

# Digital signal classification utilizing adaptive information entropy measures and machine learning

---

Vranković Lacković, Ana

Doctoral thesis / Disertacija

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Engineering / Sveučilište u Rijeci, Tehnički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:190:848447>

Rights / Prava: [Attribution 4.0 International](#)/[Imenovanje 4.0 međunarodna](#)

Download date / Datum preuzimanja: **2024-12-25**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Engineering](#)







UNIVERSITY OF RIJEKA  
FACULTY OF ENGINEERING

Ana Vranković Lacković

**Digital Signal Classification Utilizing  
Adaptive Information Entropy  
Measures and Machine Learning**

DOCTORAL DISSERTATION

Rijeka 2024.



UNIVERSITY OF RIJEKA  
FACULTY OF ENGINEERING

Ana Vranković Lacković

**Digital Signal Classification Utilizing  
Adaptive Information Entropy  
Measures and Machine Learning**

DOCTORAL DISSERTATION

Supervisor: Assoc. Prof. Jonatan Lerga, PhD

Rijeka 2024.



SVEUČILIŠTE U RIJECI  
TEHNIČKI FAKULTET

Ana Vranković Lacković

**Klasifikacija digitalnih signala  
korištenjem adaptivnih  
informacijskih entropijskih mjera i  
strojnoga učenja**

DOKTORSKI RAD

Mentor: izv. prof. dr. sc. Jonatan Lerga

Rijeka 2024.





Doctoral dissertation supervisor: Assoc. Prof. D. Sc. Jonatan Lerga (University of Rijeka, Faculty of Engineering, Croatia)

The doctoral dissertation was defended on \_\_\_\_\_ at the University of Rijeka, Faculty of Engineering, Croatia, in front of the following Evaluation Committee:

1. Prof. D. Sc. Ivo Ipšić, University of Rijeka, Faculty of Engineering, Croatia - Committee Chair
2. Assoc. Prof. D. Sc. Goran Mauša, University of Rijeka, Faculty of Engineering, Croatia
3. Assoc. Prof. D. Sc. Nicoletta Saulig, Juraj Dobrila University of Pula, Department of Engineering, Croatia



# ACKNOWLEDGEMENTS

Završetak ovog doktorskog puta bio je iznimno obogaćujuće i izazovno iskustvo, koje ne bi bilo moguće bez podrške, ohrabrenja i vodstva mnogih pojedinaca. Prije svega, izražavam svoju najdublju zahvalnost svom mentoru, izv. prof. dr. sc. Jonatanu Lergi, na njegovu neprocjenjivu mentorstvu i strpljenju. Njegovo stručno znanje i uvidi bili su ključni u oblikovanju mog istraživanja i akademskog rasta. Zahvaljujem i profesoru Dombiu sa Sveučilišta u Szegedu na potpori i pomoći u upoznavanju sa novom tematikom tijekom mog doktorskog studija. Duboko sam zahvalna Zavodu za računarstvu, čiji su akademska zajednica i resursi bili temelj podrške tijekom mojih studija. Posebne zahvale idu mojim prijateljima i kolegama Diegu i Franku, čije druženje, ohrabrenje i zajednički trenuci borbe učinili su ovaj put ne samo podnošljivim već i ugodnim.

Iskreno zahvaljujem udruzi Kitten Safe House, udruzi koja mi je pružila emocionalnu podršku potrebnu za prevladavanje najzahtjevnijih faza mog istraživanja. Njihov rad me svakodnevno inspirira, i zahvalna sam za njihov doprinos mojem mentalnom stanju.

Mojim voljenim mačkama, Saši i Elizi, hvala vam za vaše stalno društvo i bezbroj trenutaka utjehe i distrakcije tijekom mojih studija. Vaša prisutnost bila je izvor radosti i opuštanja.

Zahvaljujem svojoj obitelji za njihovu podršku i poticaj koji su bili ključni za moje akademske uspjehe. Vaše povjerenje u mene motiviralo me da nastavim ići naprijed.

Najvažnije, na kraju, zahvaljujem svome suprugu, Dariu, za njegovu postojanu podršku i strpljenje tijekom ovog procesa. Njegovo razumjevanje i podrška bili su izuzetno važni, i zahvalna sam na njegovoj ulozi u mom životu.



# ABSTRACT

This thesis proposes a new approach for preprocessing method for signal classification based on blind source separation of signal components from noisy data in the time-frequency domain. The method is based on the local entropy, which is calculated within adaptive, data-driven 2D regions. One of the advantages of the proposed technique is that it requires no prior knowledge of the signal, its components or the noise, but the processing is performed on the noisy signal mixtures. Furthermore, the method is robust to the selection of time-frequency distributions and entropy measures. The proposed approach is demonstrated using several examples. The first example is synthetic signals with different signal-to-noise ratios. This example is selected to demonstrate the usage of the proposed method and the analysis of the results. The second example is a real example of speech signals. This example was selected to demonstrate the ability to use the proposed method on real-life signals. In addition, a novel entropy measure was introduced and adapted to improve the detection within the method. Finally, the method was presented as a preprocessing tool for signal detection of seismology signals, and the results of time-frequency representations and entropy maps were compared.

The results obtained by evaluating different machine learning models for classification show excellent classification performance of the proposed approach, with classification accuracy, area under the receiver operating characteristic curve, F1-score and Matthews correlation coefficient up to 97.55%, 97.72%, 98.07% and 94.88%, respectively. Moreover, the proposed approach significantly outperforms the model trained on the time-frequency distributions in terms of all considered metrics, with statistical significance confirmed by Cochran's Q and McNemar tests. The obtained results indicate that the proposed technique improves the classification of seismological signals and has the potential to be extended to other practical applications of signal classification in other areas of research.

**Keywords: Entropy, Time-frequency distributions, Adaptive thresholding,  
Signal classification**

# PROŠIRENI SAŽETAK

Analiziranje nestacionarnih signala predstavlja kompleksan zadatak u različitim istraživačkim područjima zbog njihovog promjenjivog frekvencijskog spektra kroz vrijeme. Takva analiza zahtijeva korištenje naprednih alata kako bi se istovremeno prikazali u vremenskoj i frekvencijskoj domeni, što nadilazi standardne tehnike analize signala u tim pojedinačnim domenima. Osim toga, nestacionarni signali u stvarnim situacijama često su višekomponentni i dodatno su izloženi šumovima iz vanjske okoline.

U ovom radu razvijena je i testirana metoda za odvajanje komponenti signala od okolnog šuma bez da su potrebne dodatne informacije o prirodi signala i šuma. Metoda je bazirana na adaptivnim izračunima lokalnih entropija unutar vremensko-frekvencijske distribucije signala.

Metoda je testirana na tri različita sintetička signala. Svakom signalu dodane su tri različite razine šuma čime se dobilo devet slučajeva. Za svaki od njih dobiveno je pet vremensko-frekvencijskih distribucija što je razultiralo s 45 slučajeva. Za svaki slučaj metoda je provedena koristeći tri klasične entropijske mjere i mjeru neizravne entropije koja je bila adaptirana upravo za rad s predloženom metodom. Konačno, test je imao 180 slučajeva. Rezultati su pokazali kako predložena metoda može razdvojiti signal i šum s točnošću od 89.8% do 99.5% za slučajeve kada je u signalu prisutno više šuma, odnosno u odnosu signal-šum šum prevladava, te od 89.7% do 99.9% kada je šum slabiji, odnosno signal nadjačava šum. F1 mjera se kreće između 61.2% i 90.8% za slučajeve s jačim šumom te između 64.8% i 97.3% kod slučajeva s više šuma. Ukupno gledano najbolje rezultate ostvaruje predložena adaptacija entropijske mjere s najvišom točnošću između različitih vremensko-frekvencijskih distribucija između 94.5% i 99.9% te F1 mjerom između 78.2% i 97.3%. Nakon toga, metoda je testirana na primjeru audio signala iz javno dostupne baze s različitim izvorima šuma i različitim odnosima signala i šuma. Dobiveni rezultati imaju



točnost između 97.78% i 99.03%, preciznost između 98.94% i 99.65% te odziv između 98.96% i 99.65%.

Završni cilj je bio pokazati kako se predložena metoda može koristiti kao metoda za predobradu signala sa svrhom klasifikacije signala. Testni slučaj je bila klasifikacija potresa i šuma kod seizmografskih signala. Rezultati su pokazali kako korištenje entropijske mape dobivene iz predložene metode mogu poboljšati rezultate klasifikacije u odnosu na dosad korišteni vremensko-frekvencijski prikaz signala. Postignuta su poboljšanja do 2.64% za točnost, do 2.34% poboljšanje kod površine ispod krivulje odnosa specifičnosti i osjetljivosti, F1 mjera je poboljšana do 2.16% i Matthew-ov koeficijent korelacije je imao povećanje do 5.38%.

Analiza dobivenih rezultata pokazuje da je predloženi pristup primjene adaptivnog prozora za izračun lokalne entropije uspješan u odvajanju signala od šuma. Također ukazuju i na to da je entropijska mapa dobivena predloženom metodom korisna kao korak predobrade signala za unaprijeđenje kvalitete klasifikacije seizmoloških signala. Predložena metoda za predobradu signala može se proširiti i za korištenje na drugim područjima znanstvenog istraživanja različitih vrsta signala.

**Ključne riječi:** Entropija, Vremensko-frekvencijske distribucije, Adaptivni prag, Klasifikacija signala

# CONTENTS

<b>Acknowledgements</b>	<b>I</b>
<b>Abstract</b>	<b>III</b>
<b>Prošireni sažetak</b>	<b>V</b>
<b>1. Introduction</b>	<b>1</b>
1.1. Motivation . . . . .	1
1.2. Research Objectives and Scientific Contributions . . . . .	3
1.3. Research Methodology . . . . .	4
1.4. Organization of the Thesis . . . . .	5
<b>2. TIME-FREQUENCY DIGITAL SIGNAL ANALYSIS</b>	<b>7</b>
2.1. Non-Stationary Signals . . . . .	7
2.2. Time-Frequency Distributions . . . . .	10
<b>3. ENTROPY MEASURES</b>	<b>17</b>
3.1. Classical Entropy Measures . . . . .	18
3.1.1. Shannon entropy . . . . .	18
3.1.2. Rényi Entropy . . . . .	20
3.1.3. Tsallis Entropy . . . . .	21
3.2. Fuzzy Entropy Measures . . . . .	23
<b>4. CONVOLUTIONAL NEURAL NETWORKS</b>	<b>27</b>
4.1. Classification . . . . .	27

4.2. Convolutional Neural Networks . . . . .	29
<b>5. OVERVIEW OF RELATED WORK</b>	<b>37</b>
5.1. Methods in signal-noise separation . . . . .	38
5.1.1. Classical representation . . . . .	38
5.1.2. Time-frequency representation . . . . .	39
5.2. Methods for seismology signal classification . . . . .	41
5.2.1. Seismology signal as a time-series . . . . .	41
5.2.2. Seismology signal as an image . . . . .	43
<b>6. PROPOSED 2DLEM METHOD FOR SIGNAL SEPARATION</b>	<b>45</b>
6.1. Methodology . . . . .	46
6.1.1. First step: signal decomposition into TFD . . . . .	46
6.1.2. Second step: entropy calculation . . . . .	46
6.1.3. Third step: RICI . . . . .	48
6.1.4. Fourth step: entropy maps to masks utilizing RICI . . . . .	53
6.1.5. Performance metrics . . . . .	56
6.2. Novel entropy for the 2DLEM method . . . . .	58
6.2.1. Dombi operators . . . . .	60
6.2.2. Novel Dombi entropy . . . . .	61
<b>7. RESULTS AND DISSCUSSION</b>	<b>63</b>
7.1. 2DLEM applied to synthetic data . . . . .	64
7.1.1. Dataset - synthetic signals . . . . .	64
7.1.2. 2DLEM with Shannon entropy . . . . .	73
7.1.3. 2DLEM with Rényi entropy . . . . .	79
7.1.4. 2DLEM with Tsallis entropy . . . . .	82
7.1.5. 2DLEM with Dombi entropy . . . . .	88
7.1.6. Comparison of 2DLEM with different entropy measures on synthetic data . . . . .	93
7.2. 2DLEM applied to real-world data . . . . .	96
7.2.1. 2DLEM for noisy speech signals . . . . .	96
7.2.2. 2DLEM with CNN for seismic signal classification . . . . .	101

<b>8. CONCLUSION</b>	<b>109</b>
<b>Bibliography</b>	<b>111</b>
<b>List of Symbols</b>	<b>133</b>
<b>List of Figures</b>	<b>135</b>
<b>List of Tables</b>	<b>141</b>
<b>Appendices</b>	<b>143</b>
<b>CURRICULUM VITAE</b>	<b>157</b>
<b>LIST OF PUBLICATIONS</b>	<b>159</b>



# 1. Chapter

## INTRODUCTION

### Contents

---

<b>1.1. Motivation</b> . . . . .	<b>1</b>
<b>1.2. Research Objectives and Scientific Contributions</b> . . . . .	<b>3</b>
<b>1.3. Research Methodology</b> . . . . .	<b>4</b>
<b>1.4. Organization of the Thesis</b> . . . . .	<b>5</b>

---

This chapter gives an introduction to the research detailed in this thesis. It begins by giving scientific motivation. It articulates the research hypotheses, main objectives, and the significant contributions made within this thesis. Additionally, it outlines the methodology employed in the research and provides a concise overview of the thesis structure.

### 1.1. Motivation

Time-domain signal representation, which plots signal amplitude as a function of time, is commonly used in signal analysis because it depicts the mode in which the majority of signal measurements are acquired. The Fourier transform-based frequency-domain representation of signals, which gives information on the signal's frequency content but not its temporal localization, is another often-used signal representation. To evaluate stationary signals typical signal analysis techniques such as the time-domain and frequency-domain representations are appropriate. These one-dimensional representations in one domain, however, are insufficient for the analysis of nonstationary signals, or signals whose fre-

quency content varies with time. Furthermore, nonstationary signals in real-world applications are frequently multi-component and noise-corrupted, which makes analysis much more difficult. Two-dimensional time-frequency distributions (TFDs) result from the need for sophisticated and reliable tools for simultaneous signal representation in the joint time-frequency domain in non-stationary signal analysis [1]. Insight into the signal energy distribution as a function of time and frequency is provided by TFDs, which are more informative signal representations. The separation of noise and signal is one area of research that has been dominated by time and frequency analysis. Recently, efforts in this area have shifted to the time-frequency domain, motivated by [2]. Methods investigating the use of entropy measures in the separation of the useful signal component have also been investigated in many studies [3, 4, 5, 6]. In the work by Saulig et al. [7] proposes an automatic adaptive method for the detection and separation of useful information contained in the time-frequency domain. The main idea is based on the K-means clustering method, which performs a partitioning of the data. Instead of a hard threshold, the authors use a blind separation of the useful information from the background noise using the local Rényi entropy. The advantage of this approach is that no prior knowledge of the signal is required. The results show that this method works as an optimal automatic hard threshold selector. The use of probabilistic entropies, such as Shannon's and Rényi's, requires several assumptions about the system in order for their calculation to be possible. Probabilistic entropies are not suitable for quantifying the resulting uncertainty in scenarios where ambiguity, uncertainty or missing information play a role. Since the concept of probability was not sufficient for the unambiguous modelling of an uncertainty system, Zadeh proposed a new theory, fuzzy set theory, as a generalization of classical set theory [8]. The fuzzy entropy measure is based on the degree of membership, which is different from probability and therefore has a different form than probabilistic entropy. De Luca and Termini introduced the fuzzy entropy function as a generalization of Shannon entropy and introduced axioms that the fuzzy entropy measure must satisfy [9]. After De Luca and Termini, many authors have proposed generalizations of fuzzy entropy. The application of algorithms based on fuzzy logic is widely used in signal analysis [10, 11, 12].

Motivated by the discovery of the usefulness of entropy measures in signal analysis, this thesis presents a new method for blind signal-to-noise separation based on entropy measures and the adaptation of a fuzzy entropy measure for signal-to-noise separation.

The method was applied to the classification of seismological signals, where the proposed method showed promising results as a preprocessing method.

## **1.2. Research Objectives and Scientific Contributions**

The main objective of the research was to develop a method that effectively extracts useful content from the time-frequency domain of a signal based on entropy analysis. Several objectives were achieved in the course of this work. The first step was to develop a specialized method for calculating 2D entropy. This method was further improved by integrating adaptive windows that can dynamically adapt to changing conditions. In addition, the research focused on extracting masks from the derived entropy maps that allow further analysis and insights into the signal content. Another important aspect was the detailed comparison of the performance of different entropy measures. Furthermore, a new and innovative entropy measure was introduced and adapted to improve the separation of the signal and the noise. Finally, the proposed approach was employed as a preprocessing method for signal classification tasks. Based on these objectives, the research hypotheses are as follows.

The first hypothesis states that the use of adaptive 2D entropy within the method significantly improves the process of extracting useful content from the signal. It is hypothesized that the adaptive nature of the entropy calculations will improve the accuracy and F1-score of identifying relevant signal components.

The second hypothesis is that the entropy map will provide additional insights into the signal properties and thus enable more accurate classification results.

The contribution of this work is a novel method specifically developed for the separation of signal components from noisy data. In addition, a fuzzy entropy measure is adapted and applied in the study to fit the proposed method. Furthermore, it is shown that the developed method can be used as a preprocessing step in classification tasks, enabling effective extraction of useful content from the signals.

The research methodology used to develop and validate the proposed method is described below.



### **1.3. Research Methodology**

The study consisted of several phases of research and development. In the first phase, an overview of the use of entropy measures and time-frequency distributions in signal processing was given. The different types of representations used in the decomposition of signals into the time-frequency domain were explored such as spectrogram, Wigner-Vile, Choi-Williams, etc. In addition, a comprehensive understanding of the methods of time-frequency distribution, both in theory and in practical implementation, was gained. This included categorizing the advantages and disadvantages of the different signal representations. In addition, an overview of previous research on the application of entropy measures in signal analysis was given, with a particular focus on the application to signals in the time-frequency domain. The available tools for signal analysis in the time-frequency domain and the functions for entropy calculations were selected. In particular, those tools were packages in Matlab that allowed the calculation of various time-frequency distributions and probabilistic entropies.

The second phase focused on the development of a novel method suitable for the analysis of time-frequency signals using local entropy measures. The proposal included a method for calculating the local 2D entropy with an adaptive window for the time-frequency representation of the signal. Preliminary research results were obtained by applying the proposed approach in signal analysis. The method was applied to synthetic noisy, non-stationary and multi-component signals and the results were compared with existing methods using the time-frequency representation of signals and based on entropy measures. In addition, the new fuzzy entropy measure was configured to ensure its compatibility with the proposed method. By incorporating fuzzy logic principles into the entropy calculation, an attempt was made to capture the uncertainty and fuzziness inherent in the signal characteristics. This adaptation was intended to enable the method to deal with complex and ambiguous signals and to further improve the quality of the extracted components. The aim was to extract valuable information from time-frequency representations of noisy, non-stationary and multi-component signals. Furthermore, the effectiveness of different entropy measures and time-frequency distributions within the proposed method was analyzed. The final phase involved the application of the developed methods to real signals. A suitable problem to which the proposed method could be

applied was carefully selected on the basis of the quantitative results obtained so far. To facilitate this process, available tools for signal classification and image classification in the Python Torch framework were investigated. In addition, the performance of the proposed method was tested as a data preprocessing technique for an existing signal classification problem. Through the implementation of this comprehensive research plan, a contribution was made to the further development of noise separation in the time-frequency signal domain and its application in practical scenarios.

## 1.4. Organization of the Thesis

This thesis is divided into eight chapters that provide a structured description of the research conducted. The first chapter discusses the academic motivation for the research offered, outlines the primary aims and contributions of the thesis and briefly describes the research methodology. In addition, this chapter provides an overview of the framework of the thesis.

Chapter 2 introduces the basic concepts of non-stationary signal analysis, focusing on the definition and mathematical properties of different time-frequency distributions. Examples are also given to illustrate the concept.

A description of classical entropy measures, including the Shannon, Rényi and Tsallis entropies, and fuzzy entropy measures, emphasizing the differences between them, is given in Chapter 3.

Chapter 4 explores fundamental aspects related to classification problems, focusing on the use of convolutional neural networks. The chapter describes the three different CNN architectures used in this thesis.

Chapter 5 provides an overview of methods for separating signal from noise, including the time representation and the time-frequency representation, as well as methods for classifying seismological signals represented as both time series and images.

One of the main parts of the thesis is in Chapter 6, which presents a novel technique for distinguishing signals from noise and an entropy measure modified specifically for this purpose. It analyzes the method and discusses the operators, steps and algorithms involved.

The results obtained with the proposed method are presented in Chapter 7, accompa-

nied by a detailed analysis and discussion of several examples.

Finally, Chapter 8. outlines the conclusions and possibilities for future work.

