Application of Machine Learning Methods for Enhancing the Quality of Medical Audio Recordings: Comparative Analysis of Classical and Modern Approaches

Nataliya Boyko
Lviv Polytechnic National University,
Lviv, Ukraine

Petro Slobodian
Lviv Polytechnic National University,
Lviv, Ukraine

ABSTRACT

The aim of the study is to solve the problem of noise in audio recordings and improve sound quality using existing machine learning methods; compare different existing methods. In order to test, analyze and compare methods of machine learning based on sound processing problem, it is proposed to use several different approaches. The work will use both classical methods of audio signal processing, such as the wiener filter and spectral subtraction, and more modern ones, namely convolutional neural networks. Each of these methods has its own pros and cons that will be analyzed during experiments, in order to determine in which case which method will be useful. Using these methods will allow for in-depth analysis and comprehensive results for audio processing. Based on the research, it was determined that Spectral subtraction performs slightly better than the Wiener filter. This is evidenced by both the PESQ scores for the two methods and the audiovisual analysis. Among all the selected methods, convolutional neural networks perform the best, and based on the metrics, conclusion was made that the best results for CNN’s can be achieved using L1/L2 regularization and Dropout. Further research may include investigating new CNN architectures for audio de-noising, exploring the possibilities of using other types of neural networks such as Recurrent Neural Networks, Generative Adversarial Networks for audio de-noising.

General Terms

Convolutional Neural Networks, Mean Square Error, Mean Absolute Error, Structural Similarity Index, Peak Signal-To-Noise Ratio, Perceptual Evaluation of Speech Quality.

Keywords

Audio, De-noising, Wiener filter, Spectral subtraction, Audio processing, Speech enhancement, Noise estimation, Machine learning model.

1. INTRODUCTION

The work is dedicated to sound processing, namely, de-noising of individual audio materials. In the modern world, sound is an integral part of our lives. It accompanies us everywhere: music, voice messages in various social networks, videos, etc. Sound is not just air vibrations perceived by our ears; it is one of the key aspects of information transfer between people. The use of sound covers a large number of areas of our lives. First of all, communication, because sound is one of the main components of communication; medicine, because sound is used to diagnose diseases (ultrasound); technology - sound signals are used in many technical devices, from cars to computers, and help us navigate in space, receive information about the operation of devices, and control them. Art, media, education, entertainment, etc. - the list goes on. Sound is an integral part of most people's lives [3, 7].

The work is dedicated to sound processing, namely, de-noising of individual audio materials. In the modern world, sound is an integral part of our lives. It accompanies us everywhere: music, voice messages in various social networks, videos, etc. Sound is not just air vibrations perceived by our ears, it is one of the key aspects of information transfer between people. The use of sound covers a large number of areas of our lives. First of all, communication, because sound is one of the main components of communication; medicine, because sound is used to diagnose diseases (ultrasound); technology - sound signals are used in many technical devices, from cars to computers, and help us navigate in space, receive information about the operation of devices, and control them. Art, media, education, entertainment, etc. - the list goes on. Sound is an integral part of most people's lives [1, 5].

Modern technology makes it possible to record and transmit sound. However, due to various reasons, such as a poor-quality recording device, poor recording conditions, etc., the quality of the audio often deteriorates and noise - unwanted sound signals – occurs [13].

The aim of this study is to solve the problem of noise in audio recordings and improve the sound quality using existing machine learning methods. The conclusions obtained during the development process can be useful in various areas of our lives [9].

Objectives of the research:

- To study and summarize the theoretical foundations of audio de-noising: using related scientific works and research, to study the methods of audio de-noising.
- Analyze existing de-noising methods: investigate different algorithms for de-noising audio recordings.
- Collect data and prepare it for analysis.
- Using the analyzed and studied machine learning methods, develop various models for de-noising audio recordings.
- Evaluate and compare the effectiveness of different algorithms: using standard evaluation metrics, determine the effectiveness of the developed models, compare and determine the best model.

The resulting trained models and their comparative analysis can be used by a wide range of people: the final results can be useful [10]:

- In the media and entertainment industry, as sound engineers and producers will be able to use the ready-made models to improve sound quality.
- In telecommunications, as the results can be used to improve the quality of voice messages, calls, or video conferences.
- In the military sphere: de-noising audio recordings can help in the investigation of criminal cases. The models can also

be used in the military, as during war, radio signals are intercepted, which can be noisy or distorted due to obstacles, poor conditions, or enemy defenses. These methods could help reduce noise and recover important audio signals to facilitate analysis.

Relevance: In today's world, audio recordings are used in many fields, such as music, speech, sound recording, cinematography, and others. However, the quality of audio recordings can often be degraded by noise, which can be caused by various factors, such as background sounds, electronics noise, wind noise, etc. De-noising audio recordings is an important task that allows to improve the sound quality and make it clearer and more pleasant to listen to.

When discussing the relevance of the chosen topic, the following points can be highlighted:

Growing use of audio: Audio content is becoming more and more popular as people listen to it at home, at work, on the road, and in the gym.

Noisy environment: Life takes place in a noisy world filled with various sources of noise, such as traffic, construction, and human activity.

Sound quality requirements: People expect clear, noise-free sound, making noise reduction an important consideration for many applications.

Accessibility: People with hearing loss may have difficulty understanding speech in noisy environments. Noise reduction can improve the accessibility of audio content for the hearing impaired.

2. RELATED WORKS

In the process of researching the topic of audio de noising, many publications were found that offer innovative or existing methods to solve this problem. Here are some of them with a brief analysis and additions:

Navneet Upadhyaya, and Abhijit Karmakarb in their paper “Speech Enhancement using Spectral Subtraction-type Algorithms: A Comparison and Simulation Study” [1] (2015) write that spectral subtraction is historically one of the first algorithms proposed to improve single-channel speech. In this method, the noise spectrum is estimated during speech pauses and subtracted from the spectrum of the noisy speech to estimate the clean speech. This is also achieved by multiplying the spectrum of the noisy speech by a gain function and then combining it with the phase of the noisy speech. The disadvantage of this method is the presence of processing distortions called residual noise. In recent years, a number of variations of the method have been developed to eliminate this disadvantage. These variations form a family of spectral subtraction algorithms. Their results have shown that modified forms of the spectral subtraction method significantly reduce residual noise, and the improved speech contains minimal speech distortion.

Researchers Jonathan Le Roux and Emmanuel Vincent in their paper «Consistent Wiener Filtering for Audio Source Separation» [2] (2013) write that the Wiener filter is one of the most common tools in signal processing, in particular for signal de-noising and audio source separation. In the context of audio, it is usually applied in the frequency-time domain using the Short-Time Fourier Transform (STFT). Such processing usually does not take into account the relationship between STFT coefficients in different time-frequency bands due to STFT redundancy, which they call coherence. The authors propose to take this relationship into account when designing a Wiener filter, either as a hard constraint or as a soft penalty. They derive two conjugate gradient algorithms to compute the filter coefficients and show improved separation performance of sound sources compared to the classical Wiener filter.

Thanh Tran Sebastian Bader, Jan Lundgren in their paper “De-noising Induction Motor Sounds Using an Autoencoder” [3] (2022) describe the problems of conventional de-noising methods and point out that deep learning can solve such problems. The authors write that although traditional de-noising methods have achieved good results in reducing noise in images, sound, and speech, they still have some drawbacks. These traditional methods are only effective at low noise levels. In addition, noise estimation and assumptions about aggregate statistical models are fundamental to these traditional methods. Consequently, these algorithms often underestimate or overestimate the noise, resulting in insufficient noise removal (noise is audible in the filtered result) or in sound distortion caused by excessive noise removal. In addition, determining the gain of a Wiener filter requires knowledge of the power spectral densities (PSDs) of the noise and the desired signals at a particular frequency. The calculation of SVD is slow and has a high computational complexity. Choosing the right wavelet of the right wavelets when an audio signal is noisy, for example, using a wavelet transform, can be time-consuming. By using deep neural networks (DNNs) for de-noising, these problems can be solved. DNN-based methods use paired data of noisy sounds and their corresponding clean sounds to train a noise reduction model. They outperform conventional filters for noisy images and signals [12].

J.S. Ashwin, N. Manoharan in «Audio De-noising Based on Short Time Fourier Transform» [4] (2017) propose a solution to the problem of sound de-noising using STFT. The proposed system uses the block thresholding method of STFT. It is used to effectively de-noise the audio signal for efficient noise removal. The proposed architecture uses a novel approach to estimate the adaptive estimation of ambient noise from speech. In this architecture, the original speech signals are given as an input signal. Using AWGN, noise is added to the signal. Then, the noisy signals are cleaned of noise using STFT techniques. Finally, the signal-to-noise ratio (SNR), peak-to-peak signal-to-noise ratio (PSNR) for the noisy and clear signals [10].

