Universidade do Minho
Escola de Engenharia

Diogo Miguel Cunha Fernandes

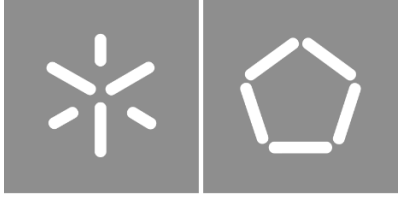
FPGA accelerated TFR generator for applications based on CNN techniques

FPGA accelerated TFR generator for applications based on CNN techniques

Diogo Miguel Cunha Fernandes

UMinho|2023

janeiro de 2023



Universidade do Minho
Escola de Engenharia

Diogo Miguel Cunha Fernandes

FPGA accelerated TFR generator for applications based on CNN techniques

Dissertação de Mestrado
Mestrado em Engenharia Eletrónica Industrial e Computadores
Sistemas Embebidos e Computadores

Trabalho realizado sob a orientação do
Professor Doutor Rui Pedro Oliveira Machado

janeiro de 2022

Contents

List of Figures

List of Abbreviations

CAD	Computer-Aided Design.
CNN	Convolutional Neural Network.
CWT	Continuous Wavelet Transform.
DFT	Discrete Fourier Transform.
DL	Deep Learning.
DSP	Digital Signal Processing.
DWT	Discrete Wavelet Transform.
EMD	Empirical Mode Decomposition.
ESC	Environmental Sound Classification.
FFT	Fast Fourier Transform.
FPGA	Field-Programmable Gate Array.
FSLL	First Side-Lobe Level.
FT	Fourier Transform.
HHT	Hilbert-Huang Transform.
LGS	Log-Gammatone Spectrogram.
LMS	Log-Mel Spectrogram.
ML	Machine Learning.
PCB	Printed Circuit Board.
PDS	Power Density Spectrum.
PSLL	Peak Side-Lobe Level.
RFSLL	Rate of Fall-off of Side-Lobe Level.
RPM	Revolutions per Minute.
STFT	Short-Time Fourier Transform.

TFR	Time-Frequency Representation.
WA	Wavelet Analysis.
WPT	Wavelet Packet Transform.
WT	Wavelet Transform.

Chapter 1: Introduction

This chapter aims to provide a brief overview of the dissertation's theme. It will start with its contextualization, making a problem statement for a better understanding of the relevance and significance of the topic. Then it is presented the motivation for the study of this matter. Finally, one defines the objectives of this dissertation.

1.1 Contextualization

Nowadays, Deep Learning (DL) methods are everywhere and allow us to design systems that automatically solve complex tasks and problems that before could only be solved by humans. They have constantly evolved, closing the gap between human and device capacities. Traditional methods of Machine Learning (ML) require a set of features manually obtained, frequently generating unreliable predictions. DL methods have such a powerful feature learning ability that they overcome this problem, through the capacity of learning hierarchical representations from unprocessed data [1]. There are many applications for DL methods, like object detection and identification, voice assistance, autonomous vehicles, and faults diagnosis, among others. A DL method widely used for image and video recognition is the Convolutional Neural Network (CNN) [2]. The advantage brought by DL models and the high accuracy achieved, made the CNN artificial neural network an arising technology in recent years of ML [3].

A Time-Frequency Representation (TFR) is a type of representation that displays the frequency content of a signal as a function of time and is obtained through a time-frequency domain transform. It allows for the analysis of a signal's spectrum over time, rather than just at a single interval in time as in a traditional frequency domain analysis. TFRs provide a more detailed and complete understanding of the characteristics of a signal, particularly when it is non-stationary, meaning its frequency content changes over time. This way, a TFR creates a visual representation of a signal's time-frequency domain, where the abscissa represents time and the ordinate represents frequency. The energy of the signal is distinguished using a color map.

The TFR technique is useful when analyzing analog signals obtained from sensors because it allows for a better understanding of the sampled data in relation to time and because different patterns in the time-frequency domain are specific to different conditions of signals. Therefore, a signal's TFR can be

applied to a CNN model in order to detect and identify the patterns in the captured signal.

A TFR can be used in the context of many applications that employ a DL method or similar artificial intelligence methods. In the automotive industry, a microphone can be used to detect the alarm sirens of emergency vehicles trying to pass, allowing the drivers to know, in advance, their presence [4]. Also in this segment, it can be used an accelerometer to detect a minor crash when parked, activating the alert system and, eventually, taking a picture of the car's registration plate. Predictive maintenance is another application for TFR. These applications intend to ensure that machine faults can be detected and diagnosed [2]. When the system detects a fault, the production line can be stopped, allowing a much faster repair than it was compared to a forced stopping, reducing its costs. Regarding audio processing, TFR may be the feature extraction algorithm for an animal tracker [5], in which one can identify and geographically locate them. Furthermore, seismology is another area where one can apply TFR techniques to detect earthquakes and tectonic movements. Figure 1.1 illustrates six TFRs recorded in six different locations of a 4,9 magnitude earthquake that occurred on the northern California coast [6]. The red color represents high energy and the blue color represents low energy. An earthquake can be divided into two waves: the first P-wave and the second S-wave. As one may see in the TFRs, the P-waves have higher frequency components than the S-waves, which have higher energy.

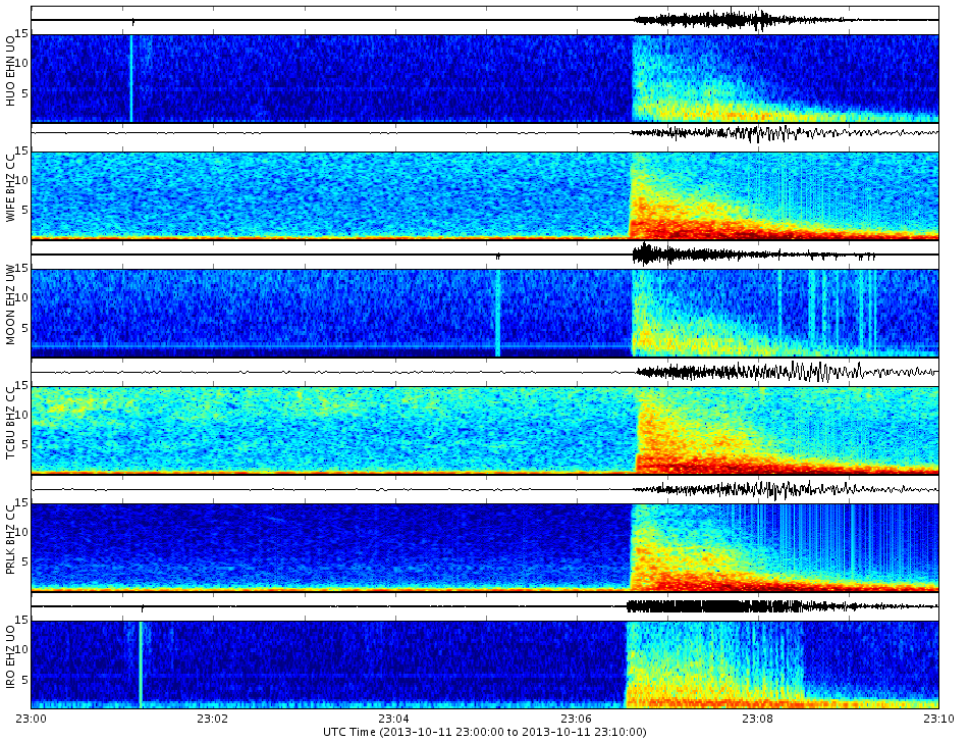


Figure 1.1: TFR of a 4,9 magnitude earthquake occurred in the northern California coast, obtained from six different locations [6].

