SPEECH DENOISING BY LISTENING TO NOISE

Anonymous authors
Paper under double-blind review

ABSTRACT

Speech denoising is the task of obtaining clean speech from the speech signal corrupted by background noise. Except in high end recording studios, we do not get clean speech signal as some background noise, or noise due to the recording device is always present. We propose an approach to denoise noisy speech signal by modeling the noise explicitly. Existing approaches model speech, potentially of multiple speakers, for denoising. Such approaches have an inherent drawback as a separate model is required for each speaker. We show that instead of modeling speaker(s), modelling the noise helps obtain a unified speaker independent denoiser, cf. speaker dependent ones in existing popular approaches. In addition to a novel speech denoising network, we also propose a large scale noise dataset, `AudioNoiseSet`, derived from Audioset dataset, to train our model. We show that our model outperforms prior approaches by significant margin in a large scale, in the wild speech datasets, *i.e.* AVspeech, with standard quantitative metrics. In addition we show with multiple human ratings that the method is preferred over state-of-the-art approaches. The user study also points towards limitations of the metrics used, which we discuss. We also provide many qualitative results to demonstrate our better results.

1 INTRODUCTION

Speech signals often contain noise which is introduced either due to environmental conditions such as honking, crowd *etc.* or get corrupted during recording and transmission of the signal. The quality of the speech signal not only impacts the intelligibility for human listeners but also many downstream machine learning tasks such as automatic speech recognition, speech enhancement, speaker identification, speech emotion recognition, language detection *etc.* While humans have a remarkable ability to interpret speech even with heavy background noise (Kell & McDermott, 2019; King & Walker, 2020), the performance of downstream machine learning tasks is highly dependant on the quality of input speech signal. Therefore, to aid interpretation by humans as well as downstream machine learning tasks, speech denoising *i.e.* extracting intelligible speech signal from observed noisy speech signal is an important problem. To this end, we propose an end to end deep learning network to extract clean speech from noisy speech audio.

We aim at denoising speech signal with high noise interference. Prior works (Park & Lee, 2016; Rethage et al., 2018; Luo & Mesgarani, 2019; Pascual et al., 2017) utilize encoder-decoder style networks that take a noisy audio as input and output a clean audio in frequency or time domain. In general, such methods are designed to learn the properties of speech signal, and isolating it from the overall noisy signal. Unlike earlier works, we hypothesize that directly learning to model a noise signal, and disentangling it from the overall noisy signal to predict a clean speech signal, is a more practical approach to speech denoising.

Recent approaches to speech denoising (Yang et al., 2022; Fang et al., 2021; Xiang et al., 2022) either explicitly model clean speech for individual speakers or tend to model general representation of a speech signal. However, speaker dependent modelling impacts the generalization ability of such methods. Further, high variation in standards for intelligibility of speech varies with speaker, language, culture, emotion and perception (Jacewicz et al., 2010; Gupta, 2021; Puglisi et al., 2021; Winn & Teece, 2021), which makes generalized modelling of a speech signal challenging. To overcome this, instead of attempting to model speech, we propose to model the noise and estimate noise signal from the noisy audio, which we then utilize to obtain the denoised speech signal.

The output from methods directly modelling speech signals usually have residual noise in the predicted speech output, indicating a correlation between the speech and the residual signal. Earlier attempts at exploiting this correlation involve explicitly disentangling additive noise from noisy speech inputs (Odelowo & Anderson, 2018; Xu et al., 2020). However, these methods suffer from low-SNR speech outputs and contain some amount of residual noise in the prediction. Jointly modelling speech and noise signals (Zheng et al., 2021) often results in losing latent information leading to poor performance on downstream tasks (Hu et al., 2022). This motivates us towards the need of a more robust modelling of noise signal.

We work with melspectrogram representation of audio, as it has been found beneficial in many audio tasks and its inherent representation of fine grained representation of lower frequencies makes it perceptually closer to human hearing. We use a deep neural network to estimate a model for noise. We propose to use feature disentanglement to obtain features of clean audio from the features of noisy/mixed audio. Our method first extracts the noise features from the mixed signal. It also embeds the mixed signal in the same feature space. The clean signal features are then obtained by subtracting the projection of the noise features on the mixed feature vector, from the mixed feature vector. The resulting vector is the clean speech feature, which we use to estimate the clean speech signal, by first using a network to predict the melspectrogram, followed by a pre-trained vocoder to predict the audio waveform.

To train our network, we need a dataset of noise audio samples which do not contain speech. Hence, in addition to the novel network, we also propose a large scale in-the-wild noise dataset, *i.e.* `AudioNoiseSet`, curated from the Audioset (Gemmeke et al., 2017) dataset. Prior works (Xu et al., 2020; Gao & Grauman, 2021; Ephrat et al., 2018), have used a subset of Audioset dataset (after removing the examples labelled as `speech` in the dataset) as background noise. But on manual observation we found that there are a significant number of samples that contain speech signal even if they are not labelled as speech. Using such samples as noise harms the training of denoising network greatly. We filter out samples which are not labeled as speech, but do contain speech, and obtain a relatively cleaner large scale in-the-wild noise dataset. We will make the dataset publicly available upon acceptance and hope it will be an useful resource for the community.

In summary, we make the following contributions.

- We propose a novel method for speech denoising, which works by explicit noise modeling and removing the noise from the mixed signal.
- We propose a large scale in-the-wild noise dataset, curated from Audioset dataset, for training and evaluation of speech denoising tasks.
- We show with quantitative results that the proposed method obtains better results than current state of the art methods.
- We provide a user study which also shows that our proposed method is perceived to give better quality results when postprocessed with a state of the art enhancement method, while the existing methods fail on many high noise inputs.
- We also show quantitatively that our noise modelling network is powerful enough and can classify environmental sounds at par with the best performing methods, after finetuning for a small number iterations.

2 Related Works

Speech denoising (Benesty et al., 2006; Loizou, 2007) has been a long standing problem in the area of audio processing. A wide range of approaches has been proposed using both deep learning and non-deep learning based techniques. Prior to the deep learning era, spectral subtraction (Boll, 1979; Kamath et al., 2002; Vaseghi, 1996) was one of the initial methods for the speech denoising. Here, the noise profile is first estimated and then subtracted from the mixed signal to get the final denoised output. This approach has also been extended to multi-channel audio (Furuya & Kataoka, 2007; Meyer & Simmer, 1997; Miyazaki et al., 2014) and other related tasks such as dereverberation (Wang et al., 2012; Lebart et al., 2001; Zhang et al., 2014). In another line of approach wiener filtering (Wiener et al., 1949) is used for speech denoising (Lim & Oppenheim, 1978; Sreenivas & Kirnapure, 1996; Almajai & Milner, 2010; Lin et al., 2002), where the mean squared error between the clean speech and reconstructed speech is minimized.

The recent groundbreaking success of deep learning for various visual tasks (Krizhevsky et al., 2012) have prompted researchers to apply the same for various audio domain tasks as well. Specifically for the task of audio denoising (Park & Lee, 2016; Pascual et al., 2017; Xu et al., 2017; Qian et al., 2017; Rethage et al., 2018; Luo & Mesgarani, 2019; Pandey & Wang, 2019), the network follows the standard setting of supervised learning where the system has access to both the degraded audio and the corresponding clean audio. In most of the approaches (Park & Lee, 2016; Rethage et al., 2018; Luo & Mesgarani, 2019) the network follows as encoder-decoder framework where the network takes the degarded/mixed audio as input and is tasked to predict the clean audio as output. Inspired by the recent generative models for audio (Oord et al., 2016; Vasquez & Lewis, 2019; Kumar et al., 2019), different variants of generative architectures (Qian et al., 2017) are proposed for the task of denoising. Further to improve the quality of the denoised audio, several approaches (Pascual et al., 2017; Liu et al., 2022; Su et al., 2020) have used adversarial losses along with reconstruction loss for training denoising task.