The Table 1 below provides an analysis of research on the topic.

Table 1. Review of related papers

Title of the work (author)	Method	Pros of methodology	Cons of methodology
Navneet Upadhyaya, Abhijit Karmakarb	Spectral subtraction	One of the first and simplest methods	The effectiveness of the method is highly dependent on accurate noise estimation, which is quite a challenge; In case of insufficiently good noise estimation, residual noise and sound distortion occur

Title of the work (author)	Method	Pros of methodology	Cons of methodology
Jonathan Le Roux, Emmanuel Vincent	Wiener Filter	Reliably removes noise due to its statistical optimality and adaptability to different conditions, making it effective in a wide range of applications	Does not take into account the relationship between STFT coefficients, which can lead to inaccuracy
Thanh Tran Sebastian Bader, Jan Lundgren	Deep learning (DNN)	Outperform traditional de-noising methods by effectively removing noise and minimizing distortion	More difficult to develop, requires large amounts of data, can be difficult to transfer to new data
J.S. Ashwin, N. Manoharan	STFT with block threshold	Includes a new approach to estimating adaptive ambient noise that improves signal-to-noise ratio (SNR) and peak-to-peak signal-to-noise ratio (PSNR)	STFT can have limited time and frequency resolution, especially when using short analysis windows. This can lead to a loss of information about the exact timing of events in the signal or the accuracy of determining the frequencies of signal components.

The above methods can be briefly described:

Spectral subtraction: This method, one of the first, estimates the noise spectrum during speech pauses and subtracts it from the spectrum of the noisy speech. The disadvantage is the presence of residual noise. Modified forms of the method significantly reduce it, but can lead to speech distortion.

Wiener filter: This common method uses the Short-Term Fourier Transform (STFT) to de-noise and separate sound sources. The classic Wiener filter does not take into account the relationship between STFT coefficients, which can lead to inaccuracy. Advanced algorithms take this consistency into account, improving results [14].

Deep learning: Methods based on deep neural networks (DNNs) are trained on pairs of noisy and clean sounds. They outperform traditional de-noising methods by effectively removing noise and minimizing distortion.

STFT with block threshold: This method uses STFT to effectively de-noise an audio signal. It incorporates a new approach to estimating adaptive ambient noise, which improves the signal-to-noise ratio (SNR) and peak-to-peak signal-to-noise ratio (PSNR) [15].

To summarize, there are quite a few machine learning methods for audio de-noising, which can be divided into two categories:

Traditional methods: Filtering, equalization, compression, spectral subtraction, Wiener filter.

Machine learning methods: Convolutional neural networks (CNN), recurrent neural networks (RNN), spectral analysis methods, hybrid methods, deep learning.

Advantages of machine learning methods: Efficiency; Flexibility; Continuous improvement [9].

It is important to note that there is no universal de-noising method that is suitable for all cases. The choice of method depends on the type of noise, the complexity of the audio recording, and the desired sound quality. Throughout the study, there will be comparison of the existing methods and determination under what circumstances which method is more effective.

3. RESEARCH METHODOLOGY

In order to train a machine learning model well, a good dataset is required. In this work, there were used two different datasets.

LibriSpeech [5] is a large English speech dataset consisting of thousands of hours of spoken audio. It is divided into training, validation, and test sets, which provide a wide range of speech data for model training. The dataset includes a variety of speakers, genders, and accents, which can help the model generalize well across different types of audios. In addition, the dataset contains different types of noise added to the audio signals to create a de-noising task, making it an ideal dataset for audio de-noising.

ESC-50 [6] is a dataset of environmental sound samples. The dataset contains a total of 2000 sound samples grouped into 5 classes: animal, nature, human, interior/household, and outdoor. The dataset is labeled with 50 classes and is also available with noise added to the audio signals, making it ideal for de-noising tasks.

Subsequently, these two datasets will be randomly combined to create two datasets: a dataset with clean sound and a noisy dataset, which will allow for better model training. This will happen as follows: random sounds from the ESC-50 dataset will be superimposed on the LibreSpeech dataset. Thus, the result is a dataset with clean audio recordings and a noisy dataset. By using these two datasets, the model will have a diverse range of audio data to train on, including both speech and environmental sounds. This will help the model to generalize well to different types of audios and noise, which will lead to better performance in the de-noising task [11].

In addition, using these datasets will allow us to evaluate the model's performance on a diverse set of audio signals and noise, which will help us to assess the model's performance under different conditions.

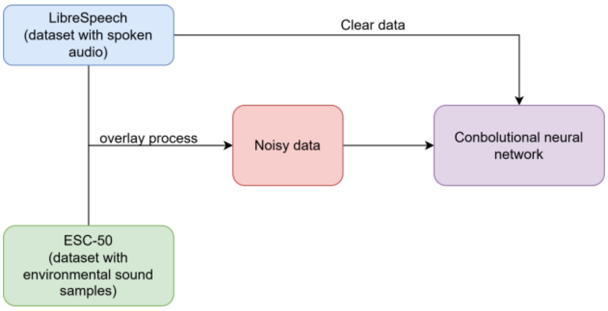

LibreSpeech
(dataset with spoken audio)
Clear data
overlay process
Noisy data
Conbolutional neural network
ESC-50
(dataset with environmental sound samples)

Fig 1: Process of datasets overlay and usage

Now, let's consider the methods of audio data de-noising shown in Table 1 in more detail. Start with the spectral subtraction algorithm.

First of all, it is worth understanding the algorithm of the spectral subtraction method after which the mathematical basis of this algorithm will be discussed.

The following 6 stages can be distinguished in the spectral subtraction algorithm:

Frame splitting: the noisy audio signal is divided into short time intervals (frames), during which the noise characteristics are considered stationary [14].

Calculating the spectrum: For each frame, the amplitude-frequency response (spectrum) is calculated using the Fourier transform. This enables the representation of the signal in the frequency domain, where the energy distribution between the audio signal and noise becomes clearly visible.

1. Noise estimation: The noise spectrum is estimated using techniques such as low-resolution frame analysis or the use of an a priori noise model.
2. Noise subtraction: The noise spectrum is subtracted from the spectrum of the noisy signal. This step leads to the suppression of noise components and the enhancement of the audio signal spectrum.
3. Signal recovery: Based on the modified spectrum obtained in the previous step, the signal is reconstructed using the inverse Fourier transform. This process transforms the signal from the frequency domain back to the time domain.
4. Overlay and addition: The cleaned signal fragments obtained from the previous steps are overlapped and added to obtain the resulting filtered audio signal.

Spectral subtraction method has a number of advantages that make it a popular choice for audio signal noise reduction [13].

The advantages of this method are as follows:

- Simplicity of implementation: The algorithm is based on simple mathematical operations and does not require complex calculations;
- High efficiency in removing stationary noise, such as background noise or engine noise;
- Relatively low computing resource requirements, which makes the algorithm suitable for implementation in real-time systems.

The disadvantages of this method are as follows:

- With non-stationary noise, such as impulse noise, Spectral subtraction can lead to distortion of the audio signal; also, in complex acoustic environments, SV may not always clearly separate speech from noise, which can lead to residual noise artifacts.

After a superficial look at the spectral subtraction algorithm, it is now time to examine the algorithm from a mathematical and theoretical perspective [15].

In their paper “Speech Enhancement using Spectral Subtraction-type Algorithms: A Comparison and Simulation Study” [1], Navneet Upadhyaya and Abhijit Karmakarb describe the spectral subtraction algorithm as one of the first and easiest audio de-noising algorithms, but one that has disadvantages in the form of residual noise and sound distortion due to insufficiently good noise estimation.

From a mathematical point of view, the spectral subtraction algorithm works as follows:

First of all, a noisy audio signal is received as input using the following Equation (1):

$$y[n] = s[n] + d[n], \quad (1)$$

where: $y[n]$ is a noisy audio signal;

$s[n]$ is a clear sound signal;

$d[n]$ - additional noise.

The article states that the additional noise has a zero mean value and does not correlate with the pure sound signal. It is important to understand that the sound signal is non-stationary and changes in time, and therefore it is worth considering not the entire input sound signal but individual frames. Frame by frame - just as the text is broken into words, the algorithm breaks the noisy audio signal into short fragments. This will be the first stage of the algorithm's work: splitting into frames. The representation in the short-term Fourier transform looks like this (Equation (2)) [10]:

$$y(w,k) = s(w,k) + d(w,k), \quad (2)$$

where: k is a number of frames.

