Audio denoising papers

Noise Reduction Techniques and Algorithms For Speech Signal Processing (Algo_Speech.pdf)

Link

Different types of noise :

- Background noise
- Echo
- Acoustic / audio feedback (Mic capture loudspeaker sound and send it back)
- Amplifier noise
- Quantization noise when transformning analog to digital (round values), neglectable at sampling higher than 8kHz/16bit
- Loss of quality due to compression

Linear filterning (Time domain) : Simple convolutation

Spectral filtering (Frequency domain) : DFT and back

ANC needs a recording of the noise to compare it to the audio

Adaptive Line Enhancer (ALE) doesn't need it.

Smoothing : noise is often random and fast change, so smoothing can help again white and blue (high freq) noise.

A Review of Adaptive Line Enhancers for Noise Cancellation (ALE.pdf)

Doesn't need recording of noise. Adaptive self-tuning filter that can spearate periodic and stochastic component. Detect low-level sin-waves in noise

…

A review: Audio noise reduction and various techniques (Techniques.pdf)

Some filters : Butterworth filter, Chebyshev filter, Elliptical filter

Employing phase information for audio denoising (Phase.pdf)

Audio Denoising by Time-Frequency Block Thresholding (Block_Threshold.pdf)

Speech Denoising with Deep Feature Losses (Speech_DL.pdf)

Fully convolutional network, work on the raw waveform. For the loss, use the internal activation of another network trainned for domestic audio tagging, and environnement detection (classification network). It's a little bit like a GAN.

Most approaches today are done on the spectrogram domain, this one not. Prevents some artefacts due do IFT. Methods that are in the time domain often use regression loss between input and output wave. Here, the loss is the dissimilarity between hidden activations of input and ouput waves. Inspired from computer vision (-> Perceptual_Losses.pdf)

Details of The main network are given in papers, section II-A-a.Different layers in the classification/feature/loss network correspond to different time scales. The classification network is inspired by VGG architecture from CV, details in paper II-B-a. II-B-b explain how to transoorm activations / weights to a loss.

Train the feature loss network using multiple classification tasks (scene classification, audio tagging). Train the speech denoising using the [1] database. They used the clean speeches and some noise samples and created the training data by combining them together, then they are downsampled.

Experimental setup : compared with Wiener filterning pipeline, SEGAN, and the WaveNet based one used as a baseline. Used different score metrics (overall (OVL), the signal (SIG), and the background (BAK) scores)). It was better than all the baselines. Also evaluated with human testers, also better than the others.

Now this is for speech, and it might not work as well for general sound/music

Recurrent Neural Networks for Noise Reduction in Robust ASR (RNN.pdf)

SPLICE algorithm is a model that can reduce noise by finding a joint distribution between clean and noisy data, ref to article in the paper's reference, but could not find it online for free.

We could simply engineer a filter, but it's hard, and not perfect .

Basic idea : We can use L1 norm as the loss function. This type of network is known as denoising autoencoder (DAE). Since input has variable length, we train on a small moving window

More advanced : Deep recurrent denoising audtoencoder, we add conection "between the windows" $\implies$ Input is [0 1 2] [1 2 3] [2 3 0], we give each one one to a NN with e.g. 3 hidden layer, with layer 2 recursively connected, and it gives [1] [2] [3] as the output. Uses Aurora2 corpus, with noisy variants synthetically generated

Investigating RNN-based speech enhancement methods for noise-robust Text-to-Speech (RNN_Speech_Enhancement.pdf)

Shows the two alternative approaches (time vs frequency) on a graph

...

Audio Denoising with Deep Network Priors (DN_Priors.pdf)

Combines time and frequency domain, unsuppervised, you try to fit the noisy audio and since we only partialy fit the output of the network helps to find the clean audio. Link to github repo with some data and the code https://github.com/mosheman5/DNP.

Usually we first create a mask that tells us what frequency are noise, then we use an algo that removes those frequencies.

Here the assumption is that it is hard by definition to fit noise, so the NN will only fit the clean part of the input.

Technique already used in CV. Diff : in CV, the output is already the cleaned image, not here, so they create a spectral mask from the ouput to then denoise the audiio. Better than other unsupervised methods, almost as good the the supervised ones.

=> Probably not usefull for GANs.

Spectral and Cepstral Audio Noise Reduction Techniques in Speech Emotion Recognition (Spectral_Cepstral.pdf)

They will compare their method to methods of "Spectral substraction", where you remove the noise spectrum from the audio spectrum.

Need to look in more details at "Spectral domain", "Power Spectral"; "log-sepstral", "cepstral domain", ...

One again no NN are used here, this is mostly some signal processing, so I don't think it will be very usefull.

They also talk about "accurancy measures", e.g. "Itakura distance", "Toeplotz autocorrelation matrix", "euclidian distance between two mel-frequency cepstral vectors".

Probably more informations about signal processing techniques in the references.