Along with the reconstruction loss, inspired by recent research showing similarities between feature representation in deep neural network and human brain (Yamins & DiCarlo, 2016; Kell & McDermott, 2019), researchers have used perceptual loss where the intermediate features between the clean audio and predicted audio obtained from different layers of the network is minimized (Saddler et al., 2020; Germain et al., 2018; Su et al., 2020; Hsieh et al., 2020; Kataria et al., 2021). Similarly, taking another inspiration from the area of image denoising, where the task of denoising is performed without having a clean image (Lehtinen et al., 2018), authors have applied similar idea in the area of speech denoising (Alamdari et al., 2021; Kashyap et al., 2021) as well. Here the task of speech denoising is handled in a completely unsupervised manner without having the access clean ground truth audio.

The other generative model VAE (Kingma & Welling, 2013) has also been used for the task of speech denoising. In VAE based approach for audio denoising (Fang et al., 2021; Xiang et al., 2022), a two step approach is followed where in the first step an encoder decoder model is used for learning the parameters of the audio from clean audio. In the second step, a mapping function is learnt to get the parameters of the clean audio from the noisy one and once the parameters are obtained, the decoder trained in the first step is used for getting the corresponding clean audio. Although the methods were trained for speaker independent cases, but it was shown recently (Yang et al., 2022) that these class of methods give improved performance if individual models are trained for each speaker. Our proposed approach is the inverse of these approach where instead of modelling the audio we model the noise and then estimate the noise from the mixed signal. Once the noise is estimated we use both the estimated noise and the mixed signal to onatin the denoised output.

3 Approach

Problem Formulation We aim to obtain a clean speech signal from a noisy/mixed speech signal. The mixed signal can be considered as a mixture of clean speech signal and a noise signal. Let $\mathbf{a_i} \in \mathbb{R}^t$, $\mathbf{n} \in \mathbb{R}^t$, $\mathbf{a} \in \mathbb{R}^t$ be the input clean audio, noise and the mixed signals respectively and t be the length of each signal. Let $P_{\mathbf{a_i}}$ and $P_{\mathbf{n}}$ represent the power of input clean audio and noise signal respectively and S be the SNR of the mixed signal. The mixed signal $\mathbf{a}$ is then given by,

$$\mathbf{a} = \mathbf{a_i} + \frac{\mathbf{n}}{r} \quad \text{where,} \quad r = \sqrt{P_{\mathbf{a_i}}} \Bigg/ \sqrt{\frac{P_{\mathbf{n}}}{10^{\frac{S}{10}}}} \tag{1}$$

Our objective is to predict the clean signal $\mathbf{a_i}$ from the mixed signal $\mathbf{a}$, given just the mixed signal.

3.1 Overview

The detailed architecture of our approach is shown in Fig. 1. It consists of three components: (i) noise modelling, (ii) audio denoising, and (iii) vocoder. We learn the network in steps. In the first step, we model the noise signal only. We use a Vector Quantized VAE (VQVAE) (**?**) model to learn quantized representation of the noise giving only the pure noise signal as the input. In the second step, we use the trained noise network to get the noise signal from the mixed signal. We then use the estimated noise and the mixed signal to obtain the denoised feature, using the disentanglement network. The denoised feature is further decoded into melspectrogram representation using a recurrent

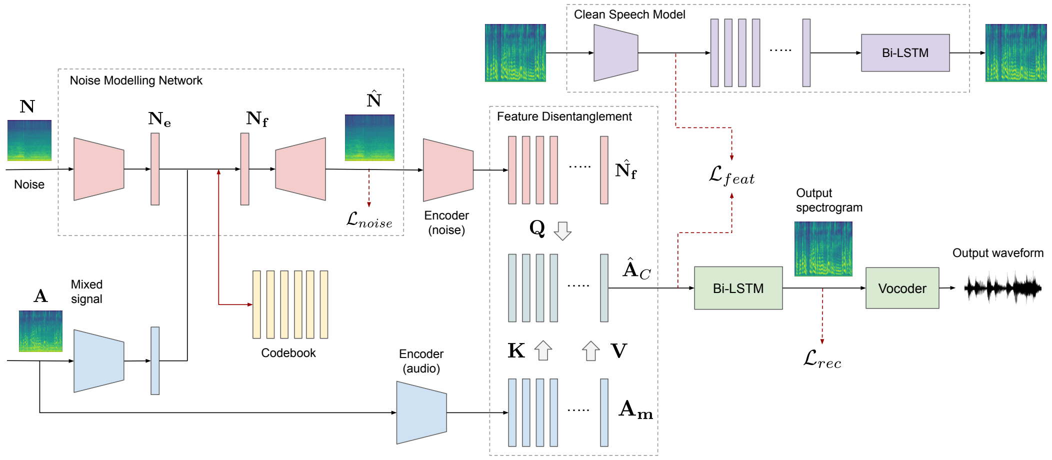


Figure 1: **Proposed network and approach.** The proposed network consists of a noise modeling network (VQ-VAE), a clean audio model (recurrent encoder decoder) and a feature disentanglement network (cross-attention inspired network). The noise and clean speech modeling networks are first pretrained on noise and clean speech samples respectively, and then fixed. While training to denoise, the mixed signal is first passed through the noise modeling network to obtain noise melspectrogram, which is encoded in a feature space by an encoder for noise. The mixed signal melspectrogram is also encoded into the same feature space by another encoder. The disentanglement network then separates the noise from the mixed feature to obtain the clean feature. The clean feature is further passed through a recurrent decoder followed by a vocoder to obtain clean speech waveform.

decoder, which is eventually converted to the clean audio via a pre-trained vocoder. We describe in detail, each of the network components in the following sections.

3.2 Noise Modelling

The noise modelling component of the network is inspired from the recent VQ-VAE network (Van Den Oord et al., 2017), which has been successfully applied for quantized representation of images. It follows a two-step approach which involves learning a discrete representation of the data and subsequently the same data is reconstructed by sampling from a distribution with discrete representation as its parameters.

We follow a similar approach: in the first stage we learn quantized representation of noise signal and then in the next stage, we learn a mapping from the noisy signal to its corresponding quantized noise representation. We describe each stage below in detail.

In the first stage of noise only modelling, we use an encoder-decoder architecture which takes noise as input and reconstructs the same noise as output. Let $\mathbf{n} \in \mathbb{R}^t$ be the noise signal, where t is the length of the signal, and $\mathbf{N} \in R^{T \times F}$ be its melspectrogram representation, where T is the number of time window and F is the number of mel-frequency bins in the signal. We feed the mel-spectrogram representation of the signal $\mathbf{N}$ to the encoder part of this network and obtain a representation $\mathbf{N_e} \in R^{M \times K}$, where M is a compact time window representation as compressed by the encoder and K is the feature dimension for each time window. We then obtain discrete representation $\mathbf{N_f} \in R^{M \times N}$ from the encoded representation $\mathbf{N_e}$ using the codebook, $\mathbf{C} \in \mathbb{R}^{K \times N}$. The codebook can be considered as the compact representation of all noise signals. We obtain the discrete representation individually for each time window by sampling from Gumbel-softmax distribution (Jang et al., 2016) and then selecting corresponding index from codebook, *i.e.* $\mathbf{N_f}[i,:] = \mathbf{C}[j,:]$ where $j = \text{Gumbel-Softmax}(\mathbf{N_e}[i,:])$.