Next step – specter calculation.

The sound signal is uncorrelated with the background noise, and therefore the short-term power spectrum $y[n]$ has no cross terms. Hence (Equation (3)) [10, 12]:

$$|y(w,k)|^2 = |s(w,k)|^2 + |d(w,k)|^2. \quad (3)$$

The audio signal can be determined by subtracting the estimated noise from the received signal (Equation (4)):

$$|\hat{s}(w,k)|^2 = |y(w,k)|^2 + \left|\hat{d}(w,k)\right|^2. \quad (4)$$

The next step is step 3: noise estimation. To determine the estimated noise $\left|\hat{d}(w,k)\right|^2$, the last frames of the sound pauses need to be averaged (Equation (5)):

$$\left|\hat{d}(w,k)\right|^2 = \frac{1}{M}\sum_{j=0}^{M-1} y_{sp_j}|(w,k)|^2, \quad (5)$$

where M is the number of consecutive frames of sound pauses (SP).

If the background noise is stationary, it converges to the optimal estimate of the noise power spectrum, since it takes the longer average value.

Spectral subtraction can also be thought of as a filter. This requires that it can be expressed as the product of the noisy sound spectrum and the spectral subtraction filter (SSF) as follows (Equation (6)):

$$|\hat{s}(w)|^2 = \left(1 - \frac{|d(w,k)|^2}{|y(w,k)|^2}\right)|y(w)|^2 = H^2(w)|y(w)|^2, \quad (6)$$

where $H(w)$ is a gain function and a well-known spectral subtraction filter (SSF).

$H(w)$ is a zero-phase filter with a value in the range $0 < H(w) \leq 1$.

To restore the received audio signal, it is also necessary to estimate the phases of the sound. To achieve this, the phase of the noisy audio signal can be taken as the phase of the estimated clean speech signal, assuming that the short-term phase is relatively unimportant. Therefore, the audio signal in the frame will be calculated as (Equation (7)):

$$\hat{s}(w) = |\hat{s}(w)|e^{j<y(w)} = H(w)y(w). \quad (7)$$

The estimated speech signal shape is recovered in the time domain using the inverse Fourier transform $S(\omega)$ using overlay and addition approaches.

An example of the algorithm's performance on the input noisy audio recording:

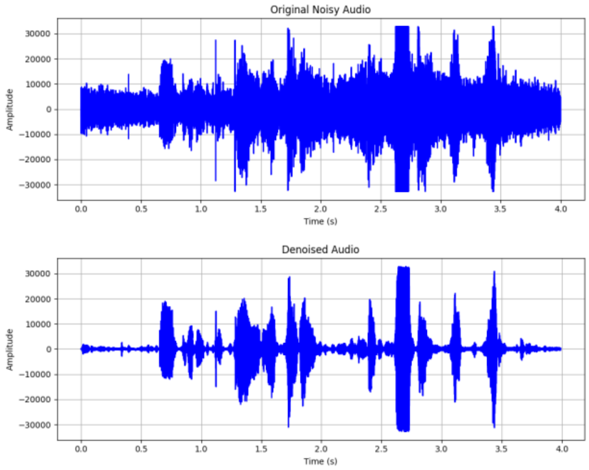


Fig 2: Results of audio de-noising, using example of people's speech with stationary background noise

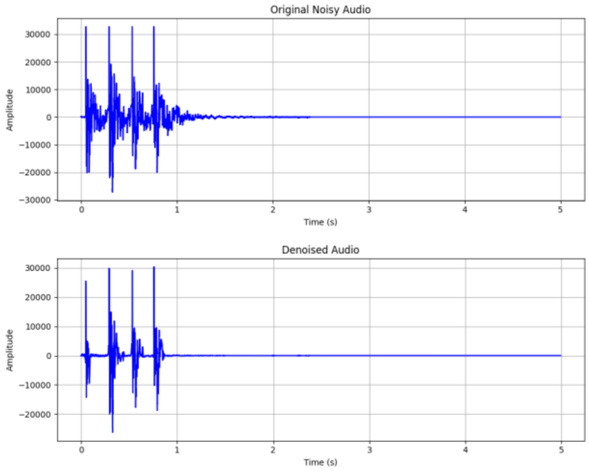


Fig 3: Results of audio de-noising, using example of people's speech with stationary background noise

The images illustrate how the algorithm removes noise by calculating its spectrum and subtracting it from the noisy audio recording, resulting in a de-noised signal. In particular, Figure 2 shows that the frequency range around zero is much larger compared to the figures of the de-noised audio recordings.

3.1. Wiener Filter and its modifications

The classic Wiener algorithm, developed by Norbert Wiener in the 1940s, is one of the most powerful traditional methods for separating audio sources. It is based on the spectral characteristics of both the audio signal and noise, allowing for the estimation of individual sound sources from a mixture [9, 7]. The Wiener filter is an optimal linear filter used to remove noise or distortion from a signal caused by a known function. It finds wide application in various fields including image processing, signal processing, and telecommunications.

The Wiener filter method is based on minimizing the root mean square error between the estimated and original signals. It uses the spectral information of both the useful signal and the noise to obtain the optimal filter for noise removal.

It is important to note that the Wiener filter gives optimal results only when the distortion function is known exactly, i.e., the filter is dependent on a known distortion function.

The distortion function in the Wiener filter can be specified in several ways:

1. Measurement of noise characteristics:

- Acoustic measurements: Measure the acoustic characteristics of a room, such as reverberation time and noise spectral distribution. This data can be used to create a noise model that will serve as a distortion function.
- Analyze recording data: Analyze the recording made in the room to extract the characteristics of the noise. This analysis can include spectral analysis, statistical analysis, and machine learning techniques.

2. Using off-the-shelf models:

- Existing room noise models: There are a number of off-the-shelf room noise models that can be used. These models are usually based on statistical data or physical models of room acoustics.
- Audio processing software: Some audio processing software has built-in room noise models that can be used in the Wiener filter.

3. Approximation:

- Simple approximation: A simple approximation can be used, such as adding white noise to the recording. This may not be accurate, but it can be useful for some applications.
- Frequency-dependent approximation: frequency-dependent approximation can be used, where the noise level depends on the frequency. This is more accurate than the simple approximation, but it may require more information about the noise characteristics.

The choice of method depends on the specific application and available data.

It is important to note that the accuracy of the room noise distortion function will affect the results of the Wiener filter. A more accurate distortion function will result in better noise removal and improved audio quality [13].

The Wiener filter can work even when the distortion function is not known exactly, however, the accuracy and efficiency of the filter will be significantly lower compared to the case when the distortion function is known, for the following reasons:

- If the distortion function is unknown, it will need to be estimated. This estimate is likely to be inaccurate, leading to filtering errors.
- The Wiener filter can amplify noise if it does not know how to remove it properly. This can lead to a deterioration in signal quality.
- The Wiener filter will not be able to give optimal results if it does not know which distortion function is affecting the signal.

Now let's move on to the wiener filter algorithm description [11].

Description of the Wiener filter algorithm:

1. Recording an audio signal: The first step involves recording an audio signal that contains a mixture of sound sources that need to be separated. It can be a recording of a conversation in a noisy environment, a recording of a

music concert, or any other audio signal containing several sources.

2. Conversion to the frequency domain: Next, the audio signal is transformed from the time domain to the frequency domain using the Discrete Fourier Transform (DFT) (Equation (8)). This gives us a spectral representation of the signal that shows the amplitude and phase of the signal at each frequency.

$$X[k] = \sum_{n=0}^{N-1} x[n] * \mathrm{e}^{-\mathrm{j}2\pi\mathrm{nm/N}}, \qquad (8)$$

where:

$X[k]$ – spectral coefficients of the signal;

$x[n]$ – discrete values of the signal;

N – signal length;

j – imaginary unit.

3. Assessment of the noise spectrum: In order to accurately separate sound sources, it is important to correctly estimate the spectral characteristics of the noise present in the recording. This can be done using various methods, such as analyzing the signal during moments of silence or using ready-made room noise models.
4. Calculating the spectrum of the estimated signal: The heart of the Wiener algorithm is a formula that allows to calculate the spectrum of the estimated signal for each frequency. This formula is based on the audio signal, noise, and signal-to-noise ratio (SNR) spectra.

