given the importance of realistic datasets, this paper focuses on developing a GAN that effectively models noise and creates synthetic but highly credible training sets. It should be noted that this work does not focus on directly employing GANs to support SE, but it concentrates on a preliminary but vital step, namely on how to best make synthetic noisy speech training sets that could be used to train SE algorithms. Thus, the design of the SE network itself or its quality is out of this study's scope.

To evaluate the precision and recall of NM-GAN for real noise samples, we adopt several quantitative measures [11–14] and perform two basic experiments, "GAN-train and GAN-test" [15]. In these trials, we calculate the distance between the generated samples and a real data manifold in terms of precision and recall. In the "GAN-train" and "GAN-test" comparative experiments, we utilize three different training sets. The first involves real-world recordings referred to as the "real" training set. The second training set is a synthesized set using only limited real noise samples, entitled the "limited" training set, and the third exploits our proposed method. The three training sets are used to train the same SE network, which is then evaluated on real noisy speech signals. The experimental results demonstrate that the SE model trained on our synthesized training set presents a robust SE effect on a real noisy speech, while the SE model trained on the real training set and the synthesized noisy speech presented a similar performance. Besides, the NM-GAN-prepared training sets better train the DNN-based SE models than the "limited" training sets, demonstrating our method's feasibility and reliability. Further details are presented in the comparative experiments carried out in Section 4.

1.3. Related Works

To the best of our knowledge, this work is the first to focus on GAN-based noise modeling to create training sets for speech enhancement. Although similar ideas appear in image enhancement, these are not applied in speech enhancement. In addition, although data augmentation, as a method of preparing training sets, has been widely used to process images and speech [16,17], it is still based on label preserving transformations. Data augmentation schemes typically deform existing training sets and expand them to create more training sets. This strategy enhances the adaptability of DNN-based networks and solves the problem of small or limited training sets. Moreover, data augmentation is commonly used in image recognition, where transformations such as translation, rotation, scaling, and reflection can significantly improve recognition accuracy [18]. However, speech enhancement tasks commonly involve limited numbers of noisy speech samples or no training set.

In conclusion, our research presents some originality in solving the DNN-based SE problem with a limited noisy speech dataset. The proposed method increases the robustness

of the SE model training when applied to actual speech and for blind speech denoising tasks when the noise type is unlabeled, and the clean speech signal is unavailable.

The remainder of this paper is organized as follows: Section 2 introduces the basic principles of GAN and the architectures that evolved from the original network. Section 3 analyzes the NM-GAN architecture and the proposed method for building paired training sets, while Section 4 presents the information regarding the noise modeling and evaluation experiments, together with the experimental results and their analysis. Finally, Section 5 concludes this work and suggests some future research directions.

2. GAN: Basic Principles and Evolved Architectures

This paper focuses on developing an advanced GAN for noise modeling and constructing paired training sets. In this chapter, we briefly present the basic principles associated with GANs and the most successful recent architectures.

In its original formulation, a GAN [4] comprises a pair of adversarial neural networks, namely 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 originates from a set of training data rather than G. The generator is trained to create samples that will try to fool the discriminator and an adversarial game is played between the two networks.

A GAN is used for data generation and given a dataset $\{(x_1, z_1), (x_2, z_2), (x_3, z_3) \cdots (x_N, z_N)\}$ comprising N signal pairs, a target signal x and a random signal z are generated. Learning the generator's distribution P_g involves finding a mapping $P_z(z)$: $z \rightarrow x$, to map the random input signal z to the target signal x. Conforming to the principle of a GAN, the adversarial learning process is formulated as a min-max game between G and D using the $V(D, G)$ function [4]:

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

where $G(z)$ is a generated sample from the learned generator distribution and D maximizes the probability of the correct label being assigned to both the training examples x and the samples from G. G and D are trained simultaneously [14], and we adjust parameters for G to minimize $log(1 - D(G(z)))$ and for D to minimize $logD(x)$. Ultimately, this process is a two-player min-max game with the value function $V(D, G)$.