## 2. Chapter

# TIME-FREQUENCY DIGITAL SIGNAL ANALYSIS

### Contents

---

<b>2.1. Non-Stationary Signals . . . . .</b>	<b>7</b>
<b>2.2. Time-Frequency Distributions . . . . .</b>	<b>10</b>

---

In the field of signal analysis, understanding the connection between time and frequency components is essential for unravelling the dynamic behaviour of signals. This chapter describes the main concept of non-stationary signal analysis with an explanation of the need for joint time-frequency representation. Time-frequency signal distributions with a focus on quadratic distributions are introduced. The concepts of time-frequency signal analysis and the need for it are also demonstrated in the examples of the multi-component non-stationary signal.

### 2.1. Non-Stationary Signals

Not all signals describing real-world phenomena exhibit predictable, consistent behaviour. Some signals exhibit trends, fluctuations, or variations that change over time. These “non-stationary” signals present special difficulties and opportunities in the case of their analysis. A signal is basically a picture of information that changes over time or location. A stationary signal is one whose statistical characteristics, such as mean,

variance, and autocorrelation, do not vary over time and that generally remain constant. On the other hand, the statistical characteristics of a non-stationary signal change over time. Non-stationarity can be caused by a variety of factors, including external forces, underlying dynamics, and changing environmental conditions. One such external influence is noise, which is found in the majority of signals in real-world applications.

The time domain is the most natural way to represent a signal. The signal  $s(t)$  in this case reflects the signal amplitude as a function of time.

The Fourier transform (FT) is a bijective function that allows for transformation between the time and frequency domains and vice versa. Transformation to frequency domain is

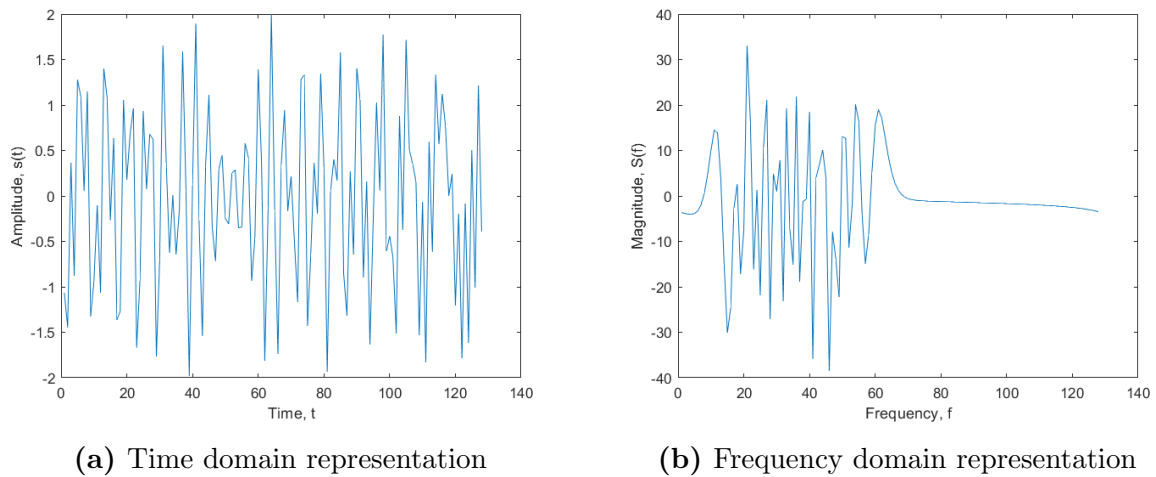
$$S(f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi ft} dt \tag{2.1}$$

Time signal can be recovered from the FT as

$$s(t) = \int_{-\infty}^{\infty} S(f)e^{j2\pi ft} df \tag{2.2}$$

The time domain and frequency domain are mutually exclusive; the time domain does not contain information about the frequency domain. The frequency domain shows the overall strength of every frequency in the signal; it does not show information about the time localization of spectral components. These limitations associated with one-dimensional signal representations are showcased in the context of a synthetic non-stationary signal containing two linear components. The signal manifestations are illustrated in Figure 2.1, where Figure 2.1(a) exhibits the signal's presentation in the time domain, and Figure 2.1(b) portrays the frequency domain.

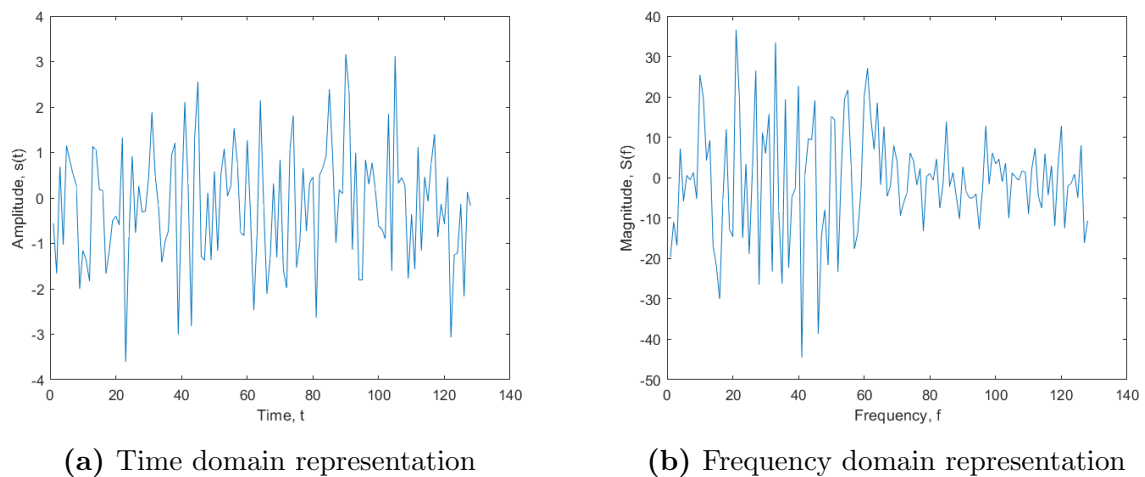
Furthermore, real-world non-stationary signals often encountered in practical applications frequently contain noise, complicating their analysis through conventional methodologies. The impact of this noise on one-dimensional signal depictions is exemplified in Figure 2.2. This figure offers both the time-domain and frequency-domain representations of the previously examined signal. However, in this instance, the signal is additionally affected by additive white Gaussian noise, introducing a signal-to-noise ratio (SNR) of 5 dB. As evident from Figure 2.2, the noise's presence amplifies the complexity of analyzing the aforementioned non-stationary signal. Consequently, relying solely on separate time-domain and frequency-domain depictions becomes impractical for deciphering noisy



**Figure 2.1:** Example of a two-component non-stationary signal

signals characterized by fluctuating frequency content.

This restriction of the one-dimensional approach can be removed by describing the time-frequency structure of a signal by a surface over the time-frequency plane.



**Figure 2.2:** Example of a two-component non-stationary signal with added noise(SNR=5 dB)

## 2.2. Time-Frequency Distributions

Traditional methods of spectral analysis, such as the FT, work best with stationary signals but have difficulty with signals that change over time. Time-frequency distributions (TFDs) provide a thorough illustration of how the frequency components of a signal change over time. TFDs provide a more thorough and dynamic perspective than the static frequency information provided by the FT and allow the evaluation of signals with varying frequency content, such as non-stationary signals. The basic idea of the method is to create a combined time and frequency function that can simultaneously characterize the energy density of a signal in time and frequency [1, 13, 14]. The time-frequency distributions theoretically define the properties of a signal in a two-dimensional time-frequency plane.

All linear TFD's satisfy the superposition principle: if  $x(t)$  is a linear combination of signal components, then the TFD of the  $x(t)$  is the same linear combination of the signal components TFDs

$$x(t) = c_1x_1(t) + c_2x_2(t) \rightarrow T_x(t, f) = c_1T_{x_1}(t, f) + c_2T_{x_2}(t, f) \quad (2.3)$$

Short-time Fourier transform is the most simple and commonly used linear TFD. This thesis will concentrate on the quadratic TFDs where non-linearity is present.

The first attempt at observing signals in both domains simultaneously was the short-time Fourier transform(STFT) proposed by Gabor in 1964 [15]. The STFT at time  $t$  is defined as the FT of the signal  $x(\tau)$  multiplied by a shifted window  $\omega(t - \tau)$  centred around  $t$ .

$$STFT = \int_{-\infty}^{\infty} z(\tau)\omega(t - \tau)e^{-j2\pi f\tau} d\tau \quad (2.4)$$

Calculation of the squared magnitude of the STFT of the signal is the spectrogram representation [1, 13, 16].

$$S_z(t, f) = |STFT_z(t, f)|^2 = \left| \int_{-\infty}^{\infty} z(\tau)\omega(t - \tau)e^{-j2\pi f\tau} d\tau \right|^2 \quad (2.5)$$

The spectrogram introduces a nonlinearity in the time-frequency representation. Fur-

thermore, the spectrogram of the sum of two signals does not equal the sum of the spectrograms of the two signals. This lack of additivity is one of the reasons why the spectrogram is considered a bilinear TFD rather than a linear TFD, even though it is derived from the STFT. In addition, the spectrogram plot depends on the window function  $\omega(t)$ . A smaller window gives better time resolution, while a wider window gives better frequency resolution. In other words, the observation window  $\omega(\tau)$  allows the spectrum to be localized in time but also blurs the spectrum in frequency. The lack of additivity is one of the limitations of the spectrogram, especially for complex signals that contain multiple components, since the STFT involves a windowed FT that results in cross terms that lead to interactions between the frequency components of the two signals[1].

The spectrogram often causes great difficulty when used to analyze rapidly changing signals. If the analysis window is short enough to capture rapid changes in the signal, it becomes impossible to resolve frequency components of the signal that are close in frequency during the duration of the analysis window[17]. To solve this problem, the Wigner-Vile distribution (WVD), a quadratic TFD, was introduced in [18] as an alternative.

The WVD is obtained by applying the FT to the instantaneous autocorrelation function (IAF) of the signal  $s(t)$ . The IAF  $R_s(t, \tau)$  of the signal  $s(t)$  is defined as

$$R_s(t, \tau) = s\left(t + \frac{\tau}{2}\right)s^*\left(t - \frac{\tau}{2}\right) \quad (2.6)$$

where  $s^*(t)$  denotes the complex conjugate of the signal  $s(t)$  and  $\tau$  is the continuous lag. Therefore, the WVD is obtained as

$$WVD_s(t, f) = STFT(R_s(t, \tau)) \quad (2.7)$$

On the one hand, the WVD provides a high-resolution representation in time and frequency for a nonstationary signal. On the other hand, its energy distribution is nonnegative and often leads to strong cross-terms between components in different  $(t, f)$  regions [19]. These cross terms can make visual interpretation of the TFD difficult. Therefore, to obtain interpretable time-frequency representations, the unwanted cross-terms should be attenuated. Attenuating the cross terms is possible by smoothing the WVD using filter functions, also known as kernels [20].



The pseudo-Wigner-Ville distribution (PWVD) is derived by windowing the IAF through the time window  $h(t)$ . The PWVD is smoothed in the frequency domain by multiplying it by the window function in the time domain [21], which results in attenuation of the cross terms oscillating in the frequency direction and improvement of the frequency resolution of the signal auto-terms. The PWVD is calculated as:

$$PWVD_s(t, f) = \int_{-\infty}^{\infty} h(\tau) s(t + \frac{\tau}{2}) s^*(t - \frac{\tau}{2}) e^{-j2\pi f\tau} d\tau \quad (2.8)$$

The PWVD has no effect on cross-terms oscillating in the time direction but suppresses cross-terms oscillating in the frequency direction. The smoothed pseudo-Wigner-Ville distribution (SPWVD) addresses the problem of residual cross-terms. The PWVD is smoothed in the time direction by the additional application of the time-smoothing window  $g(t)$  in this TFD. Therefore, the smoothing of the PWVD in the time and frequency domains can be changed independently by adjusting the lengths of the windows  $h(t)$  and  $g(t)$  in the SPWVD [22]. However, as is common with interference-reduced distributions, there is still a trade-off between the level of the cross-terms in the representation and the resulting time-frequency resolution [22]. The SPWVD is defined as

$$SPWVD_s(t, f) = \int_{-\infty}^{\infty} h(\tau) \int_{-\infty}^{\infty} g(u - t) s(u + \frac{\tau}{2}) s^*(u - \frac{\tau}{2}) du e^{-j2\pi f\tau} d\tau \quad (2.9)$$

Different distributions in the quadratic class of TFDs can be obtained by selecting different kernel functions.

Choi and Williams introduced one of the earliest distributions [23] that was different from the WVD, which they called the Exponential Distribution. This new distribution overcomes several drawbacks of the spectrogram and WVD, providing high resolution with suppressed interferences.

The exponential Distribution kernel is defined as  $g(u, \tau) = e^{-\frac{u^2\tau^2}{\sigma}}$ . Choi-Williams distribution (CWD) is obtained as

$$CWD_s(t, f) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \sqrt{\frac{\sigma}{\pi}} \frac{1}{2|\tau|} e^{-\frac{\sigma u^2}{16\tau^2}} s(t + u + \frac{\tau}{2}) s^*(t + u - \frac{\tau}{2}) du e^{-j2\pi f\tau} d\tau. \quad (2.10)$$

While the emphasis on the development of distribution such as Choi-William distribution was to meet marginal conditions and other properties, in order to introduce finite

time support and reduce cross terms, Zhao-Atlas-Mark distribution(ZAMD) [24, 25], was designed with a cone-shaped kernel that performs the directional band-pass filtering of the WVD. The ZAMD is defined as

$$ZAMD_s(t, f) = \int_{-\infty}^{\infty} h(\tau) \int_{t-\frac{|\tau|}{2}}^{t+\frac{|\tau|}{2}} s(u + \frac{\tau}{2})s^*(u - \frac{\tau}{2})dud e^{-j2\pi f\tau} d\tau \quad (2.11)$$

The reduced interference distribution (RID) is a quadratic TFD in which the cross-terms are constricted relative to the auto-terms. It can use different functions. One of the functions is the Bessel function of the first kind [26]. The distribution is defined as:

$$RIDB_s(t, f) = \int_{-\infty}^{+\infty} h(\tau)R_s(t, \tau)e^{-j2\pi f\tau} d\tau \quad (2.12)$$

where  $h$  is the frequency smoothing window and  $R_s$  represents the kernel

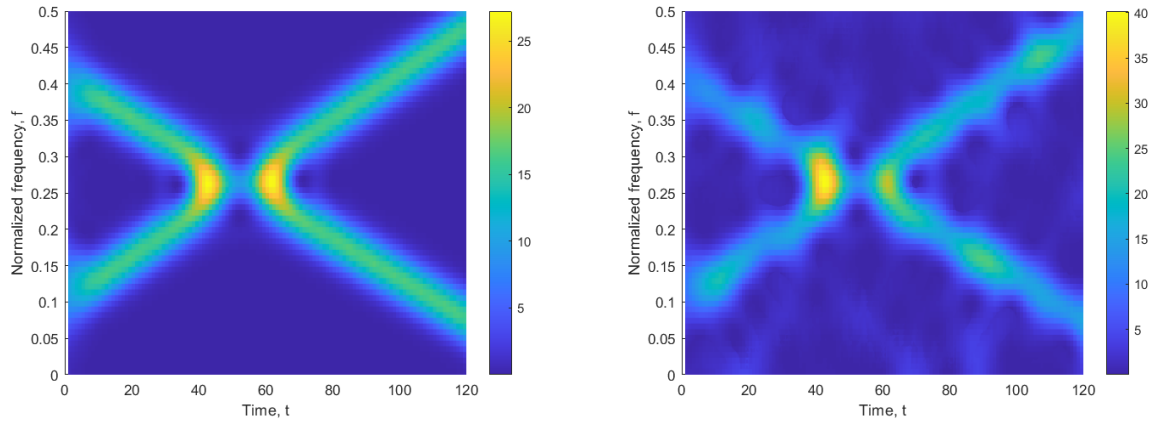
$$R_s(t, \tau) = \int_{t-|\tau|}^{t+|\tau|} \frac{2g(v)}{\pi|\tau|} \sqrt{1 - (\frac{v-t}{\tau})^2} s(v + \frac{\tau}{2})s^*(v - \frac{\tau}{2})dv \quad (2.13)$$

$g$  is the time smoothing window and  $s^*$  denotes the complex conjugate of  $s$ .

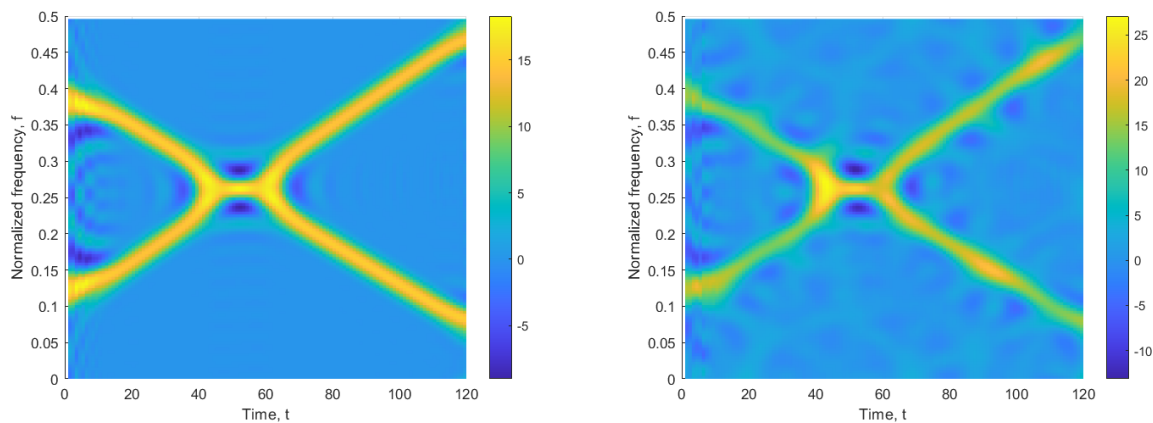
The quadratic TFDs discussed earlier are used in this thesis. Figures 2.3 to 2.7 present the visual representations of five distinct quadratic TFDs: spectrogram, SPWVD, CWD, ZAMD, and RIDB. These figures showcase the characteristics of these TFDs using an illustrative example involving a two-component non-stationary signal.

Both noiseless and noisy TFDs (with an additional noise at a signal-to-noise ratio (SNR) of 5 dB) are shown in the figures. It's worth noting that the presence of noise leads to an overlap between the inherent components of the signal and the added noise. This phenomenon is visually evident in the figures where the cross-terms between the signal and noise components can be observed, highlighting the impact of noise on the TFD plots.

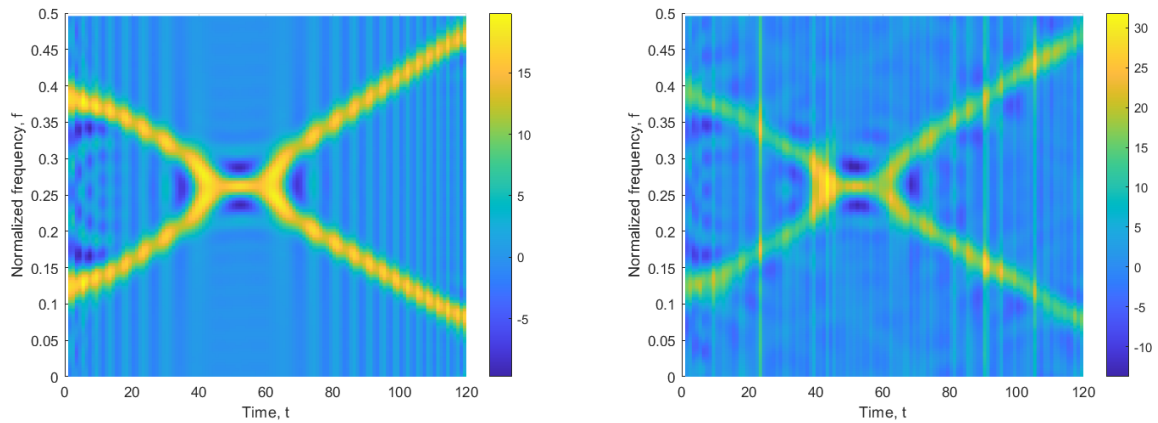
The examples clearly illustrate that the time-frequency approach enhances the distinction between noisy and clean signals compared to classical methods. In this thesis, the digital signal analysis will concentrate on signal representation in the time-frequency domain, and also on the entropy measures discussed in the subsequent chapter.