The Equation (9) of the Wiener algorithm:

$$\hat{\mathrm{X}}[\mathrm{k}] \ = \ \mathrm{X}[\mathrm{k}] \ / \ (|\mathrm{X}[\mathrm{k}]|^2 / \ \sigma_\mathrm{n}^2 \ + \ 1), \qquad (9)$$

where:

$\hat{X}[k]$ is the spectrum of the estimated signal;

σ_n^2 - noise power spectral density.

5. Conversion back to the time domain: After calculating the spectrum of the estimated signal, it must be transformed back to the time domain using the inverse DFT (Equation (10)). This gives us an estimated audio signal that contains only one of the sound sources that were present in the original recording.

$$\mathrm{x}[\mathrm{n}] = \frac{1}{\mathrm{N}} * \sum_{n=0}^{N-1} \mathrm{X}[\mathrm{k}] * \exp\left(\frac{2\pi\mathrm{jkn}}{\mathrm{N}}\right). \qquad (10)$$

Wiener's algorithm is based on simple mathematical formulas, which makes it easy to understand and implement. The algorithm can be implemented using efficient algorithms such as the Fast Fourier Transform (FFT), making it practical for real-time audio signal processing. The Wiener algorithm gives optimal results for separating audio sources if the SNR is known accurately.

The classical Wiener filter, although a powerful tool for separating audio sources, has certain limitations. It does not take into account the relationship between the coefficients of the spectral representation of the signal at different frequencies and in different time intervals. This can lead to inaccuracies when the filtered sound sources are converted back to the time domain.

This is exactly what Jonathan Le Roux and Emmanuel Vincent describe in their paper “Consistent Wiener Filtering for Audio Source Separation”. They propose a Consistent Wiener Filter. The Consistent Wiener Filter is an improved version of the classic filter that takes this consistency into account. It ensures that separated audio sources are free of artifacts and consistent with each other after being converted back to the time domain.

This filter works by minimizing a certain cost function, provided that the estimated sound sources in the frequency domain must match the Short-Time Fourier Transform (STFT) of their reconstructed versions in the time domain. In other words, the filter ensures that the information in the frequency domain corresponds to a signal that can be reconstructed in the time domain.

The paper proposes an iterative algorithm based on the conjugate gradient method (Fig. 4). This algorithm seeks the minimum of the cost function subject to a consistency constraint. It uses a special preconditioner method to speed up convergence and exploits the properties of the forward and inverse short-time Fourier transform (STFT and iSTFT).

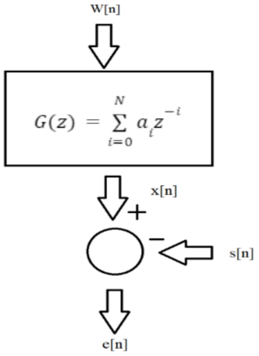


Fig 4: Schematic representation of the Wiener filter

Advantages of the coherence-constrained Wiener filter:

- By taking coherence into account, this filter provides a more accurate separation of audio sources than the classical Wiener filter.
- The resulting audio sources have fewer artifacts and distortions because their frequency domain information matches their time domain information.

The coherence constraint makes the algorithm a more powerful tool for separating sound sources than the classical Wiener filter. It provides better quality and consistency of the separated signals.

So, summing up, the advantages and disadvantages of the Wiener Filter:

Advantages:

- Simplicity and accessibility;
- Efficiency;
- Optimality (with accurate SNR);
- Wide application.

Disadvantages:

- Dependence on SNR (accuracy);
- Does not take into account nonlinear noise;
- Does not take into account the relationship between spectral components;
- Possible artifacts.

The Wiener filter is a powerful tool for separating audio sources that has a number of advantages, such as simplicity, efficiency, and optimality. However, it is important to take into account its limitations, such as its dependence on SNR, inability to account for nonlinear noise, and the relationship between spectral components. In some cases, these limitations can lead to inaccuracies or artifacts in the obtained sound sources.

3.2. Using neural networks to de-noise audio

One of the most powerful methods of audio de-noising nowadays is the use of neural networks. In particular, Convolutional Neural Networks provide an opportunity to quite accurately identify and eliminate noise in audio recordings.

Let's consider the advantages of using neural networks to solve the problem [12].

Neural networks, in particular CNNs, offer a number of significant advantages for audio de-noising compared to various traditional methods:

- CNNs are capable of recognizing and removing various types of noise with high accuracy, including background noise, wind noise, machine noise, hum, and other acoustic artifacts. Their ability to learn from large amounts of data allows them to analyze audio signals in depth and clearly separate speech or other important sounds from noise components.
- CNNs can be customized to de-noise audio recordings with different noise characteristics and acoustic environments. This makes them a versatile tool that can be used to process audio from a variety of sources, such as telephone conversations, microphone recordings in noisy environments, music recordings, etc.
- CNNs can be continuously improved and trained on new data, allowing them to adapt to new types of noise and acoustic conditions. This ensures that their performance does not deteriorate over time, but rather improves continuously.
- CNNs can be trained to recognize and remove certain specific types of noise, such as wind noise, machine noise, or engine hum. This makes them a valuable tool for applications where it is important to detect certain non-stationary noises.

However, despite their advantages, neural networks have their drawbacks. These include the following points:

- Writing neural networks is a more difficult task than using traditional audio de-noising methods.
- Training and using CNNs can be computationally intensive, especially for processing large amounts of audio data.
- In some cases, CNNs can remove not only noise, but also some important sounds from an audio signal, especially if these sounds have similar acoustic characteristics to the noise.
- CNNs require large amounts of data for training to achieve high accuracy and efficiency.
- CNNs are highly dependent on the quality of the data they are trained on. Noise or other artifacts in the training data can negatively affect their ability to remove noise from new audio recordings.

The use of CNNs has its own advantages and disadvantages.

Let's take a deeper look at the method of using neural networks.

In the experimental part will be used model of encoder-decoder type. The encoder-decoder model in CNNs (convolutional neural networks) is a special architecture consisting of two interconnected subnets: encoders and decoders (Fig. 5).

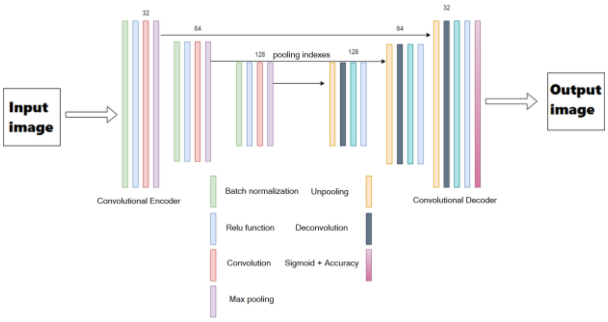


Fig 5: Encoder-Decoder model architecture

The encoder is the first part that takes the audio signal as input and compresses it into a lower dimensional hidden space. The hidden space contains the main functions and characteristics of the input audio signal.

The decoder is the second part that takes this hidden space as input and tries to restore or generate the original audio signal based on the information received. In other words, the decoder decodes the hidden space, returning to the original size or even generating a new audio signal based on these encoded features.

2 architectures of the coder-encoder model can be distinguished (Fig. 6):

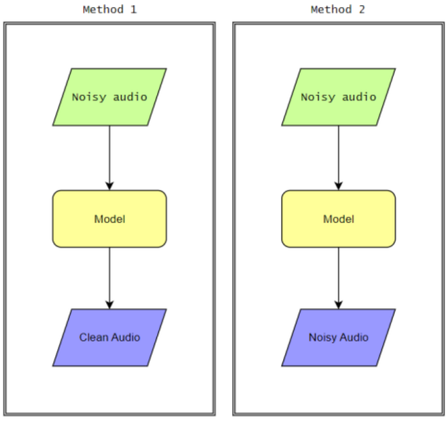


Fig 6: CNN model training approaches

Both approaches are worth analyzing.

The first approach is that the model gives us a noise-free audio signal as an output. This approach is simpler because the model only needs to clean the audio signal. The model will learn to directly output the clean signal, which can be more efficient and give better results than trying to estimate the noise and subtract it. Disadvantages - The model will not be able to learn about the characteristics of the noise, which can be useful for other tasks (for example, to identify and remove certain types of noise). Also, if the noise is very different from one training example to another, it may be difficult for the model to learn a universal noise removal function.

In the case of the second approach, the model outputs noise and a clean signal. This approach is more complex because the model needs to output both noise and clean signal, but it allows the model to learn about the characteristics of the noise, which can be useful for identifying and removing certain types of noise. This model is more versatile by estimating noise separately from the clean signal (Fig. 7).