We then use the discrete representation $\mathbf{N_f}$ to get back the original noise signal. Denoting $\hat{\mathbf{N}}$ as the reconstructed signal from the decoder, we minimize the mean squared error between $\mathbf{N}$ and $\hat{\mathbf{N}}$,

$$\mathcal{L}_{noise} = \|\mathbf{N} - \hat{\mathbf{N}}\|_1, \tag{2}$$

to obtain encoder, decoder and codebook jointly.

Once the noise only model is learnt, we fix the decoder and codebook and learn the encoder only for the noisy audio. We learn the mapping from the noisy audio to the noise signal using the learnt decoder and codebook obtained in the previous step. This can be considered as a regression between the noisy audio to the discrete representation of the noise it contains.

3.3 Speech Denoising

The speech denoising network comprises of two parts; (i) obtaining the noise profile from the noisy audio, and (ii) denoising the noisy audio using both the noise profile and noisy audio using an encoder-decoder architecture.

In the first part of the network, our goal is to extract noise from the noisy audio as it has been shown earlier that explicit noise estimation helps in the overall denoising process (Xu et al., 2020). We use the decoder and codebook learnt in the previous step to obtain the noise profile from the noisy mixed signal $\mathbf{a} \in R^t$. In the first part of audio denoising network we use an autoregressive encoder which takes noisy melspectrogram representation of $\mathbf{a}$ *i.e.* $\mathbf{A} \in R^{T \times F}$, where T and F is the time-window and frequency-bins for spectrogram calculation and estimates the noise present in the signal, *i.e.* $\hat{\mathbf{N}} \in R^{T \times F}$.

Once the noise has been estimated, the second step involves using it along with the mixed audio signal for the denoising process. We encode both the noise and mixed signal melspectrograms individually with series of 1D convolutional layers and obtain features for mixed audio, *i.e.* $\mathbf{A_m} \in R^{T \times K_1}$ and estimated noise features *i.e.* $\hat{\mathbf{N_f}} \in R^{T \times K_1}$. We then disentangle the noise features from the mixed features using a cross-attention inspired mechanism. In order to disentangle the features we use three linear layers, $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in R^{K_1 \times K_2}$. We then use $\mathbf{Q}$ to project noise features $\hat{\mathbf{N_f}}$ and obtain $\hat{\mathbf{N_Q}} \in R^{T \times K_2}$. Similarly, we also project the mixed audio features $\mathbf{A_m}$ individually using $\mathbf{K}$ and $\mathbf{V}$, and obtain $\mathbf{A_K} \in R^{T \times K_2}$ and $\mathbf{A_V} \in R^{T \times K_2}$ respectively. We then estimate the extent of noise component in the mixed signal, and then remove it from the mixed signal to obtain feature of the clean signal *i.e.* $\hat{\mathbf{A}}_c \in R^{T \times K_2}$. We obtain $\hat{\mathbf{A}}_\mathbf{C}$ as,

$$\hat{\mathbf{A}}_\mathbf{C} = \mathbf{A_V} - (\frac{\hat{\mathbf{N}}_Q^T \mathbf{A_K}}{\sqrt{K_2}}) * \mathbf{A_V}. \tag{3}$$

The above formulation is equivalent to projecting the noise feature vector along the mixed feature vector and then removing the noise component along the direction of mixed feature vector. In order to enforce that the estimated clean features become closer to the ground truth clean audio features, we use knowledge distillation, where we use a teacher network to help train a student network. We accomplish this by training a separate speech only network which takes a clean speech samples as input and reconstructs the same. We obtain the clean audio features after the encoder layer of this network and denote it as $\mathbf{A_C}$. This clean audio prediction network is the teacher network for the denoising network. We use the following loss function for training:

$$\mathcal{L}_{feat} = \|\mathbf{A_C} - \hat{\mathbf{A}}_\mathbf{C}\|_1 \tag{4}$$

Eq. 4 can also be considered similar to the case where it enforces the dot product of the speech features to be similar to the speech content in the mixed signal and the dot product with the noise features to be as dissimilar as possible.

In the next step of denoiser, we use a bidirectional LSTM followed by linear layers, that takes estimated clean audio features $\hat{\mathbf{A}}_\mathbf{C}$ and predicts the denoised mel spectrogram $\hat{\mathbf{A}}_i \in R^{T \times F}$. We use a final reconstruction loss between the ground truth and estimated melspectrogram:

$$\mathcal{L}_{rec} = \|\mathbf{A}_i - \hat{\mathbf{A}}_i\|_1. \tag{5}$$

We train the network by combining all the losses,

$$\mathcal{L}_{tot} = \mathcal{L}_{rec} + \alpha \mathcal{L}_{feat} + \beta \mathcal{L}_{noise}, \tag{6}$$

where, α and β are the hyperparameters to control the weight of each loss.

Training. We follow a two step training where, in the first step, we train the network to model the noise only using Eq. 2. Once the noise modelling network is trained, in the second step we fix (i) the codebook and (ii) decoder learnt in previous step, and learn all other modules in an end to end manner using Eq. 6.

3.4 VOCODER

As we perform our denoising in the melspectrogram domain, our next goal is to convert the estimated melspectrogram to a time domain signal. To this end, we learn an inverse mapping function, *i.e.* $\mathcal{F} : \mathbf{A} \rightarrow \mathbf{a}$. As observed in prior studies (Liu et al., 2022), learning the vocoder separately with clean audio helps in reconstruction even from the estimated denoised melspectrogram. This is beneficial for the denoising task also, as it learns to generate speech signals from the melspectrogram and potentially model the speech signal which can further help in restoring the estimated memlspectrogram. We utilize TFGAN (Tian et al., 2020) which has reconstruction loss along with both frequency and time discriminator for high fidelity speech synthesis.

4 EXPERIMENTS

4.1 IMPLEMENTATION DETAILS

Dataset Details We use two different datasets for noise modelling, *i.e.* UrbanSound8k Salamon et al. (2014) and AudioNoiseset, curated from a large scale mulit-label audio classification dataset AudiosetGemmeke et al. (2017). We also use Audioset Gemmeke et al. (2017), a relatively large scale dataset and contains videos downloaded from YouTube for modelling noise. As Audioset was designed for general purpose audio classification, it contains a wide variety of 632 different audio classes covering both speech and non-speech signals. We perform various steps to obtain noisy only samples from Audioset without any speech content. Finally, we obtain a dataset containing 106331, 5096 and 5025 samples for train, val and test set respectively.

For the clean speech, we download a subset of AVSPEECHEphrat et al. (2018), a relatively clean speech only dataset containing around 10000 samples for training and 2000 samples each for validation and testing. We purposefully selected the dataset as it contains in-the-wild videos downloaded from YouTube and contains a variety of languages covering different accents and speaking style. In order to have a fair comparison, we created a fixed val and test set where we fix the audio and noise sample. We created such fixed-eval and fixed-test set for both UrbanSound8K and AudioNoiseset and report all our results in the fixed-test set. We request the readers to refer to the supplementary material for more details on dataset preparation.

Input Representation and Metrics We resample both speech and noise signal to 22.05 kHz and perform min-max normalization and then mix the signal with specific SNR value. We convert the time domain mixed waveform to mel-spectrogram with FFT window length of 1024, hop length of 256 and 80 number of mel bands. Following the prior works Fu et al. (2019), we evaluate the results using two speech quality estimation metrics *i.e.* Perceptual Evaluation of Speech Quality (PESQ) and Short-Time Objective Intelligibility (STOI). PESQ gives a perceptual measure of the estimated speech signal and STOI measures the intelligibility of the denoised signal with respect to the clean input signal. We only report the perceptual metrics here as our work is of generative nature which does not produce a sample level alignment of predicted waveform with the ground truth one and more strict measures like SNR and SDR penalizes heavily in this case as also reported in prior generative approaches Liu et al. (2022); Kumar et al. (2020).