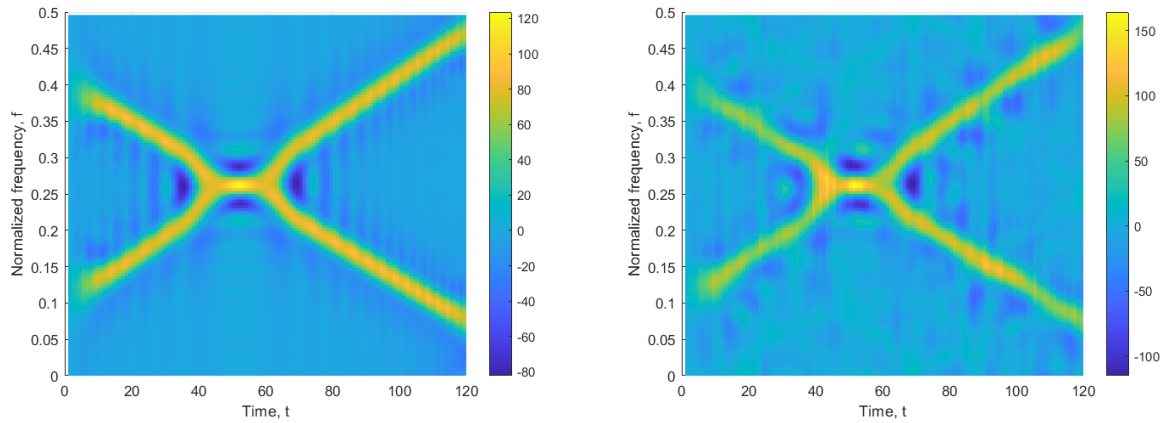
**Figure 2.3:** Spectrogram of the example of a two-component non-stationary signal  $s(t)$  with a parabolic and a linear frequency modulated component with and without added noise



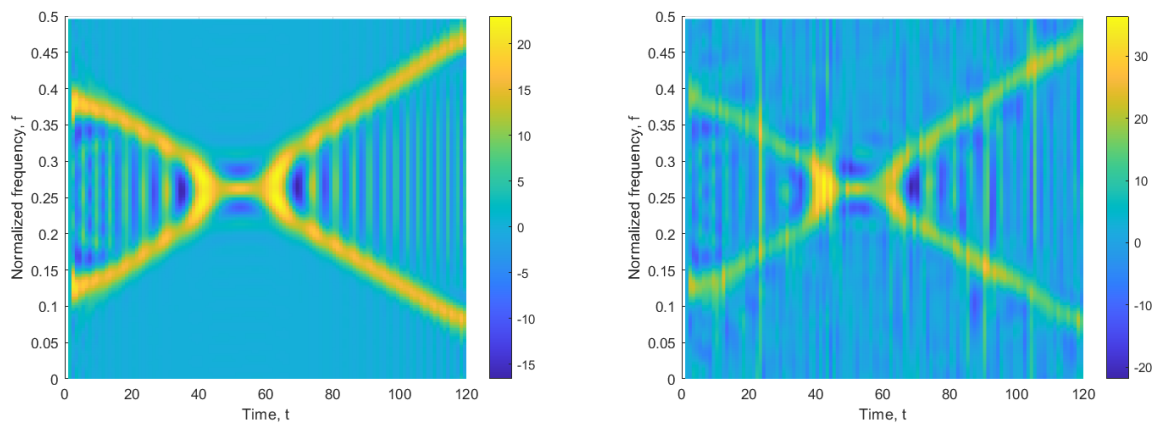
**Figure 2.4:** SPWVD of the example of a two-component non-stationary signal  $s(t)$  with a parabolic and a linear frequency modulated component with and without added noise



**Figure 2.5:** CWD of the example of a two-component non-stationary signal  $s(t)$  with a parabolic and a linear frequency modulated component with and without added noise



**Figure 2.6:** ZAMD of the example of a two-component non-stationary signal  $s(t)$  with a parabolic and a linear frequency modulated component with and without added noise



**Figure 2.7:** RIDB of the example of a two-component non-stationary signal  $s(t)$  with a parabolic and a linear frequency modulated component with and without added noise



## 3. Chapter

# ENTROPY MEASURES

### Contents

---

<b>3.1. Classical Entropy Measures</b> . . . . .	<b>18</b>
3.1.1. Shannon entropy . . . . .	18
3.1.2. Rényi Entropy . . . . .	20
3.1.3. Tsallis Entropy . . . . .	21
<b>3.2. Fuzzy Entropy Measures</b> . . . . .	<b>23</b>

---

Entropy is a term used in thermodynamics, statistical mechanics, information theory, and other fields to describe how complex or unpredictable a system is. However, the meanings vary depending on the field of study. Shannon [27] introduced the term “entropy” in order to describe communication networks, which is also known as the Shannon entropy, and it significantly influenced the foundation of information theory.

Let’s first consider a finite set  $X$  of mutually exclusive events. In probability theory, this uncertainty about the event occurring is expressed by a function  $P : X \rightarrow [0, 1]$ , for which we can define a probability distribution function

$$\sum_{x \in X} p(x) = 1 \tag{3.1}$$

For each  $x \in X$ , the  $p(x)$  represents the degree of support that  $x$  occurs.  $X$  variable is called a random variable as its states can be associated with the probability. The

information associated with the outcome  $X = x$  is denoted by:

$$I(x) = \log_b \frac{1}{p(x)} \tag{3.2}$$

The choice of the value  $b$  determines the unit in which the information is measured. For  $b = 2$ , the resulting unit of measurement is bit. If  $b = e$ , then the resulting unit of measure is nat.

The amount of information acquired due to the observation of event  $x$  follows from Shannon’s solution of the fundamental properties of information[13]

- **I(x) is monotonically decreasing in p**, an increase in the probability of an event decreases the information from an observed event, and vice versa.
- **I(1) = 0**, events that always occur do not communicate information.
- **I(p<sub>1</sub>, p<sub>2</sub>) = I(p<sub>1</sub>) + I(p<sub>2</sub>)**, the information learned from independent events is the sum of the information learned from each event.

### 3.1. Classical Entropy Measures

#### 3.1.1. Shannon entropy

As already mentioned, Shannon in [27] introduced the concept of information from a discrete source without memory as a function that quantifies the uncertainty of a random variable. The average of the information is known as the Shannon entropy measure. Shannon entropy measure is formulated as

$$H_S(p_1, p_2, \dots, p_n) = -c \sum_{i=1}^n p_i \log_b(p_i), \tag{3.3}$$

where  $b$  and  $c$  are positive constants, and  $b \neq 1$ . Each choice of the values  $b$  and  $c$  determines the unit in which the uncertainty is measured.

By definition, for  $p_i = 0$ ,  $p_i \log_b(p_i) = 0$ .

Let us consider a simple case of  $n = 2$ :

$$H(\mathbf{p}) = H_s(p, 1 - p) = -(p \log_2(p) + (1 - p) \log_2(1 - p)). \tag{3.4}$$

For  $p = 1$  or  $p = 0$ ,  $H(1) = H(0) = 0$ .

The formulation of Shannon entropy is based on a set of axioms that provide a solid foundation for its properties and calculations. These axioms are as follows:

### 1. Non-Negativity

The Shannon entropy of a random variable  $p$  is always non-negative:  $H_s(p) \geq 0$ , i.e.

If  $p = 1$ , then  $H_s(p) = 0$ .

### 2. Expansibility

When an event with probability zero is added to the distribution, the entropy should not change

$$H_s(p_1, p_2, \dots, p_n) = H_s(p_1, p_2, \dots, p_n, 0) \quad (3.5)$$

### 3. Symmetry

The entropy is a symmetric function of its arguments, i.e., the entropy  $H_s(p_1, p_2, \dots, p_n)$  does not depend on the order of  $p_i$ :

$$H_s(p_1, p_2, \dots, p_n) = H_s(p_{\sigma_1}, p_{\sigma_2}, \dots, p_{\sigma_n}), \quad (3.6)$$

where  $\sigma_1, \dots, \sigma_n$  is a permutation of  $(1, 2, \dots, n)$ .

### 4. Maximum

Maximum entropy is reached when all probabilities are equal.

$$H_s(p_1, p_2, \dots, p_n) \leq H_s\left(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n}\right) = 1. \quad (3.7)$$

### 5. Additivity

If two events  $X$  and  $Y$  are independent, then the entropy of their joint distribution is the sum of their individual entropies

$$H_s(X, Y) = H_s(X) + H_s(Y) \quad (3.8)$$

### 6. Subadditivity

The combined uncertainty or information content of two random variables together is never greater than the sum of their individual uncertainties.



$$H_s(X, Y) \leq H_s(X) + H_s(Y) \tag{3.9}$$

**7. Continuity**

$H_s$  should be continuous in all its arguments  $p_1, p_2, \dots, p_n$ .

**8. Concavity**

$H_s$  is a concave function of all its arguments.

The subset of Non-Negativity, Maximum Entropy, Additivity, and Concavity are the best-known examples of axiomatic characterization of the entropy measure.

There are a number of collections of axioms that are sufficient to characterize the Shannon entropy uniquely. It has been proven that the Shannon entropy is the only functional that satisfies any of these sets of axioms.

**3.1.2. Rényi Entropy**

Alfréd Rényi, a Hungarian mathematician, was one of the first to define new measures of uncertainty. He sought the most general definition of uncertainty measures that would preserve additivity for independent events and be compatible with the axioms of probability. His information measure, known as the Rényi entropy [28], was defined as

$$H_R^{(\alpha)}(p_1, p_2, \dots, p_n) = \frac{1}{1 - \alpha} \log_2 \left( \sum_{i=1}^n p_i^\alpha \right). \tag{3.10}$$

Let us here summarize the properties of Rényi's entropy measure:

**1. Symmetry**

$H_R^{(\alpha)}(X)$  is a symmetric function of its variables.

**2. Continuity**

$H_R^{(\alpha)}(X)$  is a continuous function of  $X$ .

**3. Maximum**

$$H_R^{(\alpha)}\left(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n}\right) = 1.$$

**4. Additivity**

$$H_R^{(\alpha)}(X, Y) = H_R^{(\alpha)}(X) + H_R^{(\alpha)}(Y)$$

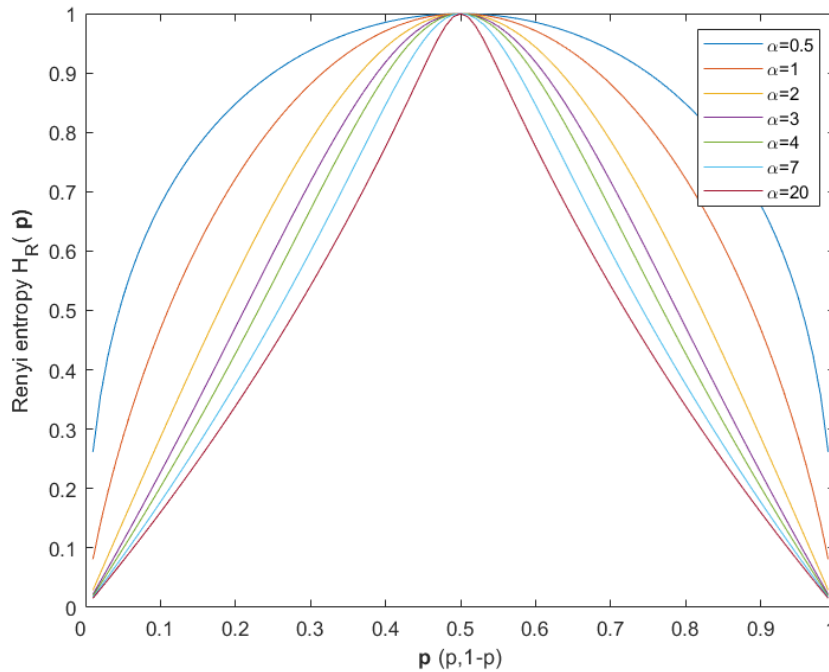
5. When  $\alpha \rightarrow 1$ , the Rényi entropy becomes the Shannon entropy. That is

$$\lim_{\alpha \rightarrow 1} H_R^{(\alpha)}(X) = H_S(X). \quad (3.11)$$

The above conditions 1 to 4 characterize the Rényi entropy, i.e., they are necessary and sufficient axioms for the Rényi entropy.

Rényi entropy is characterized as a continuous family of entropy measures depending on the value of the parameter  $\alpha$ [28].  $\alpha$  controls the degree of sensitivity of the entropy towards particular probability distribution functions[28].

A plot of Rényi entropy for different  $\alpha$  values is given in Figure 3.1 for cases when only two states are possible ( $n = 2$ ). We can see the change in the degree of sensitivity as the  $\alpha$  increases.



**Figure 3.1:** Rényi entropy( $H_R^{(\alpha)}$ ) for different  $\alpha$  values when  $n = 2$ ; for  $\alpha = 1$  then  $H_R^{(\alpha)} = H_S$

### 3.1.3. Tsallis Entropy

Tsallis entropy, also known as  $q$ -entropy, is a generalization of Shannon entropy introduced by Constantino Tsallis in 1988 [29]. Tsallis entropy introduces a parameter  $q$

that allows for a more flexible characterization of the distribution's properties, including fat-tailed and long-range dependence behaviours. It is defined as

$$H_T(x) = \frac{1}{q-1} \left( 1 - \sum_{i=1}^n p_i^q \right) \tag{3.12}$$

The Tsallis entropy has a similar axiomatic foundations as the Shannon entropy [30, 31] but unlike Shannon entropy, Tsallis entropy is non-additive. There is also a simple transformation between the Tsallis entropy and the Rényi entropy

$$H_R^{(\alpha)} = \frac{1}{1-q} \log(1 + (1-q)H_T(X)) \tag{3.13}$$

The Tsallis entropy has four important properties[32]

**1. Minimum**

If  $p = 1$ , then  $H_T(p) = 0$

**2. Pseudo-additivity or non-additivity**

Let X and Y be two independent systems in the probability theory, i.e.,

$$p_{ij}^{X+Y} = p_i^X + p_j^Y$$

then the following holds:

$$H_T(X + Y) = H_T(X) + H_T(Y) + \frac{1-q}{1} H_T(X)H_T(Y) \tag{3.14}$$

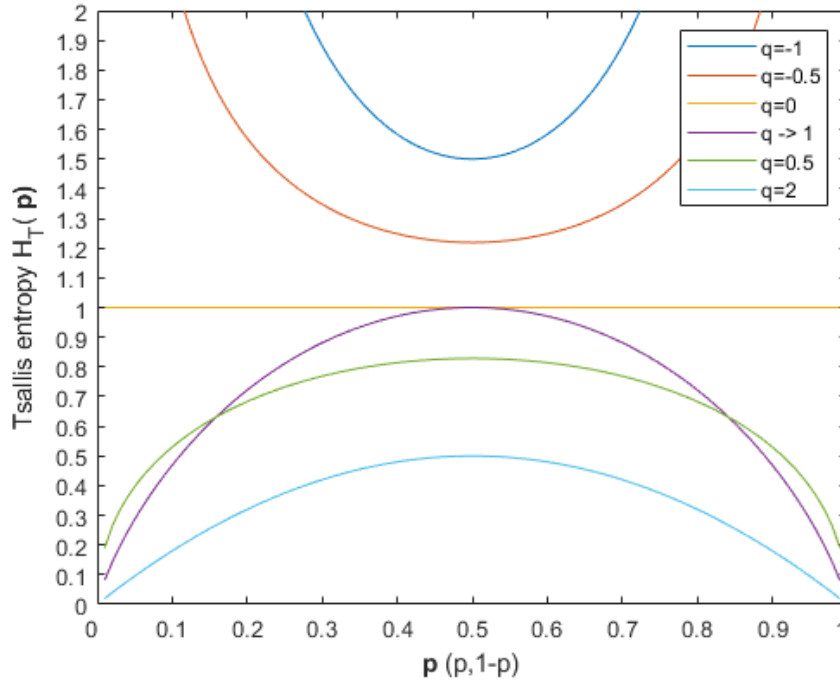
**3. Maximum**

$$H_T^{(\alpha)}\left(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n}\right) = 1$$

**4. If  $q = 1$  we get the Shannon entropy**

A plot of Tsallis entropy for  $q = -1, -0.5, 0, 0.5, 1$  and  $2$  is given in Figure 3.2. For  $q < 0$ , the Tsallis entropy is concave, and for  $q > 0$ , it becomes convex. For  $q = 1$ , it converges to the Shannon entropy. For all cases, the Tsallis entropy decreases as  $q$  increases.

Although the entropy is named after Tsallis, others studied it long before the 1988 paper in which Tsallis first mentioned it. For example, Havrda and Charvar [33] had



**Figure 3.2:** Tsallis entropy for different  $q$  values when  $n = 2$ ; when  $q \rightarrow 1$  then  $H_T = H_S$

already introduced a similar formula, adapted to base 2 logarithms, in the 1967 paper, and it was also mentioned by Patil and Taillie [34] in 1982.

## 3.2. Fuzzy Entropy Measures

For the calculation of classical entropy measures, such as Shannon or Rényi entropy, the sum of all probabilities associated with the events or outcomes in the set needs to equal 1. This requirement ensures that the probabilities represent a valid probability distribution.

For a set of events  $x_1, x_2, \dots, x_n$  with associated probabilities  $p_1, p_2, \dots, p_n$ , the sum of probabilities must satisfy 3.1.

This condition ensures that the probabilities cover all possible outcomes and form a normalized probability distribution. It is a fundamental prerequisite for entropy calculations because entropy measures the uncertainty or information content associated with a probability distribution, and the probabilities must sum to 1 to represent the entire distribution of possible events accurately.

The requirement that the sum of probabilities must equal 1 is not needed when cal-

culating fuzzy entropy. Fuzzy entropy is a concept used in fuzzy set theory, which deals with degrees of membership rather than traditional probabilities.

In fuzzy set theory, the sum of membership degrees (similar to probabilities in classical set theory) does not have to equal 1. Instead, each element in a fuzzy set is associated with a degree of membership between 0 and 1, which indicates the degree to which an element belongs to the set. These membership degrees can be fuzzy and do not need to be normalized.

Fuzzy entropy measures are designed to work with fuzzy sets and the corresponding degrees of membership. They capture the uncertainty or fuzziness inherent in these degrees of membership.

A fuzzy set  $A$  is defined on a universe of discourse  $X = x_1, x_2, \dots, x_n$  by Zadah[35] as

$$A = \{x_i | \mu_A(x_i) : i = 1, 2, \dots, n\}, \tag{3.15}$$

where  $\mu_A(x)$  is a membership function:

$$\mu_A(x) = \begin{cases} 0, & \text{x does not belong to A ,} \\ 1, & \text{x belongs to A ,} \\ < 0, 1 >, & \text{x is partially member of A.} \end{cases} \tag{3.16}$$

The idea of fuzzy entropy was first developed by Zadah [8] as a fuzziness metric. He created a weighted Shannon entropy by combining the membership function and probability theories; however, fuzzy sets did not respond well to the measure. The amount of fuzziness brought on by a fuzzy set’s ambiguity is measured by its fuzzy entropy. It is a crucial idea when discussing how to quantify the fuzziness of a fuzzy set. De Luca and Termini [9] defined the axioms that a fuzzy entropy must adhere to in order to be classified as a fuzzy entropy:

1.  $H(A)$  attains a minimum if and only if  $A$  is a crisp set, i.e.,  $\mu_A(x) = 0$  or  $1 \forall x$ .
2.  $H(A)$  attains a maximum if and only if  $A$  is the fuzziest set, i.e.,  $\mu_A(x) = 0.5 \forall x$ .
3.  $H(A^*) \leq H(A)$  where  $A^*$  is a sharpened version of  $A$ .
4.  $H(A) = H(\neg A)$  where  $\neg A$  is a complement set of  $A$ .

The measure they proposed was the first correct measure that defined the entropy of a fuzzy set based on Shannon's function. They suggest the following measure

$$H_f(A) = - \sum_{i=1}^n [\mu_A(x_i) \log \mu_A(x_i) + (1 - \mu_A(x_i)) \log(1 - \mu_A(x_i))] \quad (3.17)$$

Since then, there have been a number of new fuzzy entropy measures, including non-parametric and parametric  $((\alpha)$  and  $(\alpha, \beta))$  definitions.

Sharma and Tenaja developed one of the earliest entropies of type  $(\alpha, \beta)$  based on the generalization of the functional equation [36] in [37] as

$$H_\alpha^\beta(A) = \frac{1}{2^{1-\alpha} - 2^{1-\beta}} \left( \sum_{i=1}^n \mu_A(x_i)^\alpha - \sum_{i=1}^n \mu_A(x_i)^\beta \right), \quad (3.18)$$

where  $\alpha \neq \beta$ . The authors subtracted two averages to obtain this generalized entropy. Based on the generalized exponential entropy of a probability distribution presented in [38], Fan and Mal suggested a fuzzy entropy in [39]. It is

$$H_\alpha^\beta(A) = \left[ \prod_{i=1}^n \frac{\mu_\alpha(x_i)^\alpha - (1 - \mu_A(x_i))^\alpha}{\mu_\alpha(x_i)^\beta - (1 - \mu_A(x_i))^\beta} \right]^{\frac{1}{\beta - \alpha}}, \quad (3.19)$$

where  $\alpha \neq \beta$ .