An example of using convolutional neural networks for de-noising audio data:

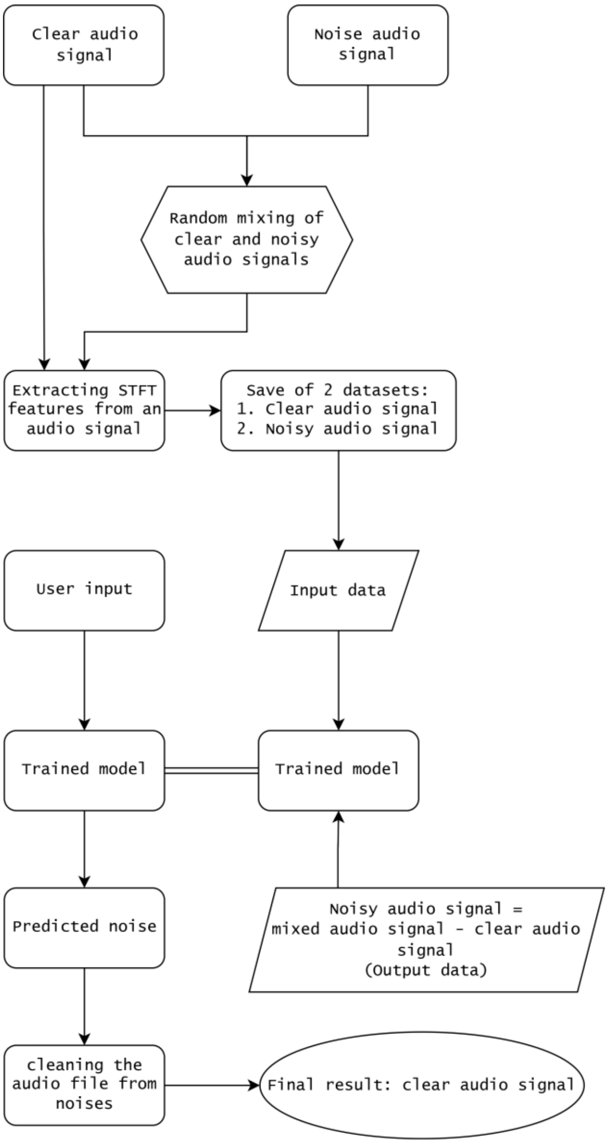

Clear audio signal
Noise audio signal
Random mixing of clear and noisy audio signals
Extracting STFT features from an audio signal
Save of 2 datasets:
1. Clear audio signal
2. Noisy audio signal
User input
Input data
Trained model
Trained model
Predicted noise
Noisy audio signal = mixed audio signal - clear audio signal
(Output data)
cleaning the audio file from noises
Final result: clear audio signal

Fig 7: Block diagram of the audio de-noising algorithm using CNN

To summarize, neural networks are one of the most popular data de-noising methods because, despite all its drawbacks, it is the most versatile and allows processing audio recordings with both stationary and non-stationary noise. This is the method, that will be used in the experiments described below.

4. EXPERIMENTAL DATA SETUP

This section will explore various data de-noising methods, such as spectral subtraction, convolutional neural networks (CNNs), and the Wiener filter.

4.1. Experiments using the spectral subtraction algorithm

Initially, the spectral subtraction algorithm was investigated. As mentioned in Section 2, this algorithm has the following advantages and disadvantages:

The advantages of this method are as follows:

- Relatively low computing resource requirements, which makes the algorithm suitable for implementation in real-time systems.
- Simplicity of implementation: The algorithm is based on simple mathematical operations and does not require complex computations; High efficiency in removing stationary noise such as background noise or engine noise.

The disadvantages of this method are as follows:

- With non-stationary noise, such as impulse noise, SV can lead to distortion of the audio signal; also, in complex acoustic environments, SV may not always clearly separate speech from noise, which can lead to residual noise artifacts.

The audio recordings were taken from LibriSpeech [5]. These are random audios with different types of noise.

To evaluate the results, first of all, a visual and comparative evaluation of the sound was performed.

Visual evaluation was performed by displaying two images of the signal.

Example 1:

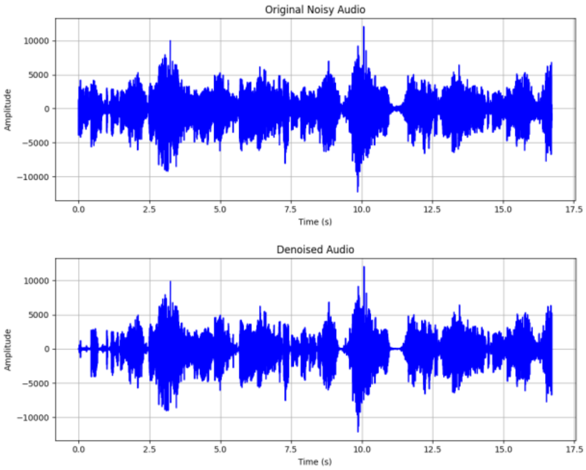


Fig 8: Visualization of the signal of noisy (a) and unnoisy (b) audio recordings using spectral subtraction (example 1)

Figure 8 (a) shows the original audio signal containing a significant amount of noise, as the signal amplitude fluctuates over a fairly wide range of values. Figure 8 (b) shows that the amplitude has become smoother and more stable compared to the original noisy audio. Listening to the 2 audio recordings, one could feel that there was less noise after the algorithm was applied, but the sound quality was also partially degraded.

Perceptual Evaluation of Speech Quality (PESQ) is a family of standards that provides a test methodology for automated evaluation of speech quality as experienced by a telephony system user. It was standardized as ITU-T Recommendation P.862[1] in 2001. PESQ is used for objective voice quality testing by handset manufacturers, network equipment vendors, and telecommunications operators. This industry standard for voice quality is objective and internationally recognized. It takes into account various critical characteristics, including: sound clarity; call volume; background noise; variable delay; sound delay; clipping; interference.

This test compares two audio recordings. This comparison provides a completely unbiased and objective measure of the actual sound that people hear. This is much more accurate than other methods of measuring sound quality, which often rely on predictions or even subjective assessments. The PESQ score ranges from 1 to 4.5. The higher the score, the better the sound quality level.

Experiments were conducted on different audio recordings with different types of noise in the background. The results are as follows Table 2.

Table 2. Spectral subtraction results

№	№ of audio in dataset:	Type and description of noise	PESQ:
1	730-358-0061	Non-stationary (inhalation /exhalation noise)	2.06
2	2182-181173-0033	Stationary (regularly repeated barking of a dog)	2.4
3	4406-16883-0021	Stationary (keyboard sound)	2.3
4	5808-54425-0016	Non-stationary (engine sound)	1.1
5	7059-88364-0015	Stationary (loud noise)	1.2
6	8063-274115-0002	Stationary (engine sound)	1.5
7	8747-293952-0005	Stationary (knocking sound)	2.2

Signal graphs for the experiments:

Experiment 1: Non-stationary noise (inhalation /exhalation):

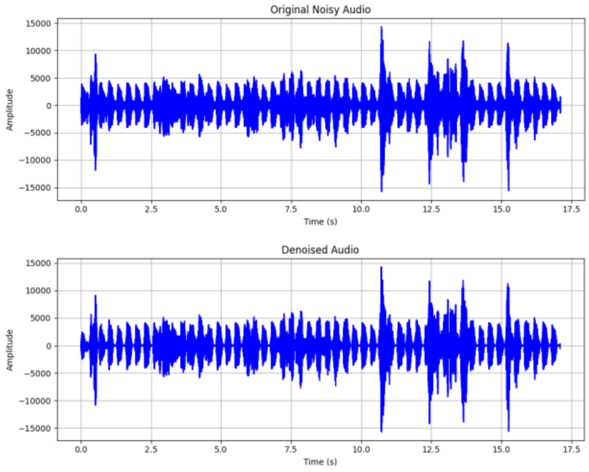
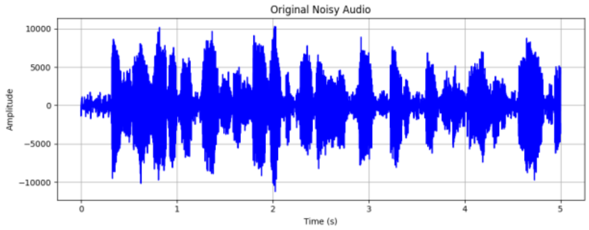


Fig 9: Signal visualization of noisy (a) and de-noised (b) audio recordings using spectral subtraction for experiment 1

Experiment 2: Stationary noise (regularly repeated dog barking)