In practice, Equation (1) may not provide a definitive way for G to learn effectively. Therefore, the Conditional Generative Adversarial Network (CGAN) [19] has been proposed, an extension of GAN that includes a conditional model where both the generator and discriminator are conditioned according to some extra information x_c. This extra information can be any kind of auxiliary information according to the different targets to be generated, avoiding needless data generation. Such conditioning could be based on class labels on some parts of the data for inpainting or different modalities. For a CGAN, the objective function [19] is:

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

Nevertheless, if G is ineffective during the learning process, D may reject samples with high confidence because they are different from the training data. In this case, $log(1 - D(G(z, x_c), x_c))$ becomes saturated. To solve this problem, the least-squares GAN (LSGAN) [20] employs a least-squares function with binary coding to replace the cross-entropy loss in the original GAN formulation. This helps to stabilize the training process and increases the quality of the generated samples in G. LSGAN's objective function is:

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

Although existing GANs are capable of generating large amounts of synthetic data, the NM-GAN structure presented below is especially well suited to noise generation for the following reasons:

(1) By drawing upon a combined U-Net and LSTM block, NM-GAN can learn complex noise distributions. The generator uses a symmetrical U-Net, where the encoder is a stack of convolutional and pooling layers. This strategy affords to extract high-level features from the input noise. Moreover, the decoder has the same structure as the encoder but, in reverse, effectively mapping the low-resolution feature maps from the encoder to full-size input noise feature maps. This network structure better extracts the input data characteristics than other GANs as it fully exploits the information of various features.

(2) NM-GAN operates end to end without involving any hand-crafted features and without considering explicit assumptions about raw noise.

(3) NM-GAN operates directly in the time domain, permitting integrated modeling of the phase information and considering contextual information.

3. Noise Modeling and Building the Training Sets

This chapter introduces the proposed NM-GAN and presents the construction of the paired training sets.

3.1. NM-GAN Structure

The developed NM-GAN structure learns to translate a random data sequence into a real noise sample. Its purpose is to generate noise samples that mimic real-world noise samples while remapping data z to real noise data x is a form of end-to-end mapping.

Similar to CGAN, NM-GAN considers that a Gaussian distribution governs noise. The conditioning is met by feeding x_c , $x_c \sim N(0,1)$ into the discriminator and to the generator as an additional input layer. The entire noise generation process is illustrated in Figure 2.

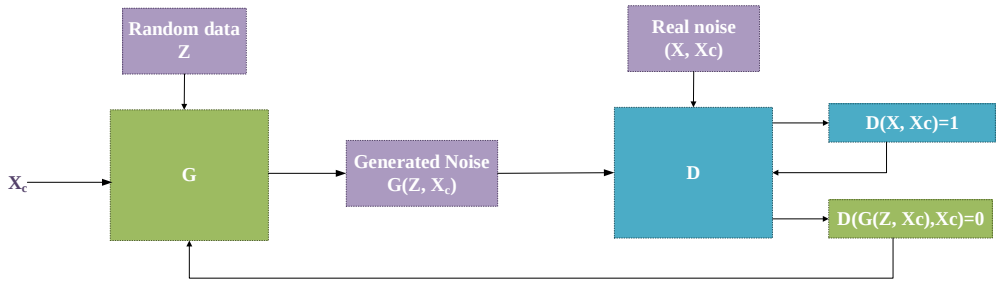

Figure 2. Noise generation process.

The detailed NM-GAN architecture is depicted in Figure 3. Specifically, the generator *G* relies on a symmetrical U-Net, where the encoder is a stack of convolutional and pooling layers extracting the high-level features from the input noise. The decoder has the encoder's reverse structure and maps the low-resolution feature maps from the encoder to full-size input noise feature maps. As the model's input and output share the same underlying structure, *G* contains skip connections that connect each encoding layer to its homologous decoding layer, enhancing the input data feature extraction process according to the various feature dimensions. Moreover, a bidirectional LSTM [3] at the middle of the encoder and decoder enables context information to be considered. The *G* network is convolutional, and thus only some nodes between any two adjacent layers are connected.

