Preparatory Work for the Master Thesis

Pascal Tribel

2022

Contents

1 Introduction

1.1 Problem description

Audio recording has known many improvements during the past decades. We can claim that the first existing audio recording we have was done on the 9^{th} of April 1860 [1]. The first major recording technique that appeared was the *Phonograph*, which has been quite fast replaced by the *Gramophone*. Finally, during the 20^{th} century, the digitalization of audio recording made those tools lapsed.

However, unfortunately, in music history, many composers and musicians have been only recorded on such materials, and we can assume that the resulting sound is quite harshly deformed compared to the original one, as much by the recording process as by the deterioration of the storage device.

In many other fields, like *photography* or *speech recordings*, Machine Learning techniques have shown an interesting ability to recover missing information, and to clean messy data. Old musical audio recordings could benefit from the use of such techniques.

The production of more accurate audio recordings starting from messy ones could help music historians in their research process, as well as musicians in their artistic work. Even more, many Music Information Retrieval (MIR) researches could see their results improved when being able to work on cleaner musical recordings.

1.2 Approach

In the following work, we assume that three main steps are needed to retrieve a clean signal starting from a messy one:

- Finding a representation of audio signal: this representation has to be complex enough to lose the smallest amount of information, and yet to be usable by Machine Learning techniques,
- Removing the noise: this is the most remarkable characteristic of *old* audio recordings, the somehow white noise (that we will define later in this work), which seems induced by the recording method,
- Completing the missing frequencies: old recording tools did not cover the entire frequencies spectrum, nor the whole humanly detectable frequencies ($20Hz$-$20kHz$).

[1] Hear the recording on Youtube

2 State of the Art

2.1 Audio representation in Machine Learning

The following section follows the presentation of A. Natsiou and S. O'Leary in [21].

2.1.1 Raw audio

The most informative form of digital representation of audio signal is **raw audio**. In such a representation, an audio signal is sampled, and is then represented by a sequence of amplitudes discretized at a determined frequency. As given by C. Shannon in [31], this frequency should be at least two times higher than the maximal desired sampled frequency.

This representation is extremely precise (the precision depending on the sampling frequency and on the size of the encoding of the amplitudes), even though it discretizes a continuous audio signal.

This signal can be processed to be quantified (while being less informative, the signal becomes more usable by Machine Learning techniques), by linear transformations, or by non-linear ones such as the μ-law, a parametric non-linear transformation of the signal, which is used in a huge diversity of works ([24], [23], [3]):

$$f(x) = sgn(x)\frac{ln(1+\mu|x|)}{ln(1+\mu)}$$

2.1.2 Spectrograms

Spectrogram is a visual representation of an audio signal obtained by the application of the Short-Time Fourier Transform. In a spectrogram, the phases obtained by the Fourier Transform are discarded (only the absolute values are kept). The visual representation is obtained by plotting the frequencies over the time.

However, Mel-spectrogram (see figure 1) have been widely used in the literature: the previously presented spectrogram is transformed by the mel-scale relation

$$mel(f) = \frac{1000}{log_{10}2}log_{10}(1+\frac{f}{1000})$$

where f stands for the frequency (in $Hertz$). This transformation (or the *log* of this transformation) mimics the perception of the human auditory filters and has shown interesting results in music classification and analysis ([7], [27], [13]).

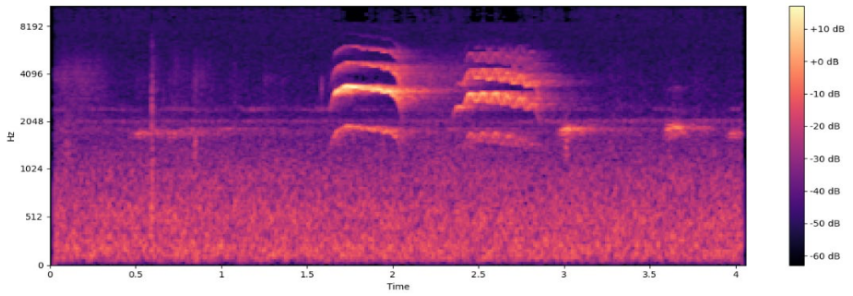

Figure 1: Mel-spectrogram derived from audio. Image from [13]

2.2 Noise cancellation

The first part to recover audio from a noisy recording consist in denoising (i.e. removing artefacts added either by the recording technique, or the recording support, or the deterioration of the recording support).

In the literature, three different kind of noises are pointed out:

- White noise, which is spread on all the frequency spectrum and over time ([33]), and where the data is slightly randomly modified,
- Impulse noise, where there are abnormal points in the data: this might be either:
  - Salt and Pepper Impulse Noise (SPIN),
  - Random Valued Impulse Noise (RVIN).

The technique used to denoise depends on the kind of noise we face. Some of the techniques will perform great when applied to a certain kind while being extremely disastrous on another.

However, some researchers took in the literature the stronger assumption that noise could be considered as anything that bothers the comprehensibility of the information. As an example, in [10], noise is considered to be anything decreasing the clarity of a speech signal. Therefore, they define *speech enhancement* as the task of maximizing the **perceptual** quality of speech signals.