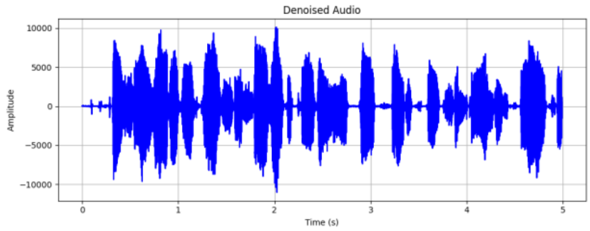


Fig 10: Signal visualization of noisy (a) and unnoisy (b) audio recordings using spectral subtraction for experiment 2

Experiment 3: Stationary noise (keyboard sound)

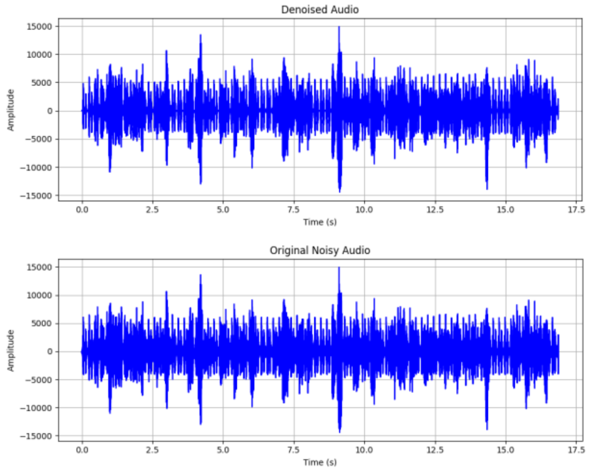


Fig 11: Signal visualization of noisy (a) and unnoisy (b) audio recordings using spectral subtraction for experiment 3

Experiment 4: Non-stationary noise (engine sound)

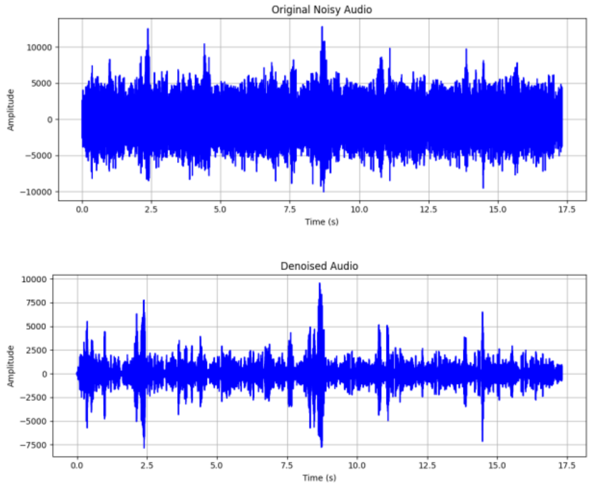


Fig 12: Signal visualization of the noisy and de-noised audio recordings using spectral subtraction for experiment 4

Experiment 5: Stationary noise (loud noise)

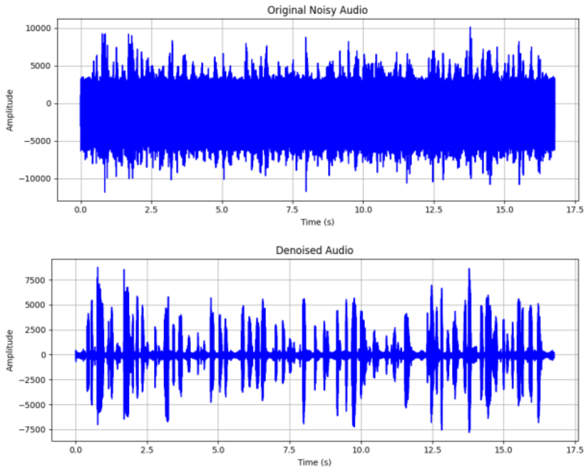


Fig 13: Signal visualization of the noisy and de-noised audio recordings using spectral subtraction for experiment 5

Experiment 6: Stationary noise (engine sound)

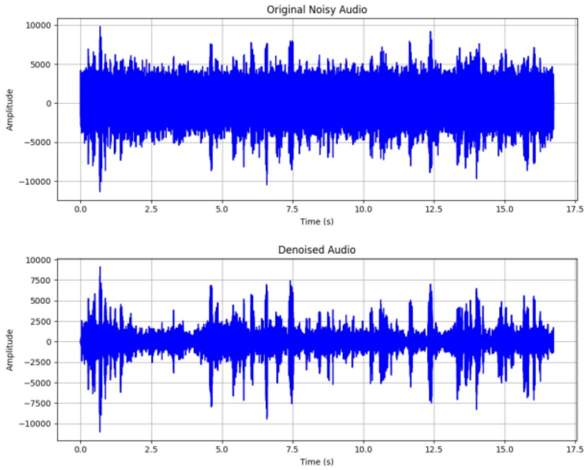


Fig 14: Signal visualization of the noisy and de-noised audio recordings using spectral subtraction for experiment 6

Experiment 7: Stationary noise (knocking sound)

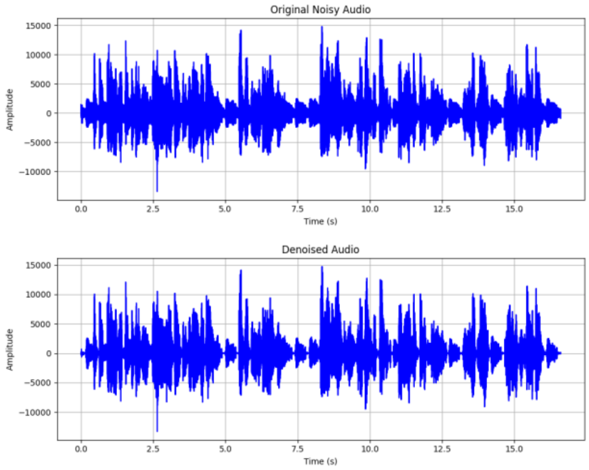


Fig 15: Signal visualization of noisy and unnoisy audio recordings using spectral subtraction for experiment 6

Analyzing the results, it can be distinguished, that the algorithm copes best with stationary noise, and it is important to note that if the noise is loud, the PESQ score is lower, and therefore the result can be considered worse. In general, the advantages of using the spectral subtraction method include the following points:

1) The method is very fast: for example, it takes 1.5373 seconds to de-noise a 20-second audio recording;
2) The method copes well with stationary noise, but the result is worse when the noise volume is high.
3) The algorithm does not require a large amount of data: only one audio recording is enough.

Disadvantage of the method: The method performs rather mediocrely in the presence of non-stationary noise: many artifacts remain in the audio recording. In general, the method does not give the best results in de-noising.

4.2. Experiments using the Wiener filter algorithm

The Wiener filter is a statistical signal de-noising method widely used in audio signal processing. It belongs to the group of linear filters based on minimizing the mean square error (MSE) between the original and filtered signals.

The Wiener filter is based on the assumption that the useful signal and noise are statistically independent. This means that their spectral characteristics do not overlap.

The filter uses the spectral information about the signal and noise to separate them and remove the noise.

Table 3. Wiener Filter results

№	Type and description of noise	PESQ:	Impressions from listening
1	Non-stationary (engine sound)	1.4	It did a good job, the noise has become quieter, but not gone
2	Stationary (wind noise)	1.2	The noise is quite loud
3	Stationary (environment sound)	1.5	Partially muffled, but not completely eliminated
4	Non-stationary (engine sound)	1.2	Among all the examples, it did the worst job, only increased the noise
5	Stationary (loud noise)	1.4	The noise is quite loud in the background

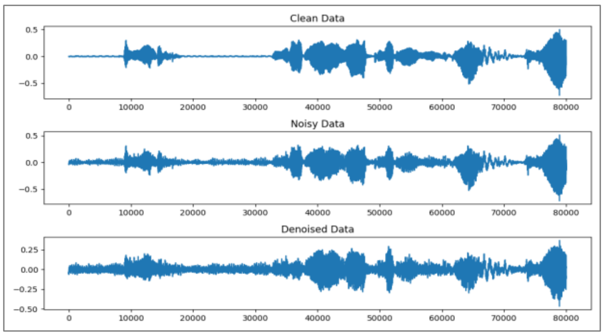


Fig 16: Signal visualization of the clean, noisy and de-noised audio recordings using Wiener filter for experiment 1