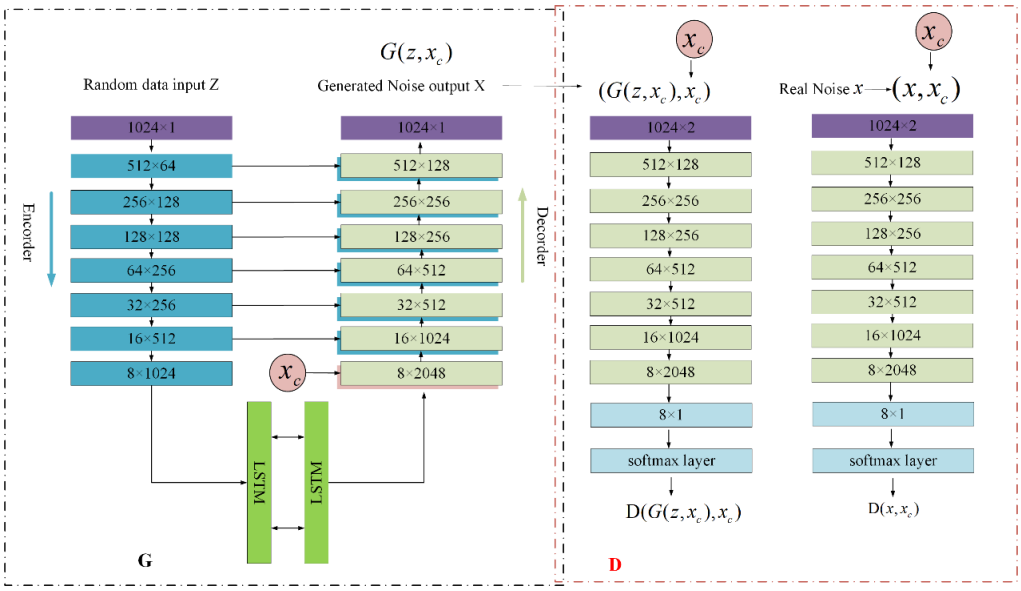

Figure 3. NM-GAN structure.

During the encoding stage, the input signal (1024 × 1) is projected and compressed through seven convolutional layers before being passed to a Batch Normalization and Parametric Rectified Linear Unit. The dimensions (sampling number × feature map) of each layer are 1024 × 1, 512 × 64, 256 × 128, 128 × 128, 64 × 256, 32 × 256, 16 × 512, and 8 × 1024, respectively. The latent feature vectors sequence in the encoder is modeled by two bidirectional LSTM layers, which do not change the data shape because the LSTM layer's output sequence is subsequently converted back to the original input shape by the decoder. Then, we add the extra information x_c to direct the data generation process, ultimately providing the encoding layer's output of dimension 8 × 2048.

The decoding process is the reversed encoding utilizing seven deconvolutions, followed by a Batch Normalization and Parametric Rectified Linear Unit. The dimensions (sampling number × feature map) of each decoding layer are 8 × 2048, 16 × 1024, 2 × 512,4 × 512,128 × 256, 256 × 256, 512 × 128, and 1024 × 1, respectively.

The *D* network's function is discriminating and classifying the real and the generated noise output by the *G* network. D has a similar architecture to the generator's encoder part but involves a two-channel input and uses a virtual batch-norm before activating a LeakyReLU with $\alpha = 0.2$. The dimensions (sampling number × feature map) per layer are 1024 × 2, 512 × 128, 256 × 256, 128 × 256, 64 × 512, 32 × 512, 16 × 1024, and 8 × 2048, respectively. In addition, *D* is topped up with a one-dimensional convolutional layer involving a filter of width one, i.e., a 1 × 1 convolution, to reduce the last convolutional output size from 8 × 1024 to eight features before classification at the softmax layer.

Our experiments indicated that adding a secondary component to the *G* network helps minimize the distance between the generated and the real examples. Moreover, we employ a least-square loss to replace the discriminator's cross-entropy loss and set $l_1 = 120$. Hence, the loss functions in *G* and *D* become:

$$\min_G V(D,G) = \frac{1}{2}E_{z\sim f_z(z),\ x_c\sim f(x_c)}\left[(D(G(z,x_c),x_c)-1)^2\right] + 120\times\|G(z,x_c)-x\|_1 \quad (4)$$

$$\min_D V(D,G) = \frac{1}{2}E_{x,\ x_c\sim f(x_c)}\left[(D(x,x_c)-1)^2\right] + \frac{1}{2}E_{z\sim f_z(z),\ x_c\sim f(x_c)}\left[D(G(z,x_c),x_c)^2\right] \quad (5)$$

where $f(x_c) \sim N(0,1)$ is the conditional information, and z and x are 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 its outputs are imitative noise samples $G(z,x_c)$. The l_1 distance between the clean sample x and the generated sample $G(z,x_c)$ is $\|G(z,x_c)-x\|_1$,

encouraging G to generate more fine-grained and realistic results [21–23]. Additionally, D learns to classify the $(G(z, x_c), x_c)$ pair as real and the (x, x_c) pair as fake, while G tries to fool D such that D classifies the $(G(z, x_c), x_c)$ pair as real. Over time, the output samples of G become increasingly like the real noise samples, x.

3.2. Building the Training Sets

The overall process involved in building training sets from noisy real-world inputs is illustrated in Figure 4. The first step is to extract noise samples from noisy speech signals to establish a noise dataset. After that, NM-GAN is trained to learn the noise distribution and generate many noise samples, followed by noisy speech synthesis.