We claim that this notion of perception is important in music signal denoising as well, and thus we propose a definition for music signal denoising which takes it into account:

Music signal denoising is the task of maximizing the musical (melodic, harmonic, rhythmic and dynamic) perceptual quality of a music signal.

Therefore, as we will see in section 3.1, the perceptual aspect also has to be considered when judging the quality of the produced results by any music signal denoising method.

2.2.1 Classical techniques

Siya, Gouri and Ugam reported in [20] a great sum up of the existing techniques to denoise audio. They studied the particular application case of *podcasts*, for which a correct audio quality was difficult to get during the COVID-19 pandemic, as they could only be recorded by separate sources of recordings.

Such audio recordings contains the three kinds of noise we have described above.

Their sum up focuses then on all of those three types of noise. They make the statement that any audio recording bears noise, assuming that "pure clean" audio does not exist with the current existing recording techniques. Thus, they conclude that audio denoising has not only fields in *harshly* noised recordings, but in any context of use of audio recordings.

They present the following methods:

- Kalman filter ([26]),
- Boll Spectral Substraction ([26]),
- White Gaussian Noise Filter ([26]),
- Least Mean Squared (LSM) Adaptative Filter ([16]),
- SDROM ([22]),
- Spectral Gate Algorithm ([29]).

Moreover, we will describe a filtering method, extend from Kalman Filter, presented in [34], and a method based on Wavelet transform, proposed by Srikar, Gundu and Rajendra Prasad in [35]. A wavelet (see figure 2) is a wave-like oscillation, starting at 0, and oscillating around it.

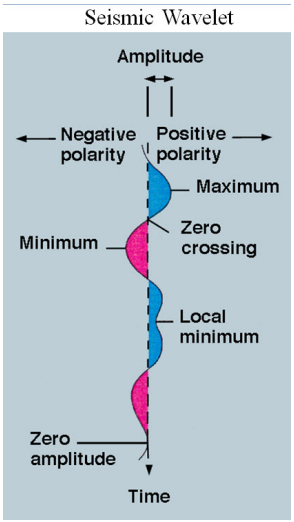

Figure 2: An example of wavelet (Image from Wikipedia)

Kalman Filter Kalman Filter is a filter which estimates a signal, based on noised measures. It is a recursive estimator, which uses all the previous observations in order to predict a cleaner version of the actual observation. It is used in a huge diversity of fields, including radars, oceanography, economy,...

The splitting of an audio recording to a bunch of samples produces data which is suitable for this method. Das and Orchisama proposed in a conference ([8]) an application of such a filtering technique on noisy speech recordings.

Extension In [34], Harinarayanan and al. presented a new filtering method based on Kalman Filtering. In order to preserve the significant part of the audio signal, they

- Compute the inner correlation of the audio signal, assuming that speech or music has a greater correlation, or a more remarkable periodicity than noise,
- Detect the fricatives, in the case of speech, to avoid removing them as they were impulse noises,
- Measure the average energy of the audio signal, assuming that the lower the energy is, the more likely the observed signal is to be noise.

Those three criteria are necessary to consider a given frame of the audio signal to be noise. A frame which is marked as noisy is then used to evaluate the multiple parameters of the Kalman filter.

Boll Spectral Substraction Introduced in [4], Boll's Spectral Substraction is a method to remove a defined signal from another, substracting the Short-Time Fourier Transform of the main signal and the one to remove. In [28], the noise of an audio signal is evaluated on frames using a gaussian window instead of a classical fixed window, then the noise is removed from the original noisy signal using Boll's Spectral Substraction.

White Gaussian Noise Filter In [33], Singh and Laxmi propose an algorithm to remove white noise from audio signals and speech signals. They use Discrete Wavelet Transform to convert audio signal into the wavelet domain, and assume that the low amplitudes in the wavelet space correspond to the noise.

Many transformation exist, such as Discrete Wavelet Transform, to represent audio signals with a wavelets series. They threshold those coefficient to obtain less noisy data, and obtain the denoised audio by performing an inverse transformation from the wavelet data to audio signal. By trying different thresholds, and different wavelet techniques, they achieve an accuracy of 98.3%.

Least Mean Squared (LSM) Adaptative Filter Least Mean Squared (LSM)[16] (for Least Mean Squares) algorithm is an algorithm that determines the most efficient filter by steepest descent on the error function, evaluated between the predicted signal and the clean one. This method requires the clean signal in order to compute the error and to modify the filter.

SDROM According to [22], the proposed method is intended to target impulse noises. They point out that classical filtering methods do not make difference between corrupted (noisy) or clean samples, so they propose a method based, first, on the detection of the samples containing such impulse noises, and then, on the estimation of their clean version.

They consider impulse noises as being completely corrupted samples, i.e. the n^{th} sample $x(n)$ is described as

$$x(n) = \begin{cases} s(n) & \textit{with probability } 1 - p_e \\ \eta(n) & \textit{with probability } p_e \end{cases}$$

where $s(n)$ is the clean n^{th} sample, $\eta(n)$ is *identically distributed, independent random process with an arbitrary underlying probability density function* and p_e is the probability for a sample to be corrupted.

They compute the differences of rank in a window of 5 samples surrounding the observed ones and decide that the sample is effectively an impulse noise if one of the difference exceeds a defined threshold. Then, a detected impulse is replaced by the rank-order mean.