4.2 ROLE OF NOISE MODELLING IN DENOISING TASK

Here, we show the importance of noise modelling in the task of denoising. We report the performance in Tab.7. In the first case `Denoiser Only`, we only train the denoiser part of the network that takes directly the mixed melspectrogram and produces the denoised speech signal. The architecture here is exactly equivalent to the denoiser part of the network propose in Fig.1 excluding feature disentanglement part. This is a basic configuration for denoising and for our case serves as the lower bound for the task. In the second case, `w. GT Noise`, we use the same denoiser only but instead of giving the mixed audio only as input, we provide the ground truth noise signal along with it. We experiment with `w. GT Noise`, to experiment whether adding GT noise to the network improves the task of denoising and this serves as the upperbound for our approach and reinforces our claim of using noise signal helps in predicting the denoised audio better. We observe that on an average the PESQ and STOI value increases by 14% and 11% respectively after using ground truth noise *vs.* directly denoising from the mixed audio. Finally, we give two different variant of the proposed

Method	SNR=-10		SNR=-7		SNR=-3		SNR=0		SNR=3		SNR=7		SNR=10		Avg.	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
`Denoiser Only`	1.654	0.560	1.938	0.647	2.201	0.697	2.267	0.728	2.361	0.746	2.447	0.791	2.546	0.796	2.202	0.709
`w. GT Noise`	2.116	0.701	2.318	0.752	2.516	0.779	2.547	0.799	2.635	0.809	2.690	0.836	2.787	0.836	2.515	0.787
`Ours (Concat)`	1.717	0.562	1.974	0.650	2.248	0.703	2.300	0.735	2.413	0.752	2.513	0.797	2.637	0.800	2.258	0.714
`Ours (FDD)`	1.787	0.572	2.029	0.660	2.356	0.714	2.408	0.746	2.551	0.761	2.671	0.815	2.800	0.819	2.372	0.727

Table 1: **Effect of noise estimation on final denoising performance**. PESQ and STOI for different noise combination strategy in the network for UrbanSound8K dataset.

Method	SNR=-10		SNR=-7		SNR=-3		SNR=0		SNR=3		SNR=7		SNR=10		Avg.	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Noisy	1.118	0.392	1.356	0.460	1.573	0.535	1.683	0.582	1.834	0.618	2.015	0.700	2.179	0.730	1.680	0.574
SEGAN Pascual et al. (2017)	0.668	0.285	0.759	0.365	1.133	0.471	1.168	0.526	1.385	0.588	1.639	0.689	1.868	0.734	1.232	0.523
VoiceFixerLiu et al. (2022)	1.264	0.421	1.561	0.548	1.885	0.644	1.931	0.683	2.168	0.717	2.215	0.758	2.390	0.764	1.916	0.648
Denoiser Defossez et al. (2020)	1.101	0.450	1.645	0.614	1.980	0.683	2.135	0.746	2.335	0.772	2.463	0.827	2.729	0.840	2.056	0.705
MetricGAN+ Fu et al. (2021)	1.598	0.354	1.745	0.432	1.972	0.528	2.144	0.591	2.301	0.632	2.415	0.717	2.656	0.745	2.119	0.571
LSSXu et al. (2020)	1.656	0.516	1.929	0.615	2.256	0.683	2.383	0.738	2.566	0.767	2.732	0.827	2.949	0.844	2.353	0.713
Ours (concat)+HiFi Kong et al. (2020)	1.420	0.516	1.662	0.606	1.907	0.653	1.951	0.679	2.034	0.698	2.081	0.737	2.165	0.743	1.889	0.662
Ours (concat)+GL	1.646	0.500	1.941	0.582	2.208	0.630	2.265	0.648	2.388	0.669	2.478	0.704	2.579	0.705	2.215	0.634
Ours (Concat)	1.717	0.562	1.974	0.650	2.248	0.703	2.300	0.735	2.413	0.752	2.513	0.797	2.637	0.800	2.258	0.714
Ours (FDD)	1.787	0.572	2.029	0.660	2.356	0.714	2.408	0.746	2.551	0.761	2.671	0.815	2.800	0.819	2.372	0.727
GT Audio	3.359	0.923	3.378	0.926	3.406	0.925	3.363	0.925	3.377	0.925	3.382	0.926	3.400	0.924	3.381	0.925

Table 2: **Audio Denoising**. Performance of different methods for the task of audio denoising for UrbanSound8K dataset.

network, where `Ours (concat)` is the simple denoiser that takes the concatenated mixed audio and predicted noise to obtain the denoised audio and `Ours(FDD)` is our full network with feature disentanglement in the denoiser. We observe that the simple `Ours(concat)` approach gives a performance boost of around 2.6% and `Ours(FDD)` gives a boost of 7.8% for PESQ over the baseline approach of `Denoiser Only`. Similarly for STOI, we observe around 1% and 2.6% for `Ours(concat)` and `Ours(FDD)` respectively over the baseline of `Denoiser Only`.

4.3 Comparison with prior approaches

In this section, we report the results to show how does our method compare with prior approaches. We compare our results with four existing state-of-the-art methods. We report all the results for the prior methods by using the pre-trained models made publicly available by the authors of the respective methods.

As our approach works in the melspectrogram domain, we first use a well known vocoder HiFi-GAN Kong et al. (2020) to convert our predicted melspectrogram to time domain waveform and mark it as ours(concat)+HiFi in Tab.2. Similarly we also use a well-known signal processing algorithm for converting melspectrogram to waveform, *i.e.* Griffin-Lim and report it as Ours(concat)+GL in Tab.2. We observe that Ours(concat)+GL gives quantitatively better performance on average in terms of PESQ, but slightly lower in terms of STOI over the HiFi GAN method, *i.e.* 2.215, 0.634 *vs.* 1.889, 0.662. However qualitatively inspecting the audio, we observe that HiFi GAN changes the accent of the speaker as it was mostly trained native english speaker data. Similarly for GL algorithm we observe a buzzing sound in almost all the examples. We then give the resulsts for both of our approach Ours(concat) and Ours(FDD) where in the first case input noise is concatenated with the mixed audio where as in the later case we perform our full approach of feature disentanglement. In this two methods we use our vocoder trained previously with clean speech data alone. We observe the best performance of 2.372 and 0.727 for PESQ and STOI and is also comparable to the best performing method LSS haivng PESQ and STOI of 2.353 and 0.713 respectively. We also observe that our method performs especially better in reconstructing speech for high noise cases which the prior methods fail to do. Further, qualitatively we observe that the noise leaks into the output in case of LSS which our method handles gracefully. We request the reader to have a look at the qualitative results for a better understanding. Finally we also report the vocoder capability of reconstructing the GT melspectrogram and mention it as GT Audio in Tab.2. The metrics for this case serves as an upperbound for the results.