This dissertation proposes to design and implement a Field-Programmable Gate Array (FPGA) accelerated system capable of sampling an input signal, generating the respective TFR, and sending it to another system to be analyzed by a CNN model. The system must sample two signals: one from an accelerometer and one from a microphone. The sampling frequency needs to be adequate for the type of signals to be studied. Knowing that the limits of human hearing are roughly between 20 and 20 kHz, one can define the minimum sample frequency for the microphone as 40 kHz, which corresponds to two times higher than the maximum frequency of a signal. The sampling frequency for the accelerometer may be lower than the last one because the vibration signals have lower frequency components than sound signals.

1.2 Motivation

As motivation, one can emphasize the constant evolution of technology and industry. The number of sensors used in a car has increased exponentially in the last decade [7], bringing the necessity of systems that process some of the data produced by them. This system is an adaptable solution given that it can be applied to whatever analog sensor signal that one wants to use, decreasing the final product's market value and time-to-market.

The computation of a TFR of an input signal is a complex process, as it generates lots of data and it includes complex mathematical operations. This way, it takes a significant amount of time to generate a signal's TFR using a software-based application. Thus, it is important to design a system that is accelerated in a FPGA platform, that calculates the signal's TFR and outputs it to another system for application of an appropriate CNN model. Beyond that, one knows that CNN models are a time-consuming process as well. Therefore, this system proposition can be extended to embed a DL classification model, implemented in FPGA.

Moreover, the advance of DL models facilitates its implementation and adaption to different datasets and TFRs of different types of input signals. Thereby, this system may be easily adjusted to other types of signals, making it a modular-like system.

1.3 Objectives

The main objective of this dissertation is, as previously stated, to implement a TFR generator accelerated in a FPGA platform. However, one split it into more specific objectives, being enumerated below:

- **Review the state-of-the-art TFRs:** identify, explore and compare the leading technologies to

generate TFRs from an analog signal;

- **Design and implement a Printed Circuit Board (PCB)** to integrate the sensors (microphone and accelerometer). First, study the best-fit sensors for the application's requirements and constraints. Test the functionality of the sensors with a microcontroller. Design the PCB layout, using a Computer-Aided Design (CAD) software, and fabricate it. Assemble the sensors by soldering them following the manufacturer's instructions;
- **Design and implement a FPGA-based hardware accelerator for generation of TFRs**, according to the state-of-the-art techniques. Optimize the hardware design of the system for improved performance. Evaluate the FPGA resource utilization. Investigate the potential for scaling the system to handle a greater number of sensors or higher data rates;
- **Develop a communication interface between the FPGA and a computer**, and a desktop application that converts TFR generated data into an image;
- **Evaluate the performance of the proposed system**. Develop a software-based or high-level application that generates a TFR based on the sensor's data. Compare the performance to other state-of-the-art TFR generators and the software-based application;
- **Document the design, implementation and results** of the system in this dissertation.

Chapter 2: State-of-the-Art

This chapter presents a state-of-the-art review that focuses on how to obtain a TFR that will be input to a CNN algorithm. It is important to find a good TFR method that best represents an analog time domain signal, in an effort of obtaining the highest accuracy with the CNN model. It is also relevant to reach the most efficient method of time-frequency domain analysis, in order to reduce memory usage and computation time.

2.1 Time-Frequency Domain Analysis

The frequency domain analysis is an excellent method for analyzing periodic signals [8]. Its magnitude provides the spectral components of the signals and its phase includes information about time distribution. Still, in the frequency domain, it is difficult to infer the time distribution of a signal, which isn't a problem when analyzing periodic signals since the spectral components are stationary. However, not all signals have periodic components, and some may have a complex or random pattern over time. The presence or absence of periodic components affects how a signal is analyzed and processed. This way, it is essential to consider this aspect when performing signal processing, as it may be relevant to study the non-stationary components of the signal. For example, to analyze the condition of rotating machines is used a microphone to capture data during a speed sweep. The motor can either be started from a low Revolutions per Minute (RPM) and increased to a high RPM, or it can be started from a high RPM and decreased to a low RPM. In this case, it will produce non-stationary signals, as their frequency content is changing over time [9].

The time-frequency domain analysis has been one of the most effective approaches to solve signal-processing problems, considering it allows the identification and qualification of oscillatory components present in non-stationary waveforms, which are prevalent in real-life signals. It is useful to study a signal's structure in time and frequency simultaneously, through constructed projections of the signal in the time-frequency plane. The combination of these projections is called Time-Frequency Representation (TFR). TFRs are used extensively in many areas of science and are known for their powerful ability to analyze signals in the time-frequency domain. They are consistently applied to image processing, finance, geo-physics, and biological science, among other fields. The widespread use of TFRs to analyze non-stationary signals has made them an essential tool for studying the time-varying properties of complex systems. A

TFR can be used by other algorithms, like ML algorithms, that detect patterns in the time-frequency domain representation of the signal, in order to identify them [10, 11, 12]. Avery Wang [13], responsible for developing the Shazam algorithm, that identifies songs using a mobile phone microphone, uses TFRs to extract a fingerprint of a sampled audio file. The fingerprint consists of a "constellation map" obtained by drawing out the peak points present in the TFR of the audio file and associating them through hash tokens. To identify the music, the extracted fingerprint is compared to all fingerprints in a database.

Several time-frequency domain analysis techniques have been developed like Short-Time Fourier Transform (STFT), Wavelet Transform (WT), Empirical Mode Decomposition (EMD) [14], Hilbert-Huang Transform (HHT) [15], among others. However, the STFT and the WT are the most used ones because of their performance in feature extraction and their potential to generate a graphical distribution of the TFR, which is essential for the CNN model. In this section, these two methods are described in terms of advantages and disadvantages and their respective related works.

2.1.1 Short-Time Fourier Transform

The Short-Time Fourier Transform (STFT), first introduced by Gabor [16], is an extension of the Fourier Transform (FT) [17], being a widely used method for studying non-stationary signals. The STFT is a method that analyses a signal by breaking it up into shorter frames and computing the FT of each frame. The FT of the STFT is the DFT if the signal is a discrete-time signal. This allows studying the frequency components of the signal over time (time-frequency domain analysis), instead of just its overall frequency content. In practice, the Fast Fourier Transform (FFT) is often used to perform the DFT of the STFT [18, 19], as the FFT is a more efficient (fast) algorithm to compute the DFT. Reminding from the frequency domain analysis, the DFT, applied to a signal, x, is defined by equation 2.1,

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}nk} \tag{2.1}$$

where N is the total number of samples in the signal x and k represents frequency. Equation 2.1 gives the spectrum computation of one section, which size corresponds to the size of the entire sampled signal, N. To study the spectral components over the time domain with STFT, one has to divide the signal into small frames. Ideally, the frame should be very small, so that the signal, in that small chunk of time, would be periodic. This would avoid the spectral leakage, caused by the discontinuities between consecutive frames. One can determine the running spectrum by adapting the equation 2.1, to the

equation 2.2.

$$STFT\{x(n)\} = STFT_{x(n)}(m,k) = X(m,k) = \sum_{n=0}^{N-1} x(n+mH)w(n)e^{-j\frac{2\pi}{N}nk} \tag{2.2}$$

The equation 2.2 computes the Fourier coefficients for the k^{th} frequency at the m^{th} frame of the signal. Figure 2.1 shows an explanation of the parameters used in the equation 2.2. The parameter m represents the frame number to be analyzed, and k is the frequency component. Furthermore, H represents the hop size, i.e., how many samples the next frame will be shifted from the previous frame, so the mH can be seen as the beginning of a frame. This way, the DFT is applied to each frame of the signal, with size N. Thus, the range of each frame is $[mH; mH + (N-1)]$.