We point out their final remarks, which states that the scratches and glitches appearing on gramophone recordings can be modeled by impulse noises, and that their method is really suitable for detecting (and removing) them. Their solution does not produce the best results in the literature but they have the advantage to be computationally simple, and to focus on specific kind of noises.

Spectral Gate Algorithm Spectral Gate Algorithm [29] is a filtering method, where the a noise signal is divided into to different signal, the splitting being determined by a threshold on the frequency spectrum. This threshold is determined by the use of statistics on the Fast Fourier Transform (FFT) applied on a signal containing a prototypical (an approximation of) the noise to remove. Once the transformed noise signal is computed, it used as a substracting mask on the noisy signal. Finally, the transformed and masked noisy signal is reverted to audio signal.

Interestingly, Siya, Gouri and Ugam reported that this method reports the best Signal Noise Ratio (SNR) (see 3.1) value from all the previously introduced denoising method.

DWT Variants As stated earlier, Direct Wavelet Transform (DWT) describes a lot of techniques based on the transformation of the signal to the wavelet space. Those techniques are mainly based on the following algorithm:

- Translate the signal into wavelets,
- Threshold the wavelets,
- Transform the wavelets into signal.

In [35], Srikar and Prasad studied three variants: Double Density Dual Tree Discrete Wavelet Transform, Double-Density Discrete Wavelet Transform and Dual-Tree Discrete Wavelet Transform. Those variants explore different ways to transform the signal into wavelets.

Comments As we can see, many techniques exist for audio denoising, without using Machine Learning. They all present their advantages, but they all need a first assumption on the nature of the noise to remove. Those approaches are experience-driven, not data-driven. However, they showed interesting results, and are efficient to compute. Those existing methods will help to benchmark any new method, would it be purely analytic, or a Machine Learning technique.

In [20], Siya, Gouri and Ugam concluded Spectral Gate Algorithm to give the best results. Our goal is to define a method that outperforms this one.

2.2.2 Machine Learning techniques

Machine Learning applied on audio literature is extremely recent. All the available literature make use of network architecture, although their topology, structure and learning dynamics vary.

Also, the main audio source that is used is speech: never music is used. Speech denoising (and speech enhancement) has some complexities (as the presence of fricatives, see [34]). The important challenges that communication faces (see [17]), as speech diffusion or communication, hearing aids design, or speech recognition, may justify the greater attention speech enhancement receives in comparison with music.

We claim that such differences is important to take into account, as music often implies polyphony. Polyphony describes the presence of multiple pitches occurring together (the more often, those pitches can be related in time (as counterpoint), or in pitch (as harmony)..

The upcoming section presents some of those existing techniques:

- Deep Networks,
  - Using supervised learning ([37] and [39]),
  - Using unsupervised learning ([19]),
- Dense Networks ([15]),
- Auto-encoders/decoders ([10]),
- Generative Adversarial Network (GAN) ([25]).

Deep Networks with supervised learning In [37], Xu, Du, Dai and Lee have tried to denoise speech recordings using a Deep Network architecture. Their main motivation was the removal of the artifact noises that classical denoising techniques usually bring. They train their Deep Network, on extracted log-spectral features, by back-propagating, with objective function being the Minimum Mean Squared Error (MMSE):

$$E = \frac{1}{N}\sum_{n=1}^{N}\sum_{d=1}^{D}(Xe_n^d(W_1^l b^l) - X_n^d)^2 \tag{1}$$

where E stands for the Mean Squared Error (MSE), $Xe_n^d(W_1^l b^l)$ stands for the d^{th} enhanced sample and X_n^d stands for the d^{th} clean signal. N stands for the batch size, and D for the size of the log-spectral feature vector. Their results were convincing in comparison with the existing classical methods,

and they evoke the possibility for Deep Neural Network (DNN) to recover the spectrum buried by the noise in the noisy signals.

In [39], they enhanced this denoising architecture. First, they figured out thanks to [32] that phase information was important, at least in human speech data. The main improvements they brought was purposed to detect and remove unforeseen impulse noises, occurring due to the recording conditions. They implemented a drop-out strategy, and a Noise-Aware Training (NAT).

Noise-Aware Training arises from the assumption that DNN can learn a better mapping when the nature of the noise is known during the learning phase. Therefore, an estimation of the noise is added here in the feeding of the network, in order to better predict the clean speech.

Other deep network structures as Long Short-Term Memory (LSMT) networks have also been tried, showing great results, as in [36].

Deep Networks with unsupervised learning In unsupervised learning, some statistical assumptions have to be done on input data in order to predict output data. One could call some of those assumptions *priors*. An example of such a *prior* might be the statistical distribution of the gaussian noise applied on the input signal.

In [19], Michelashvili and Wolf explored audio denoising using unsupervised learning, and determines as priors which part of the signal has the highest estimated level of uncertainty, i.e. the most part which is the most likely to be noised. Therefore, a mask is computed and applied on the input signal, and one of the classical denoising method is applied on the signal.

However, the main point remains to determine this uncertainty mask: the proposed algorithm takes into input a noisy signal y, in Short-Time Fourier Transform (STFT) format, and outputs a mask of the same dimensions, with values in the range $[0, 1]$. Then, using this predicted mask, they use Log-Spectral Amplitude (LSA) method as classical denoising method.