Similarly, we also report the performance for our propose AudioNoiseset in Tab.3. We observe that our method is competitive in terms of PESQ with the previous best performing method LSS, *i.e.* 2.365 *vs.* 2.373 and our method performs slightly better in terms of STOI, *i.e.* 0.730 *vs.* 0.722. For both the dataset of Urbansound8K and AudioNoiseset, our performance is at par with the best performing methods in terms of PESQ and STOI but we observe that our method produces robotic

Method	SNR=-10		SNR=-7		SNR=-3		SNR=0		SNR=3		SNR=7		SNR=10		Avg.	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Noisy	1.304	0.450	1.464	0.530	1.612	0.582	1.877	0.668	2.045	0.714	2.401	0.785	2.622	0.844	1.904	0.654
SEGAN Pascual et al. (2017)	0.733	0.319	0.948	0.413	0.992	0.469	1.293	0.571	1.505	0.640	1.801	0.728	2.014	0.771	1.326	0.559
VoiceFixerLiu et al. (2022)	1.305	0.460	1.460	0.534	1.763	0.621	2.016	0.703	2.123	0.718	2.218	0.756	2.464	0.796	1.907	0.655
Denoiser Defossez et al. (2020)	1.277	0.499	1.624	0.617	1.944	0.690	2.182	0.777	2.343	0.789	2.491	0.830	2.787	0.872	2.093	0.725
MetricGAN+ Fu et al. (2021)	1.631	0.369	1.864	0.453	1.880	0.495	2.087	0.591	2.361	0.653	2.617	0.730	2.702	0.770	2.163	0.580
LSSXu et al. (2020)	1.749	0.536	2.023	0.623	2.155	0.670	2.374	0.749	2.557	0.780	2.779	0.827	2.977	0.867	2.373	0.722
Ours (Concat)	1.754	0.573	1.950	0.647	2.116	0.687	2.308	0.753	2.396	0.764	2.533	0.792	2.641	0.848	2.243	0.723
Ours (FDD)	1.839	0.581	2.072	0.658	2.237	0.698	2.412	0.763	2.540	0.774	2.672	0.806	2.782	0.832	2.365	0.730

Table 3: **Audio Denoising**. Performance of different methods for the task of audio denoising for AudioNoiseset dataset.

Method	SNR=-10		SNR=-7		SNR=-3		SNR=0		SNR=3		SNR=7		SNR=10		Avg.	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
`SpecDiff`	1.523	0.514	1.792	0.600	2.008	0.646	2.044	0.671	2.125	0.689	2.166	0.723	2.238	0.727	1.985	0.653
`DirectInner`	1.762	0.568	1.988	0.648	2.212	0.698	2.318	0.709	2.540	0.760	2.589	0.798	2.765	0.801	2.310	0.710
`Ours(FDD)`	1.787	0.572	2.029	0.660	2.356	0.714	2.408	0.746	2.551	0.761	2.671	0.815	2.800	0.819	2.372	0.727

Table 4: **Noise Removal Strategy**. We evaluate here different strategies for unmixing the noise from the mixed signal. `SpecDiff` refers to the case where we perform unmixing at input level and `DirectInner`, `Ours(FDD)` performs unmixing at feature level for UrbanSound8K dataset.

sound at the output which we correct by adding another existing enhancement module on top of it. Please refer to user study section on detailed analysis of this.

4.4 Optimal noise removal strategy from mixed audio

In this section we evaluate different noise removal strategy from the mixed audio. We evaluate here three different strategies, *i.e.* (1) `SpecDiff`, where we directly subtract the melspectrogram of Mixed audio from estimated noise and use the resultant signal as input to the denoiser network. In (2) `DirectInner`, instead of learning the $\mathbf{Q}$,$\mathbf{K}$ and $\mathbf{V}$ matrices we directly remove the noise component after finding out the component of noise feature along the mixed feature and then removing it from the mixed feature. This operation is equivalent to $\mathbf{\hat{A}_C} = \mathbf{A_m} - (\mathbf{\hat{N}}_f^T \mathbf{A_m}) * \mathbf{A_m}$, where $\mathbf{\hat{N}}_f$, $\mathbf{A_m}$ and $\mathbf{\hat{A}_C}$ are the estimated noise signal feature, mixed signal features and obtained potentially clean features respectively. In (3), `Ours(FDD)`, we use our full method where we perform denoising after projecting the features using $\mathbf{Q}$,$\mathbf{K}$ and $\mathbf{V}$ matrices as described in eq.3. We observe that when we directly take the difference of predicted noise from the mixed signal gives the least performance in terms of average PESQ and STOI with value 1.985 and 0.653 respectively. This is mostly because after visualizing the input difference signal we observe that the input is highly sparse with most of the value being very less or zero which further makes the denoising network extremely difficult to recover the missing portion. On the contrary if we perform similar difference operation at the feature level, the performance improves significantly for both the type of difference operation, *i.e.* `DirectInner` and `Ours(FDD)` (in terms of PESQ and STOI) to 2.310, 0.710 and 2.372, 0.727 respectively. Further, we observe that our proposed approach of feature disentanglement performs the best with around 16% improvement in PESQ and 11% improvement in STOI over the baseline of direct difference at the input level.

5 User Study

We also perform a subjective evaluation to measure the perceptual quality of our enhanced speech. While giving high quantitative results, the raw output of our method, while being much better understandable, suffers from slight robotic speech artifacts. To alleviate such artefacts we post process our output with a state of the art enhancement network, VoiceFixer (Liu et al., 2022).

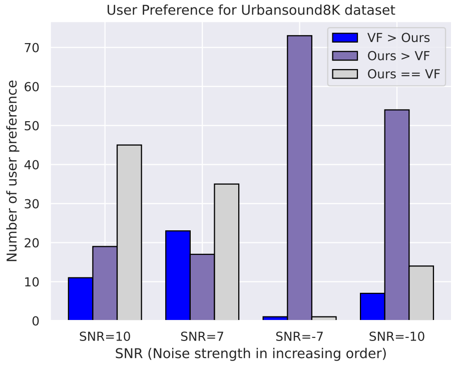
To ensure fairness, we compare all three versions, our raw output, our ouput post processed by VoiceFixer, as well the output of VoiceFixer itself, quantitatively (Tab. 5, Tab.6). We observe that our raw output performs much better than VoiceFixer, 2.365 PESQ average for ours *vs.* 1.907 for VoiceFixer on AudioNoiseSet samples. However, the post processed version of our method drops in quantitative metrics (1.846 PESQ). This is in contrast to what we observe in the user study where the post processed version of our output is strongly favored by users over the VoiceFixer outputs (discussed in the following). We also include numerous qualitative results for the reader to

Method	SNR=-10		SNR=-7		SNR=-3		SNR=0		SNR=3		SNR=7		SNR=10		Avg.	
	PESQ	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**
VoiceFixerLiu et al. (2022)	1.264	0.421	1.561	0.548	1.885	0.644	1.931	0.683	2.168	0.717	2.215	0.758	2.390	0.764	1.916	0.648
Ours (FDD)	1.787	0.572	2.029	0.660	2.356	0.714	2.408	0.746	2.551	0.761	2.671	0.815	2.800	0.819	2.372	0.727
Ours (FDD) + VoiceFixer	1.277	0.518	1.539	0.605	1.892	0.664	1.925	0.696	2.041	0.711	2.047	0.745	2.197	0.749	1.846	0.670

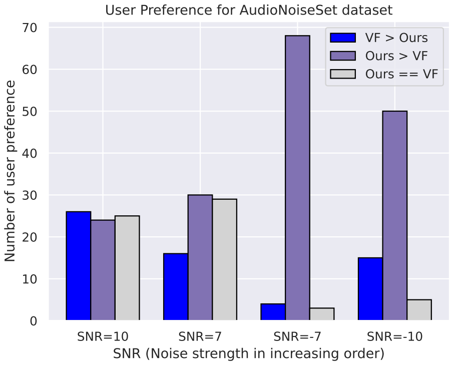
Table 5: **Degradation of metric even when perceptual quality is maintained (UrbanSound8K)**. Degradation of PESQ and STOI value of the reconstructed results even when the overall perceptual quality is maintained for UrbanSound8K dataset.