Raw Waveform-based Speech Enhancement by Fully Convolutional Networks (RawWave_CNN.pdf)

Convolutional, waveform to waveform. Mentions like most “Wiener filtering”, “spectral substraction”, “non-negative matrix factorisation”. Also mentions “Deep denoising autoencoder” from (RNN.pdf), also see (DDAE.pdf) that they are citing.

Explain that most models use the magnitude spectrogram (-> log-power spcetra), which will leave the phase noisy (as it used the phase from the noisy signal to reconstruct the output signal). Also mentions that it is important to use the phase information to reconstruct the signal afterwards. Apparently DL based models that use the raw form often get better results.

Fully connected is not very good since it won’t keep local informaton (think high frequencies). They use A “Fully convolutional network FCN” and not a CNN, see (FCN.pdf). A FCN also mean a lot less paramters.

Convolutional is considered better since we need adjacent information to make sense of frequencies in the time domain. Fully connected layers cause problems (can’t moel high and low frequencies together), so that’s why we don’t have one at the end in a FCN (FCN = CNN without fully-connected layers).

For the experiment, as some of the others papers, they took clean data and corrupted it with some noise (e.g. Babble, Car, Jackhammer, pink, street, white gaussian, . . .)

They also mention at the end the difference between the “shift step” for the input in the case of a DNN, but it’s not very clear what they did with the FCN. They say the took 512 samples from the input wave, but does it seems really low if we use e.g. 44kHz sampling for our music.

Speech Enhancement Based on Deep Denoising Autoencoder (DDAE.pdf)

They meantion a DAE where they only trained using clean speech : Clean as in and out, then when we give a noisy signal it tries to express it on the “clean subspace/basis function”, they try to model “what makes a clean speech”, need to look into that. This time, they use dirty-clean pairs, so they want to know "what is the statistical difference between noisy and clean.

Once again, they create their dataset by adding some noise artificially. They mention (RNN.pdf), which uses a recurrent network, this won’t be the case here.

The architecture looks like a classical DNN. They stack “neural autoencoders” together, and each AE seems to be layer - non-linear - layer. They also use regularization. For training, they first pretrain each AE individually which adequate parameters, then put them together and train again.

Measurement are specific to speech, they use “noise reduction”, “speech distortion” and “perceptual evalutation for speech qualty - PESQ” / not clear what this is.

For the features they use “Mel frequency power spectrum (MFP)” on 16ms intervals

Their results are mostly better than traditional methods.

SEGAN: Speech Enhancement Generative Adversarial Network (Speech_GAN.pdf)

As some other papers, mention that most use spectral form, but here they use the raw waveform.

Explains GAN : The generator which creates some data by learning the real data distribution and trying to approximate it, and the discriminator, usually a binary classifier, that tries to tell us if our sample is a real one or one generated by the generator. The goal for the generator is to fool the discriminator.

To train : D back-props a batch of real examples classified as “true”, and then a batch of fake example (generated by G) and mark them as “false”. Then we fix D’s parameters, and G does the backpropagation with the false example to try to make D missclassify. They then give more mathematical details and techniques (e.g. LSGAN).

They use a fully convolutional network (FCN), with a encoder-decoeer layout, were the signal is “compressed”, concateneted with tje latent representation (?) and then decoded. They also use skip connections so we don’t lose some details about the structure (we have speech in and out some we have some similarities). e.g. they transmit phase and alignment information. They then use some information from the D network to create their loss.

Their dataset is the usual one we saw previously [1], and they use both artificial and natural noise to create their tran/test set. THey use a sliding window of the raw data (downsampled a little bit), and they also used a minor high freqency filter.

All the code is on https://github.com/santi-pdp/segan. Results are positive and are mostly done by people’s opinions.

A Wavenet for Speech Denoising (WaveNet.pdf)

They first present the WaveNet network, which was used to synthesize natural sounding speech.

Their model is similair to WaveNet, but the convolution is “symetrically centerd” since we know both future and passed data unlinke for speech generation. They also have a different loss function, the output is not a probability but the clean data directly,

Audio super-resolution papers

Audio Super-Resolution using Neural Nets (SuperRes_NN.pdf)

Paper + webpage + github on super resolution with deep networks https://kuleshov.github.io/audio-super-res/#

Adversarial Audio Super-resolution with Unsuppervised Feature Losses (Adversarial.pdf)

Time Series Super Resolution with Temporal Adaptive Batch Normalization (TimeSerie_Batch.pdf)

Ideas from images

Perceptual Losses for Real-Time Style Transfer and Super-Resolution (Perceptual_Losses.pdf)

The Unreasonable Effectiveness of Deep Features as a Perceptual Metric (Perceptual_Metric.pdf)

Fully Convolutional Networks for Semantic Segmentation (FCN.pdf)

Datasets

1: Voice database with noisy and clean version

https://datashare.is.ed.ac.uk/handle/10283/1942

2: New version of [1], also voice

https://datashare.is.ed.ac.uk/handle/10283/2791

3: Speech database with clean and noisy

https://github.com/dingzeyuli/SpEAR-speech-database

4: Aurora2

http://aurora.hsnr.de/aurora-2.html Some script that can generate noisy data