Their method shows interesting results (outperforming most of the existing unsupervised methods), but gives lower results than the most efficient supervised methods. However, we have to keep in mind that unsupervised learning has the strong advantage of not needing any clean version of the input signal, as it does not require to compute any error measurement between the clean signal and the cleaned signal in order to learn.

Dense Network Despite the fact that Deep Network architectures have shown many interesting advances in Machine Learning, the depth of a net-

work is not the only parameter that can be explored.

A Dense Network is a network in which the layers are connected not only to the next one (see figure 3), but also to the further ones. As such, the information is used by a layer as it was in the input of the network as well as how it has been transformed by the different layers.

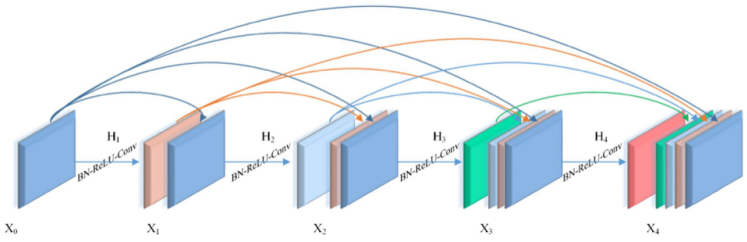

Figure 3: Example of a dense network. Image from [15]

In [15], Li, Xu, Zhang and Liu explored the use of Dense Networks for the removal of impulse noise in images. Their results outperforms from far the results of the existing techniques that use deep networks, with a notably smaller and lighter size of network.

We find this technique promising but unfortunately, this path has not been explored yet, in the literature, on audio signals.

Autoencoders and decoders The term *autoencoder* describes a huge diversity of different networks, having as main common structure a dimension reduction of the input space to a small (often extremely) representation of the data (see figure 4). In denoising application, they are often combined, and trained together, with a *decoder* which learns to recover the data from the reduced representation.

In [10], Défossez, Synnaeve and Adi presented an audoencoder/decoder method which maps a noisy signal in wave form towards the clean version.

They use an architecture called DEMUSC, proposed in [9]. This architecture was first intended for music source separation problem. Music source separation ([5]) is a MIR problem which aims at recovering the different sources of audio, when having only access to one signal with all the sources mixed together. This might be used to remix, re-balance or up-sample an audio recording.

DEMUCS consists in a *multi-layer convolutional encoder and decoder, with U-Net skip connections, and a sequence modeling network on the output of the decoder.*

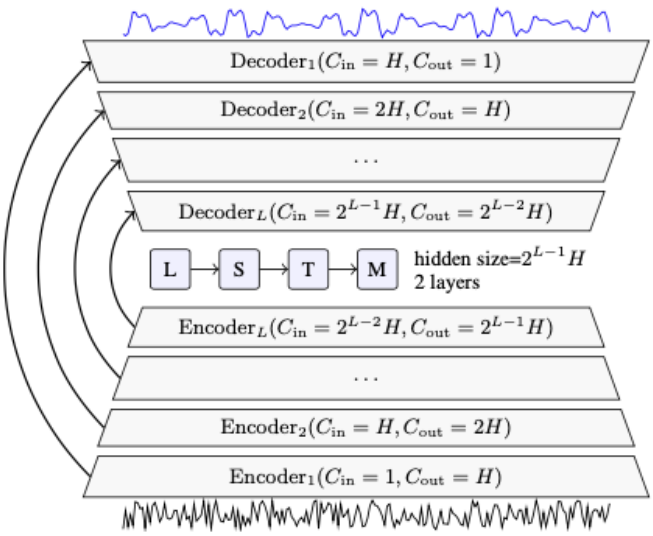

Figure 4: An example of autoencoder/decoder architecture. Image from [10]

Their architecture takes as input the audio signal in wave format, and computes two error evaluations:

- L1 loss: $||y - \hat{y}||_1$
- $L_{stft}(y, \hat{y})$): $\frac{|||STFT(y)|-|STFT(\hat{y})|||_F}{|||STFT(y)|||_F} + \frac{1}{T}||log|STFT(y)|-log|STFT(\hat{y})|||_1$

and therefore, their goal is to minimize

$$\frac{1}{T}[||y - \hat{y}||_1 + \sum_{i=1}^{M} L^i_{stft}(y, \hat{y})] \tag{2}$$

where M stands for the number of STFT losses, L^i_{stft} applies the STFT loss at different resolutions, and T is the length of the signal.

Interestingly, the results this method achieves match the state-of-the-art, but using the waveform signal directly as input: therefore, the computation is faster, and the signal does not suffer from any distortion caused by the transformation from the waveform signal to any other format.

However, the computation of the STFT loss implies the STFT. This method is thus not entirely based on wave signal.

Generative Adversarial Networks In [25], Pascual, Bonafonte and Serrà proposed Speech Enhancement Generative Adversarial Network (SEGAN), a GAN purposed to denoise speech signal. A GAN (see figure 5) has an architecture in two parts:

- A Generator, which aims to produce an output as likely as possible to be sampled from the clean signal distribution,
- A Discriminator, which aims to detect whether or not a produced input is sampled from the clean signal distribution, or is output by the generator.