Method	SNR=-10		SNR=-7		SNR=-3		SNR=0		SNR=3		SNR=7		SNR=10		Avg.	
	PESQ	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**	**PESQ**	**STOI**
VoiceFixerLiu et al. (2022)	1.305	0.460	1.460	0.534	1.763	0.621	2.016	0.703	2.123	0.718	2.218	0.756	2.464	0.796	1.907	0.655
Ours (FDD)	1.839	0.581	2.072	0.658	2.237	0.698	2.412	0.763	2.540	0.774	2.672	0.806	2.782	0.832	2.365	0.730
Ours (FDD) + VoiceFixer	1.350	0.529	1.559	0.604	1.758	0.653	1.891	0.710	2.024	0.722	2.059	0.747	2.241	0.776	1.840	0.677

Table 6: **Degradation of metric even when perceptual quality is maintained (AudioNoiseset)**. Degradation of PESQ and STOI value of the reconstructed results even when the overall perceptual quality is maintained for AudioNoiseset dataset.



(a) User study for UrbanSound8K dataset



(b) User study for AudioNoiseset dataset

appreciate this. Such observations points to the potential inadequacy in quantitative metrics used for denoising task. Designing metrics mimicking user perception is a challenging task, and we leave investigating this to the future, and move on to describing the subjective evaluation settings and results.

We perform the subjective evaluation of our result enhanced with VoiceFixer (Liu et al., 2022) *vs*. using VoiceFixer directly. We randomly selected 5 samples each for two low SNR value, *i.e.* -7 and -10 and two high SNR value 7 and 10 and asked the participants to listen to three audio samples for each data point, *i.e.* mixed audio, the reconstruction from the base network (first), and our enhanced prediction (second). We recruited 15 volunteers with varying age group, educational qualification to judge the output. We asked them to listen to the audio and mark any one of the three conditions, *i.e.* whether fist performs better than second, whether second performs better than second, or both the model perform equally.

We observe that for low SNR values *i.e.* -7 and -10, our method is a clear winner with almost 90% participants responding our method to better where as for large SNR values *i.e.* 7, 10 most of the participants gave equal preference to both the methods. This suggest that our approach can denoise in heavy noise conditions and performs at par with state-of-the art methods for less noise cases.

6 Conclusion

We propose a novel audio denoising network by explicitly modelling the noise in the signal. We show that explicitly modelling the noise and then removing the noise component at feature level from the mixed signal helps in better reconstruction of the original speech signal. We show quantitatively and qualitatively that our method is able to extract out the noise signal reliably even in very high noise conditions for which the prior methods fail to do. Finally, we show that we can recover high quality speech signal after adding an enhancement network on top of ours and we plan to investigate joint training of such frameworks in future.

References

Nasim Alamdari, Arian Azarang, and Nasser Kehtarnavaz. Improving deep speech denoising by noisy2noisy signal mapping. *Applied Acoustics*, 172:107631, 2021.

Ibrahim Almajai and Ben Milner. Visually derived wiener filters for speech enhancement. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(6):1642–1651, 2010.

Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 609–617, 2017.

Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. *Advances in neural information processing systems*, 29, 2016.

Jacob Benesty, Shoji Makino, and Jingdong Chen. *Speech enhancement*. Springer Science & Business Media, 2006.

Steven Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on acoustics, speech, and signal processing*, 27(2):113–120, 1979.

Alexandre Defossez, Gabriel Synnaeve, and Yossi Adi. Real time speech enhancement in the waveform domain. *arXiv preprint arXiv:2006.12847*, 2020.

Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. *arXiv preprint arXiv:1804.03619*, 2018.

Huajian Fang, Guillaume Carbajal, Stefan Wermter, and Timo Gerkmann. Variational autoencoder for speech enhancement with a noise-aware encoder. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 676–680. IEEE, 2021.

Szu-Wei Fu, Chien-Feng Liao, Yu Tsao, and Shou-De Lin. Metricgan: Generative adversarial networks based black-box metric scores optimization for speech enhancement. In *International Conference on Machine Learning*, pp. 2031–2041. PMLR, 2019.

Szu-Wei Fu, Cheng Yu, Tsun-An Hsieh, Peter Plantinga, Mirco Ravanelli, Xugang Lu, and Yu Tsao. Metricgan+: An improved version of metricgan for speech enhancement. *arXiv preprint arXiv:2104.03538*, 2021.

Ken'ichi Furuya and Akitoshi Kataoka. Robust speech dereverberation using multichannel blind deconvolution with spectral subtraction. *IEEE Transactions on audio, speech, and language processing*, 15(5):1579–1591, 2007.

Ruohan Gao and Kristen Grauman. Visualvoice: Audio-visual speech separation with cross-modal consistency. In *CVPR*. IEEE, 2021.

Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 776–780. IEEE, 2017.

Francois G Germain, Qifeng Chen, and Vladlen Koltun. Speech denoising with deep feature losses. *arXiv preprint arXiv:1806.10522*, 2018.

Priya Gupta. Effects of noise on speech perception in children using cochlear implants: A systematic review. 2021.

Tsun-An Hsieh, Cheng Yu, Szu-Wei Fu, Xugang Lu, and Yu Tsao. Improving perceptual quality by phone-fortified perceptual loss using wasserstein distance for speech enhancement. *arXiv preprint arXiv:2010.15174*, 2020.

Yuchen Hu, Nana Hou, Chen Chen, and Eng Siong Chng. Interactive feature fusion for end-to-end noise-robust speech recognition. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6292–6296. IEEE, 2022.

Ewa Jacewicz, Robert Allen Fox, and Lai Wei. Between-speaker and within-speaker variation in speech tempo of american english. *The Journal of the Acoustical Society of America*, 128(2): 839–850, 2010.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Sunil Kamath, Philipos Loizou, et al. A multi-band spectral subtraction method for enhancing speech corrupted by colored noise. In *ICASSP*, volume 4, pp. 44164–44164. Citeseer, 2002.

Madhav Mahesh Kashyap, Anuj Tambwekar, Krishnamoorthy Manohara, and S Natarajan. Speech denoising without clean training data: A noise2noise approach. *Proc. Interspeech 2021*, pp. 2716–2720, 2021.

Saurabh Kataria, Jesús Villalba, and Najim Dehak. Perceptual loss based speech denoising with an ensemble of audio pattern recognition and self-supervised models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7118–7122. IEEE, 2021.

Alexander JE Kell and Josh H McDermott. Deep neural network models of sensory systems: windows onto the role of task constraints. *Current opinion in neurobiology*, 55:121–132, 2019.

Andrew J King and Kerry MM Walker. Listening in complex acoustic scenes. *Current opinion in physiology*, 18:63–72, 2020.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in Neural Information Processing Systems*, 33:17022–17033, 2020.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 25(1106-1114):1, 2012.

Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brébisson, Yoshua Bengio, and Aaron C Courville. Melgan: Generative adversarial networks for conditional waveform synthesis. *Advances in neural information processing systems*, 32, 2019.

Rithesh Kumar, Kundan Kumar, Vicki Anand, Yoshua Bengio, and Aaron Courville. Nu-gan: High resolution neural upsampling with gan. *arXiv preprint arXiv:2010.11362*, 2020.

Katia Lebart, Jean-Marc Boucher, and Philip N Denbigh. A new method based on spectral subtraction for speech dereverberation. *Acta Acustica united with Acustica*, 87(3):359–366, 2001.

Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*, 2018.

Jae Lim and Alan Oppenheim. All-pole modeling of degraded speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(3):197–210, 1978.

L Lin, WH Holmes, and E Ambikairajah. Speech denoising using perceptual modification of wiener filtering. *Electronics Letters*, 38(23):1, 2002.

Haohe Liu, Xubo Liu, Qiuqiang Kong, Qiao Tian, Yan Zhao, and DeLiang Wang. Voicefixer: A unified framework for high-fidelity speech restoration. *Interspeech*, 2022.

Philipos C Loizou. *Speech enhancement: theory and practice*. CRC press, 2007.

Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 27(8): 1256–1266, 2019.

Joerg Meyer and Klaus Uwe Simmer. Multi-channel speech enhancement in a car environment using wiener filtering and spectral subtraction. In *1997 IEEE international conference on acoustics, speech, and signal processing*, volume 2, pp. 1167–1170. IEEE, 1997.