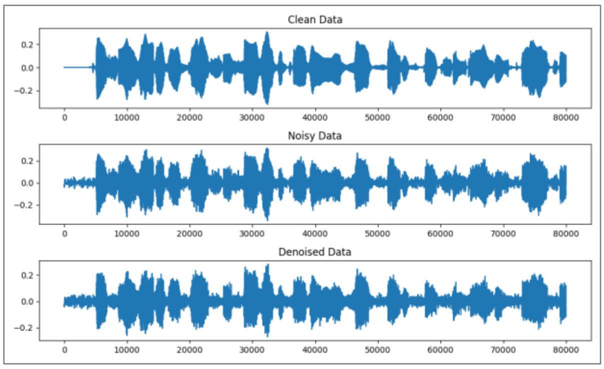


Fig 17: Signal visualization of the clean, noisy and de-noised audio recordings using Wiener filter for experiment 2

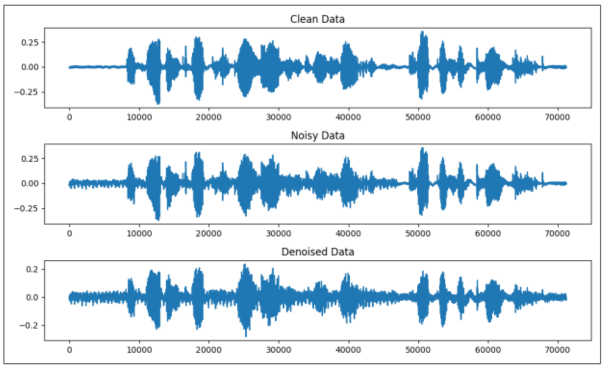


Fig 18: Signal visualization of the clean, noisy and de-noised audio recordings using Wiener filter for experiment 3

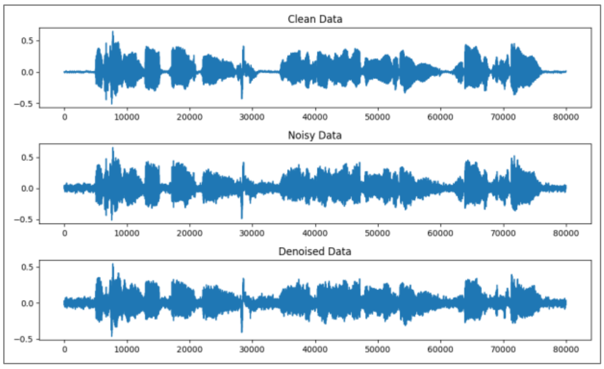


Fig 19: Signal visualization of the clean, noisy and de-noised audio recordings using Wiener filter for experiment 4

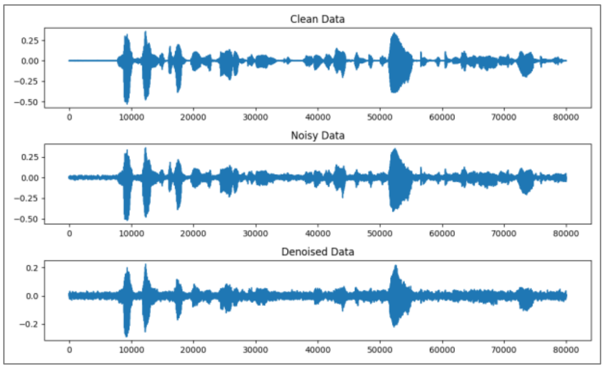


Fig 20: Signal visualization of the clean, noisy and de-noised audio recordings using Wiener filter for experiment 5

Discussing the results obtained after applying the Wiener filter, a conclusion can be made that the advantages are:

- Ease of implementation of the Wiener filter.

Disadvantages:

- The Wiener filter can lead to signal distortion if the assumptions about the statistical independence of the signal and noise are not met.
- It can be sensitive to inaccurate estimates of the spectral characteristics of the signal and noise.

4.3. Experiments using the Convolutional Neural Networks

Convolutional neural networks are now one of the most powerful tools for de-noising audio recordings.

In this study, there was used U-Net Network.

U-Net is based on the so-called “Fully Connected Networks” proposed by Long, Shelhamer, and Darrell in 2014. The main difference is that U-Net replaces pooling operations with upsampling operations in the output layers of the contracting path of the network. This allows increasing the resolution of the original image. The next convolutional layer can then be trained to collect the exact output based on this information.

An important modification in the U-Net is the large number of feature channels in the resolution part. This allows the network to transmit contextual information to layers with higher resolution. As a result, the expanding path is more or less symmetrical to the contracting path, which gives the network a U-shape. The network uses only the useful part of each convolutional layer without fully connected layers. To predict pixels in the border region of the image, the missing context is extrapolated by mirroring the input image. This fragmentation strategy is essential for applying the network to large images, as otherwise the resolution would be limited by GPU memory.

MAE (Mean Absolute Error) [8]: the average absolute error used to estimate the differences between the pixels of the original and restored images. It is calculated as the average absolute difference in pixel values between two images. The smaller the MAE value, the better the de-noising model reproduces the original image (Equation (11)).

$$MAE = \sum_{i=1}^{n} |y_i - y_i|. \qquad (11)$$

The ratio of peak signal to noise [9], which is used to assess the quality of images after compression or processing. PSNR is measured in decibels (dB) and characterizes the dynamic range of an image. The higher the PSNR value, the better the image quality (Equation (12)).

$$PSNR = 10 * log(MAX^2/MSE). \qquad (12)$$

A structural similarity index [10] used to evaluate the similarity of structural details between two images. SSIM takes into account the brightness, contrast, and structural details of the images. The higher the SSIM value, the higher the structural similarity between the images (Equation (13)).

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \qquad (13)$$

where:

- μ_x and μ_y are the mean values of images x and y;
- σ_x^2 and σ_y^2 are the variances of images x, y;
- σ_{xy} is the covariance between x and y;

- C_1 and C_2 are small constants to avoid division by zero.

MAE, PSNR, and SSIM are three important metrics used to evaluate the quality of images after processing. Each of these metrics has its own advantages and disadvantages, so it is important to use a combination of these metrics to obtain a comprehensive assessment of image quality.

The results of CNN's training see in the Table 4.

Table 4. Comparative table of Unet performance with different parameters (test data)

Parameters			Metrics and measurements				
N, data	Epochs number	Batch size	Loss	MAE	PSNR	SSIM	Runtime
500	25	32	0.0216	0.1069	18.2159	0.2263	105.88616
500	50	32	0.0209	0.0950	18.9880	0.3112	91.37419
500	100	32	0.0244	0.1067	17.7399	0.2833	167.82782
500	25	64	0.0197	0.1051	18.2654	0.1699	52.683226
500	50	64	0.0198	0.1093	17.9111	0.1087	92.025675
500	100	64	0.0321	0.1281	16.4224	0.2880	211.297902
1000	25	32	0.0335	0.1452	15.5555	0.1197	90.386527
1000	50	32	0.0232	0.1073	17.2842	0.3022	211.815254
1000	100	32	0.0164	0.0858	19.4960	0.3777	311.73863
1000	25	64	0.0205	0.1042	18.4482	0.1556	89.6746056
1000	50	64	0.0253	0.0885	17.6580	0.2627	213.68908429
1000	100	64	0.0496	0.1743	13.7136	0.1426	333.563862
5000	25	32	0.0258	0.1120	17.0949	0.2897	459.9007775
5000	50	32	0.0216	0.1060	17.6949	0.2840	817.169139
5000	25	64	0.0407	0.1586	14.6541	0.1255	455.5682692
5000	50	64	0.0251	0.1105	17.2120	0.3104	816.09427595

The main problem with the trained network was that it performed poorly on the test data. To fix this, L2 regularization was applied to the model and Dropout was added.

L1/L2 regularization: methods to prevent overfitting in machine learning.

- L1: adds a penalty to the loss proportional to the sum of the absolute values of the model's weights (can lead to zero weights).
- L2: adds a penalty to the loss proportional to the square of the norm of the model weights (does not lead to zero weights).

Dropout: a method to prevent overfitting in neural networks.

- Randomly removes some neurons from the network during training.
- Forces the model to learn representations independent of individual neurons.

The results were as follows in the Table 5.

Table 5. Comparative table of Unet performance results with different parameters (with L1&L2 regularization and Dropout)

Parameters			Metrics and measurements				
N, data	Epochs number	Batch size	Loss	MAE	PSNR	SSIM	Runtime
5000	25	32	0.0241	0.0241	0.0241	0.0241	0.0241
5000	50	32	0.0925	0.0925	0.0925	0.0925	0.0925

Let's look at the results in Table 5. It is worth considering them from different perspectives.