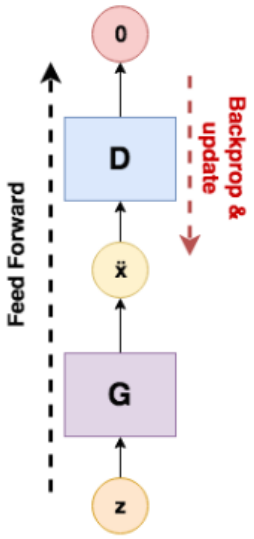

Figure 5: An example of GAN. Image from [25]

In their case, the generator has a structure of autoencoder-decoder. It takes at input the noisy audio signal in waveform signal, and outputs a signal in waveform that is used as input for the discriminator.

The generator learns by minimizing the L_1 error, and the discriminator learns by minimizing the absolute error between the predicted and the expected class.

They show great results evaluated by statistical observations as well as subjective evaluations.

2.3 Frequencies completion and audio inpainting

This second part focuses on the spectrum completion. We assume that old recordings not only suffer from extraneous noises, but that old recording techniques along with recording material deterioration cuts into the frequency spectrum. Moreover, all of those audio signals are recorded on mono channel, and continuous recording.

The reconstruction of missing data inside a given vector is well known in image treatment as *image inpainting*. This problem, in the case of images, is concerned by the reconstruction of a portion of the image, based on the available content.

In audio processing, *audio inpainting* has been studied since 2011. The idea came from the image treatment, and Adler, Emiya, Jafari, Elad, Gribonval and Plumbley proposed the application of such a problem on audio, in [2]. Their assumption, in this paper, was to consider noisy samples as missing, and therefore to reconstruct them starting from the surrounding context.

2.3.1 Classical techniques

Before the rise of Machine Learning applied on audio, many problems that could require audio inpainting were solved using a huge diversity of methods. As an example (given in [2]), impulse noises, scratches, clipping (audio clipping occurs in wave signals, when the maximum range in an acquisition system is exceeded [2], and the audio waveform has to be truncated (see figure6), and those problems have been solved by an huge diversity of different techniques, as interpolation ([14]), extrapolation ([1]), imputation ([6]), bandwidth expansion ([11]), ...

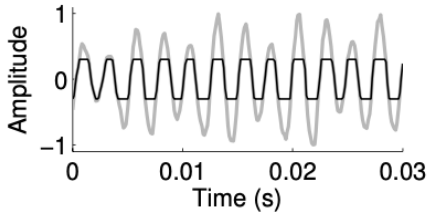

Figure 6: Clipped version of an audio signal. Image from [2]

However, those methods have been outperformed, by far, by Machine Learning techniques that we will present later on, and were all solved by

the same methods: therefore, we refer the reader to the presented sources ([14], [1], [6], [11]) for historical information, but we will not focus on those methods in our work.

Orthogonal Matching Pursuit In [2], Adler, Elad, and Plumbley propose a method for audio inpainting. In their work, they consider noisy frame as missing, and therefore, they need a sparse representation of the data. Their audio inpainting method is based on sparse dictionaries.

They study the impact of the kind of audio signal (music, speech) processed, as well as the impact of the sampling rate. They also show the impact of the length of the missing samples, which is greater than the impact of the proportion of the missing samples. This might be explained by the fact that the reconstruction is performed with the help of the surrounding information, and the longer are the missing samples, the further the known neighbours. The information the neighbours is (intuitively) decreasing with the distance in time they are separated by.

The results defined the initial state-of-the-art for the audio inpainting. However, as it will be presented in the next section, the use of linear methods to solve this problem is by far outperformed by non-linear methods as the one used in the framework of Machine Learning.

2.3.2 Machine Learning techniques

Machine Learning helps the audio inpainting problem in adding non-linearity in the reconstruction process. As in the denoising section, the art is quite recent, and therefore the existing approaches use only Networks, exploring the different possible architectures. We will explore some of them:

- Deep Networks ([30], [18]),
- Generative Adversarial Network ([12]),
- Autoencoders/decoders ([38]).

Deep Networks In [30], Serra, Busson, Guedes and Colcher proposed a Deep Network model to inpaint audio samples. In order to corrupt the data, they randomly delete audio frequencies on the STFT version of the audio data. The audio inpainting therefore becomes a special case of image inpainting one could name *spectrogram inpainting*.

They described three different deep network architectures:

- U-Net (and a variant U-Net V2), we already presented, being a convolutional network with skip connection between the layers,
- Res-U-Net (Res-U-Net V2), a particular case of U-Net, where the input data goes through an U-Net (U-Net V2), and is again injected into the last layer of the network,
- Attention U-Net, another particular case of U-Net, where the network uses the skip connection through *attention blocks*, non-linearly summing those skip connections with the up-sampling information.

All the metrics used in the evaluations of those different methods used in [30] showed that Res-U-Net V2 is the best inpainting method among all the implemented methods. Although this paper does not compare the results with other paper's ones (probably, due to the lack of possible comparison), this defines the state-of-the-art results for audio inpainting using Deep Networks.