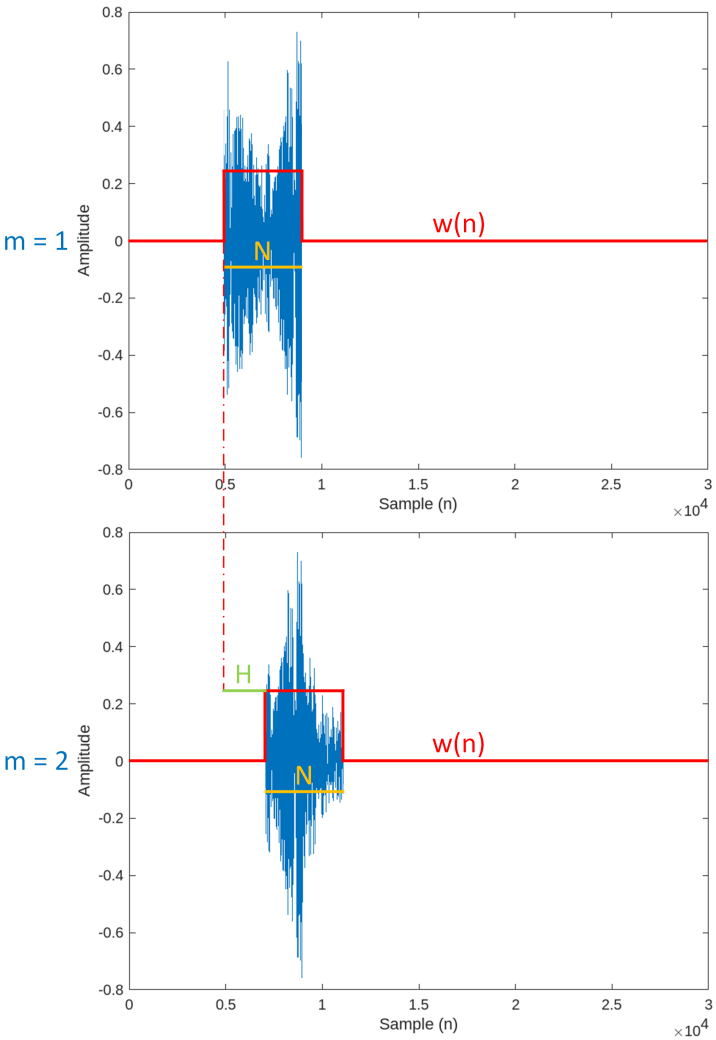


Figure 2.1: Explanation of the parameters used in the mathematical computation of the STFT.

The output of the STFT is a two-dimensional spectral matrix. The x-axis represents the frame number, being its maximum value N_{frames}. The y-axis represents the frequency bin number, and its maximum

value is $N_{freqbins}$. Each element of the matrix contains complex Fourier coefficients. The number of frequency bins can be calculated using the equation 2.3 and the number of frames is derived from the equation 2.4. The $N_{freqbins}$ may be explained by the symmetrical property of the DFT that introduces redundancy in the computed coefficients above the central frequency, which corresponds to the Nyquist frequency, $\frac{f_s}{2}$ [17]. Thus, one may consider only the first part of the coefficients and discard the other part. As the DFT is applied over a frame, the first part of the coefficients matches to $\frac{N}{2}$. Regarding the frames, one can predict that the greater the value of N_{frames}, the more computation time will be needed.

$$N_{freqbins} = \frac{N}{2} + 1 \tag{2.3}$$

$$N_{frames} = \frac{SamplesNum - N}{H} + 1 \tag{2.4}$$

As an example, consider a signal with $30\ kSamples$, which STFT is computed with a frame size of 2000 and hop size of 500. Substituting in the equation 2.3, the $N_{freqbins}$ will be 1001, so the frequency range, which is between $0\ Hz$ and $\frac{f_s}{2}\ Hz$, is divided into 1001 equal frequency bins. Resolving the equation 2.4, the number of frames, N_{frames}, will be equal to 57. This means that m ranges from 0 to 57. Hence, the output shape of the STFT is a two-dimensional array with size $(1001, 57)$.

Window

As stated before, if the signal is not periodic, discontinuities towards the edges of the signal are created when applying the DFT, introducing new high frequencies. This process is called spectral leakage, which isn't desirable in a time-frequency domain analysis. Spectral leakage happens because the DFT interprets the portion of the analyzed signal as a continuous repetition of that signal. When the signal is aperiodic (both ends of the signal are different in amplitude), which happens in most real-world situations, it will create a discontinuity in the subsequent repetitions of the signal, as seen in figure 2.2.

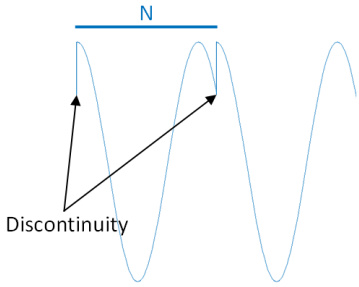


Figure 2.2: Discontinuity caused by application of DFT to an aperiodic signal, in the time domain.

In order to avoid discontinuities, one has to attenuate both signal edges, applying a window function. The window function to choose from depends on the type of application, but the most common ones include the Hann window, the Hamming window, and the Kaiser window. The Hamming and Hann windows are most suited for audio and vibration signal analysis [20, 19, 21, 22], due to their noise performance and lower side lobes. Although they are two similar window functions, the Hamming window performs better at canceling the First Side-Lobe Level (FSLL), whereas the Hann window has a better Rate of Fall-off of Side-Lobe Level (RFSLL). The Kaiser window is a general-purpose window, as it allows more customization of the window response for the type of application [22]. According to Prabhu [23], there are four ways to identify the necessary window function to be used in a specific application. This dissertation requires the analysis of two signals: one from a microphone and another from an accelerometer. The microphone signal fits "Case 3" because one wants to consider far-away frequency components with unequal strengths. For this case, the best-fit window is the Hann window, because of its high RFSLL, which makes the side lobes fall off faster. The case of the accelerometer signal should match "Case 2", which corresponds to the analysis of signals with near-frequency components with unequal strengths. In this case, the most appropriate window is the Hamming window, as the FSLL is smaller than the Peak Side-Lobe Level (PSLL). The rectangular window, used in figure 2.1, is not appropriate for computing the STFT because it creates discontinuities on the edges of the window. The equation 2.5 represents mathematically the Hamming window response in the time domain. Figure 2.3 illustrates the time and frequency domain response of the Hamming window of size 50. One can confirm that the second side lobe is more attenuated and the attenuation through the rest of the side lobes is approximately constant.

$$w(n) = \begin{cases} 0,54 - 0,46\cos\left(\dfrac{2\pi n}{N}\right), & 0 \leq n \leq N-1 \\ 0\,, & otherwise. \end{cases} \tag{2.5}$$

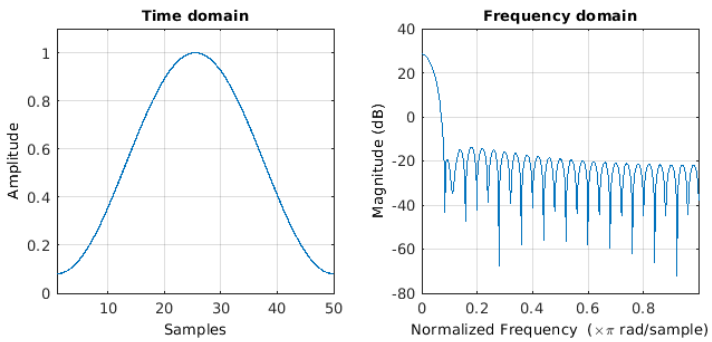


Figure 2.3: Hamming Window: Time and Frequency Domain Response with $N = 50$.