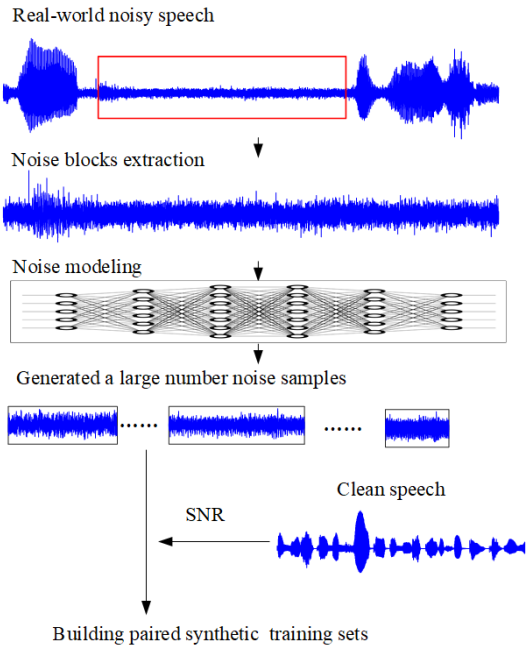

Figure 4. Training set construction process.

3.2.1. Noise Dataset

Before building the paired training dataset, a set of approximate noise blocks (or patches) must be extracted from the given noisy speech recordings. In this way, the noise distribution becomes the principal learning objective for the NM-GAN, enhancing its accuracy. Thus, a Speech Endpoint Detection Method [24] can extract the noise segments from the noisy speech.

3.2.2. Noise Modeling

The NM-GAN is trained using the method proposed in Section 3.1 with real noise samples. During training, the constructed data is segmented frame by frame (1024 points, 50% overlap) before being synthesized. The learning rate for G and D is set to 0.0002, and within a minibatch, all training samples are padded with zeros to ensure 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 according to the loss measurement. Once this process is complete, the last-obtained model is further fine-tuned, with the batch size doubled, and the learning rate lowered to 0.00001. This is re-iterated until 50 epochs without any improvement in the validation loss. Finally, the model with the best validation loss is selected. In this way, the generator can generate countless noise samples.

3.2.3. Noisy Speech Synthesis

Assuming that additive noise can create noisy speech, we create a noisy speech database by adding noise to clean speech at real Signal to Noise Ratio (SNR) levels. In this

way, we overcome the data scarcity problem mentioned above by synthesizing pure-noisy speech pairs comprising pure and realistic synthesized noisy speech.

4. Experiments and Discussion

In order to compare our noise modeling-based synthesized datasets against finite noise synthesis datasets, we adopt quantitative measures and employ the "GAN-train", and "GAN-test" approaches to judge the competitor methods. "GAN-train" and "GAN-test" were first proposed by Shmelkov to measure the quality of data generated by GANs by calculating the distance between the generated samples and a real data manifold in terms of precision and recall [15]. The "GAN-train" refers to training the SE model on our synthetic training sets and evaluating its performance on real noisy samples. The synthetic samples are considered sufficiently diverse if an SE model trained on them can provide appealing noise reduction on a real noisy speech. The "GAN-test" utilizes an SE model trained on real training sets but tested on synthetic samples. Here, if an SE model trained on a real training set can provide appealing noise reduction on a noisy synthetic speech, the generated samples are considered a realistic approximation of the natural samples' (unknown) distribution. The "GAN-train" and "GAN-test" experiments aim to measure the diversity and similarity between synthetic and real noisy samples and to assess the extent to which NM-GAN can match real noise distributions.

The experimental setup involves two groups of comparative "GAN-train" and "GAN-test" experiments. In the first "GAN- train" experimental group, the SE model is trained on our synthetic training set but tested on real noisy samples. In the first "GAN- test" experimental group, the SE model is trained on a real training set but tested on our synthesized noisy samples. In contrast, in the second group of "GAN-train" experiments, we employ the "limited" training set based on the direct synthesis of speech and limited real noise samples to train the same SE model and then test it on the real noisy set. Similarly, for the second group of "GAN-test" experiments, the SE model is trained on the real training set and tested on the "limited" set.

In order to perform the two "GAN-train" and "GAN-test" comparative experiments, we prepare three different training sets. The first is a real-world recording, referred to as the real training set. The second training set is a synthesized set using limited real noise samples, referred to as the "limited" training set, while the third is created using our proposed method. In the first group of "GAN-train" experiments, an SE model is trained on our synthesized training set but tested on a set of real noisy samples. The preparation method of the three datasets is illustrated in Figure 5.