Hooda in [40] proposed the following measure of fuzzy entropy based on the [41]:

$$H_\alpha^\beta(A) = \frac{1}{1-\beta} \sum_{i=1}^n \left[ (\mu_A^\alpha(x_i) + (1 - \mu_A(x_i))^\alpha)^{\frac{\beta-1}{\alpha-1}} - 1 \right], \quad (3.20)$$

where  $\alpha \neq \beta$ ,  $\alpha, \beta > 0$ , and  $\alpha \neq 1$ .

One of the more recent propositions of exponential fuzzy entropy of order  $(\alpha, \beta)$  is given by Joshi and Kumar [42] as

$$H_\alpha^\beta(A) = \frac{1}{n(e^{1-0.5^\alpha} - e^{1-0.5^\beta})} \sum_{i=1}^n \left[ (\mu_A(x_i)e^{(1-\mu_A(x_i))^\alpha} + (1 - \mu_A(x_i))e^{(1-(1-\mu_A(x_i))^\alpha)}) \right. \\ \left. - (\mu_A(x_i)e^{(1-\mu_A(x_i))^\beta} + (1 - \mu_A(x_i))e^{(1-(1-\mu_A(x_i))^\beta)}) \right] \quad (3.21)$$

where either  $\alpha > 1$ , and  $0 < \beta < 1$ , or  $0 < \alpha < 1$ , and  $\beta > 1$ .

This measure generalizes the Verma and Sharma entropy [43], Pal and Pal exponential

entropy [44], and De Luca and Termini logarithmic entropy [9]. Additional overview of generalized entropic forms, including entropies of order  $(\alpha, \beta)$  is given in [45]

Several fuzzy entropy measures exist, categorized into one-parameter and two-parameter variants. This section emphasizes two-parameter measures as Chapter 7 will introduce a novel fuzzy entropy measure, which also falls within the two-parameter category.

## 4. Chapter

# CONVOLUTIONAL NEURAL NETWORKS

### Contents

---

<b>4.1. Classification</b> . . . . .	<b>27</b>
<b>4.2. Convolutional Neural Networks</b> . . . . .	<b>29</b>

---

One of the goals of the thesis is the design of the method for classifying non-stationary signals. This chapter provides an introduction to machine learning classification, with a particular emphasis on convolutional neural networks in the domain of image classification. In addition, the chapter describes the specific convolutional neural networks utilized in the thesis.

### 4.1. Classification

Machine learning is a growing field of research today. It incorporates algorithms that can automatically learn from provided data examples to solve new tasks [46]. Learning is defined as improving the algorithm's performance at some tasks with experience, which is measured by different performance measures[47]. Some of the machine-learning tasks include classification, regression, transcription, machine translation, anomaly detection, and many more [47].

In machine learning and data analysis, classification is the process of categorizing or



labelling data into predefined classes or categories based on its attributes or features. It involves building a model that automatically assigns labels to new, unseen data points based on patterns and relationships learned from a labelled training dataset (supervised learning). The primary goal of classification is to make accurate predictions or decisions about the class membership of data instances.

When an algorithm learns from training data with known target responses or labels, which can be a categorical or a numeric value, this is known as supervised learning. Through training, a model is created that anticipates the response when a new example is presented.

Mathematically, supervised learning can be described as learning a function  $f$  that maps input data  $X$  to output data  $Y$ :

$$f : X \rightarrow Y \tag{4.1}$$

in a dataset of  $N$  samples, each with  $M$  features and corresponding labels. The input data can be represented as a matrix  $X \in \mathbb{R}^{N \times M}$ , where each row represents a sample and each column represents a feature:

$$X = [x_1, x_2, \dots, x_M]_N \tag{4.2}$$

The label can be denoted as a vector  $y \in \mathbb{R}^N$ , where each element corresponds to the label for the corresponding sample.

$$Y = [y_1, y_2, \dots, y_N] \tag{4.3}$$

The idea behind the classification task is to learn a function  $f$  that maps inputs to outputs. This function is typically represented as a model with a set of parameters denoted by  $w$ :

$$f(X; w) \approx y \tag{4.4}$$

The model is trained to reduce the discrepancy between the actual output and what was expected. A loss function, indicated by the symbol  $L$ , is generally used to calculate this inaccuracy since it measures the discrepancy between the output that was predicted

and the actual output:

$$L(y, f(X; w)) \quad (4.5)$$

The optimization problem is finding the set of parameters  $w$  that minimizes the loss function over the training data:

$$\min_w : L(y, f(X; w)) \quad (4.6)$$

An optimization algorithm like gradient descent can be used to address this optimization problem. The weights are updated in the direction of the negative gradient by the optimization process, which calculates the gradient of the loss function with respect to the weights:

$$w_{ij} = w_{ij} - \eta \frac{\partial L}{\partial w_{ij}} \quad (4.7)$$

where  $w_{ij}$  is the weight connecting neuron  $j$  to neuron  $i$ ,  $\eta$  is the learning rate (a small positive scalar), and  $\frac{\partial L}{\partial w_{ij}}$  is the gradient of the loss function with respect to the weight  $w_{ij}$

Once trained, the model can be used to forecast the results with fresh input data. To achieve this, new input data is fed into the trained model, which generates a predicted value. Decision Trees, the Random Forest algorithm, Support Vector Machines, Artificial Neural Networks, and Convolutional Neural Networks are the most popular supervised learning approaches that can be used for classification problems.

Next, we focus on Convolutional Neural Networks describing the idea of beginning the Convolutional Neural Networks-based classification.

## 4.2. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a subset of deep learning algorithms that are particularly adept at visual data-intensive applications like image and video analysis. They have substantially gained popularity in the fields of machine learning and artificial intelligence, changing a wide range of fields like computer vision and object recognition.

The first design of a CNN was proposed in work by Yann LeCun [48] under the name LeNet. It was developed for the problem of recognizing handwritten digits. The

developments of new CNNs happened in 2010 when a competition in image classification was announced under the name ‘ImageNet large scale visual recognition challenge’. It led to a considerable effort from academics to compare their computer vision and machine learning models on a shared dataset (ImageNet), particularly for picture categorization.

The design of CNNs is based on three different layers: convolutional, pooling, and fully connected (FC) layers.

### **Convolutional Layer**

The convolutional layer is the core building block of a CNN, where most computation occurs. It requires input data (input image), a filter, and results as a feature map. It applies a set of learnable filters, or kernels, to localized regions of the input data to produce a set of output feature maps. This localized connectivity enables the layer to focus on specific patterns and features, effectively capturing details within the data. Each filter slides over the input data, computing the dot product between its weights and a small patch of the input data, effectively capturing local spatial patterns.

Every kernel has its receptive field, which refers to the local region of the input data. In CNNs, the kernel size determines the dimensions of the receptive field.

Given an input matrix  $I$  and a filter matrix  $K$ , the convolution operation is defined as:

$$S(i, j) = (I \circledast K)(i, j) = \sum_m \sum_n K(i + m, j + n) \circledast I(m, n), \quad (4.8)$$

where  $S(i, j)$  represents the output feature map  $S$  at the position  $(i, j)$  in the output feature map, and  $\circledast$  denotes the convolution operation [49].

In a neural network, the activation function is responsible for transforming the summed weighted input from the neuron into the activation of the neuron or output for that input. ReLU, which stands for Rectified Linear Unit, is one of the activation functions commonly used in convolutional layers and other neural network layers. ReLU introduces non-linearity to the model. This is important because it allows the network to learn complex patterns in the data. Without non-linear activation functions like ReLU, the entire neural network would behave like a linear model. ReLU can also introduce sparsity into the network. Since it sets negative values to zero, it encourages some neurons to be inactive, effectively reducing the model’s complexity. This can help prevent overfitting.

## Pooling Layer

The pooling layer, also known as a downsampling layer, is used for dimensionality reduction. Just like the convolutional layer, the pooling operation moves a filter across the entire input. However, unlike convolution, this filter does not have any weights. Instead, it uses a simple calculation to combine the values in the area, and it collects the result in the output array. In CNNs, two primary types of pooling are commonly employed: Max Pooling and Average Pooling.

- **Max Pooling:** The biggest value present in each receptive field is chosen and preserved in the output
- **Average Pooling:** The average value of the data in the receptive field is calculated, and the average value is then assigned to the appropriate location in the output. Average pooling, as opposed to max pooling, considers the region's overall feature intensity.

## Fully connected layer

The fully connected layer gathers data from the entire feature map, in contrast to convolutional layers, which concentrate on spatial hierarchies and local patterns. A fully connected network is one in which every neuron in a fully connected layer is linked to every neuron in the layer above it. Typically, the fully connected layer is the last layer in the network as it generates the final output, which can be utilized to make decisions, such as categorizing an image into specified categories. The fully connected layers usually use a softmax activation function for classification, producing a probability from 0 to 1. The softmax function begins by normalizing the raw output values by exponentiating them all. This exponentiation emphasizes larger values, making them more distinct. They are then added together for all classes. This summing acts as a normalizing factor, ensuring that the final numbers fall between 0 and 1. Finally, the normalization factor is applied to the exponentiated data. This division yields a set of values that can be translated into probabilities. Each value represents the probability that the input belongs to a given class.

In this thesis, we will implement three different CNN architectures: Resnet, VGG,

and DenseNet. All models that were used were pre-trained on more than a million images from the ImageNet database.

## VGG

One of the most significant contributions of 2014 was the creation of a new architecture known as the VGGNet [50]. The VGGNet is an architecture developed by the Visual Geometry Group at Oxford University. It was believed that by making CNN deeper, one might better solve problems and achieve a reduced error rate on the ImageNet classification task. They worked on the notion that increasing depth allows one to simulate more non-linearities in one's function, and so the important contribution was to view depth as a significant component in the design. Other important design features included homogeneous architecture and reduced receptive fields. VGG's design primarily focuses on using small 3x3 convolutional filters and stacking them one after the other, which allows it to capture intricate features in images. Despite its straightforward architecture, VGG achieved remarkable results in image classification tasks, and its principles have influenced the development of subsequent, more complex CNN models. VGG16 is a specific variant of the VGG architecture consisting of 16 layers. These layers comprise 13 convolutional layers and 3 fully-connected layers. The architecture of VGG16 is straightforward, consisting of a series of convolutional layers with rectified linear unit (ReLU) activations, followed by max-pooling layers for spatial downsampling. The fully-connected layers at the end are typically used for classification tasks. The input of VGG is set to an RGB image of 224x224 size. The average RGB value is calculated for all images on the training set image, and then the image is input as the input to the VGG convolution network. A 3x3 or 1x1 filter is used, and the convolution step is fixed. The list of layers is given in Table 4.1.

Due to its strong performance and generalizability, VGG16 is often used as a pre-trained model. Researchers and practitioners fine-tune VGG16 on specific tasks or datasets, leveraging the knowledge learned from ImageNet.

Layer Name	Type	Output Shape	Stride	Activation
Input Image	Input	[224, 224, 3]	-	-
2X Convolution 3x3	Conv2D	[224, 224, 64]	1	ReLU
2X Max Pooling 2x2	MaxPooling2D	[112, 112, 64]	2	-
2X Convolution 3x3	Conv2D	[112, 112, 128]	1	ReLU
2X Max Pooling 2x2	MaxPooling2D	[56, 56, 128]	2	-
3X Convolution 3x3	Conv2D	[56, 56, 256]	1	ReLU
2X Max Pooling 2x2	MaxPooling2D	[28, 28, 256]	2	-
3X Convolution 3x3	Conv2D	[28, 28, 512]	1	ReLU
2X Max Pooling 2x2	MaxPooling2D	[14, 14, 512]	2	-
3X Convolution 3x3	Conv2D	[14, 14, 512]	1	ReLU
2X Max Pooling 2x2	MaxPooling2D	[7, 7, 512]	2	-
Flatten	Flatten	[25088]	-	-
3X FC	Dense	[4096]	-	ReLU
Output	FC	[1000]	-	Softmax

**Table 4.1:** Layers in the VGG16 Architecture

## Resnet

ResNet, short for “Residual Networks,” was developed by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun at Microsoft Research. The ResNet was introduced in the “Deep Residual Learning for Image Recognition” paper in 2015 [51]. It was created to address a common problem of the vanishing gradient. ResNet is known for its extreme depth, often exceeding 100 layers. This depth allows the model to learn increasingly abstract and intricate features from input data, which is particularly advantageous in computer vision tasks. The innovation of ResNet lies in its residual blocks. These blocks introduce skip connections, also known as shortcut connections or residual connections, that bypass one or more layers. Instead of trying to learn the desired output directly, ResNet focuses on learning the residual, or the difference between the desired output and the actual output from the skipped layers. This approach enables the model to learn more effectively and helps mitigate the vanishing gradient problem. In residual blocks, if the desired output is equivalent to the input (i.e., no change is required), the model can learn to represent the identity mapping, allowing gradients to flow smoothly through the network. ResNet has several variants, including ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152, which differ in terms of depth. Layers for Resnet-50 are described in Table 4.2.

Layer Name	Type	Output Shape	Stride	Activation
Input Image	Input	[224, 224, 3]	-	-
Conv1	Conv2D	[112, 112, 64]	2	ReLU
Max Pooling	MaxPooling2D	[56, 56, 64]	2	-
2X Conv2x 3x3	Conv2D	[56, 56, 64]	1	ReLU
2X Conv2x 3x3	Conv2D	[56, 56, 256]	1	ReLU
2X Conv2x 3x3	Conv2D	[56, 56, 512]	1	ReLU
3X Conv3x 3x3	Conv2D	[28, 28, 1024]	2	ReLU
4X Conv3x 3x3	Conv2D	[14, 14, 2048]	2	ReLU
Global Avg Pooling	AveragePooling2D	[2048]	-	-
FC	FC Dense	[1000]	-	-
Output	FC	[1000]	-	Softmax

**Table 4.2:** Layers in the ResNet-50 Architecture

### DenseNet

DenseNet, short for “Densely Connected Convolutional Networks,” is an innovative neural network architecture designed to enhance the flow of information and gradient throughout deep networks. It was introduced by Gao Huang, Zhuang Liu, and Laurens van der Maaten in their 2016 paper, “Densely Connected Convolutional Networks” [52].

The core idea behind DenseNet is the concept of “dense block”. In these blocks, each layer is connected to every other layer in a feed-forward fashion. This dense connectivity enables feature reuse and promotes the flow of gradients during training. Essentially, each layer receives as input not only the feature maps from the preceding layer but also those from all previous layers within the same dense block. To manage computational complexity, DenseNet often employs bottleneck layers within dense blocks. These bottleneck layers use 1x1 convolutions to reduce the number of feature maps before applying 3x3 convolutions. This approach helps to strike a balance between model capacity and computational efficiency. Between dense blocks, transition layers are typically used to downsample feature maps and reduce spatial dimensions. Transition layers include a combination of 1x1 convolutions for dimension reduction and average pooling for down-sampling. DenseNet101 is the original DenseNet architecture. It consists of four dense blocks and is relatively shallow compared to later versions. Its layers are described in Table 4.3.

The selection of these three Convolutional Neural Networks in this thesis (ResNet, DenseNet, and VGG) is based on their exceptional performance across a spectrum of

Layer Name	Type	Output Shape	Stride	Activation
Input Image	Input	[224, 224, 3]	-	-
Conv1	Conv2D	[112, 112, 64]	2	ReLU
Max Pooling	MaxPooling2D	[56, 56, 64]	2	-
4X Dense Block 1	Dense Block	[56, 56, 256]	-	ReLU
Transition 1	Conv2D	[28, 28, 128]	2	ReLU
4X Dense Block 2	Dense Block	[28, 28, 512]	-	ReLU
Transition 2	Conv2D	[14, 14, 256]	2	ReLU
23X Dense Block 3	Dense Block	[14, 14, 1024]	-	ReLU
Transition 3	Conv2D	[7, 7, 512]	2	ReLU
15X Dense Block 4	Dense Block	[7, 7, 2048]	-	ReLU
Global Avg Pooling	AveragePooling2D	[2048]	-	-
FC	FC Dense	[1000]	-	-
Output	FC	[1000]	-	Softmax

**Table 4.3:** Layers in the DenseNet-101 Architecture

computer vision applications, with the purpose of showing that these CNN architectures combined with time-frequency representations (images) of digital signals and derived entropy masks from time-frequency images can be used to improve classification of nonstationary signals. Chapter 5 gives a comprehensive review of related work involving the application of CNNs in signal classification, followed by Chapter 6, where we propose our novel nonstationary signal classification technique. The experiments and achieved classification results are elaborated on in Chapter 7.





## 5. Chapter

# OVERVIEW OF RELATED WORK

### Contents

---

<b>5.1. Methods in signal-noise separation . . . . .</b>	<b>38</b>
5.1.1. Classical representation . . . . .	38
5.1.2. Time-frequency representation . . . . .	39
<b>5.2. Methods for seismology signal classification . . . . .</b>	<b>41</b>
5.2.1. Seismology signal as a time-series . . . . .	41
5.2.2. Seismology signal as an image . . . . .	43

---

The related work in the field of signal-noise separation is briefly summarized in this chapter as an introduction to the proposed technique combining machine learning classification, time-frequency distributions, and information-entropy-based analysis of time-frequency images of nonstationary signals. The chapter is divided into two main sections, each of which addresses a significant contribution of this thesis. The first section focuses on the techniques used to separate signal from noise. We start by looking at the methods that work in one domain. We then proceed to more modern techniques that improve signal-noise separation by using time-frequency distributions. The classification of seismology signals is examined in detail in the second section since this is the application chosen to demonstrate the efficiency of here-proposed novel method designed to separate signal from noise. Seismology signals are examined from two unique angles: first, as time series data, and second, as two-dimensional representations resembling images.

## 5.1. Methods in signal-noise separation

A perfect signal-noise separation extracts only the signal and ignores all of the noise. In the real world, achieving that without apriori knowledge about the signal is a challenging task. This chapter describes related work on blind-source separation (BSS) with different tools, including time-frequency distributions and entropy measures.

### 5.1.1. Classical representation

In the case of the approaches that work with representation in one domain, the BSS methods are often based on independent component analysis (ICA). It is an iterative method with the goal of optimizing a function. In this case, it is a contrast function that allows for the separation of different sources in a multi-component signal. There are a number of different upgrading algorithms and optimizing methods that have been applied for this problem [53]. This chapter will mention only some.

The most popular method that works with time or frequency representation is Principal Component Analysis (PCA) using Singular Value Decomposition (SVD). PCA was introduced as a tool in signal processing in paper [54]. From then on, it was combined with SVD in [3]. This approach has found widespread application in BSS, especially within the realm of EEG signal processing [55, 56, 57, 58]. PCA with SVD is a powerful technique for the BSS, but it performs best when the source signals are statistically independent and the noise is uncorrelated with the sources as it leverages the statistical properties of the data [59].

PCA is based on a greedy approach. A greedy approach in BSS refers to a class of algorithms that iteratively estimate and separate the source signals one at a time. These algorithms are called “greedy” because they make local decisions at each step with the goal of maximizing some criterion without considering the global optimization of the entire separation problem [60]. Greedy algorithms are often used when the BSS problem is complex and finding a global solution is computationally infeasible. However, one problem with this approach is that the order in which sources are estimated can affect the results. The other problem is that the convergence is not always guaranteed, and the algorithm may get stuck in local minima [61].

Another popular approach is the relaxation approach. The relaxation approach in BSS combines elements of optimization theory, constraint satisfaction, and iterative refinement [62, 63]. It allows for incorporating prior knowledge about the sources and the mixing process.

Iterative approaches based on ICA are particularly effective when the sources have different statistical properties, such as non-Gaussian distributions, and when they are mixed in a way that preserves their statistical independence. In addition, ICA has limitations, such as the assumption of linear mixing and the requirement of a sufficient number of observations to estimate the sources accurately. Additionally, ICA is sensitive to the order in which the sources are estimated, which can affect the results significantly [53].

BSS techniques utilizing entropy measures often rely on one-domain representations for source separation. These approaches aim to leverage the information-theoretic concept of entropy to distinguish and separate sources based on their statistical properties.

Shannon entropy can be employed to assess the statistical properties of sources. Sources with distinct entropy profiles can often be separated effectively. For example, sources with significantly different entropy values may be separated using thresholding or clustering techniques. In paper [64], such a technique, utilizing Shannon entropy and BSS, was employed for detection.

An extensive analysis of entropy methods for instantaneous blind source separation was presented in [5]. The analysis included Shannon entropy and Rényi entropy.

Depending on the parameter choice, Rényi entropy measures can capture different aspects of source statistics. In one-parameter representation, a specific  $\alpha$  value is selected to emphasize particular statistical features for source separation. Another approach based on Rényi entropy was introduced in [65] with variable order of entropy. Similar work was performed in [4]. A method for blind source separation of sources by minimizing the mutual information between segments was presented, utilizing both Rényi and Shannon entropy.