First of all, let's consider the effect of the number of epochs: Increasing the number of epochs from 25 to 50 results in improved MAE and PSNR for most dataset sizes and batch sizes, however, further increasing the number of epochs to 100 may result in worse PSNR and longer runtimes for some configurations.

The effect of the batch size: Using a larger batch size (64) may result in improved Loss and MAE for some configurations, but may negatively impact PSNR and runtime.

Effect of dataset size: Increasing the dataset size from 500 to 1000 results in improved PSNR and SSIM for most configurations, but increases the runtime.

The best performing model is the one with the parameters (1000 data, 100 epochs, 32 batch size) (Fig. 21):

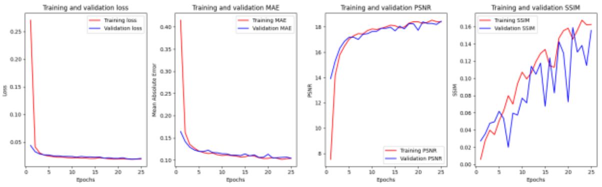


Fig 21: The best model obtained with parameters (1000 data, 100 epochs, 32 batch size)

The loss and MAE on both training and testing data gradually decrease. Similarly, the PSNR for the model grows quite smoothly, which is not the case with SSIM, but this value eventually reaches its maximum value compared to other models with other parameters.

Increasing the dataset size to 5000 can lead to a degradation in PSNR and a significant increase in runtime.

The worst model is the model with the parameters (1000 data, 100 epochs, 64 -batch-size) (Fig. 22).

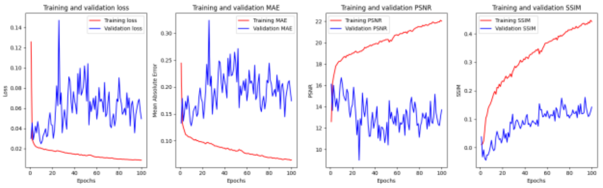


Fig 22: The worst model obtained with parameters (1000 data, 100 epochs, 64 batch size)

The graphs show that the model is not trained, as evidenced by the metrics. loss, MAE do not decrease, but rather increase, PSNR decreases, and SSIM does not increase, which indicates a clear overtraining.

Analysis of the obtained results for CNNs with L1/L2 regularization and Dropout. Initially, the application of regularization and Dropout resulted in a significant improvement in PSNR on the test data.

Dropout also led to a slight improvement in SSIM for the 25 epoch configuration.

From the graphs, it can be observed that in general, the metrics results for the test data became more normalized and less unpredictable.

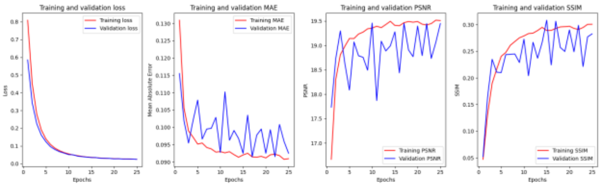


Fig 23: Results of model training with parameters (5000 data, 25 epochs, 32 batch-size)

The graphs show that the loss drops off to zero rather slowly and smoothly; the PSNR and SSIM values increase more steadily upward compared to the model above, eventually reaching a maximum value of about 19.5 for PSNR and 0.3 for SSIM, which is the best result among all models.

5. CONCLUSIONS

Having selected 3 de-noising algorithms, a software implementation was developed, after which the experiments conducted were described in Section 3. As expected, due to their simplicity, the spectral subtraction and Wiener filter algorithms did not cope very well with the task, however, using convolutional neural networks CNN, it was possible to achieve a fairly good result. In Section 3, all the metrics used to assess the accuracy of the models were described, and a comparative analysis of the three methods was also carried out. Based on the study, it can be concluded that using convolutional neural networks is currently the best way to de-noise audio recordings. An analysis of the results obtained from the metrics leads to the following conclusion: Spectral subtraction performs slightly better than the Wiener filter, as evidenced by both PESQ estimates and audiovisual analysis. Among all methods, convolutional neural networks demonstrate the best performance. The best results are achieved using L1/L2 regularization and Dropout.

The choice of an audio de-noising algorithm depends on the type of noise, signal characteristics, and available resources. Wiener filter: Recommended for simple types of noise when available computing resources are limited; Spectral subtraction: Can be used as a quick and easy method to remove some types of noise; CNN: Recommended for complex types of noise when significant computing resources are available.

Further research may include researching new CNN architectures for audio de-noising and exploring the possibilities of using other types of neural networks.

6. ACKNOWLEDGEMENTS

The study was created within the framework of the project financed by the National Research Fund of Ukraine, registered No. 30/0103 from 01.05.2023,"Methods and means of researching markers of ageing and their influence on post-ageing effects for prolonging the working period", which is carried out at the Department of Artificial Intelligence Systems of the Institute of Computer Sciences and Information Technologies of the Lviv Polytechnic National University.

7. REFERENCES

[1] Le Roux J., Vincent E. 2013. Consistent Wiener Filtering for Audio Source Separation, IEEE Signal Processing Letters, Vol. 20, No. 3, pp. 217–220. https://doi.org/10.1109/lsp.2012.2225617

[2] Tran T., Bader S., Lundgren J. 2023. Denoising Induction Motor Sounds Using an Autoencoder, IEEE Sensors Applications Symposium (SAS), Ottawa, ON, Canada, pp. 18–20. https://doi.org/10.1109/sas58821.2023.10254150.

[3] Upadhyay N., Karmakar A. 2015. Speech Enhancement using Spectral Subtraction-type Algorithms: A Comparison and Simulation Study, Procedia Computer Science, Vol. 54, pp. 574–584. https://doi.org/10.1016/j.procs.2015.06.066.

[4] Ashwin J. S., Manoharan N. 2018. Audio Denoising Based on Short Time Fourier Transform, Indonesian Journal of Electrical Engineering and Computer Science, Vol. 9, No. 1, pp. 89. https://doi.org/10.11591/ijeecs.v9.i1.pp89-92.

[5] Junfeng L., Masato A., Yoiti S. 2010. A Two-Microphone Noise Reduction Method in Highly Nonstationary Multiple-Noise-Source Environments, IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences. https://doi.org/E91A. 10.1093/ietfec/e91-a.6.1337.

[6] Edmonson J W. 2002. Tucker, Digital Signal Processing System for Active Noise Reduction, Vol. 1, p. 49.

[7] Boll S. 2014. Suppression of acoustic noise in speech using spectral subtraction, IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 27, No. 2, pp. 113-120. https://doi.org/10.1109/TASSP.1979.1163209.

[8] Xu Y., Du J., Dai L., Lee C. 2015. A Regression Approach to Speech Enhancement Based on Deep Neural Networks, IEEE/ACM Transactions on Audio, Speech, and Language Processing, Vol. 23, No. 1, pp. 7-19. https://doi.org/10.1109/TASLP.2014.2364452.

[9] Valin J. 2018. A Hybrid DSP/Deep Learning Approach to Real-Time Full-Band Speech Enhancement, IEEE 20th International Workshop on Multimedia Signal Processing (MMSP), pp. 1-5, https://doi.org/10.1109/MMSP.2018.8547084.

[10] Keshavarzi M. 2018. Use of a Deep Recurrent Neural Network to Reduce Wind Noise: Effects on Judged Speech Intelligibility and Sound Quality, Trends in Hearing. https://doi.org/10.1177/2331216518770964.

[11] Omaima A. 2015. Removing Noise from Speech Signals Using Different Approaches of Artificial Neural Networks, International Journal of Information Technology and Computer Science, Vol. 7, pp. 8-18. https://doi.org/10.5815/ijitcs.2015.07.02.

[12] Taal C. H., Hendriks R. C., Heusdens R. and Jensen J. 2010. A short-time objective intelligibility measure for time-frequency weighted noisy speech, IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 4214-4217. https://doi.org/10.1109/ICASSP.2010.5495701.

[13] Sun C., Zhang M., Wu R. 2021. A convolutional recurrent neural network with attention framework for speech separation in monaural recordings, Vol. 11, pp. 1434. https://doi.org/10.1038/s41598-020-80713-3.

[14] Boyko N. 2023. Models and Algorithms for Multimodal Data Processing, WSEAS Transactions on Information Science and Applications, ISSN / E-ISSN: 1790-0832 / 2224-3402, Vol. 20, pp. 87-97. https://doi.org/10.37394/23209.2023.20.11.

[15] Boyko N. 2023. Evaluating Binary Classification Algorithms on Data Lakes Using Machine Learning, Revue d'Intelligence Artificielle, Vol. 37(6), pp. 1423–1434. https://doi.org/10.18280/ria.370606