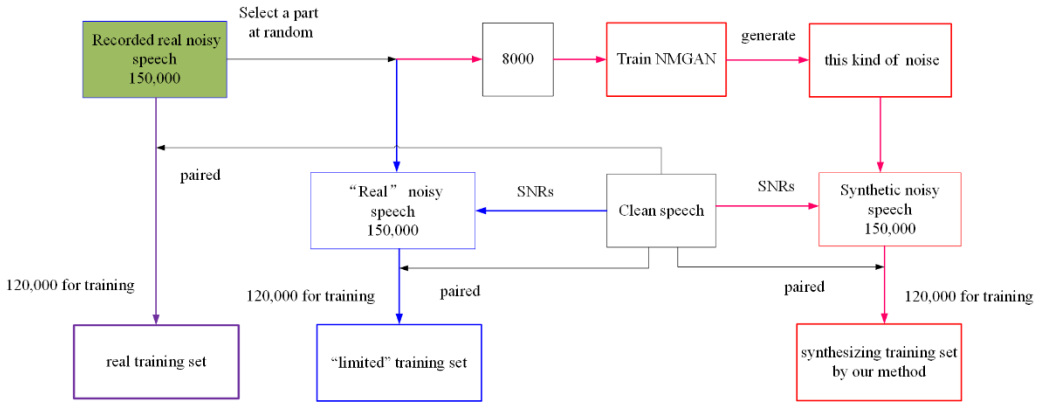

Figure 5. Preparation method of the three datasets.

The noisy speech recordings of the real training set are undertaken in a laboratory with almost no external interference. The laboratory's volume is 80 m^3 with a reverberation time of 0.45~0.6 s (500 Hz). Two loudspeakers are placed in the same horizontal position, where one played a pure voice and the other played noise. A recorder 0.5 m away from each loudspeaker recorded the noisy voice. For the entire process, apart from the speakers

playing the voice and the noise, the entire laboratory is quiet without interference from other sound sources. Moreover, we adjust both loudspeakers' volumes to obtain noisy speech with different SNRs (−5, 0, and 5 dB), providing 150,000 real noisy speech samples under these SNR levels. We randomly select 8000 noisy samples from each SNR, which with the 150,000 pure speech samples from the "limited" noisy speech for each SNR (one noise sample is reused), and from this we construct the "limited" training set. At the same time, the 8000 selected samples are used to train NM-GAN to generate 150,000 noise samples, which are then synthesized with pure speech according to the different SNRs. Ultimately, we create three datasets, each containing 150,000 sample pairs for each SNR. We split 80% of the dataset into a training set, and the remaining 20% forms the test set. That is, 120,000 pairs of the three datasets per SNR are used to train the same SE model, with the remaining pairs used for testing. Additionally, we employ the CSTR VCTK Corpus [25] and Demand database [26] as the original pure speech and noise datasets for the recordings because these datasets are publicly available. Finally, we adopt the SEGAN [5] model as the SE model for training and testing because its code is publicly available and is widely used in current research.

To compare the quality of the speech enhancement, we use the following objective evaluation metrics:

PESQ [27]: Perceptual evaluation of speech quality (from −0.5 to 4.5).
CSIG [28]: MOS prediction of the signal distortion (from 1 to 5).
CBAK [28]: MOS prediction of the background noise intrusiveness (from 1 to 5).
COVL [28]: MOS prediction of the overall effect (from 1 to 5).
SNR [27]: Signal-to-Noise Ratio (from 0 to ∞).

In our experiments, the results are calculated by comparing the enhanced signal against the pure signal. In order to maximize the accuracy of the experimental results and reduce any possible errors, all metrics are based on an average of 30 groups of speech signals. Table 1 presents the results for the first "GAN-train" and "GAN-test" experimental group, while Table 2 shows the corresponding results of the second group. From Tables 1 and 2, we observe the following:

Table 1. Evaluation results for the first group of "GAN-train" and "GAN-test" experiments.