Generative Adversarial Networks In [12], Ebner and Eltelt proposed a GAN for audio inpainting, in order to complete missing samples in an audio recording.

Despite the interesting results they got, the focus of such a method was not on the frequency reconstruction but rather on the whole samples reconstruction (see figure 7). However, two strategies could be explored:

- Either considering that a sample where a frequency is missing is corrupted, and therefore considering it as missing,
- Or using the same GAN structure, to output the completed sample.

In the case of old audio recordings, the first strategy does not seem to be interesting: all the samples tend to have the same missing frequency bandwidth. However, we could explore the second strategy in order to recover those missing frequencies.

Autoencoders/decoders Yeh, Chen, Lim, Schwing, Hasegawa-Johnson and Do proposed in [38] an Autoencoder/decoder architecture for image inpainting.

They have shown extremely great results (outperforming, from far, the existing inpainting methods for large gapes in images). However, this is the current state-of-the-art of Autoencoders applied on inpainting, and therefore, this path could as well be explored for audio inpainting.

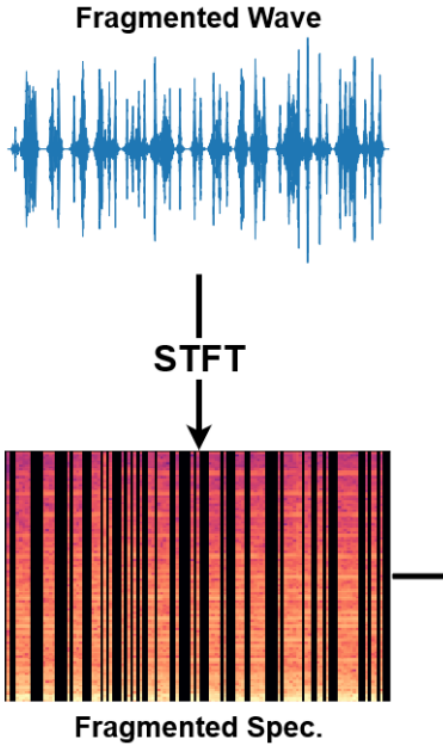

Figure 7: An example of fragmented audio: some sample are missing in the wave signal, and therefore are missing in the associated spectrogram. Image from [30]

The architecture of an Autoencoder/decoder has already been described earlier. Their strategy is to reconstruct the missing data by learning a input-to-input mapping: the missing information is then considered as a slight variation of the input, and is estimated from the rest of the available data.

An interesting point their strategy seems to target is the use of large-scale information for retrieving the missing data, and not only the surrounding neighbour information.

3 Experiment process

3.1 Evaluation methods

In [20], the others evaluated the performances of the denoising algorithm by the computation of the SNR, given by the formula

$$SNR = \frac{Power\ of\ the\ signal}{Power\ of\ the\ noise} \tag{3}$$

We can also point the use of the Root Mean Square Error (RMSE)

$$\sqrt{\frac{\sum_{i=1}^{n}(r_i - s_i)^2}{n}} \tag{4}$$

used along the SNR in [35], or the MMSE, given in 1.

Let us also introduce some metrics used in [30]:

- Peak Signal-to-Noise Ratio (PSNR): $PSNR(x, y) = 10log_{10}(\frac{MAX^2}{MSE(x,y)})$, where MAX is the maximal possible value of the data (in their case, the maximal pixel value of the spectrogram),
- Normalized Root Mean Square Error (NRMSE): $NRMSE(x, y) = \frac{\sqrt{MSE(x,y)}}{y_{max} - y_{min}}$

In [10], Défossez, Synnaeve and Adi evaluated their speech signal denoising by computing the Word Error Rates (WER) between the a priori text and the text output by the best Automatic Speech Recognition (ASR) systems in the state-of-the-art. We could also use such techniques to evaluate any music signal denoising method. However, further research should be done in order to find the best recording-to-score software. Moreover, music information is not entirely represented by those transcriptions, the more often, only pitch and rhythm is transcribed. This form of evaluation needs to be explored.

In [34], Harinarayanan and al. used a group of five *expert listeners* to compare the produced denoised samples produced by their method, with other available techniques. As it is also stated in [10], this subjective evaluation should be taken into account in order to evaluate the quality of the produced results.

3.2 Datasets

3.2.1 Data generation

In the literature, the most used technique to have noisy data consists in adding white (gaussian) noise to the data before it is processed. In order to target more specifically the typical white and crackling noise of old gramophone recording, we could also use extracts of *silence* recorded on gramophones, and consider the recorded noise as being the noise we add to clean signals.

In [39], Xu, Du, Dai and Lee showed that training a Machine Learning model (in their case, a Deep Network) on synthetically noised data produced efficient models for real-life recording.

We can also decide to remove a defined bandwidth of the audio signal to train any Machine Learning model to predict the missing frequencies.

3.3 Methodology

Of course, this section is a small prevision of the plan of the work.

We consider *old musical audio recordings* as being the ones prepared as stated in section 3.2.1.

As stated in the introduction, we claim that the recovering of old musical audio recordings requires two main steps (we consider that the audio representation highly depends on the kind of model we implement):

- The denoising of the recording,
- The completion of the recording.