Ryoichi Miyazaki, Hiroshi Saruwatari, Satoshi Nakamura, Kiyohiro Shikano, Kazunobu Kondo, Jonathan Blanchette, and Martin Bouchard. Musical-noise-free blind speech extraction integrating microphone array and iterative spectral subtraction. *Signal processing*, 102:226–239, 2014.

Babafemi O Odelowo and David V Anderson. A study of training targets for deep neural network-based speech enhancement using noise prediction. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5409–5413. IEEE, 2018.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

Ashutosh Pandey and DeLiang Wang. A new framework for cnn-based speech enhancement in the time domain. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(7): 1179–1188, 2019.

Se Rim Park and Jinwon Lee. A fully convolutional neural network for speech enhancement. *arXiv preprint arXiv:1609.07132*, 2016.

Santiago Pascual, Antonio Bonafonte, and Joan Serra. Segan: Speech enhancement generative adversarial network. *arXiv preprint arXiv:1703.09452*, 2017.

Karol J. Piczak. ESC: Dataset for Environmental Sound Classification. In *Proceedings of the 23rd Annual ACM Conference on Multimedia*, pp. 1015–1018. ACM Press. ISBN 978-1-4503-3459-4. doi: 10.1145/2733373.2806390. URL http://dl.acm.org/citation.cfm?doid=2733373.2806390.

Giuseppina Emma Puglisi, Anna Warzybok, Arianna Astolfi, and Birger Kollmeier. Effect of reverberation and noise type on speech intelligibility in real complex acoustic scenarios. *Building and Environment*, 204:108137, 2021.

Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, Dinei Florêncio, and Mark Hasegawa-Johnson. Speech enhancement using bayesian wavenet. In *Interspeech*, pp. 2013–2017, 2017.

Dario Rethage, Jordi Pons, and Xavier Serra. A wavenet for speech denoising. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5069–5073. IEEE, 2018.

Mark R Saddler, Andrew Francl, Jenelle Feather, Kaizhi Qian, Yang Zhang, and Josh H McDermott. Speech denoising with auditory models. *arXiv preprint arXiv:2011.10706*, 2020.

Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 1041–1044, 2014.

TV Sreenivas and Pradeep Kirnapure. Codebook constrained wiener filtering for speech enhancement. *IEEE Transactions on Speech and Audio Processing*, 4(5):383–389, 1996.

Jiaqi Su, Zeyu Jin, and Adam Finkelstein. Hifi-gan: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks. *arXiv preprint arXiv:2006.05694*, 2020.

Silero Team. Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier. https://github.com/snakers4/silero-vad, 2021.

Qiao Tian, Yi Chen, Zewang Zhang, Heng Lu, Linghui Chen, Lei Xie, and Shan Liu. Tfgan: Time and frequency domain based generative adversarial network for high-fidelity speech synthesis. *arXiv preprint arXiv:2011.12206*, 2020.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Saeed V Vaseghi. Spectral subtraction. In *Advanced Signal Processing and Digital Noise Reduction*, pp. 242–260. Springer, 1996.

Sean Vasquez and Mike Lewis. Melnet: A generative model for audio in the frequency domain. *arXiv preprint arXiv:1906.01083*, 2019.

Longbiao Wang, Kyohei Odani, and Atsuhiko Kai. Dereverberation and denoising based on generalized spectral subtraction by multi-channel lms algorithm using a small-scale microphone array. *EURASIP Journal on Advances in Signal Processing*, 2012(1):1–11, 2012.

Norbert Wiener, Norbert Wiener, Cyberneticist Mathematician, Norbert Wiener, Norbert Wiener, and Cybernéticien Mathématicien. *Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications*, volume 113. MIT press Cambridge, MA, 1949.

Matthew B Winn and Katherine H Teece. Listening effort is not the same as speech intelligibility score. *Trends in Hearing*, 25:23312165211027688, 2021.

Yang Xiang, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen. A deep representation learning speech enhancement method using β-vae. *arXiv preprint arXiv:2205.05581*, 2022.

Ruilin Xu, Rundi Wu, Yuko Ishiwaka, Carl Vondrick, and Changxi Zheng. Listening to sounds of silence for speech denoising. *Advances in Neural Information Processing Systems*, 33:9633–9648, 2020.

Yong Xu, Jun Du, Zhen Huang, Li-Rong Dai, and Chin-Hui Lee. Multi-objective learning and mask-based post-processing for deep neural network based speech enhancement. *arXiv preprint arXiv:1703.07172*, 2017.

Daniel LK Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience*, 19(3):356–365, 2016.

Karren Yang, Dejan Marković, Steven Krenn, Vasu Agrawal, and Alexander Richard. Audio-visual speech codecs: Rethinking audio-visual speech enhancement by re-synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8227–8237, 2022.

Zhaofeng Zhang, Longbiao Wang, and Atsuhiko Kai. Distant-talking speaker identification by generalized spectral subtraction-based dereverberation and its efficient computation. *EURASIP Journal on Audio, Speech, and Music Processing*, 2014(1):1–12, 2014.

Chengyu Zheng, Xiulian Peng, Yuan Zhang, Sriram Srinivasan, and Yan Lu. Interactive speech and noise modeling for speech enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 14549–14557, 2021.

A Dataset Details

We use two different datasets for noise modelling, *i.e*. UrbanSound8k Salamon et al. (2014) and AudioNoiseset, curated from a large scale mulit-lable audio classification dataset AudiosetGemmeke et al. (2017). UrbanSound8k contains environmental sounds downloaded from `Freesound`. It contains a total of 8732 noise samples with each sample being ≤ 4 sec. and is annotated into 10 different classes. As we are modelling the noise in the first step, we require noise that do not contain speech samples. So, we ignore two classes *i.e*., `street_music` and `children_playing` as we manually verfiy that the noise sample from these classes contain significant amount of speech mixed with it. So, for the dataset of UrbanSound8k we train and evaluate our model with the noise samples from the remaining 8 classes.

We also use Audioset Gemmeke et al. (2017), a relatively large scale dataset and contains videos downloaded from YouTube for modelling noise. As Audioset was designed for general purpose audio classification, it contains a wide variety of 632 different audio classes covering both speech and non-speech signals. We first select a subset of audio that were not marked as speech in the audioset dataset and were able to download a total of 242269 videos after ignoring the missing links. It is well known that the audioset dataset contains noisy annotations and we also observe that some of the audios do contain background speech signal even if it is not explicitly marked with speech label. We then perform a filtering process to get high quality non-speech only audio for the background noise. We use an open-source voice activity detector (VAD), *i.e*. Silero VAD Team (2021) to find out if there is any speech signal. We perform this by dividing the audio signal into chunks of 10ms and output a binary value indicating if each chunk contains speech or not. We then estimate the relative duration of the speech in the audio. We show the histogram of audio samples by their relative speech duration in the Fig.3. We observe from the figure that although the histogram is highly skewed towards zero, there are a few audios that contain quite a lot of speech. We use a strict threshold of 5% to select non-speech audio, *i.e*. we discard an audio to be considered as background noise if more than 5% of the total chunks are marked as speech by VAD. We use a very heavy threshold *i.e*., we discard one second audio if 0.5 second is marked as speech to ensure high quality noisy samples and have also verified manually that having such a large threshold gives almost perfect noisy samples. We finally had a dataset with 142281, 7096 and 7567 samples for train, val and test set respectively. We further observe that the after removing the samples containing speech samples, the samples were mostly biased towards `music instrument` class. We then remove some of the samples from this class and finally obtain a dataset containing 106331, 5096 and 5025 samples for train, val and test set respectively.