Trained Model/SNR/Test on \ Metric			PESQ	CSIG	CBAK	COVL	SNR
GAN-train (SE trained on our synthetic training set)	−5 dB	Synthetic noisy by NM-GAN	1.35	2.81	2.90	2.80	3.67
		Real noisy	1.09	2.76	2.72	2.29	3.11
	0 dB	Synthetic noisy by NM-GAN	2.52	3.15	3.27	2.93	8.15
		Real noisy	2.32	3.15	3.21	2.63	7.94
	5 dB	Synthetic noisy by NM-GAN	2.60	3.79	3.72	3.25	7.12
		Real noisy	2.45	3.56	3.47	3.16	6.88
GAN-test (SE trained on real training set)	−5 dB	Synthetic noisy by NM-GAN	1.01	2.26	2.32	2.19	3.07
		Real noisy	1.04	2.32	2.41	2.25	3.18
	0 dB	Synthetic noisy by NM-GAN	2.15	2.85	3.18	2.14	7.36
		Real noisy	2.21	3.04	3.17	2.21	7.40
	5 dB	Synthetic noisy by NM-GAN	2.03	3.12	3.23	3.23	7.82
		Real noisy	2.05	3.17	3.29	3.37	8.00

Table 2. Evaluation results for the second group of "GAN-train" and "GAN-test" experiments.

Trained Model/SNR/Test on \ Metric			PESQ	CSIG	CBAK	COVL	SNR
GAN-train (SE trained on "limited" training set)	−5 dB	"Limited" Synthetic noisy	1.47	2.93	2.88	3.06	5.01
		Real noisy	1.01	2.33	2.14	1.74	1.89
	0 dB	"Limited" Synthetic noisy	2.92	3.37	3.41	3.00	9.83
		Real noisy	2.24	2.09	2.15	1.92	5.88
	5 dB	"Limited" Synthetic noisy	2.75	3.82	3.90	3.81	11.12
		Real noisy	2.01	2.74	2.31	2.05	6.16
GAN-test (SE trained on real training set)	−5 dB	"Limited" Synthetic noisy	1.10	2.26	2.40	2.17	3.03
		Real noisy	1.04	2.32	2.41	2.25	3.18
	0 dB	"Limited" Synthetic noisy	2.07	3.11	3.24	2.15	7.28
		Real noisy	2.21	3.04	3.17	2.21	7.40
	5 dB	"Limited" Synthetic noisy	2.02	3.10	3.26	3.40	7.93
		Real noisy	2.05	3.17	3.29	3.37	8.00

(1) The "GAN-train" results in Table 1 reveal that the PESQ differences are overall 0.2. For the CSIG, CBAK, and COVL metrics, the average differences are 0.09, 0.16, and 0.3, respectively. With regard to the SNR improvement, the performance differences are 0.56, 0.21 and 0.24, for −5 dB, 0 dB and 5 dB, respectively. Considering all these results, we conclude that the SE model trained on the NM-GAN synthetic training sets provides appealing noise reduction levels for real noisy speech signals. Note that the noise and speech are synthesized through direct addition, so the noisy synthetic speech has some differences from the real noisy speech. As a result, the SE network trained on the synthetic training set achieves a slightly different noise reduction effect for the real noisy speech. Nonetheless, these results are generally encouraging despite the slightly poorer performance at the lowest SNR.

In the "GAN-test" experiment, where the SE model is trained on the real training set, the enhancement scores for the synthetic and real noisy speech test sets are almost the same. The largest difference in PESQ is only 0.06 for 5 dB. For the CSIG, CBAK, and COVL metrics, the average differences are 0.1, 0.05, and 0.09, respectively, while for the SNR, the average difference is only 0.11. These results confirm that the generated noise is similar to the real noise.

Hence, from the first "GAN-train" and "GAN-test" experimental group, we conclude that the SE model trained on our synthesized training set presents a robust SE effect for real noisy speech and that the SE model trained on the real training set achieves a similar SE effect for synthesized noisy speech. This proves that NM-GAN learned effectively from the input noise samples. Overall, NM-GAN can produce similar noise distributions that diversify as real noise.

(2) From Table 2, we observe that the SE model trained on the real training sets provides consistent noise reduction levels for the synthetic "limited" noisy speech. However, when the SE is trained on the "limited" training set, the enhancement scores for the real noisy speech test set are not ideal. Comparing the PESQ metrics, the enhancement scores are 0.46, 0.68, and 0.74 higher for −5 dB, 0 dB, and 5 dB, respectively. The biggest performance increase for the CSIG, COVL, and CBAK metrics is 1.28, 1.59, and 1.76, respectively. Moreover, for the SNR metric, the average enhancement score is 4.01 higher when the developed model is tested on the synthetic "limited" noisy speech than on real noisy recordings. These experimental results highlight that an SE model trained on a "limited" training set is not robust to real noisy speech samples. We suspect that when an SE is trained on the "limited" training set using limited noise samples, the network may remember the

specific noise waveforms, and thus its performance is less satisfactory for noisy speech with inconsistent noise waveforms.