### **5.1.2. Time-frequency representation**

Unlike the methods that work with the classic representation, TFD-based methods for signal-noise separation typically do not rely on the statistical independence assumption.

When referring to TFD used in BSS, several algorithms were developed over the years [66, 67, 68, 69, 70, 71]. In this approach, TFD matrices are typically crafted from the auto-terms and cross-terms of the observed signals. These matrices are then employed for source diagonalization, antidiagonalization, or fusion. The choice depends on the specific time-frequency point selections and the structure of the source TFD matrices. These established methods enable the separation of source signals that share identical spectral shapes but possess different time-frequency localization. However, when dealing with multi-component signals, the inherent bilinear nature introduces problematic cross-terms that complicate signal component detection and extraction. These often lead to misinterpretation of the signals.

To address this challenge, various approaches [72, 73, 74, 75] have been devised to mitigate the interference caused by cross-terms, guided by the concept of Time-Frequency Ratio of Mixtures [76, 77]. These methods draw inspiration from a paradigm where individual sources manifest exclusively in small and distinct time-frequency zones, imposing minimal constraints on source sparsity and overlap.

A variation of sparse component analysis, rooted in the time-frequency domain, was introduced in [78]. Research into blind source separation in the time-frequency domain has also been explored in [79] and [80], involving the transformation of mixed signals from the time domain to the time-frequency domain. These studies verified the effectiveness and superiority of the proposed algorithm, albeit under the assumption of multiple sensors and single-source points. Both approaches are dependent on the number of sensors. Additionally, [81] presents alternative methods for blind source separation based on the mixing matrix, also reliant on the number of sensors.

[6] investigates a technique that combines wavelet transform with time-frequency blind source separation using the smooth pseudo Wigner-Ville distribution. This approach aims to extract characteristic waves from electroencephalogram data, with the resulting output utilized in constructing a support vector machine. The drawback of this approach is either there are cross-terms that can lead the algorithms in the wrong direction, or they are dependent on the multiple numbers of sensors.

In [82], an entirely automated undetermined blind source separation approach is presented. This method relies on a signal time-frequency distribution and employs a peak detection and extraction technique. The local number of components is determined by

extracting information from the TFD Short-term Rényi entropy.

In the paper [7], the authors propose an automatic and adaptable method to find and separate useful information from time-frequency distributions. The technique uses the K-means clustering algorithm to sort the data. Instead of straightforward thresholding, it employs an algorithm that blindly separates useful data from background noise using local Rényi entropy without any prior knowledge about the signal.

While the majority of the approaches and algorithms discussed throughout this chapter primarily focus on the application of Blind Source Separation for disentangling multiple signals originating from various sources, this thesis adopts a slightly different perspective. Here, the focus of BSS is to separate the desired signals from a background of noise. This approach is motivated by the fact that, in practical signal analysis scenarios, the presence of unwanted noise can significantly impact the accuracy and reliability of signal interpretation.

The developed method was applied to the analysis of the seismograms, which is a problem tackled also by other approaches briefly summarized next.

## **5.2. Methods for seismology signal classification**

Here, we first briefly explore seismology signal classification based on time-series data. Also, we discuss entropy measures that can be utilized in classification procedures.

Next, we focus on a different approach: transforming signals into images (time-frequency distributions) for classification purposes (image-style classification).

### **5.2.1. Seismology signal as a time-series**

Seismology's automated earthquake detection and selection problems date back to the late 1970s[83]. Over the past few decades, significant advancements have been made in seismological techniques for identifying and studying earthquake events described as a time series. Some of these approaches involve constructing wave function models to replicate the physical propagation of earthquake waves through the Earth's lithosphere, enabling the examination of various patterns and characteristics[84]. Others have their roots in signal processing and statistical methods. Notably, Short-Term Averaging/Long-Term

Averaging (STA/LTA) has emerged as a widely adopted detection method in seismology, particularly in applications such as weak-motion seismology, due to its practicality [85]. This method calculates the ratio of the average absolute amplitude between a short-term window and a sliding long-term window applied to the seismic signal [86, 87].

In contrast to STA/LTA, the autocorrelation method has proven to be more effective in detecting weak seismic signals, although it is computationally intensive [88]. The autocorrelation approach operates by searching for target waveforms based on their similarity without requiring a predefined waveform template. Another method, known as template matching, strikes a balance between detection accuracy and computational efficiency. It employs a custom detection threshold that adapts to the existing earthquake events and statistical data for classification [89]. Template matching determines the correlation coefficient between the sample waveform and expert-selected templates using a “one-to-many” strategy.

Machine learning based analyses of time-series seismological data require data labeled in several classes, with supervised machine learning being used to predict or categorize the class to which a new data set belongs.

Support Vector Machines (SVM) used to be one of the most popular methods in the field of seismology for earthquake detection [90]. Through training on sample data, the classic SVM algorithm identifies an optimal hyperplane that categorizes samples into two classes, making SVM particularly well-suited for distinguishing earthquake events from non-earthquake events.

In recent times, the advent of neural networks has brought a transformative shift to the field of seismology classification. In 2015, experimental research was conducted to evaluate the potential of advanced Artificial Neural Networks (ANN) and machine learning in estimating earthquake magnitudes [91]. It demonstrated higher accuracy in predicting artificial data as opposed to historical data. ANN was also applied in [92] with a focus on the considerable data imbalance in datasets.

Recurrent neural networks (RNN), in particular Long Short-Term Memory (LSTM) Networks, have become more popular recently in the field of earthquake prediction. LSTM Networks for earthquake forecasting was first applied in 2018.[93]. The superior performance of LSTM in earthquake trend analysis compared to Feedforward Neural Networks (FFNN) was demonstrated in [94].

Short-term earthquake magnitude prediction using LSTM was introduced in [95], emphasizing the challenging nature of training networks with “zero values” or instances of zero occurrences.

Furthermore, LSTM networks were used to predict earthquake frequency from a spatio-temporal perspective in [96]. Their approach involved the subdivision of an area of interest into distinct sub-regions, referred to as the “decomposition method,” to facilitate the exploration of spatio-temporal correlations across a broader geographical area.

Notably, deep neural networks have been used to analyze seismic time series for tasks like denoising, enhancing event detection, and classifying data. A CNN called ConvNetQuake was first introduced in [97], where authors formulated earthquake detection as a supervised classification problem. Improved versions have since been offered in [98, 99, 100]. The ConvNetQuake had exceptional results in comparison to the state-of-the-art at the time but has used the waveforms in the time domain.

However, in this thesis, we propose analyzing seismograms in the transformation domain (time-frequency domain) combined with information entropy rather than training models just with time-series data. Next, we briefly summarize methods utilizing this approach.

### **5.2.2. Seismology signal as an image**

Time-frequency representation is frequently used in signal classification, either to extract features [101, 102, 103] or as an input for classifying images [104, 105, 106, 107, 108]. Recognizing seismic events within waveform time series bears a resemblance to object recognition in 2D photographs. In this analogy, the three components of a 3-component seismogram can be likened to the RGB color channels, effectively forming one image. A similar concept applies when dealing with seismic signal TFDs, involving the extraction of TFDs for each component from the three-component seismograms to construct three-channel images.

The potential of time-frequency representations for seismic data classification through machine learning approaches was explored in a recent study [109]. This study involved training three CNNs using nine different time-frequency distributions and comparing their performance with a model trained on raw waveform data. While certain distributions re-



sulted in performance below the baseline model, others exhibited marginal improvements.

Other research has investigated the application of CNN-RNN hybrid networks [110] and TFDs [111] for earthquake classification, particularly on smaller datasets. A method for analyzing GPS-derived seismic signals to determine the epicenter of far-field regional earthquakes, employing TFDs and multilateration was introduced in [112]. A rapid regional seismic damage assessment approach using CNN and leveraging TFD graphs of ground motions as detailed visual representations for near-real-time damage prediction has also recently been introduced[113].

In this thesis, we introduce the idea of upgrading time-frequency images of seismograms with information entropy measures in order to get the information maps and then utilize these for training machine-learning classifiers. The entropy measures, in addition to their widespread utilization in signal classification, notably within the realm of medical signal analysis [114, 115, 116], have found application across diverse fields for classification purposes [117, 118, 119]. The majority of these methods primarily operate on the time series data of a signal, while research regarding the utilization of entropy measures derived from time-frequency distributions is limited.

Some studies have used CNNs with entropy features extracted from TFDs for computer-aided diagnosis using EEG signals [120, 121]. In the field of medical science, more extensive research has been conducted on the utilization of entropy measures derived from TFDs, particularly in the analysis of EEG signals [108, 122, 123, 124, 125].

Motivated by the various uses of TFDs in classification tasks and recognizing the potential advantages of entropy measures, the following chapter outlines our new technique reliant on entropy measures designed to separate signal from noise. We will also demonstrate the efficiency of the proposed technique in the context of earthquake classification.

## 6. Chapter

# PROPOSED 2DLEM METHOD FOR SIGNAL SEPARATION

### Contents

---

<b>6.1. Methodology</b> . . . . .	<b>46</b>
6.1.1. First step: signal decomposition into TFD . . . . .	46
6.1.2. Second step: entropy calculation . . . . .	46
6.1.3. Third step: RICI . . . . .	48
6.1.4. Fourth step: entropy maps to masks utilizing RICI . . . . .	53
6.1.5. Performance metrics . . . . .	56
<b>6.2. Novel entropy for the 2DLEM method</b> . . . . .	<b>58</b>
6.2.1. Dombi operators . . . . .	60
6.2.2. Novel Dombi entropy . . . . .	61

---

This chapter introduces a new method for separating signals from noise, along with a customized entropy measure tailored for this method. Its procedure will be broken down in steps by explaining the algorithms, and operators involved, as well as how the entropy measure was adapted to fit the proposed approach.

## 6.1. Methodology

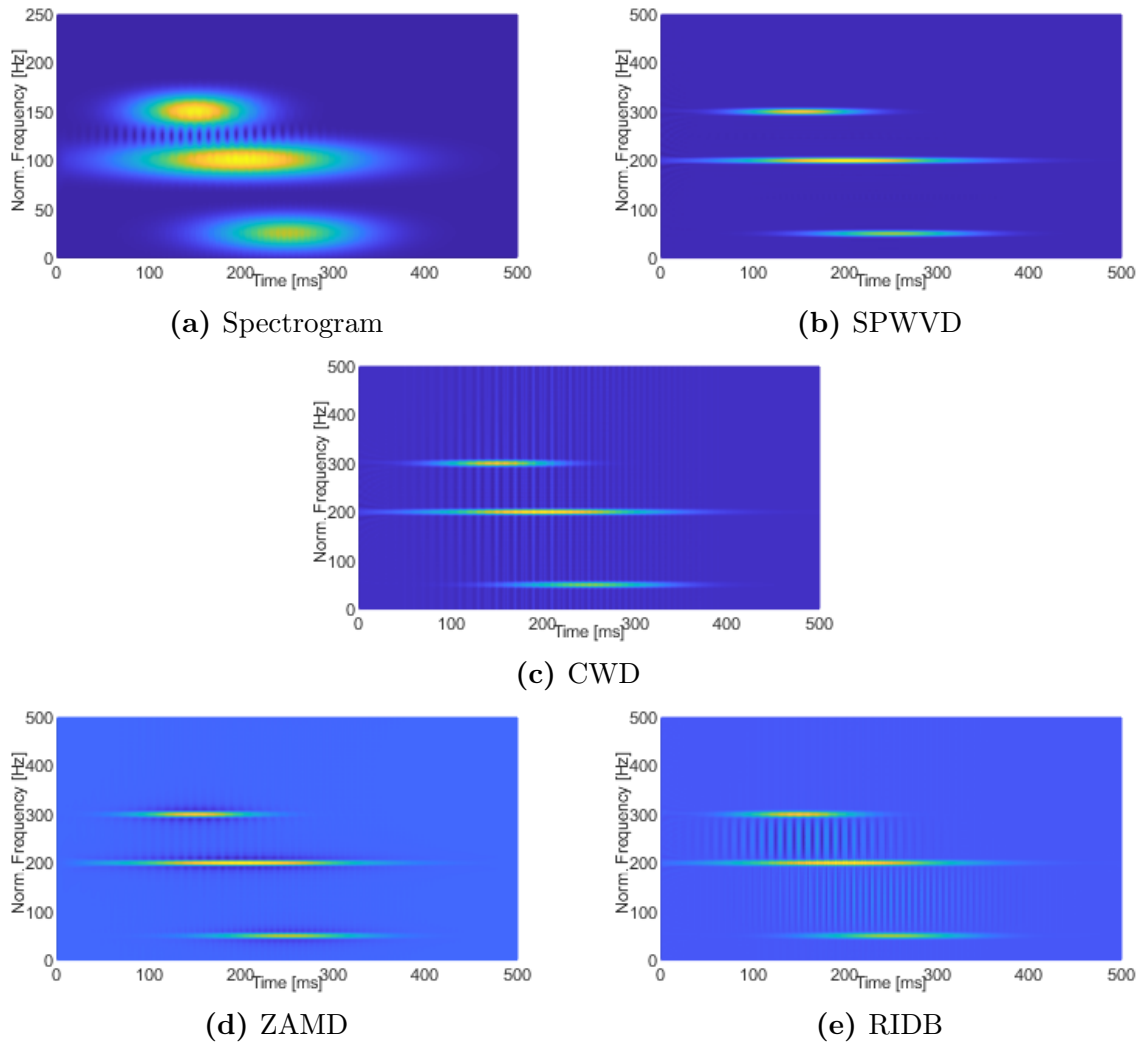
The proposed method called the 2D Local Entropy Method (2DLEM), aims to extract valuable information content from noisy signals. It's based on the premise that a two-dimensional entropy map, described in section 6.1.4, could provide an additional and more effective basis for precisely separating useful content from noise in the time-frequency domain than the traditional methods utilizing only TFDs. The method combines several steps, which will be explained in detail in the following sections. The overview of the entire method can be found in Figure 6.11.

### 6.1.1. First step: signal decomposition into TFD

The initial step in the proposed method involves generating the TFD from the signal's time series. As discussed in Chapter 2., five distributions were employed: spectrogram, SPWVD, CWD, ZAMD, and RIDB. These distributions were carefully chosen after extensive testing, considering the distinct results. For the evaluation, three synthetic signals were created. The first signal (A), in Figure 6.1, comprised three distinct components with a linear frequency modulation between -690 and 460; the second (B), in Figure 6.2, was a multi-component signal with a single point of overlap between components with a linear frequency modulation between -75 and 75; and the third (C), in Figure 6.3, was a multi-component signal with three points of overlap between components with a linear frequency modulation between -60 and 60 and another component had a parabolic frequency modulation between -100 and 100. To assess the method's effectiveness, noise was manually added to the original clean signals, with varying SNRs: -3, 0, and 5. For a comprehensive analysis, stochastic noise was generated 100 times, and the results obtained were averaged.

### 6.1.2. Second step: entropy calculation

The main part of the method is obtaining the information entropy map from a signal TFD. Originally, three entropy measures were selected: Shannon, Rényi, and Tsallis. For every point in the distribution, local, adaptive windowed entropy is calculated, this process will be explained in the next section. The adaptive part of the method refers to

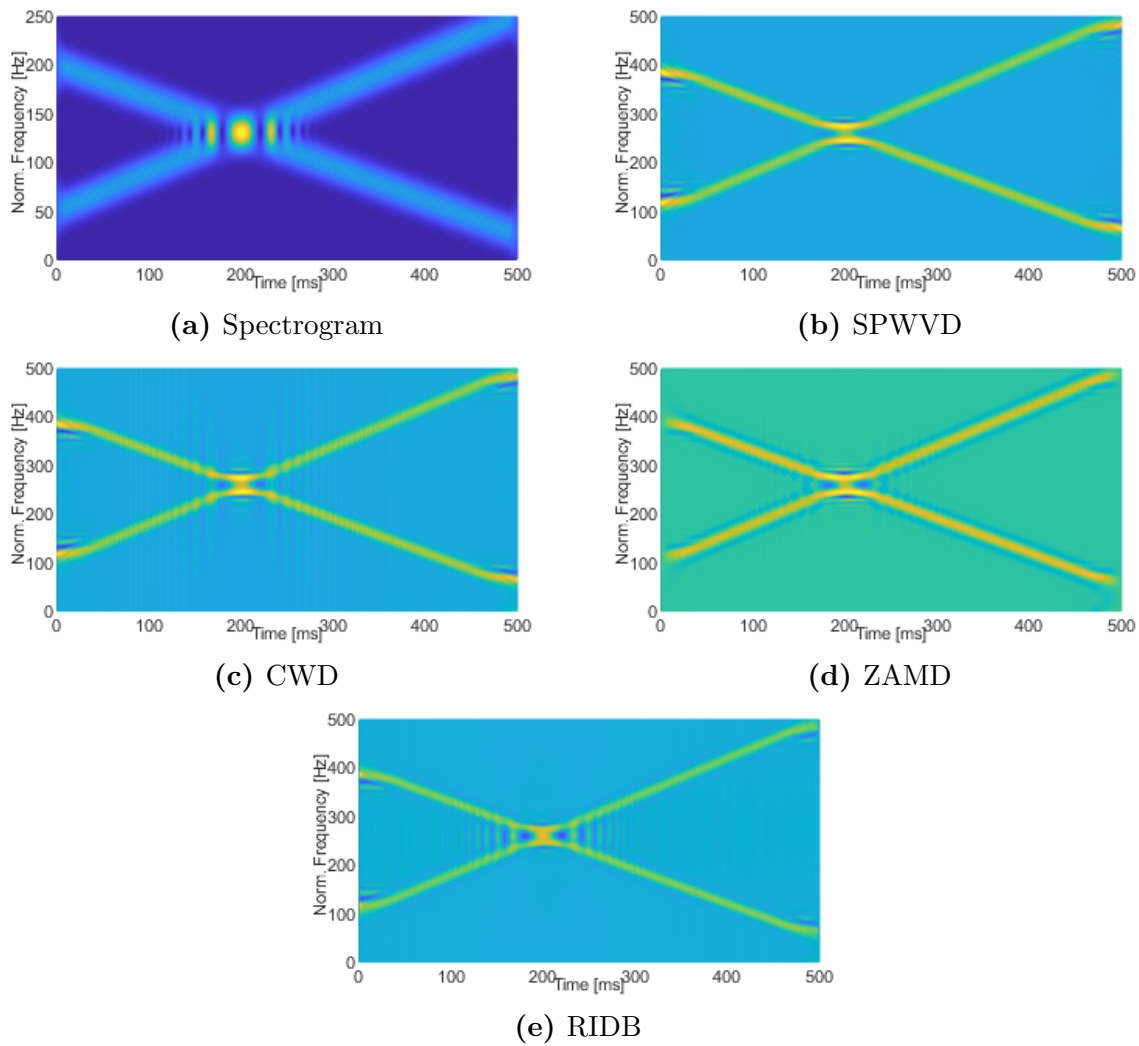


**Figure 6.1:** TFDs for signal A

the varying, data-driven size of the window. Selecting proper window sizes has a great impact on the method results. Figures 6.4, 6.5 and 6.6 show the resulting entropy map for signals A, B and C with three different fixed window sizes (window size is equal for each point in the distribution).

In the method, for every point in the distribution, local windowed entropy was originally calculated for window sizes ranging from  $2 \times 2$  to  $\frac{signalsize}{10} \times \frac{signalsize}{10}$ . Extensive experiments show no need for windows larger than  $15 \times 15$ . Depending on the window size, the entropy changes. This change is visible in one example for signal C for all three entropies in Figure 6.7.