Figure 2.4 shows the Hann window response in time and frequency domains, with a window size of 50. One may see that the responses in time and frequency of Hann and Hamming windows are very similar, however, there are some differences between the two. In the time domain, the only difference is that the Hann window reaches zero amplitude at both ends, while the Hamming window doesn't. In the frequency domain, the Hamming window has a lower FSLL than the Hann window, but the last has a higher RFSLL.

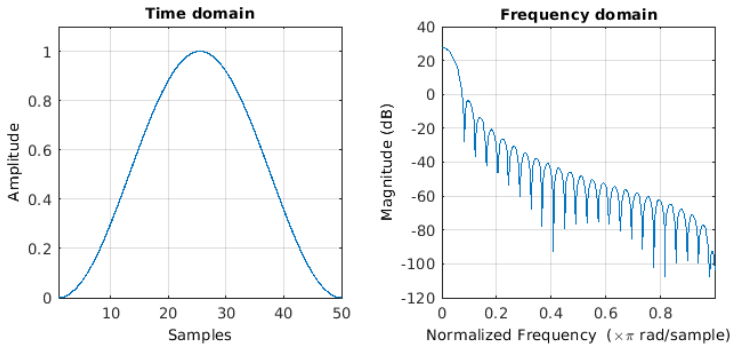


Figure 2.4: Hann Window: Time and Frequency Domain Response with $N = 50$.

Figure 2.5 represents a sampled signal to be analyzed, $x(n)$, and the result of the application of the Hamming window function, in the time domain, $x_w(n)$. As one can see, the signal gets modulated towards the ends, avoiding spectral leakage.

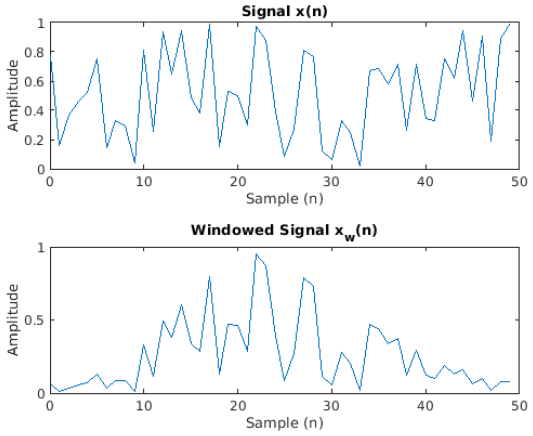


Figure 2.5: Signal, $x(n)$, and result of application of Hamming window, $x_w(n)$, with $N = 50$, in time domain.

Limitations

The computation of the STFT, as all Digital Signal Processing (DSP) techniques, has its own advantages and disadvantages. With this technique, there is a limit on the resolution plotted about the time and

frequency domains. A time series can be accurately measured in terms of when events occur in the time domain analysis but provide no information about what frequency components are present. The Fourier transform, on the other hand, can provide precise frequency components but fails to supply information about when these frequencies occur. The STFT offers a balance between time and frequency resolution, at the cost of lower resolution in each domain. This principle is called the uncertainty principle [16, 24] and demonstrates that there is a fundamental trade-off between time and frequency domain resolution. The parameter that defines the time-frequency domain resolution is the width of the window (frame size N) [25]. A large frame size gives poor time resolution but relatively good frequency resolution. A smaller frame window gives poor frequency resolution but relatively good time resolution. This trade-off depends on the type of problem to be studied. An alternative approach, called multi-resolution analysis, will be discussed in the next section (2.1.2). Figure 2.6 represents the time-frequency fixed resolution of the STFT. In this case, the time resolution is higher than the frequency resolution.

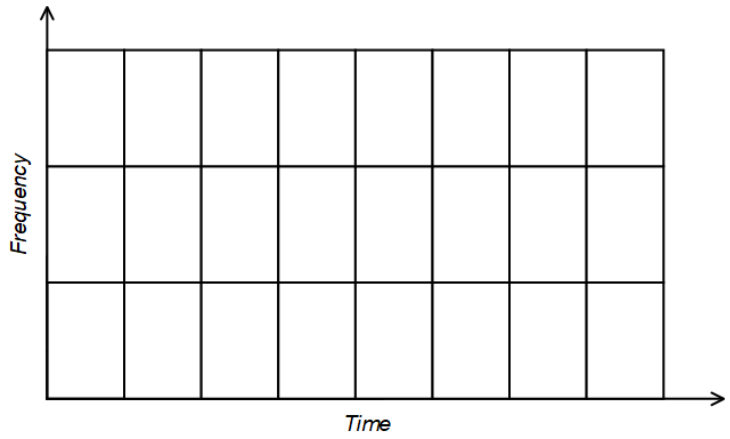


Figure 2.6: TFR Fixed Resolution of the STFT.

Spectrogram

The Power Density Spectrum (PDS) gives a representation of the distribution of the frequency components of a signal, that is easier to visualize than the complex numbers given by the transform. However, one can't use the Power Density Spectrum (PDS), as the signals to be studied won't be stationary. This way, the magnitude squared of the STFT, represented in the equation 2.6, can be calculated to estimate the PDS [26]. This estimation is called the spectrogram and is a graphical representation of a signal that estimates the energy distribution of the signal over the time-frequency domain, where the x-axis represents time and the y-axis represents frequency.

$$SPEC_{x(n)}(m,k) = |STFT_{x(n)}(m,k)|^2 \tag{2.6}$$

Figure 2.7 shows a generated spectrogram of a linear chirp function, which is a function that generates a sinusoidal waveform, that changes constantly its frequency, whether increasing or decreasing. In figure 2.7, the spectrogram shows that the frequency of the signal is increasing over time.

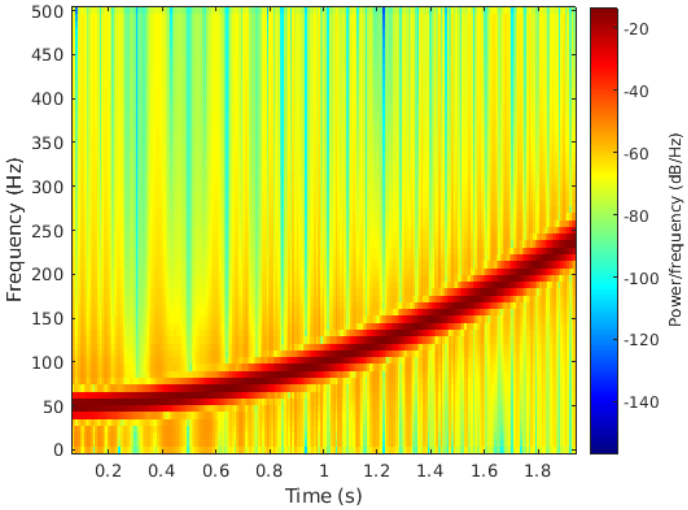


Figure 2.7: Spectrogram of a linear chirp function.