Both steps have been explored separately in the literature. The denoising part benefits from an already huge attention, and many models have shown interesting results on speech recordings, our goal is first to determine their efficiency on musical recordings. The second part being extremely recent in the literature, most of the existing network architectures have to be tested on (1.) audio and (2.) musical recordings.

Also, some methods (as Autoencoders and GAN) could learn to perform those two steps at once. We could benchmark the efficiency of stacking models together in regard to an *all-in-one* model.

Finally, the benchmark of any model will be performed using the methods described in section 3.1.

4 Conclusion

In the present document, we have defined the problem of recovering old audio recordings.

We have divided the problem into two main components, for which we have presented the existing solutions, on the classical side as well as on the Machine Learning side: the denoising and the completion of the audio recording. This second part is now referred as audio inpainting.

We have proposed a framework of study and a methodology, as well as some statistics we can use to benchmark our models.

4.1 To go further

The digital treatment of continuous signals requires their sampling. This discretization loosens the information, in contrast with continuous audio, and therefore, one could expect the best sampling (i.e. the most precise). Unfortunately, continuous information is not always available, and the sampling might be too low for a good comprehensibility. Therefore, we could explore the existing methods for up-sampling discrete signals.

5 Bibliography

References

[1] Jonathan S. Abel and Julius Orion Smith. "Restoring a clipped signal". In: *[Proceedings] ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing* (1991), 1745–1748 vol.3.

[2] Amir Adler et al. "Audio Inpainting". In: *IEEE Transactions on Audio, Speech, and Language Processing* 20 (2012), pp. 922–932.

[3] Mikołaj Bińkowski et al. *High Fidelity Speech Synthesis with Adversarial Networks*. 2019. DOI: 10.48550/ARXIV.1909.11646. URL: https://arxiv.org/abs/1909.11646.

[4] Steven F. Boll. "Suppression of acoustic noise in speech using spectral subtraction". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 27 (1979), pp. 113–120.

[5] Estefanía Cano et al. "Musical Source Separation: An Introduction". In: *IEEE Signal Processing Magazine* 36 (Jan. 2018). DOI: 10.1109/MSP.2018.2874719.

[6] Martin Cooke et al. "Robust automatic speech recognition with missing and unreliable acoustic data". In: *Speech Communication* 34.3 (2001), pp. 267–285. ISSN: 0167-6393. DOI: https://doi.org/10.1016/S0167-6393(00)00034-0. URL: https://www.sciencedirect.com/science/article/pii/S0167639300000340.

[7] David Dalmazzo and Rafael Ramirez. "Mel-spectrogram Analysis to Identify Patterns in Musical Gestures: a Deep Learning Approach". In: (Nov. 2020).

[8] Orchisama Das. "Kalman Filter in Speech and Music". In: July 2017. DOI: 10.13140/RG.2.2.36294.78403.

[9] Alexandre Défossez. "Hybrid Spectrogram and Waveform Source Separation". In: *Proceedings of the ISMIR 2021 Workshop on Music Source Separation*. 2021.

[10] Alexandre Defossez, Gabriel Synnaeve, and Yossi Adi. *Real Time Speech Enhancement in the Waveform Domain*. 2020. DOI: 10.48550/ARXIV.2006.12847. URL: https://arxiv.org/abs/2006.12847.

[11] Martin Dietz et al. "Spectral Band Replication, a Novel Approach in Audio Coding". In: *Journal of The Audio Engineering Society* (2002).

[12] P. P. Ebner and A. Eltelt. *Audio inpainting with generative adversarial network*. 2020. DOI: 10.48550/ARXIV.2003.07704. URL: https://arxiv.org/abs/2003.07704.

[13] Kevin Gunawan et al. *A transfer learning strategy for owl sound classification by using image classification model*. Nov. 2020.

[14] A. Janssen, R. Veldhuis, and L. Vries. "Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 34.2 (1986), pp. 317–330. DOI: 10.1109/TASSP.1986.1164824.

[15] Guanyu Li et al. "Densely connected network for impulse noise removal". In: *Pattern Analysis and Applications* 23 (Aug. 2020). DOI: 10.1007/s10044-020-00871-y.

[16] Yang Liu, Xiao Mingli, and Tie Yong. "A Noise Reduction Method Based on LMS Adaptive Filter of Audio Signals". In: (Nov. 2013). DOI: 10.2991/icmt-13.2013.123.

[17] Philipos C. Loizou. *Speech Enhancement , Theory and Practice*. 2013. ISBN: 9781138075573.

[18] Andrés Marafioti et al. *Audio inpainting of music by means of neural networks*. 2018. DOI: 10.48550/ARXIV.1810.12138. URL: https://arxiv.org/abs/1810.12138.

[19] Michael Michelashvili and Lior Wolf. "Audio Denoising with Deep Network Priors". In: *ArXiv* abs/1904.07612 (2019).

[20] Siya Naik, Gouri Bhatikar, and Ugam Gaude. "Analysis of Best Algorithm for Noise Reduction in Podcasting". In: *International Journal of Scientific Research in Science and Technology* (May 2021), pp. 246–250. DOI: 10.32628/IJSRST218342.