Experiments show that the optimal results are achieved when selecting a window size corresponding to the point where the entropy values exhibit a sharp increase or there



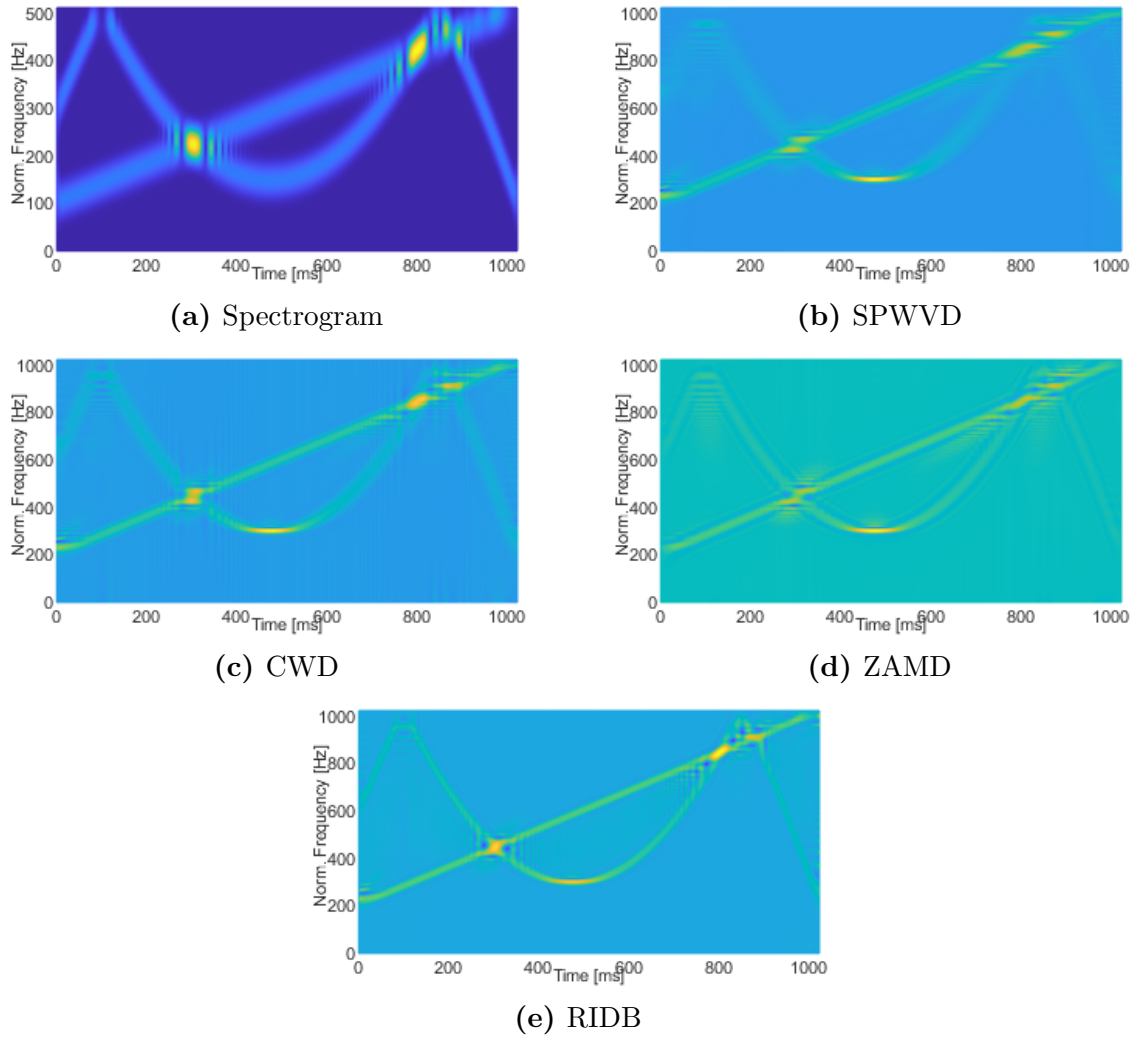
**Figure 6.2:** TFDs for signal B

is an inflection point, indicating that the noise has effectively been encapsulated by the window. As illustrated in Figure 6.7, in the example provided, a window size of 13 is selected for all entropies for this example(Signal C with SNR = 0).

### 6.1.3. Third step: RIC1

To get the proper window size, where the sharp increase or an inflection point in entropy values occurs as the window size increases, the relative intersection of confidence intervals(RICI) [126] algorithm was utilized. The original purpose of the algorithm was for signal denoising; however, it was applied here for a different objective described in the sequel.

After obtaining the TFD of the signal, local entropy( $H_{(t,f)}^{\Delta}$ ) is computed at each point

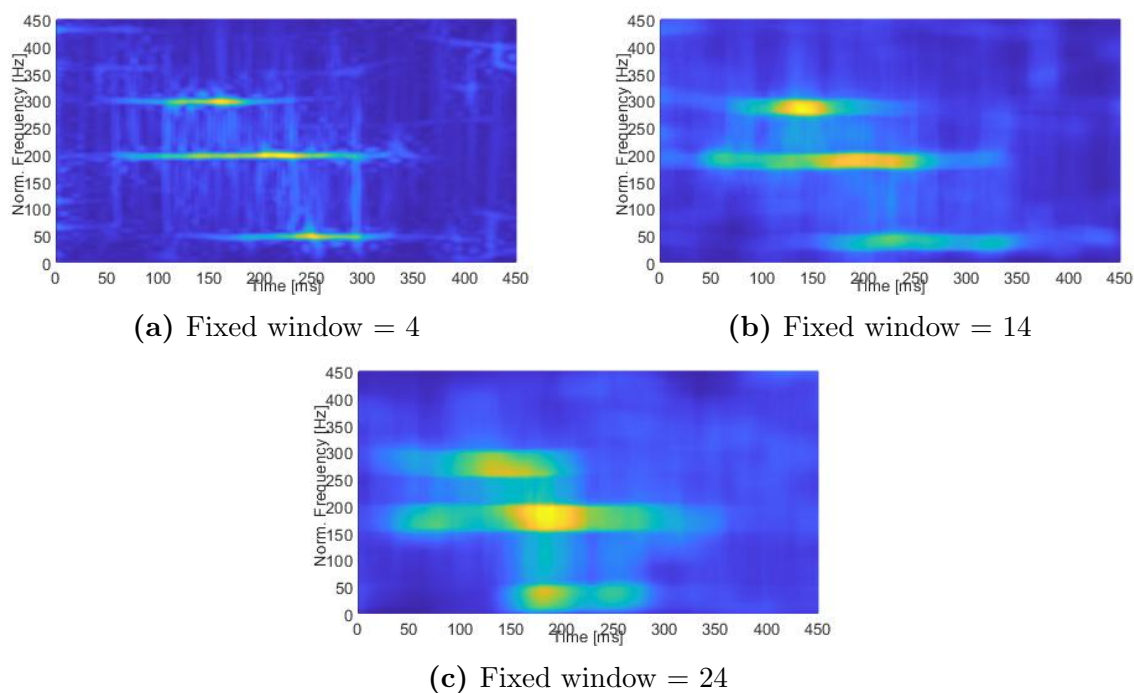


**Figure 6.3:** TFDs for signal C

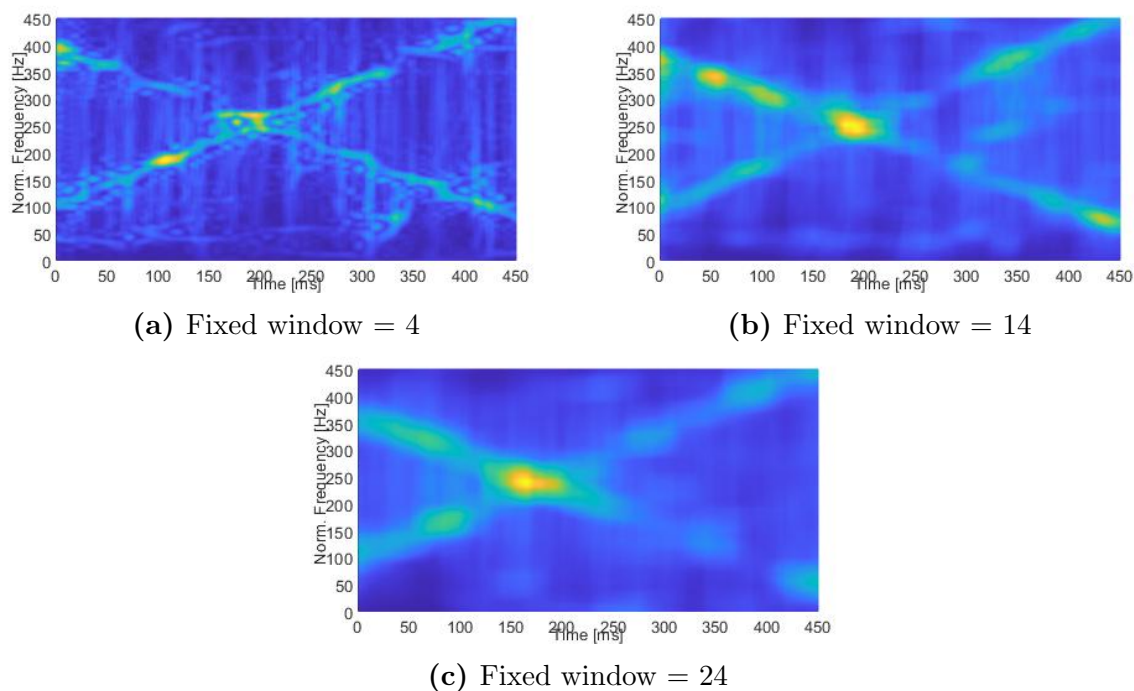
of the distribution. This computation is performed over square windows, with sizes spanning from two to one-tenth of the signal's size as  $\Delta = \Delta_2, \Delta_3, \dots, \Delta_n$  where  $\Delta_2 = 2 \times 2$  and  $\Delta_n = \frac{\text{signalsize}}{10} \times \frac{\text{signalsize}}{10}$ . The RIC algorithm takes the entropy values  $H_{(t,f)}^\Delta$  for each window size as input and selects the window size, tracking the changes in entropy. Specifically, it identifies the window size corresponding to the first slope of the entropy curve, signifying a significant shift in entropy behaviour. In this context, this shift means the point at which noise begins to impact the entropy measure significantly.

For every point  $(t,f)$  in TFD, entropy  $H_{(t,f)}^{RICI}$  is calculated as

$$H_{(t,f)}^{RICI} = RIC(H_{(t,f)}^\Delta) \quad (6.1)$$



**Figure 6.4:** Example of entropy map with different fixed window sizes for signal A

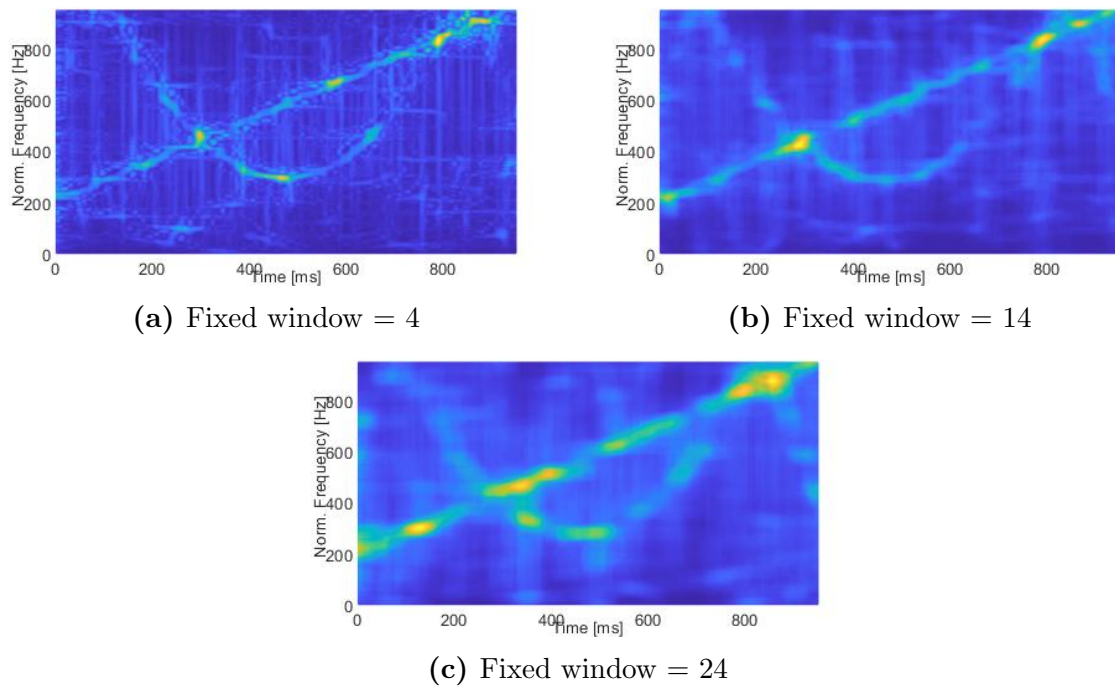


**Figure 6.5:** Example of entropy map with different fixed window sizes for signal B

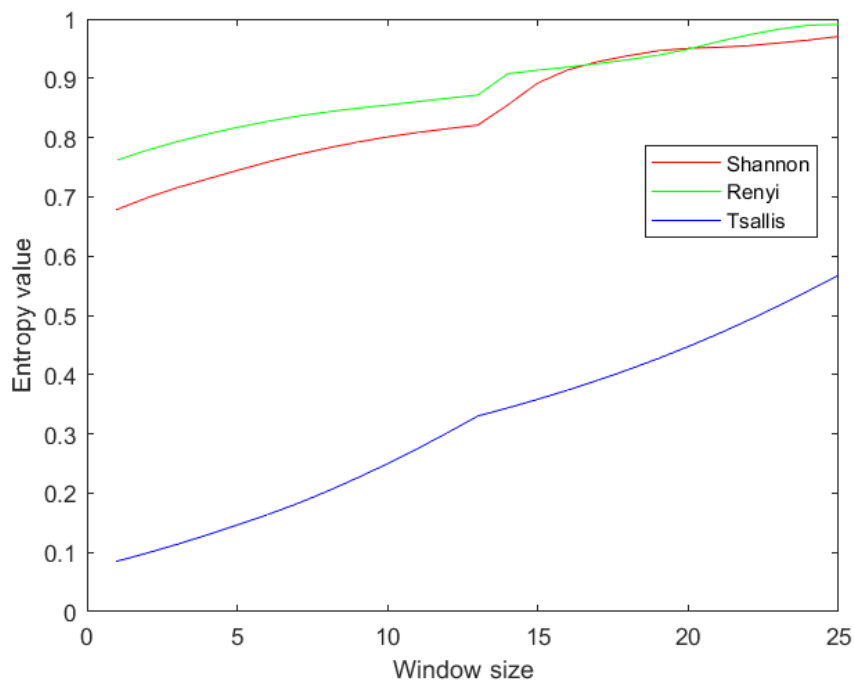
where

$$H_{(t,f)}^\Delta = \{H_{(t,f)}^{\Delta_2}, H_{(t,f)}^{\Delta_3}, \dots, H_{(t,f)}^{\Delta_n}\} \quad (6.2)$$

The algorithm generates results by examining the existence of an overlap between the



**Figure 6.6:** Example of entropy map with different fixed window sizes for signal C



**Figure 6.7:** Entropy values in one point of TFD for different window sizes

confidence intervals of signal entropy for specific window sizes.

For each window entropy calculation, a sequence of confidence intervals is calculated



with the lower and upper limit

$$D(n, \Delta) = [L(n, \Delta), U(n, \Delta)] \quad (6.3)$$

where

$$\begin{aligned} L(n, \Delta) &= H(n, \Delta) - \Gamma \cdot \sigma(n, \Delta) \\ U(n, \Delta) &= H(n, \Delta) + \Gamma \cdot \sigma(n, \Delta) \end{aligned} \quad (6.4)$$

where  $\Gamma$  is the threshold parameter of the confidence intervals.

The RICI rule introduces additional tracking of the amount of overlapping of confidence intervals, defined as

$$O(n, \Delta) = \underline{U}(n, \Delta) - \bar{L}(n, \Delta), \quad (6.5)$$

$\Delta = 1, 2, \dots, L$ . In order to obtain the value belonging to the finite interval  $[0, 1]$ ,  $O(n, \Delta)$  was divided by the size of the confidence interval  $D(n, \Delta)$  resulting in  $R(n, \Delta)$  defined as

$$R(n, \Delta) = \frac{\underline{U}(n, \Delta) - \bar{L}(n, \Delta)}{\underline{U}(n, \Delta) - \bar{L}(n, \Delta)}. \quad (6.6)$$

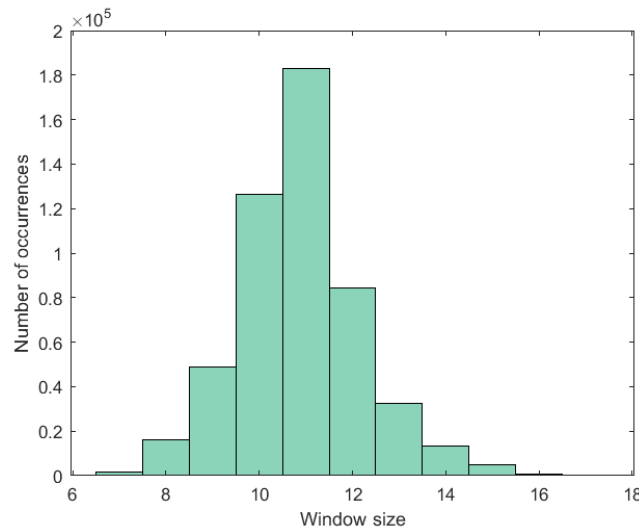
The optimal window width selection by the RICI rule can also be expressed as

$$R(n, \Delta) \geq R_c, \quad (6.7)$$

where  $R_c$  is a chosen threshold [126, 127, 128]). The window width  $\Delta^+$  obtained by the RICI rule is defined as

$$\Delta^+ = \max \{ \Delta : R(n, \Delta) \geq R_c \} \quad (6.8)$$

In Figure 6.8, the number of occurrences of different window sizes is portrayed. We



**Figure 6.8:** Number of occurrences in one example of each window size

can see that for this example, window size 11 is the most common, while there are no occurrences of window sizes less than 6 or greater than 15.

When optimal entropy size is defined for each point in the TFD, the result is the entropy map  $M(t, f)$ .

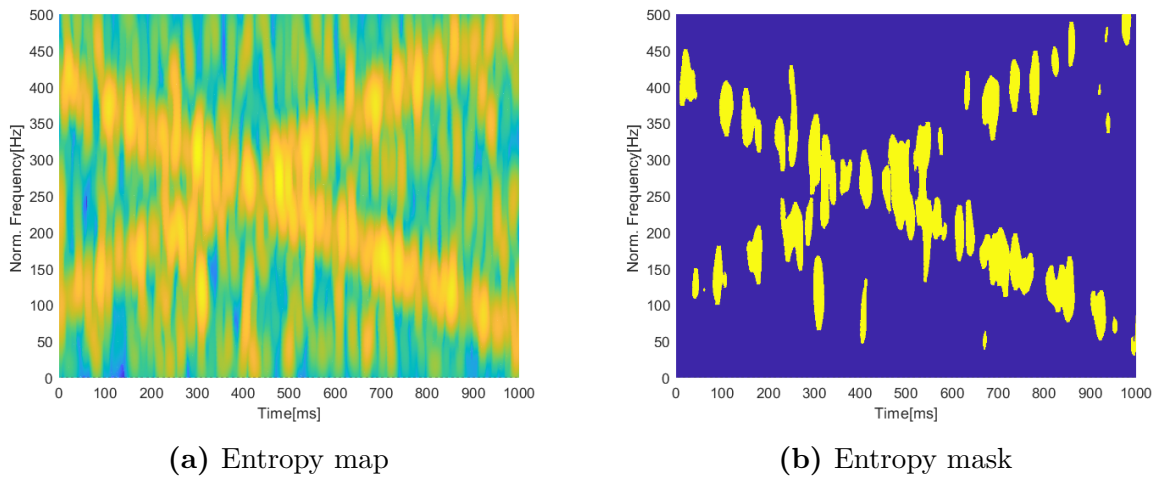
#### 6.1.4. Fourth step: entropy maps to masks utilizing RIC1

When local entropy is calculated for each point in the TFD, the entropy map, as shown in Figure 6.9a, is obtained. The next step is to obtain an entropy mask. The entropy mask is the representation of an entropy map by using zeros and ones. Ones represent points where the signal is present, and zero is a point where there is no signal presence, as demonstrated in Figure 6.12b.

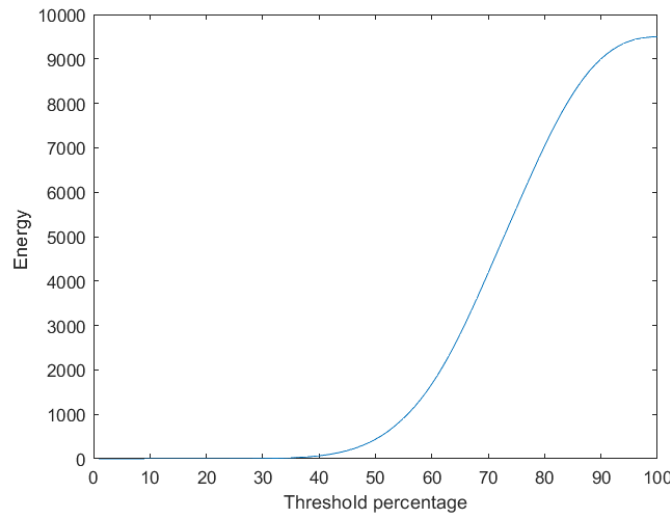
For this process, the RIC1 algorithm is once again used but with different inputs. Firstly, different thresholds are defined as

$$\tau = 0.01 \times \max(M), 0.02 \times \max(M), \dots, 0.99 \times \max(M) \quad (6.9)$$

For each  $\tau$ , the signal energy,  $E(M(t, f, \tau))$ , when an entropy map threshold is imposed, is calculated. This energy metric represents the map's energy under the selected threshold



**Figure 6.9:** Entropy map and mask of an example signals TFD



**Figure 6.10:** Energy calculations for different thresholds

$\tau$ . The RIC algorithm takes the energy calculations for the thresholds as input

$$\tau^+ = RIC\{E(M(t, f, \tau))\} \tag{6.10}$$

In Figure 6.10, we can see how signal energy changes by changing the threshold.