(3) Comparing Tables 1 and 2 reveals that the SE model trained on our synthetic set affords a better effect on noise suppression for real situations. For the PESQ metrics, the model trained on our synthetic training set and tested on real noisy recording achieves a score of 0.2 higher than the model trained using the "limited" training set. For the CSIG, COVL, and CBAK metrics, the scores are 0.77, 0.93, and 0.79 higher, respectively, while the maximum SNR difference is 2 dB higher. From the two "GAN-train" and "GAN-test" experimental groups, we conclude that the NM-GAN-prepared training sets can better train DNN-based SE models than the "limited" training sets synthesized directly from limited noise samples.

5. Conclusions

5.1. Key Outcomes

This paper highlights the problem of inadequate real training sets for training DNN-based SE models that limit the model's effectiveness and robustness. Hence, we propose a noise modeling method that creates realistic paired training sets. The main advantages of our approach are that of NM-GAN:

- Offers high precision, with the generated samples being close to the real noise manifold.
- Achieves high recall by generating "diverse" noise that does not exist in limited captured noisy speech.
- Provides synthetic training sets to train SE models that enhance effectively similar types of noisy speech, even if the noise contained in other noisy speech recordings is a little different.
- Operates end to end, not requiring hand-crafted features and not making explicit assumptions about the raw noise.
- Can learn complex noise distributions by combining U-Net and LSTM to provide the generator with powerful modeling capabilities.
- Can operate directly in the time domain, permitting integrated modeling of the phase information and considering context information.

In this work, we conduct two comparative "GAN-train" and "GAN-test" experiments to assess the proposed method's viability. In the first "GAN-train" experimental group, we verify that an SE model trained using synthetic training sets generated by the NM-GAN architecture achieves good noise reduction for real noisy speech signals. In the second "GAN-train" experiment, we find that an SE model trained using synthetic training sets generated by the NM-GAN architecture achieves an improved performance on real noisy speech samples than trained on synthesized "limited" real data with a small number of noise samples. Finally, we prove that the proposed method for generating noise and preparing training sets to train DNN-based SE models is feasible.

The proposed method increases the robustness of the SE model training involving actual speech and can also be used for blind speech denoising when the noise type is not labeled, and the clean speech signal is unavailable. Beyond this, the proposed method is effective for noise calibration of sound equipment and in special environments.

5.2. Limitations and Future Work

In our experiments, we employ the "GAN-train" and "GAN-test" performance as the criteria to judge the effectiveness of the training sets preparation. However, this method indirectly verifies the NM-GAN performance. In the future, we will explore using methods that directly evaluate the quality of the generated noise.

Second, we only used one SE model as a tool to verify our method's feasibility. Future work shall test applying the prepared datasets to more network structures. Furthermore, the number of noise samples used for NM-GAN may impact the experimental results, and therefore further in-depth research is required.

Third, given more recent public corpora, such as MUSAN, we need to test different noise types when verifying the NM-GAN's performance.

An additional limitation that should be considered is that noise is currently assumed to be additive. Beyond this, further research is still required regarding different speech synthesis methods. Although NM-GAN has a strong ability to learn noise distributions, if the synthesis method is not correct, our method's applicability in real situations is limited. Other noise and pure speech synthesis methods, such as parallel model combination (PMC), could be explored to investigate whether they provide additional advantages.