[21] Anastasia Natsiou and Seán O'Leary. "Audio representations for deep learning in sound synthesis: A review". In: *CoRR* abs/2201.02490 (2022). arXiv: 2201.02490. URL: https://arxiv.org/abs/2201.02490.

[22] Val O et al. "An Efficient Method for the Removal of Impulse Noise from Speech and Audio Signals". In: (Nov. 2000).

[23] Aaron van den Oord et al. *Parallel WaveNet: Fast High-Fidelity Speech Synthesis*. 2017. DOI: 10.48550/ARXIV.1711.10433. URL: https://arxiv.org/abs/1711.10433.

[24] Aaron van den Oord et al. *WaveNet: A Generative Model for Raw Audio*. 2016. DOI: 10.48550/ARXIV.1609.03499. URL: https://arxiv.org/abs/1609.03499.

[25] Santiago Pascual, Antonio Bonafonte, and Joan Serrà. *SEGAN: Speech Enhancement Generative Adversarial Network*. 2017. DOI: 10.48550/ARXIV.1703.09452. URL: https://arxiv.org/abs/1703.09452.

[26] S Prasadh, Sai Natrajan, and S. Kalaivani. "Efficiency analysis of noise reduction algorithms: Analysis of the best algorithm of noise reduction from a set of algorithms". In: Nov. 2017, pp. 1137–1140. DOI: 10.1109/ICICI.2017.8365318.

[27] Konstantinos Pyrovolakis, Paraskevi K. Tzouveli, and G. Stamou. "Multi-Modal Song Mood Detection with Deep Learning". In: *Sensors (Basel, Switzerland)* 22 (2022).

[28] A.SubbaRami Reddy, K. Satya Prasad, and Jyothishmathi. "Speech Enhancement using Boll's Spectral Subtraction Method based on Gaussian Window". In: 2014.

[29] Tim Sainburg, Marvin Thielk, and Timothy Q Gentner. "Finding, visualizing, and quantifying latent structure across diverse animal vocal repertoires". In: *PLoS computational biology* 16.10 (2020), e1008228.

[30] Arthur Costa Serra et al. "Quality Enhancement of Highly Degraded Music Using Deep Learning-Based Prediction Models for Lost Frequencies". In: *Proceedings of the Brazilian Symposium on Multimedia and the Web* (2021).

[31] C.E. Shannon. "Communication in the Presence of Noise". In: *Proceedings of the IRE* 37.1 (1949), pp. 10–21. DOI: 10.1109/JRPROC.1949.232969.

[32] Guangji Shi, Maryam Modir Shanechi, and Parham Aarabi. "On the importance of phase in human speech recognition". In: *IEEE Transactions on Audio, Speech, and Language Processing* 14 (2006), pp. 1867–1874.

[33] Nishan Singh and Dr. Vijay Laxmi. *Audio Noise Reduction from Audio Signals and Speech Signals*.

[34] Deepen Sinha, Shamail Saeed, and Aníbal Ferreira. "A Novel Automatic Noise Removal Technique for Audio and Speech Signals". In: *Audio Engineering Society - 123rd Audio Engineering Society Convention 2007* 3 (Jan. 2007).

[35] Gundu Srikar and Ch Rajendra Prasad. "An Enhanced Audio Noise Removal Based on Wavelet Transform and Filters". In: (Dec. 2017), pp. 3111–3121.

[36] Felix Weninger et al. "Speech Enhancement with LSTM Recurrent Neural Networks and its Application to Noise-Robust ASR". In: vol. 9237. Aug. 2015. ISBN: 978-3-319-22481-7. DOI: 10.1007/978-3-319-22482-4_11.

[37] Yong Xu et al. "An Experimental Study on Speech Enhancement Based on Deep Neural Networks". In: *Signal Processing Letters, IEEE* 21 (Jan. 2014), pp. 65–68. DOI: 10.1109/LSP.2013.2291240.

[38] Raymond A. Yeh et al. *Semantic Image Inpainting with Deep Generative Models*. 2016. DOI: 10.48550/ARXIV.1607.07539. URL: https://arxiv.org/abs/1607.07539.

[39] Lirong Dai Yong Xu Jun Du and Chin-Hui Lee. "A Regression Approach to Speech Enhancement Based on Deep Neural Networks". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23 (2015), pp. 7–19.

Glossary

ASR Automatic Speech Recognition. 19

DNN Deep Neural Network. 11

DWT Direct Wavelet Transform. 9

FFT Fast Fourier Transform. 9

GAN Generative Adversarial Network. 10, 14, 17

LSA Log-Spectral Amplitude. 11

LSM Least Mean Squared. 6, 8

LSMT Long Short-Term Memory. 11

MIR Music Information Retrieval. 2, 12

MMSE Minimum Mean Squared Error. 10

MSE Mean Squared Error. 10

NAT Noise-Aware Training. 11

NRMSE Normalized Root Mean Square Error. 19

PSNR Peak Signal-to-Noise Ratio. 19

RMSE Root Mean Square Error. 19

RVIN Random Valued Impulse Noise. 5

SEGAN Speech Enhancement Generative Adversarial Network. 14

SNR Signal Noise Ratio. 9, 19

SPIN Salt and Pepper Impulse Noise. 5

STFT Short-Time Fourier Transform. 11, 13, 16

WER Word Error Rates. 19