The output of the RIC algorithm is then the selected threshold for the obtaining of the entropy mask

$$\chi = M(t, f, \tau^+) \tag{6.11}$$

The implementation of the 2DLEM method was executed in the Python programming language. Concurrently, other components integral to the process, specifically the

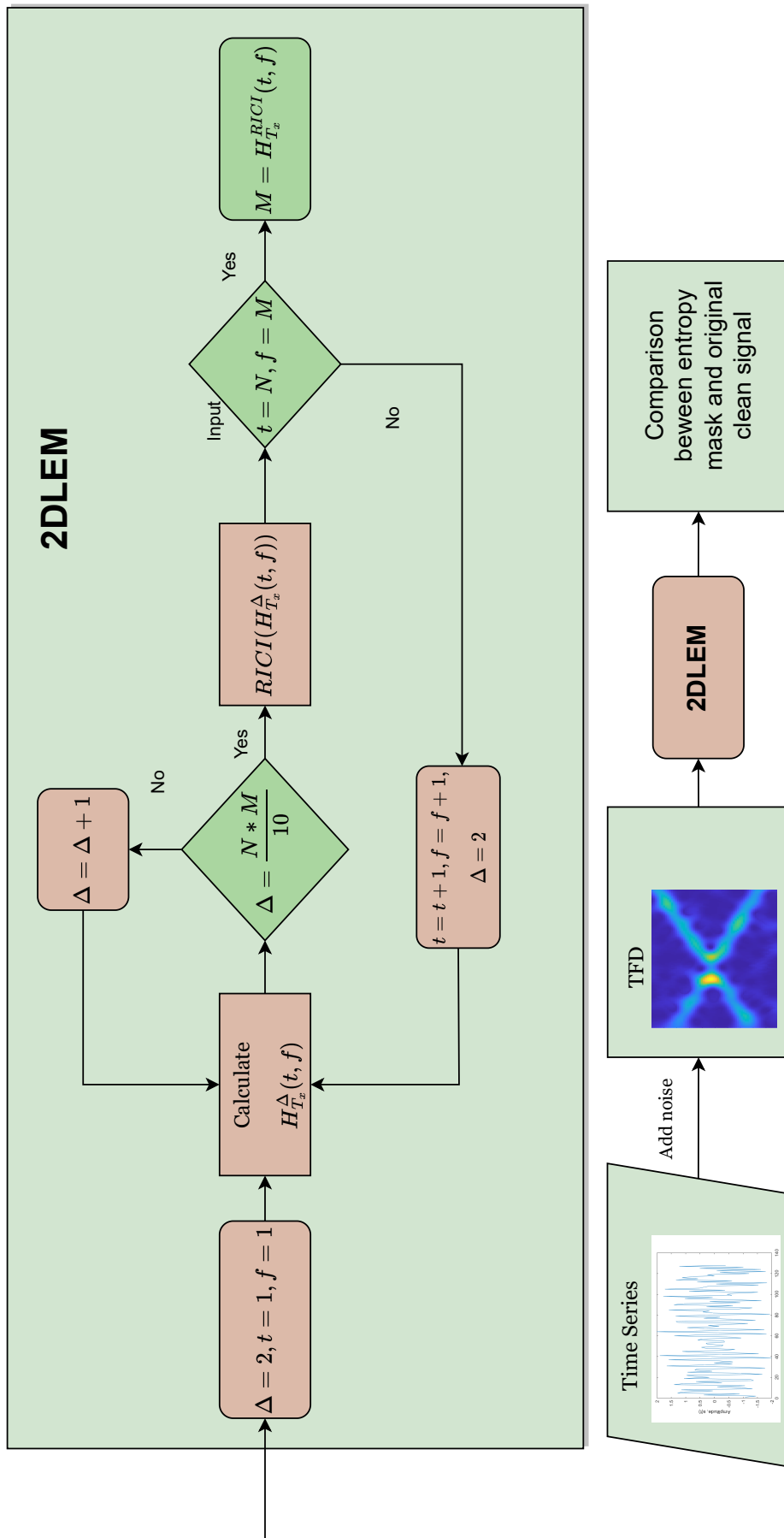


Figure 6.11: 2DLEM method steps

calculation TFD and the RIC method, were utilized from their original Matlab implementations.

### 6.1.5. Performance metrics

When comparing the ideal extraction of signal components from noise and the extraction obtained by the 2DLEM, the resulting error indicates the discrepancy between the non-zero elements in the subtraction of the mask of the noise-free signal and the mask obtained by the method. A correct extraction results in an error map with all zeros, where 1 represents a false negative, and -1 represents a false positive. Two metrics were used to evaluate the performance of the method. The first is accuracy, which is calculated as the difference between a given and correct results. In this context, points that correctly classify signal and noise are represented by 0 elements in the subtraction mask and contribute to the calculation of True Positives (TP) + True Negatives (TN) in the metric. True Positives (TP) represent correctly classified signal points, while True Negatives (TN) represent accurately classified points where the signal component is not present. False negatives (FN) correspond to points where the signal is present but are incorrectly classified as noise by the method's mask, indicated by a value of 1 in the subtraction matrix. False positives (FP) are points where the noise has been incorrectly classified as a signal and are defined by a value of -1 in the subtraction matrix. The accuracy is calculated as follows

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.12}$$

As can be seen from the above equation, the accuracy may not be ideal for unbalanced data sets, especially if the useful signal is only a fraction of the entire set. In mask extraction, where the relevant signal occupies a smaller part of the signal TFD, the F1 score provides additional information about the method's performance. The F1 score takes into account both precision and recall and represents a harmonic mean between these two aspects

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{6.13}$$

where

$$precision = \frac{TP}{TP + FP} \tag{6.14}$$

and

$$recall = \frac{TP}{TP + FN} \quad (6.15)$$

Accuracy and F1 scores commonly serve as metrics in machine learning to assess the performance of classification models. In this instance, they have been employed to evaluate how well the obtained mask aligns with the provided noise-free signal. These metrics find application in various studies related to image [129, 130] and signal processing [131], including research on EEG signals[132, 133].

In the results section when comparing machine learning classification models, accuracy and F1 score are also used but as a metric for comparing labels and obtained binary classification.

In addition to accuracy and the F1 score, the area under the receiver operating characteristic (AUROC) curve and the Matthews correlation coefficient (MCC) were used as performance indicators for classificatio results.

The AUROC is a metric for binary classification tasks that evaluates a model's ability to discriminate between classes. It quantifies the trade-off between a true-positive rate (sensitivity) and a false-positive rate (specificity) at different thresholds. Since the probability distributions for true positives and false positives are known, the ROC curve is obtained as a cumulative distribution function. The AUROC values range from 0 to 1, with higher values indicating better performance of the model in discriminating between positive and negative classes. A value of 0.5 indicates random classification, while a value closer to 1 indicates better discriminatory ability.

The Matthews correlation coefficient (MCC) is a commonly used metric in binary classification that takes into account TP, TN, FP and FN to evaluate the performance of the model. It is beneficial for unbalanced datasets as it balances the accuracy regardless of the class distribution. MCC values range from -1 to 1, where 1 stands for a perfect prediction, 0 for a random prediction and -1 for a complete mismatch between predictions and actual labels. A higher MCC value represents better overall performance on binary classification tasks. It is defined as:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (6.16)$$

A part of the results analysis are also statistical tests.

## 6.2. Novel entropy for the 2DLEM method

Classical entropies were first utilized within the method for entropy map calculation. However, here is presented a new family of Dombi operators that were adapted for the entropy measure fitting to the proposed method.

Shannon and Rényi entropies can be characterized through arithmetic and geometric means. Sharma and Mittal pioneered the investigation of averaging and nonadditivity in this context, as noted in their work [41]. More recently, similar research has been conducted by other researchers[134].

The generalized arithmetic mean is defined as

$$A^{(\alpha)}(\mathbf{x}, \mathbf{w}) = \left( \sum_{i=1}^n w_i x_i^\alpha \right)^{\frac{1}{\alpha}}. \quad (6.17)$$

where  $w_i \geq 0$  and  $\sum_{i=1}^n w_i = 1$ .

The generalized or power geometric mean is defined as

$$G(\mathbf{x}, \mathbf{w}) = \prod_{i=1}^n x_i^{w_i}. \quad (6.18)$$

A link between the generalized arithmetic mean and the generalized geometric mean has been explored in [135]:

$$\lim_{\alpha \rightarrow 0} A^{(\alpha)}(\mathbf{x}, \mathbf{w}) = G(\mathbf{x}, \mathbf{w}). \quad (6.19)$$

To make a connection to entropy, firstly, we can say that the value  $x_i$  and the weight  $w_i$  are equal, for  $x_i = w_i = p_i$ , we can write the two mean operators in the following way:

$$G(\mathbf{p}, \mathbf{p}) = \prod_{i=1}^n p_i^{p_i}, \quad (6.20)$$

$$A^{(\alpha)}(\mathbf{p}, \mathbf{p}) = \left( \sum_{i=1}^n p_i p_i^\alpha \right)^{\frac{1}{\alpha}} = \left( \sum_{i=1}^n p_i^{\alpha+1} \right)^{\frac{1}{\alpha}}. \quad (6.21)$$

Secondly, we will perform a parameter transformation, where the parameter  $\beta$  is defined as  $\alpha + 1$ . Notably,  $\alpha$  takes on positive values, implying that  $\beta$  is consistently greater than 1.

By utilizing this transformation function, we can now establish a connection between mean operators and entropy measures. Specifically, Shannon and Rényi entropies serve as monotonic transformations of  $G(\mathbf{p}, \mathbf{p})$  and  $A^\alpha(\mathbf{p}, \mathbf{p})$  when  $F(x) = k \ln(x)$  represents a monotone transformation function.

Transformation  $F(\mathbf{x})$  with the geometric mean gives Shannon entropy

$$H_S(\mathbf{p}) = F(G(\mathbf{p}, \mathbf{p})) = k \ln \left( \prod_{i=1}^n p_i^{p_i} \right) = k \sum_{i=1}^n p_i \ln(p_i). \quad (6.22)$$

If we use the arithmetic mean, where  $k = -\ln\left(\frac{1}{n}\right)$ , we obtain the Rényi entropy

$$H_R^{(\alpha)}(\mathbf{p}) = F(A^\alpha(\mathbf{p}, \mathbf{p})) = k \frac{1}{\beta - 1} \ln \left( \sum_{i=1}^n p_i^\beta \right). \quad (6.23)$$

Namely, with the transformation of the geometric power mean, we obtain the Shannon entropy, while with the transformation of the generalized arithmetic mean, we obtain the Rényi entropy.

From the mean operators, we can generate the entropy measures, while the mean operators can be generated by a generator function. With different composite equations, we can obtain different operators. While this holds true in the case of the classical Shannon and Rényi entropy, it is also possible to apply it to fuzzy theory.

Generalized mean can be defined as [136]

$$F(\mathbf{x}) = f^{-1} \left( \sum_{i=1}^n w_i f(x_i) \right), \sum_{i=1}^n w_i = 1, \quad (6.24)$$

if we define  $w_i = x_i$ , we obtain the entropy function

$$F(\mathbf{x}) = f^{-1} \left( \sum_{i=1}^n x_i f(x_i) \right), x_i = [0, 1], \quad (6.25)$$

where  $f$  is the generator function of the entropy, and it is a strictly monotonic and concave or convex function on  $[0, 1]$ . This generator function of the entropy is normally used in



fuzzy theory as a generator function of the operator [36].

A strict monotonously increasing operator in fuzzy theory then has the form

$$a(\mathbf{w}, \mathbf{x}) = f^{-1} \left( \sum_{i=1}^n w_i f(x_i) \right), \tag{6.26}$$

where  $f$  is the generator function  $f = [0, 1] \rightarrow [0, \infty], 1 \geq w_i \geq 0$ . If  $w_i = 1$  for all  $i$ , then the operator is associative, while for  $\sum w_i = 1$ , the operator is bisymmetric.

It becomes evident that each logical system produced by the function  $f$  possesses its distinct entropy measure. For example, when  $f(x) = -\ln(x)$ , it results in the Shannon entropy (where  $-\ln(x)$  serves as a generalization of the probabilistic operator  $C(x_1, \dots, x_n) = \prod_{i=1}^n x_i$ ). In the realm of fuzzy logic theory, various operators are usable, including min-max, Hamacher, Einstein, product, Frank, Lukasiewicz, Azcel-Asina, and Dombi. In the next section, we will focus on the Dombi operator family [137].

### 6.2.1. Dombi operators

The generalized Dombi operator typically involves two parameters. However, for conjunctive or disjunctive operators, just one parameter is sufficient.

Using the Dombi operator represented by  $f(x) = \left(\frac{1-x}{x}\right)^\alpha$ , we derive an entropy measure

$$H_d(\mathbf{x}) = \frac{1}{1 + \left(\sum_{i=1}^n x_i \left(\frac{1-x_i}{x_i}\right)^\alpha\right)^{\frac{1}{\alpha}}}. \tag{6.27}$$

This measure is obtained from the associativity equation solution when we define  $f(x) = \left(\frac{1-x}{x}\right)^\alpha$

$$c(x, y) = \frac{1}{1 + \left(\left(\frac{1-x}{x}\right)^\alpha + \left(\frac{1-y}{y}\right)^\alpha\right)^{\frac{1}{\alpha}}}. \tag{6.28}$$

We can replace the Dombi operator with its generalized form  $f(\mathbf{x}) = \ln \left(1 + \beta \left(\frac{1-x_i}{x_i}\right)^\alpha\right)$ , we then obtain the following measure:

$$D(\mathbf{x}) = \frac{1}{1 + \left(\frac{1}{\beta} \left(\sum_{i=1}^n x_i \ln \left(1 + \beta \left(\frac{1-x_i}{x_i}\right)^\alpha\right) - 1\right)\right)^{\frac{1}{\alpha}}}. \tag{6.29}$$

If  $x_i$  is used as weight, and if we apply the generalized means, we obtain an entropy

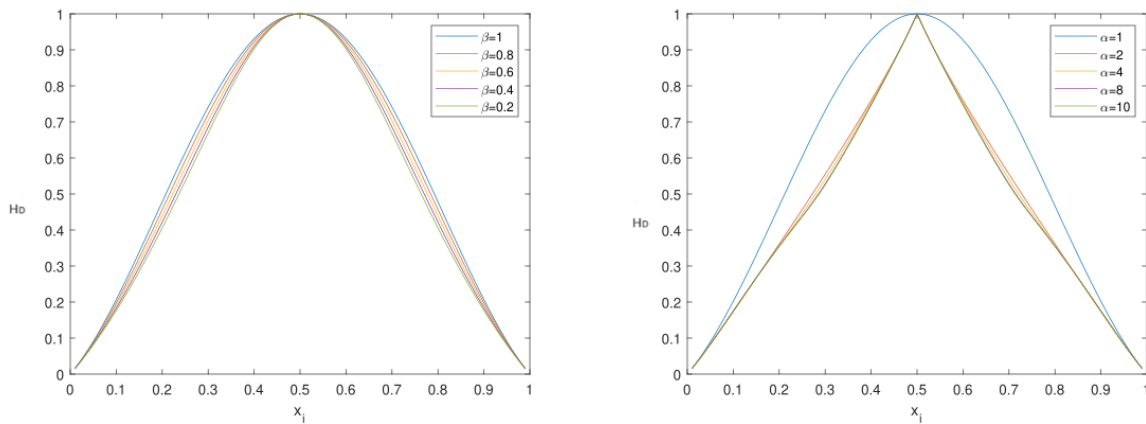
measure

$$H_D(\mathbf{x}) = \frac{1}{1 + \left(\frac{1}{\beta} \left(\prod_{i=1}^n \left(1 + \beta \left(\frac{1-x_i}{x_i}\right)^\alpha\right)^{x_i} - 1\right)\right)^{\frac{1}{\alpha}}}. \quad (6.30)$$

### 6.2.2. Novel Dombi entropy

This section will describe the behaviour of the Dombi entropy measure when compared to the classical Shannon entropy.

Figure 6.12 shows the behaviour of the entropy for different  $\alpha$  and  $\beta$  values.



(a) Entropy measure values for  $\alpha = 1$ , and  $\beta = 1, \beta = 0.8, \beta = 0.6, \beta = 0.4$  (b) Entropy measure values for  $\beta = 0.9$ , and  $\alpha = 1, \alpha = 2, \alpha = 3, \alpha = 4$

**Figure 6.12:** Dombi entropy measure for different  $\alpha$  and  $\beta$  values for  $n = 2$

The Shannon entropy function for  $n = 2$  can be written in the form

$$H_S(x) = -(x \log_2(x) + (1-x) \log_2(1-x)) \quad (6.31)$$

the approximation of the Dombi entropy measure is

$$H_D^{(\alpha)}(x) = \frac{1}{1 + \left(\frac{1}{2} \left(\left(\frac{1-x}{x}\right)^\alpha + \left(\frac{1-x}{x}\right)^{-\alpha}\right)\right)^{\frac{1}{\alpha}}}, \quad \alpha \geq 1 \quad (6.32)$$

If  $\alpha = 1$ ,

$$H_D^{(1)}(x) = 4x(1-x). \quad (6.33)$$

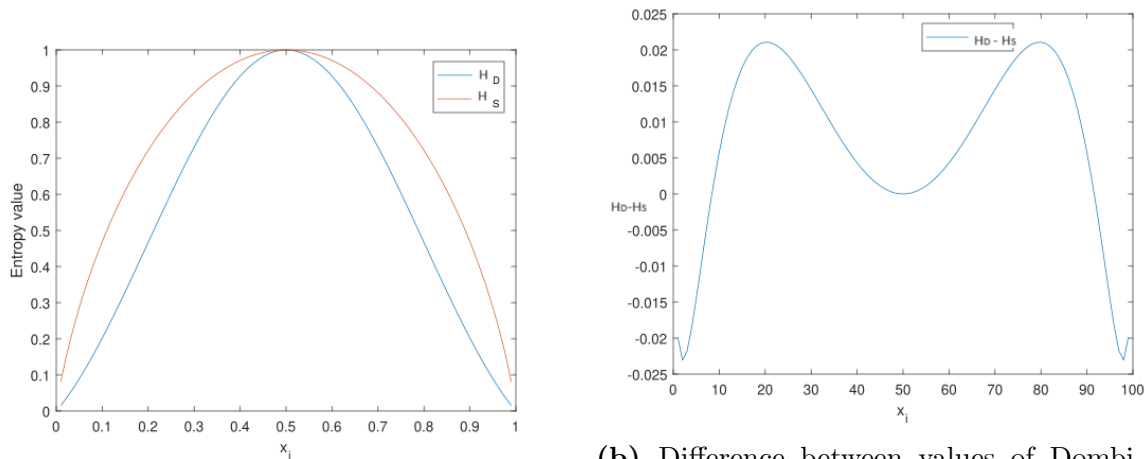
We can see that

$$\max |H(x) - H_D^{(1)}(x)| \leq 0.15. \quad (6.34)$$

When  $\alpha^* = \frac{6}{\pi^2}$ ,

$$\max |H(x) - H_D^{(\alpha^*)}(x)| \leq 0.025. \tag{6.35}$$

Figure 6.13 shows the differences between the Shannon entropy and the Dombi entropy measure.



(a) Values of Dombi entropy measure and the Shannon entropy for  $\alpha = 1$  and  $\beta = 1$

(b) Difference between values of Dombi entropy measure and the Shannon entropy for  $\alpha = \frac{6}{\pi^2}$ , and  $\beta = 1$

**Figure 6.13:** Comparison of the Dombi entropy measure and the Shannon entropy

To determine correct  $\alpha$  and  $\beta$  values for using the Dombi entropy in the proposed method, we tested several parameter values by referencing Figure 6.12. To gain larger differences in entropy value between similar values, we selected  $\alpha$  value of 2 and  $\beta$  value of 0.5 in our tests.

The next section presents the results of the adaptive, data-driven procedure for separating signal components from noise in the time-frequency domain utilizing novel Dombi entropy.