Author Contributions: Conceptualization, Y.W. (Yongbiao Wang); Data curation, X.K.; Funding acquisition, H.Z.; Investigation, Z.W.; Methodology, Y.W. (Yahui Wang); Project administration, W.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No: 62071057).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository that does not issue DOIs. Publicly available datasets (CSTR VCTK Corpus, reference number [25] and Demand database, reference number [26]) were analyzed in this study. The CSTR VCTK Corpus (accessed on 5 June 2019) can be found here: [https://datashare.ed.ac.uk/handle/10283/2651], and the Demand database (accessed on 5 June 2019) can be found here: [https://zenodo.org/record/1227121#.YgZJN3b5BzR].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Loizou, P.C. *Speech Enhancement: Theory and Practice*; CRC Press: Los Angeles, CA, USA, 2017.
2. Miao, Y.; Gowayyed, M.; Metze, F. EESEN: End-to-End Speech Recognition using Deep RNN Models and WFST-based Decoding. In Proceedings of the 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), Scottsdale, AZ, USA, 13–17 December 2015.
3. Sun, L.; Du, J.; Dai, L.R.; Lee, C.H. Multiple-target deep learning for LSTM-RNN based speech enhancement. In Proceedings of the Hands-Free Speech Communications and Microphone Arrays (HSCMA), San Francisco, CA, USA, 1–3 March 2017.
4. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *Adv. Neural Inf. Process. Syst.* **2014**, *3*, 2672–2680. [CrossRef]
5. Pascual, S.; Bonafonte, A.; Serrà, J. SEGAN: Speech Enhancement Generative Adversarial Network. In Proceedings of the INTERSPEECH 2017, Stockholm, Sweden, 20–24 August 2017.
6. Jansson, A.; Eric, J.; Humphrey, N. 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, 23–27 October 2017.
7. Ernst, O.; Chazan, S.E.; Gannot, S.; Goldberger, J. Speech dereverberation using fully convolutional networks. In Proceedings of the 2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, 3–7 September 2018.
8. Rethage, D.; Pons, J.; Serra, X. A Wavenet for Speech Denoising. In Proceedings of the ICASSP IEEE International Conference on Acoustics, Seoul, Korea, 22–27 April 2018.
9. Reddy, C.; Beyrami, E.; Dubey, H.; Gopal, V.; Cheng, R.; Cutler, R.; Matusevych, S.; Aichner, R.; Aazami, A.; Braun, S. The INTERSPEECH 2020 Deep Noise Suppression Challenge: Datasets, Subjective Speech Quality and Testing Framework. *arXiv* **2020**, arXiv:2005.13981.
10. Chen, J.; Chen, J.; Chao, H.; Ming, Y. Image Blind Denoising with Generative Adversarial Network Based Noise Modeling. In Proceedings of the IEEE/CVF Conference on Computer Vision & Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
11. 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. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6629–6640.
12. Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive growing of GANs for improved quality, stability, and variation. *arXiv* **2017**, arXiv:1710.10196.
13. Lucic, M.; Kurach, K.; Michalski, M.; Gelly, S.; Bousquet, O. Are GANs created equal? A large-scale study. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 698–707.
14. Salimans, T.; Goodfellow, I.; Zaremba, W. Improved techniques for training GANs. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 2234–2242.
15. Shmelkov, K.; Schmid, C.; Alahari, K. How good is my GAN? In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
16. Jaitly, N.; Hinton, G.E. Vocal tract length perturbation (VTLP) improves speech recognition. In Proceedings of the 30th International Conference on Machine Learning (ICML), Atlanta, GA, USA, 16–21 January 2013.

17. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Neural Inf. Process. Syst.* **2012**, *25*, 1106–1114. [CrossRef]
18. Cui, X.; Goel, V.; Kingsbury, B. Data augmentation for deep convolutional neural network acoustic modeling. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, France, 14 July 2014.
19. Mirza, M.; Osindero, S. Conditional Generative Adversarial Nets. *Comput. Sci.* **2014**, 2672–2680, arXiv:1411.1784
20. Mao, X.; Li, Q.; Xie, H.; Lau, R.Y.K.; Smolley, S.P. Least squares generative adversarial networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, France, 22–29 October 2017.
21. Baby, D.; Verhulst, S. Sergan: Speech Enhancement Using Relativistic Generative Adversarial Networks with Gradient Penalty. In Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019.
22. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5967–5976.
23. Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; Efros, A.A. Context encoders: Feature learning by inpainting. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2536–2544.
24. Burgy, L.; Consel, C.; Latry, F.; Lawall, J.L.; Palix, N.; Réveillère, L. Speech/non-speech classification using multiple features for robust endpoint detection. In Proceedings of the 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing, Proceedings (Cat. No. 00CH37100), Washington, DC, USA, 5–9 June 2000.
25. Veaux, C.; Yamagishi, J.; King, S. The voice bank corpus: Design, collection and data analysis of a large regional accent speech database. In Proceedings of the 2013 International Conference Oriental COCOSDA Held Jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE), Gurgaon, India, 25–27 November 2013.
26. Thiemann, J.; Ito, N.; Vincent, E. The diverse environments multi-channel acoustic noise database: A database of multichannel environmental noise recordings. *J. Acoust. Soc. Am.* **2013**, *133*, 3591. [CrossRef]
27. Quackenbush, S.R.; Barnwell, T.P.; Clements, M.A. *Objective Measures of Speech Quality*; Prentice-Hall: Hoboken, NJ, USA, 1988.
28. Viswanathan, M.; Viswanathan, M. Measuring speech quality for text-to-speech systems: Development and assessment of a modified mean opinion score (MOS) scale. *Comput. Speech Lang.* **2005**, *19*, 55–83. [CrossRef]