The STFT and spectrograms have been consistently applied in many studies for different types of signals. Some of them have been presented in this dissertation [12, 13]. Wibawa et al. [27] proposed a system that detects abnormal heart rhythm, using spectrograms obtained from heart beating sounds. The idea is to use the spectrograms in a CNN algorithm for the detection of anomalies, achieving an accuracy of 82,75 %. Furthermore, Chi et al. [28] proposed a system that makes Environmental Sound Classification (ESC) using two types of logarithmic spectrograms (*Log-Mel Spectrogram (LMS)* and *Log-Gammatone Spectrogram (LGS)*). The *mel* and *gammatone* spectrograms are obtained using two filterbanks that are designed to approximate the audio processing to the human auditory system, providing a more precise representation of the frequency content of a signal compared to the linear frequency scales. They converted the spectrograms to a logarithmic scale and compared their classification accuracy with a CNN model. Although the LMS got better results than the LGS, the concatenation of both types of spectrograms turned out to be the better approach. They were able to surpass other classification methods and even human performance, with an overall accuracy of 83,80 %. The proposed system is shown in figure 2.8.

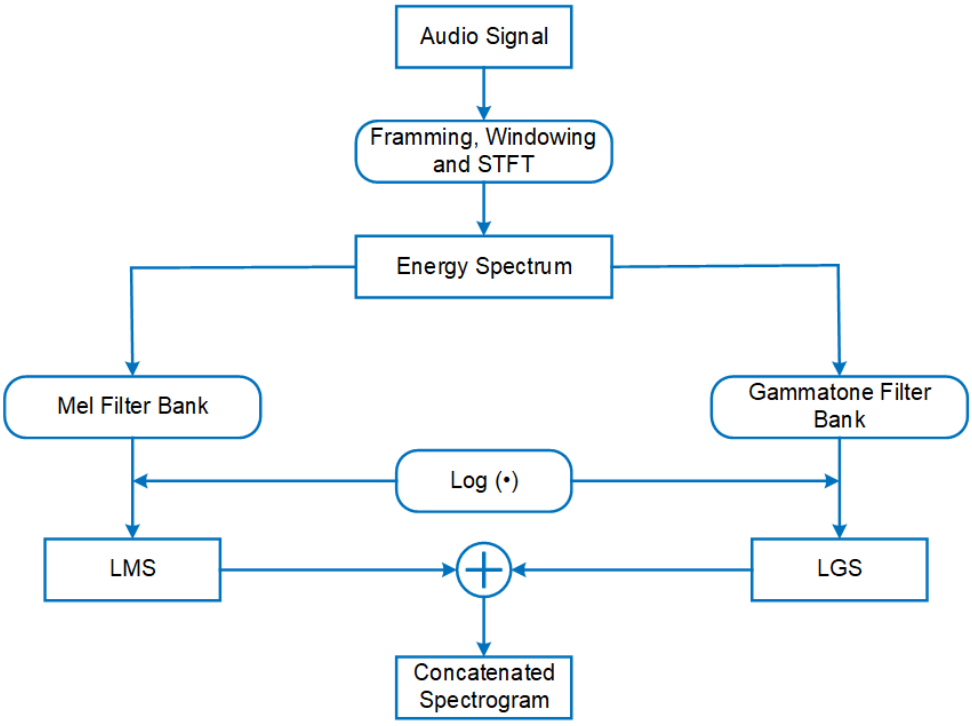


Figure 2.8: LMS and LGS concatenated spectrogram generation process [28].

2.1.2 Wavelet Transform

The Wavelet Analysis (WA) is another method used for the time-frequency domain analysis of a signal. It is mostly used to analyze time series containing non-stationary components at a wide frequency range. As seen above, the FT and the STFT use sine and cosine waveforms as analyzing functions ($e^{-j2\pi nk/N}$) through convolution with the original signal. On the other hand, the WT uses a family of wavelets as analyzing functions. With this technique, the WT partially overcomes the uncertainty principle by exploiting a multi-resolution analysis, that is illustrated in figure 2.9. This is, WT allows the analysis of signals at different frequencies with different resolutions. For example, consider a signal with a wide range of frequency components. For the lower frequencies, the data will change slower in the time domain, for which one doesn't need much resolution in the time domain, so it can be used the resolution represented by bars (c) in figure 2.9. The next level of resolution, represented by bars (b), has a higher time resolution and a lower frequency resolution. Furthermore, for the higher frequency components, one wants a better resolution in the time domain, as the signal changes faster over time. This way, it can be used the resolution represented by bars (a). Therefore, the multi-resolution analysis provided by the WT is built on the objective of reducing the uncertainty in the time or frequency domain for a specific frequency range. For high-frequency components, it is desirable to reduce redundancy over the time domain, and for low-frequency components, it is desirable to reduce the redundancy over the frequency domain. The higher

the time resolution needed, the lower should be the width of the wavelet in the time domain [29].

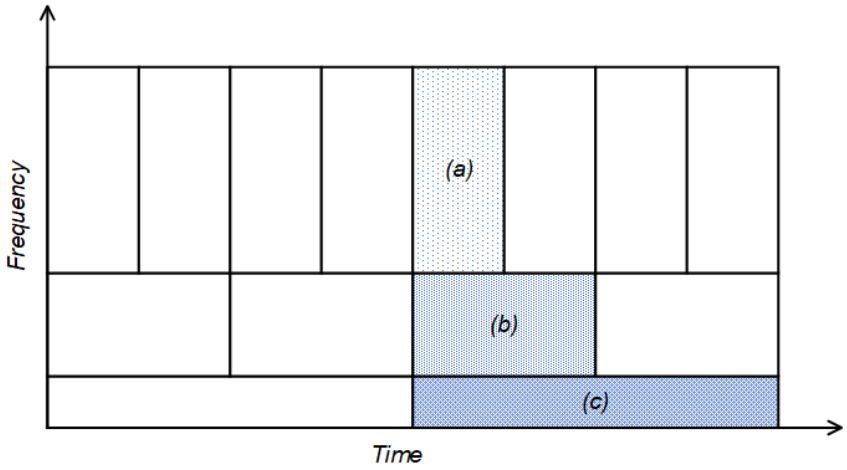


Figure 2.9: Three-level Multi-resolution Analysis.

A wavelet function or child wavelet, ψ, can be obtained in the time domain from a mother wavelet function, ψ_0, using equation 2.7, in which s is a scaling parameter and u is a translation parameter. If one doubles the s parameter, the frequency of the child wavelet will be one-half of the frequency of the mother wavelet, and vice versa. By scaling and translating a mother wavelet across the signal, it is possible to extract a multi-resolution structure that provides an optimal trade-off between time and frequency resolution.

$$\psi(t) = \psi_0 \left(\frac{t-u}{s} \right) \tag{2.7}$$

The choice of the mother wavelet function depends on the type of problem to be studied and is based on some metrics [30]. There are many mother wavelet functions with different shapes and properties. The most commonly used are the Haar, Morlet, Daubechies, and Mexican hat wavelets. In fact, one can even design a wavelet function, specifically for an application. Nevertheless, to be considered a mother wavelet, a function has to fulfill certain properties [31]. A mother wavelet has finite energy (equation 2.10), meaning that it tends to zero at both ends. A wavelet function is scalable, which makes it adaptable to a wide range of frequency- and time-based resolutions. Furthermore, the wavelet average should be equal to zero, i.e, its integral is equal to zero (equation 2.8) and the squared FT of a mother wavelet at zero frequency should be zero (equation 2.9). This is called the admissibility principle of a mother wavelet [32].

$$\int_{-\infty}^{+\infty} \psi_0(t)\, dt = 0 \tag{2.8}$$

$$|\Psi_0(w)|^2\big|_{w=0} = 0 \tag{2.9}$$

$$\int_{-\infty}^{+\infty} |\psi_0(t)|^2 \, dt < \infty \tag{2.10}$$

In spite of the existence of many wavelet functions, the most used one for feature extraction for a CNN model is the Morlet wavelet [33, 34, 35, 36] because it has a good time-frequency localization and is a non-orthogonal wavelet, which allows obtaining a better resolution in the TFR [30, 37]. Konar and Chattopadhyay [38] showed that choosing the proper wavelet is essential. They compared the Morlet wavelet with the Daubechies wavelet in a bearing fault detection system, obtaining a better result with the Morlet wavelet. Figure 2.10 shows the plot of a mother Morlet wavelet function in the time domain.