For the clean speech, we download a subset of AVSPEECHEphrat et al. (2018), a relatively clean speech only dataset containing around 10000 samples for training and 2000 samples each for validation and testing. We purposefully selected the dataset as it contains in-the-wild videos downloaded from YouTube and contains a variety of languages covering different accents and speaking style.

For training the network, for every speech sample, we randomly sample a noise signal and mix it with the speech sample for a specific SNR using eq.1. As for every instance we are randomly sampling a noise signal, we get different mixed signal each time. This will create a problem during validation and testing as we will having different samples for every run. In order to have a fair comparison, we created a fixed val and test set where we fix the audio and noise sample. We created such fixed-eval and fixed-test set for both UrbanSound8K and AudioNoiseset having 80 and 100 examples each for each SNR Value. In total we have 560 and 700 examples both in fixed-val and fixed-test set for Urbansound8K and AudioNoiseset.

B Ablations

Impact of noise loss on final reconstruction Here, we evaluate quantitatively the impact of noise estimation on the final audio reconstruction. We report here the results by varying the weights for noise reconstruction loss for both the case of `concat` and `FDD` approach for the dataset of UrbanSound8K. We observe that for both the cases increasing the noise weight to 1 gives the best performance , *i.e*. 2.258, 0.714 for `concat` and 2.372, 0.727 for `FDD` in terms of PESQ ans STOI respectively. Further, for other weight value we observe that the performance decreases which sug-

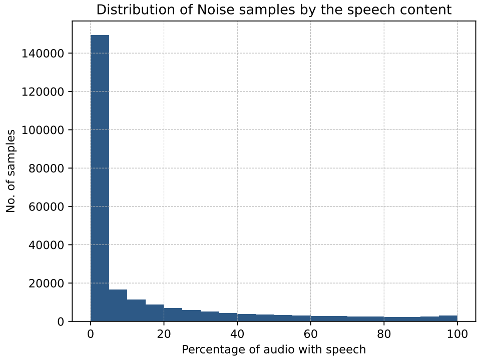


Figure 3: Audio distribution by the relative speech content

	Noise	SNR=-10		SNR=-7		SNR=-3		SNR=0		SNR=3		SNR=7		SNR=10		Avg.	
Method	weight	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
concat	0	1.654	0.560	1.938	0.647	2.201	0.697	2.267	0.728	2.361	0.746	2.447	0.791	2.546	0.796	2.202	0.709
	0.01	1.641	0.557	1.903	0.644	2.172	0.694	2.245	0.729	2.349	0.745	2.441	0.791	2.543	0.797	2.185	0.708
	1.0	1.717	0.562	1.974	0.650	2.248	0.703	2.300	0.735	2.413	0.752	2.513	0.797	2.637	0.800	2.258	0.714
FDD	0	1.719	0.566	2.022	0.657	2.286	0.709	2.379	0.743	2.504	0.761	2.613	0.811	2.735	0.813	2.323	0.723
	0.01	1.739	0.568	1.999	0.657	2.304	0.712	2.382	0.745	2.499	0.764	2.619	0.811	2.749	0.815	2.327	0.725
	1.0	1.787	0.572	2.029	0.660	2.356	0.714	2.408	0.746	2.551	0.761	2.671	0.815	2.800	0.819	2.372	0.727
	10	1.736	0.569	2.029	0.657	2.324	0.714	2.376	0.745	2.517	0.763	2.637	0.813	2.770	0.816	2.341	0.726

Table 7: **Ablation for noise weights**. PESQ and STOI for different weightage of noise in UrbanSound8K dataset.

gests that having an optimum noise weight gives a boost in performance for the task of speech denoising.

Is Vocoder all we need for denoising? Here, we evaluate the capability of vocoder itself as denoiser. As vocoder converts the melspectrograms to time-domain waveform, we experiment here whether only vocoder can give a clean time domain waveform when we feed it with a noisy melspectrogram. We train the vocoder here in two settings to evaluate the same and report the results in Tab.8 for UrbanSound8K dataset. In the first setting of Noisy2Clean, we train the vocoder by providing the noisy melspectrogram as the input and enforcing it to reconstruct the clean time-domain audio waveform. In the second setting Clean2Clean, we feed the vocoder with the clean mel spectrogram and reconstruct the corresponding clean audio waveform. We observe that on average the prediction of Clean2Clean is better than Noisy2Clean with PESQ, STOI being 1.992, 0.680 and 1.784, 0.623 respectively. We observe the approach of Clean2Clean performing better than that of Noisy2Clean as the former is able to better model the speech characteristics directly from the clean signal where as the late setting is notable to extract such characteristics with a limited network. Although the Clean2Clean setting performs better in comparison to Noisy2Clean, it lags behind our proposed approach of FDD by $~19\%$ and $~7\%$ in terms of PESQ ans STOI respectively. This experiment suggests that although the vocoder has some inherent property of denosing but it is not alone powerful enough to give better denoised result.

C Transfer Learning to other tasks

In this section, we show the capability of our noise modelling network. The environmental sound classification task can be considered similar as the audio used as background noise for speech de-

	SNR=-10		SNR=-7		SNR=-3		SNR=0		SNR=3		SNR=7		SNR=10		Avg.	
Method	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Noisy2Clean	1.503	0.496	1.628	0.574	1.823	0.623	1.822	0.639	1.868	0.653	1.887	0.689	1.956	0.689	1.784	0.623
Clean2Clean	1.515	0.534	1.757	0.621	1.992	0.673	2.045	0.698	2.127	0.716	2.204	0.756	2.300	0.763	1.992	0.680

Table 8: **Denoising capability of Vocoder**. We evaluate here the capability of vocoder alone for the task of denoising. We feed the vocoder directly with noisy melspectrogram and check whether it can reconstruct the clean audio directly as output.

noising tasks are often the environmental sounds. We use the encoder part of noise modelling network and finetune it for the task of environmental sound classification. We use a LSTM and two FC layers on top of the encoder for the final task of classification. We use the dataset ESC50 Piczak for the same. We show the performance of our network with and without pre-training of the encoder in Tab.9. We perform the standard 5-fold cross validation approach as done in the prior approaches and report the average performance of the five folds. We observe that with the unsupervised pre-training the performance on the classification tasks improves by around 12% in comparison to training the network from scratch. Although there are several methods that report higher performance in the dataset but in those cases the architecture and training startegy are carefully designed to achieve high performance exclusively for the task of sound classification. For comparison, we have provided a few approaches Aytar et al. (2016); Arandjelovic & Zisserman (2017) where the network is pre-trained in an unsupervised manner using multimodal data and then finetuned in ESC-50. For example, in Aytar et al. (2016), the network architecture is exclusively designed for the task of classification and also trained with around $20\times$ more data than ours. Similarly in case of Arandjelovic & Zisserman (2017), it is also trained with same dataset with $20\times$ more data than ours along with a multimodal self-supervised pre-training task. We also report the human level accuracy of 81.30% as an upperbound for the task. We conclude from the above experiment that our noise modelling network is competitive enough in classifying the environmental sounds which is also potentially helping us in the original task of speech denoising.

D Qualitative Results

We also provide some qualitative results in the supplementary material for both the dataset. We have also provided accompanying html file for easy listening of audio along with other baselines. We have provided results for high and low noise cases only to be within the supplementary file size limit. The filename is self explanatory for the dataset and the SNR value of the samples.

Approach	Accuracy
Scratch	61.40 %
Pre-Trained	68.60 %
Soundnet Aytar et al. (2016)	74.20 %
LLL Arandjelovic & Zisserman (2017)	79.30 %
Human Acc. Piczak	81.30 %

Table 9: **Environmental sound classification performance on ESC-50 dataset** We evaluate the performance of our unsupervised pre-training on the classification performance on ESC-50 dataset.