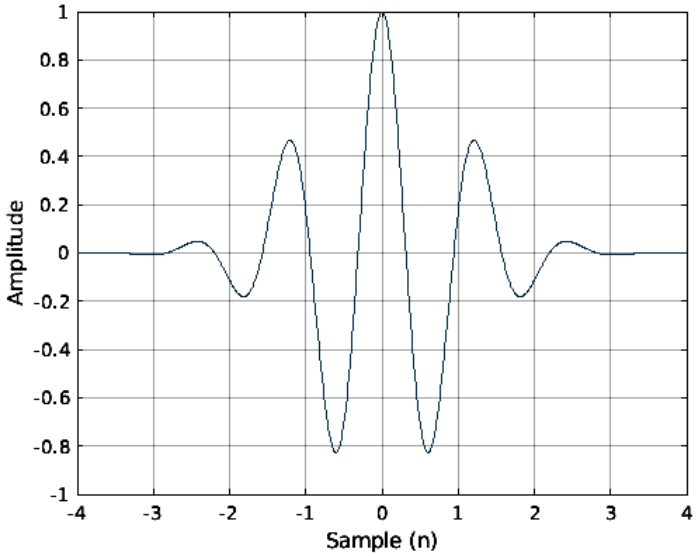


Figure 2.10: Morlet wavelet function in the time domain.

Continuous Wavelet Transform

The equation 2.11 expresses the computation of the CWT of a continuous signal $x(t)$,

$$X(s,\tau) = \int_{-\infty}^{+\infty} x(t)\psi^*\left(\frac{t-\tau}{s}\right) \, dt \tag{2.11}$$

in which the ψ^* represents the complex conjugate of ψ, scaled and translated through the parameters s and τ, respectively. This way, one can change the width and central frequency of the wavelet using these parameters. Changing the width of the wavelet is called scaling. If it is applied a small s value to the wavelet, the mother wavelet is shrunken, being more adequate for the analysis of high-frequency

components, achieving a better time resolution and a poorer frequency resolution. On the other hand, if a large s parameter is applied to the wavelet, its width is expanded and this type of wavelet is better at resolving low-frequency components of the signal, obtaining lower time resolution and higher frequency resolution. The output of $X(s,\tau)$ are coefficients that measure the correlation between the signal and the wavelet scaled and shifted by s and τ, respectively. So the higher the similarities between the signal and the wavelet function in both amplitude and frequency, the higher will be the output of $X(s,\tau)$.

The equation 2.11 can be rewritten in the form of summation for a discrete-time signal, x_n, as shown in equation 2.12

$$X(s,n) = \sum_{n'=0}^{N-1} x(n')\psi^* \left(\frac{(n'-n)\delta t}{s} \right) \tag{2.12}$$

where δt is the time step and N is the number of samples in the time series signal. This convolution can be done in two ways: time-domain convolution and frequency-domain convolution [37]. The first is computed by sliding the wavelet along the signal and computing the dot product at each time step. The second is based on the convolution theorem and is computed by taking the DFT of the signal and the wavelet, multiplying the results, and then taking the inverse FT. Although the two methods produce the same results, the frequency-domain convolution is faster, and it can be further expedited using the FFT algorithm [39]. Using the frequency-domain convolution, the CWT can be written as equation 2.13, where the angular frequency, w_k, is defined in equation 2.14.

$$X(s,n) = \sum_{k=0}^{N-1} X(k)\Psi^*(sw_k)\, e^{iw_k n\delta t} \tag{2.13}$$

$$w_k = \begin{cases} \dfrac{2\pi k}{N\delta t}, & k \leq \dfrac{N}{2} \\ \\ -\dfrac{2\pi k}{N\delta t}, & k > \dfrac{N}{2} \end{cases} \tag{2.14}$$

Scaling

When applying a scale, s, to the mother wavelet, the amplitude of the child wavelet spectrum will also be scaled. In order to ensure that the WTs at every scale, s, are directly comparable to each other, the child wavelet function must be normalized to have unit energy. This is done using equation 2.15.

$$\Psi(sw_k) = \left(\frac{2\pi s}{\delta t} \right)^{1\backslash 2} \Psi_0(sw_k) \tag{2.15}$$

In addiction, after choosing the wavelet function, one has to choose a set of scales to be used in the WT, in order to obtain the multi-resolution analysis. With the aim of computing the CWT more efficiently, it is suitable to use scaling factors as fractional dyadic numbers, as given by equations 2.16 and 2.17 [30].

$$s_j = s_0 2^{j\delta j} \ , \quad j = 0, 1, 2, ..., J \tag{2.16}$$

$$J = \delta j^{-1} log_2 \left(\frac{N \delta j}{s_0} \right) \tag{2.17}$$

in which s_0 is the smallest scale and J is the number of the last scale. The s_0 parameter should be equal to $2\delta t$ and the parameter δj depends on the type of wavelet. The lowest the δj, the finer will be the obtained resolution, and the higher will be the number of coefficients generated. For the resolution of the figure 2.9, $J = 2$ because it has three levels of resolution.

There are other forms of applying the wavelet transform such as Discrete Wavelet Transform (DWT) and Wavelet Packet Transform (WPT), that reduces significantly the number of coefficients and the necessity of computational power comparable to the CWT [19]. However, these methods are mainly used for denoising and compression of signals and images [40, 30].

Scalogram

As for the STFT, it is also important to have a representation of the coefficients computed by the WT. The coefficients of the WT are complex numbers, such that the real part defines the amplitude and the imaginary part defines the phase. In some time series analysis, like analysis of peaks and discontinuities, the real component of the transform may be enough, while for others, such as oscillatory signals, it is adequate to study both components [41]. This way, the scalogram is a graphical representation of the wavelet power spectrum, which is an analysis that includes both real and imaginary components of the wavelet transform, through the magnitude squared of the coefficients. Like the spectrogram, the x-axis represents the time domain, while the y-axis represents the frequency domain. A color map is used to represent the power of the signal in each time-frequency localization. The equation 2.18 shows how to obtain the scalogram from the wavelet transform. Figure 2.11 illustrates the scalogram of a linear chirp function

$$SCAL_x(s, n) = |X(s, n)|^2 \tag{2.18}$$

Ivan Kiskin et al. [5] designed a wavelet-conditioned CNN that achieved an accuracy of over 90 %

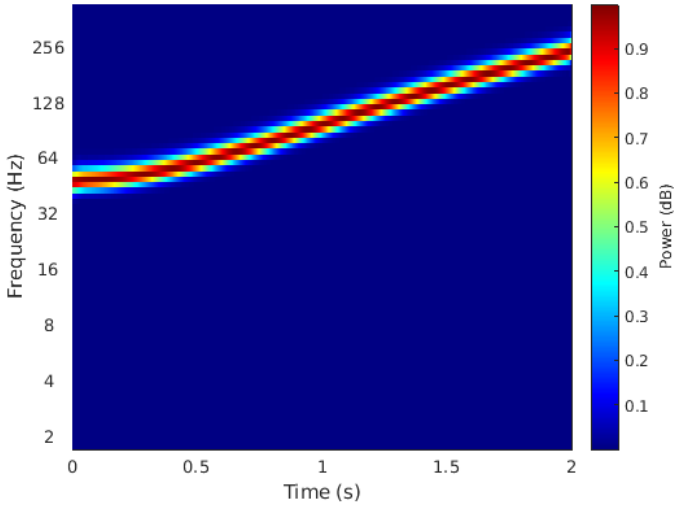


Figure 2.11: Scalogram of a linear chirp function.

in detecting mosquitoes. It is also mentionable that the designed CNN may be overextended for generic applications, as it was able to distinguish between nine bird species, with also over 90 % accuracy. Qassim et al. [42] implemented a system accelerated in FPGA that computes the CWT in the Fourier space. The proposed flow chart for the implementation is illustrated in figure 2.12. The function $g_1(t)$ and $g_2(t)$ are the signal and the wavelet function, and $G_1(w)$ and $G_2(w)$ their FT, respectively. As seen above, after multiplying the frequency domain signals, it is performed the inverse FT, obtaining the CWT of the signal at the scale s. They were able to reduce the memory space used in 89 % and to decrease the computation time in 0,57 ms. Despite the fact that the multi-resolution analysis was eliminated, these are very good results that show the importance of the application of the WT in the Fourier space using the FPGA. Furthermore, Almalki [36] proposed a technique to detect dangerous high-impedance faults in the medium voltage distribution using the CWT to get a scalogram and feeding it to a trained CNN algorithm AlexNet, obtaining a 98.54 % score in the classification process.

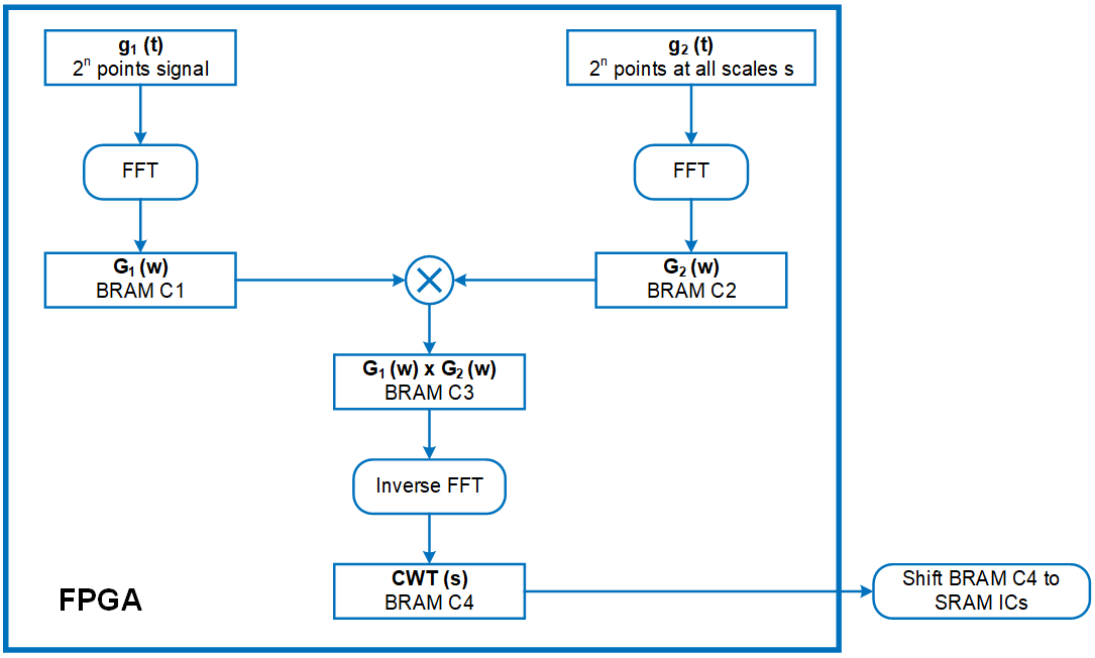


Figure 2.12: Flow chart of the digital implementation of the CWT in the Fourier space (frequency domain) [42].

2.2 Conclusion

The time-frequency domain analysis is a powerful technique that mitigates the limitations introduced by the time domain and frequency domain analysis. The TFRs have been used for studying the unsteady signals, allowing to predict where, in the time domain, the frequency components have occurred and the correspondent intensity, and providing a more complete time series analysis. This chapter brought an explanation of the two most utilized TFR techniques by other related works.

The STFT computes time and frequency information of a signal, lowering the resolution in each domain, throughout the whole signal, because of the uncertainty principle of the DFT [43], that states that one can't have both high resolution in time and frequency domains. Moreover, the fixed window parameters lead to the creation of aliasing of some frequency components that are not inserted within the frequency range of the window. This way, with a lower window size, the higher frequencies will have a high uncertainty degree, decreasing the frequency resolution. On the contrary, with a long window, the time resolution is lower, increasing the uncertainty degree about time localization. Consequently, several window lengths must be analyzed in order to determine the best-fit window for a specific application, taking into account the trade-off between time and frequency resolution. Moreover, for any window length, the $N/2\delta t$ frequencies are analyzed at each time step. These facts make the STFT an inaccurate and inefficient method for time-frequency localization. However, this is one of the most used methods for

time-frequency domain analysis, in part for being the oldest and the unfamiliarity with other methods, but mostly for providing reasonably good results, despite its drawbacks.

The other time-frequency domain method seen was the WT. This was created to reduce the problems introduced by the STFT. Even if the window size and function are the best possible for the problem, there will be a bad resolution in the time and frequency domains for some components. For this case of study, the signals have a wide range of dominant frequencies, so the best method studied is the WT, as it allows different scaling along the frequency domain, through the use of a family of wavelets. This permits to analyze the whole signal with different resolutions in time and frequency domains, arising the multi-resolution analysis.

There are many studies on time-frequency domain analysis methods. A study made by Bruns [44] showed that the WT and the STFT have approximately equal results and concluded that the most important parameter to choose is not the method of representation but the time-frequency resolution. The one made by Gupta et al. [34], where a comparison is between combinations of TFRs and ML algorithms, demonstrated that the combination CWT and CNN performed better than any other combination in speech recognition.

In conclusion, the best TFR to be used depends on the type of application but, generally, the CWT technique presents an overall better performance and results than the STFT. In this chapter was shown that for the CWT, the performance can be enhanced by deploying it to an FPGA, as the resulting coefficients and the mathematical calculations are very computationally demanding.

References

[1] S. Shao, S. McAleer, R. Yan, and P. Baldi, "Highly accurate machine fault diagnosis using deep transfer learning," *IEEE Transactions on Industrial Informatics*, vol. 15, pp. 2446–2455, 4 2019.

[2] D. Gao, Y. Zhu, X. Wang, K. Yan, and J. Hong, "A fault diagnosis method of rolling bearing based on complex morlet cwt and cnn," *Proceedings - 2018 Prognostics and System Health Management Conference, PHM-Chongqing 2018*, pp. 1101–1105, 1 2019.

[3] H. Liu, L. Li, and J. Ma, "Rolling bearing fault diagnosis based on stft-deep learning and sound signals," *Shock and Vibration*, vol. 2016, pp. 1–12, 2016.

[4] D. Pramanick, H. Ansar, H. Kumar, S. Pranav, R. Tengshe, and B. Fatimah, "Deep learning based urban sound classification and ambulance siren detector using spectrogram," *2021 12th International Conference on Computing Communication and Networking Technologies, ICCCNT 2021*, 2021.

[5] I. Kiskin, D. Zilli, Y. Li, M. Sinka, K. Willis, and S. Roberts, "Bioacoustic detection with wavelet-conditioned convolutional neural networks," *Neural Computing and Applications*, vol. 32, 02 2020.

[6] P. PNSN, "Spectrogram - regional earthquake," acedido em 23 November 2022. [Online]. Available: https://pnsn.org/spectrograms/spec-regional

[7] S. J. Prosser, "Automotive sensors: past, present and future," *Journal of Physics: Conference Series*, vol. 76, p. 012001, 7 2007.

[8] H. Ahmed and A. Nandi, *Frequency Domain Analysis*. John Wiley & Sons, Ltd, 2019, ch. 4, pp. 63–77.

[9] A. Brandt, *Noise and Vibration Analysis (Signal Analysis and Experimental Procedures)*. Wiley, 2011, ch. 12.

[10] C. Lee, C. Yoon, H.-j. Kong, H. C. Kim, and Y. Kim, "Heart rate tracking using a doppler radar with the reassigned joint time-frequency transform," *IEEE Antennas and Wireless Propagation Letters*, vol. 10, pp. 1096–1099, 2011.

[11] E. G. Strangas, S. Aviyente, and S. S. H. Zaidi, "Time–frequency analysis for efficient fault diagnosis and failure prognosis for interior permanent-magnet ac motors," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 12, pp. 4191–4199, 2008.

[12] H. Liu, L. Li, and J. Ma, "Rolling bearing fault diagnosis based on stft-deep learning and sound signals," *Shock and Vibration*, vol. 2016, p. 12, 01 2016.

[13] A. Wang, "An industrial strength audio search algorithm." 2003.

[14] P. Flandrin, G. Rilling, and P. Goncalves, "Empirical mode decomposition as a filter bank," *IEEE Signal Processing Letters*, vol. 11, no. 2, pp. 112–114, 2004.

[15] R. Yan and R. X. Gao, "Hilbert–huang transform-based vibration signal analysis for machine health monitoring," *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 6, pp. 2320–2329, 2006.

[16] D. Gabor, "Theory of communication." 2003, pp. 429–439.

[17] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 2nd ed. Prentice-hall Englewood Cliffs, 1999, ch. 8.

[18] A. V. Oppenheim, "Speech spectrograms using the fast fourier transform," *IEEE Spectr.*, vol. 7, no. 8, p. 57–62, aug 1970.

[19] H. Ahmed and A. Nandi, *Time-Frequency Domain Analysis*. John Wiley & Sons, Ltd, 2019, ch. 5, pp. 79–114.

[20] J. O. Smith, *Spectral Audio Signal Processing*. Stanford University, 2011, online book.

[21] J. Gomes and L. Velho, *Windowed Fourier Transform*. Springer, 2015, ch. 4.

[22] Tektronix, "Window functions in spectrum analyzers," acedido em 16 December 2022. [Online]. Available: https://www.tek.com/en/blog/window-functions-spectrum-analyzers

[23] K. Prabhu, *Window Functions and Their Applications in Signal Processing*. CRC Press, 10 2013.

[24] N. Bey, "Multi-resolution fourier analysis: Time-frequency resolution in excess of gabor-heisenberg limit," *Signal, Image and Video Processing*, vol. 8, 05 2014.

[25] E. F. Shair, S. A. Ahmad, A. R. Abdullah, M. H. Marhaban, and S. B. M. Tamrin, "Determining best window size for an improved gabor transform in emg signal analysis." *Telkomnika*, vol. 16, no. 4, pp. 1650 – 1658, 2018.

[26] S. Krishnan, "5 - advanced analysis of biomedical signals," in *Biomedical Signal Analysis for Connected Healthcare*, S. Krishnan, Ed. Academic Press, 2021, pp. 157–222.

[27] M. S. Wibawa, I. M. D. Maysanjaya, N. K. D. P. Novianti, and P. N. Crisnapati, "Abnormal heart rhythm detection based on spectrogram of heart sound using convolutional neural network," in *2018 6th International Conference on Cyber and IT Service Management (CITSM)*, 2018, pp. 1–4.

[28] Z. Chi, Y. Li, and C. Chen, "Deep convolutional neural network combined with concatenated spectrogram for environmental sound classification," in *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, 2019, pp. 251–254.

[29] H. A. Ali, M. M. Elsherbini, and M. I. Ibrahem, "Wavelet transform processor based surface acoustic wave devices," *Energies*, vol. 15, no. 23, 2022.

[30] C. Torrence and G. P. Compo, "A practical guide to wavelet analysis," *Bulletin of the American Meteorological Society*, vol. 79, no. 1, pp. 61–78, 1998.

[31] A. Stepanov, "Polynomial, neural network, and spline wavelet models for continuous wavelet transform of signals," *Sensors*, vol. 21, no. 19, 2021.

[32] M. Farge, "Wavelet transforms and their applications to turbulence," *Annual Review of Fluid Mechanics*, vol. 24, no. 1, pp. 395–458, 1992.

[33] A. Grinsted, J. C. Moore, and S. Jevrejeva, "Application of the cross wavelet transform and wavelet coherence to geophysical time series nonlinear processes in geophysics application of the cross wavelet transform and wavelet coherence to geophysical time series," vol. 11, pp. 561–566, 2004.

[34] P. Gupta, P. K. Chodingala, and H. A. Patil, "Morlet wavelet-based voice liveness detection using convolutional neural network," *2022 30th European Signal Processing Conference (EUSIPCO)*, pp. 100–104, 8 2022.

[35] Z. Cui, Y. Gao, J. Hu, S. Tian, and J. Cheng, "Los/nlos identification for indoor uwb positioning based on morlet wavelet transform and convolutional neural networks." *IEEE Communications Letters*, vol. 25, no. 3, pp. 879 – 882, 2021.

[36] M. M. Almalki, "A proposed fault detection using continues wavelet transform and transfer learning via alexnet," in *2022 IEEE 13th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, 2022, pp. 0124–0131.

[37] M. X. Cohen, *Analyzing Neural Time Series Data: Theory and Practice*. The MIT Press, 01 2014.

[38] P. Konar and P. Chattopadhyay, "Bearing fault detection of induction motor using wavelet and support vector machines (svms)," *Applied Soft Computing*, vol. 11, pp. 4203–4211, 9 2011.

[39] M. Fedotenkova and A. Hutt, "Research report: Comparison of different time-frequency representations." INRIA Nancy, Research Report, Dec. 2014.

[40] S. L. Brunton and J. N. Kutz, *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, 2019.

[41] R. S. Salles and P. F. Ribeiro, "The use of deep learning and 2-d wavelet scalograms for power quality disturbances classification," *Electric Power Systems Research*, vol. 214, p. 108834, 2023.

[42] Y. T. Qassim, T. R. Cutmore, and D. D. Rowlands, "Optimized fpga based continuous wavelet transform," *Computers & Electrical Engineering*, vol. 49, pp. 84–94, 2016.

[43] G. Kaiser and L. H. Hudgins, *A friendly guide to wavelets*. Springer, 1994, vol. 300.

[44] A. Bruns, "Fourier-, hilbert- and wavelet-based signal analysis: are they really different approaches?" *Journal of Neuroscience Methods*, vol. 137, pp. 321–332, 8 2004.