## 7. Chapter

# RESULTS AND DISCUSSION

### Contents

---

<b>7.1. 2DLEM applied to synthetic data . . . . .</b>	<b>64</b>
7.1.1. Dataset - synthetic signals . . . . .	64
7.1.2. 2DLEM with Shannon entropy . . . . .	73
7.1.3. 2DLEM with Rényi entropy . . . . .	79
7.1.4. 2DLEM with Tsallis entropy . . . . .	82
7.1.5. 2DLEM with Dombi entropy . . . . .	88
7.1.6. Comparison of 2DLEM with different entropy measures on synthetic data . . . . .	93
<b>7.2. 2DLEM applied to real-world data . . . . .</b>	<b>96</b>
7.2.1. 2DLEM for noisy speech signals . . . . .	96
7.2.2. 2DLEM with CNN for seismic signal classification . . . . .	101

---

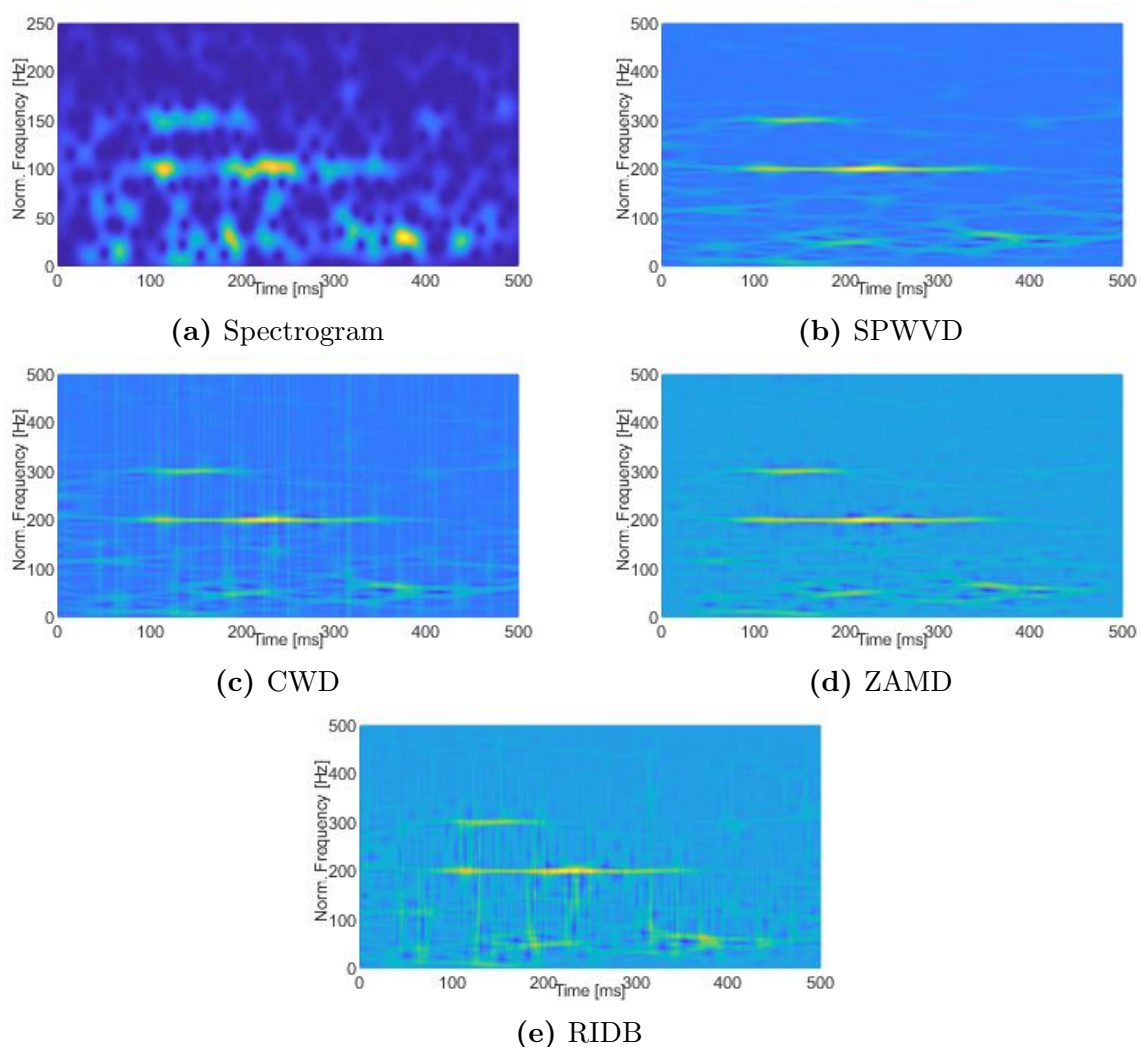
This section presents the results of applying the 2DLEM method to extract signal components from noise. For illustration, the extraction is first performed on three different synthetic signals described in section 6.1.1. with different SNR values and different TFDs (with different entropy measures being used).

## 7.1. 2DLEM applied to synthetic data

### 7.1.1. Dataset - synthetic signals

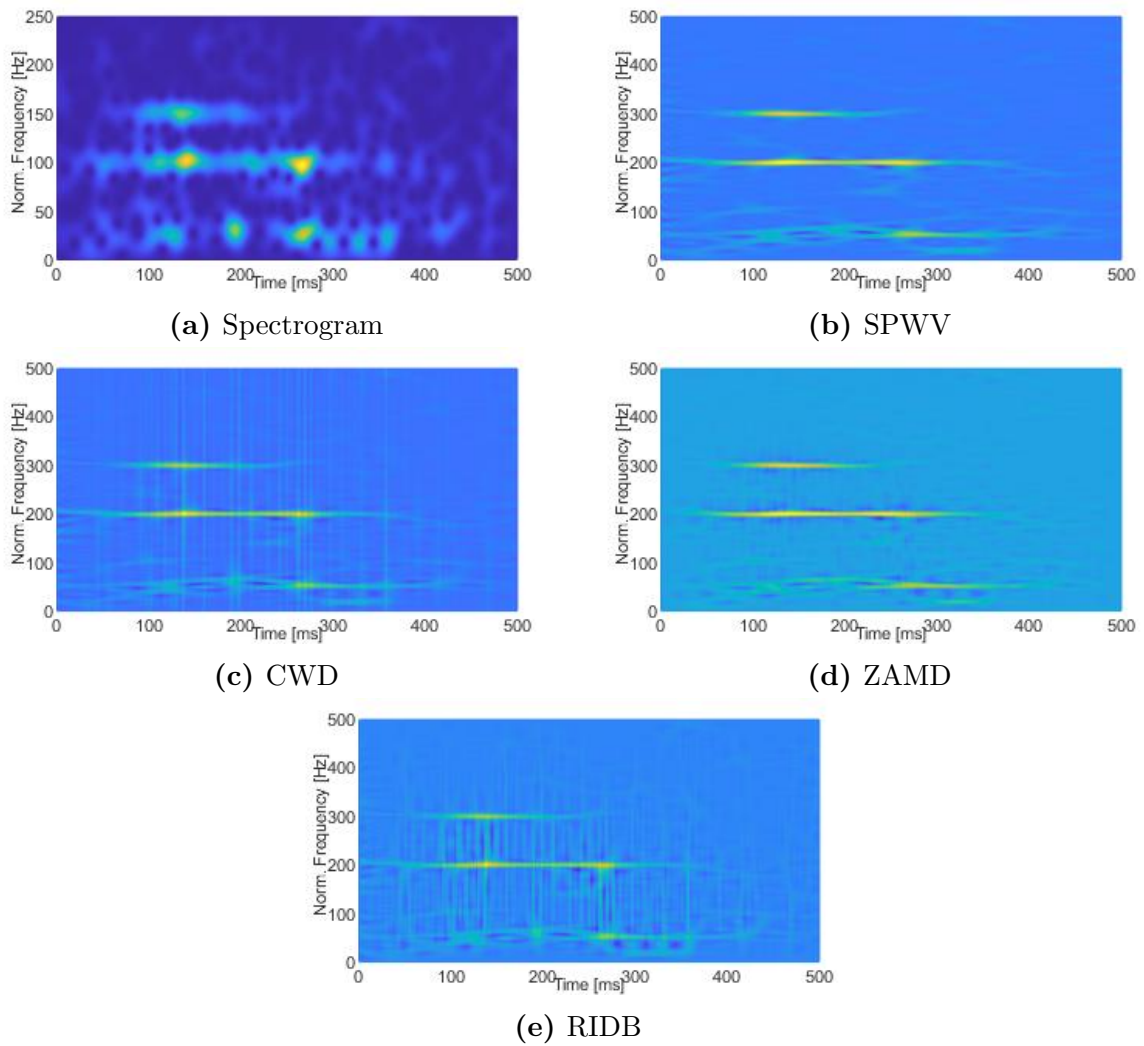
As outlined in Section 6.1.1, the initial experiment involved three synthetic signals augmented with varied levels of artificially introduced noise across different signal-to-noise ratios (SNRs).

Each signal underwent analysis across three distinct SNR levels, evaluating five different TFDs. The resulting TFDs for Signal A at  $\text{SNR} = -3$  are depicted in Figure 7.1, while  $\text{SNR} = 0$  and  $\text{SNR} = 5$  are illustrated in Figures 7.2 and 7.3, respectively.



**Figure 7.1:** Noisy TFDs of the signal A with  $\text{SNR} = -3$

Furthermore, Figure 7.4 exhibits the TFDs for Signal B at  $\text{SNR} = -3$ , with the representations for  $\text{SNR} = 0$  and  $\text{SNR} = 5$  displayed in Figures 7.5 and 7.6, correspondingly.



**Figure 7.2:** Noisy TFDs of the signal A with  $\text{SNR} = 0$

Lastly, Figure 7.7 portrays the TFDs for Signal C at  $\text{SNR} = -3$ , followed by the depictions for  $\text{SNR} = 0$  in Figure 7.8 and  $\text{SNR} = 5$  in Figure 7.9.

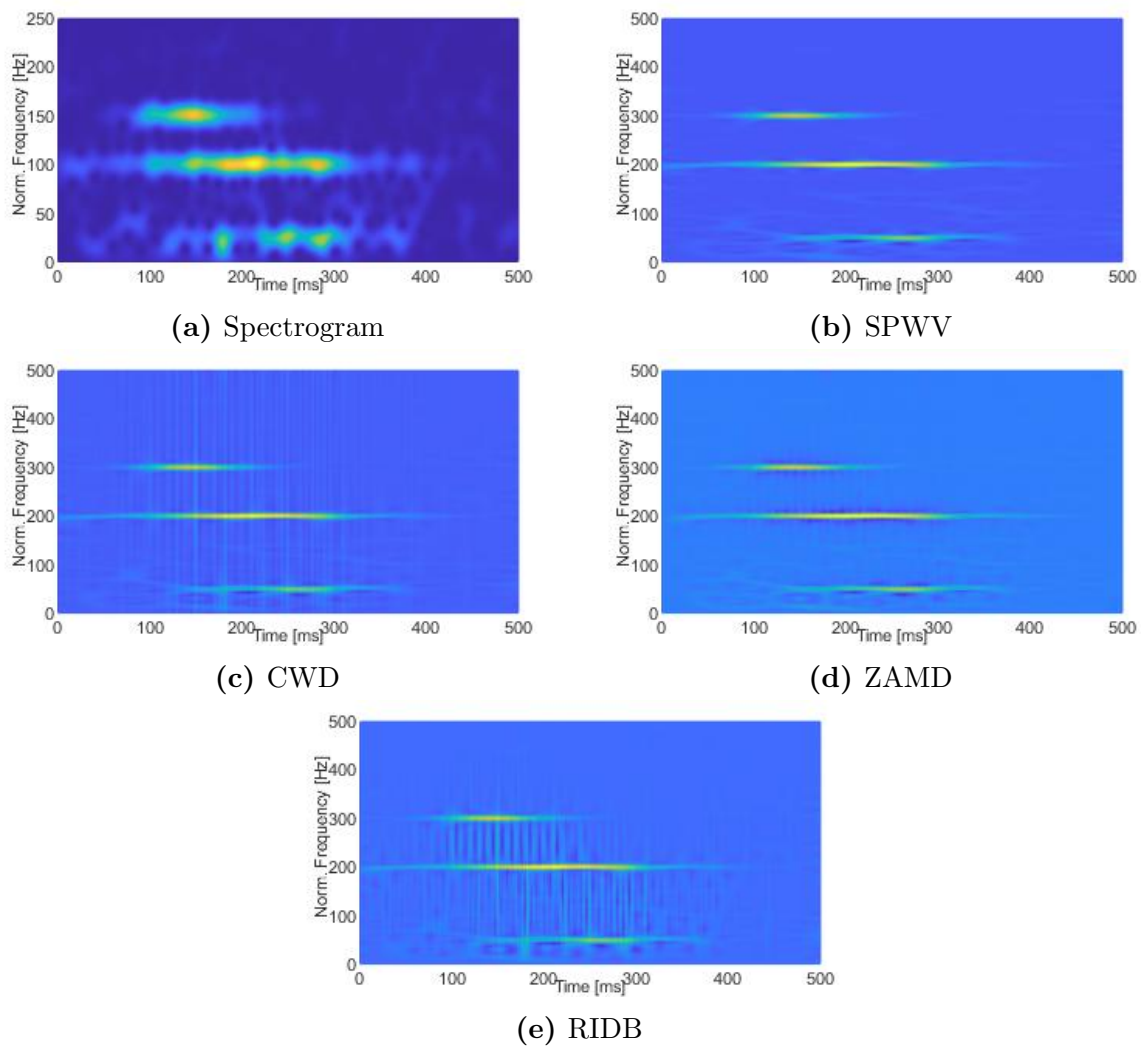
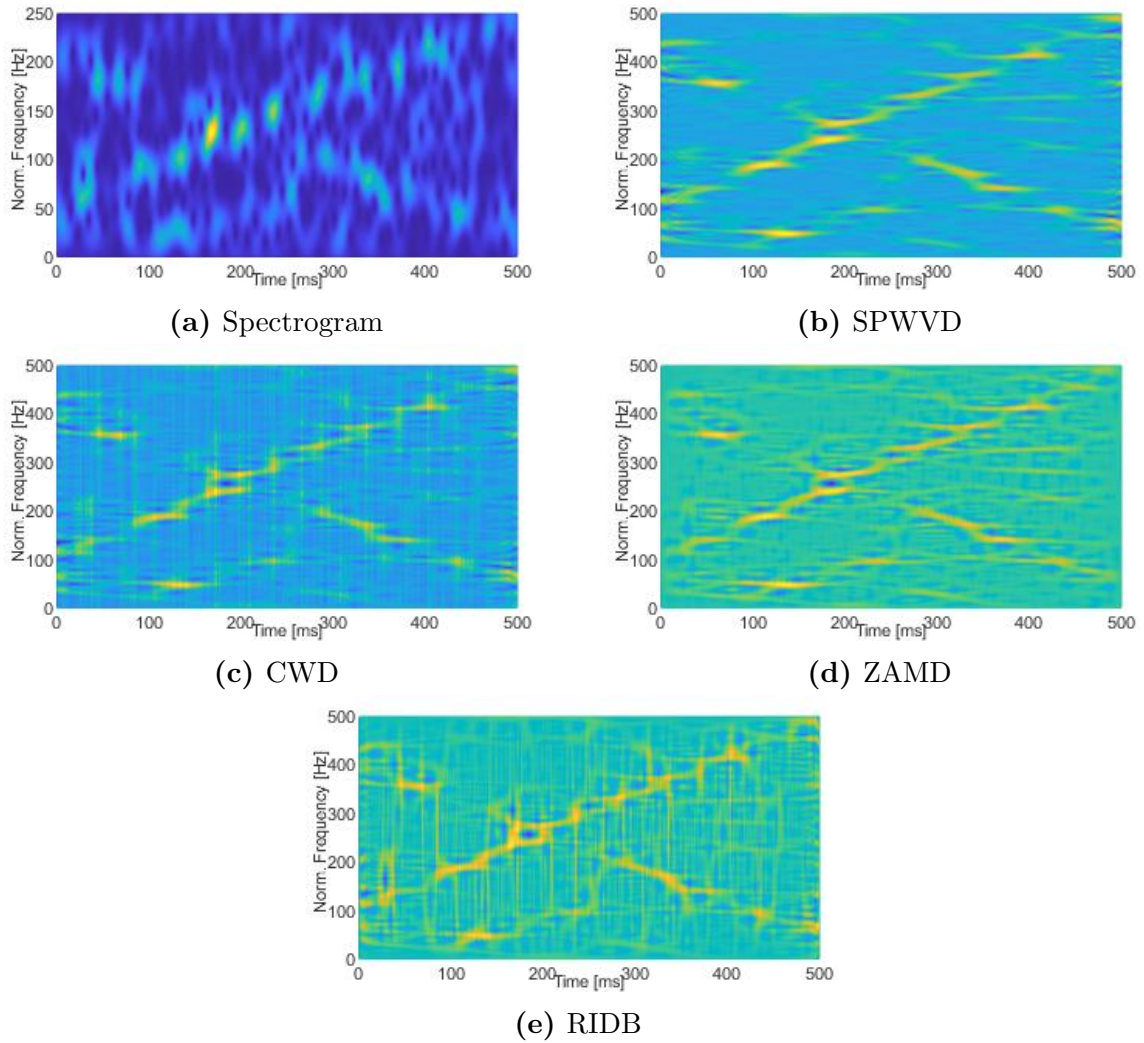


Figure 7.3: Noisy TFDs of the signal A with SNR = 5



**Figure 7.4:** Noisy TFDs of the signal B with  $\text{SNR} = -3$



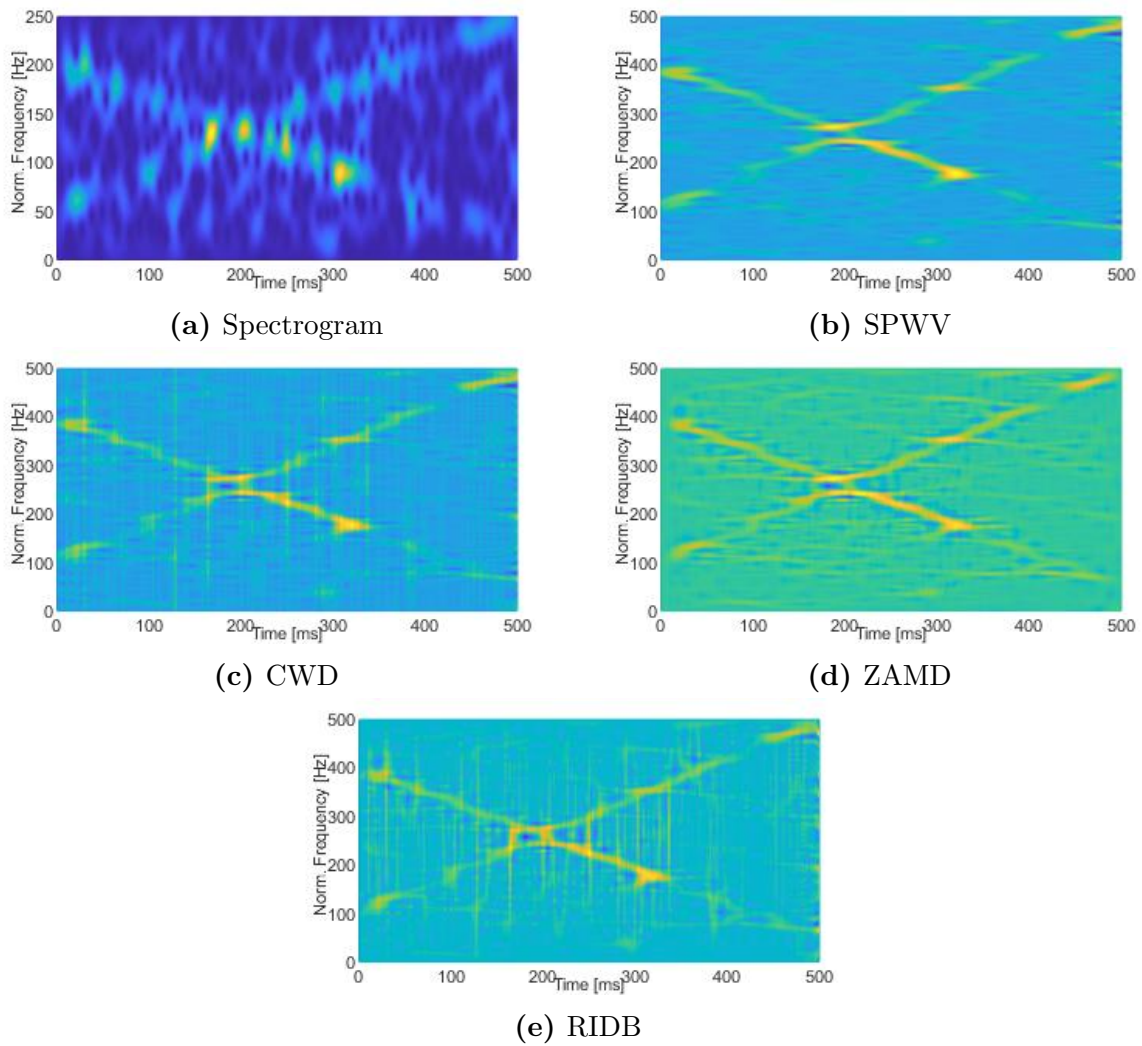
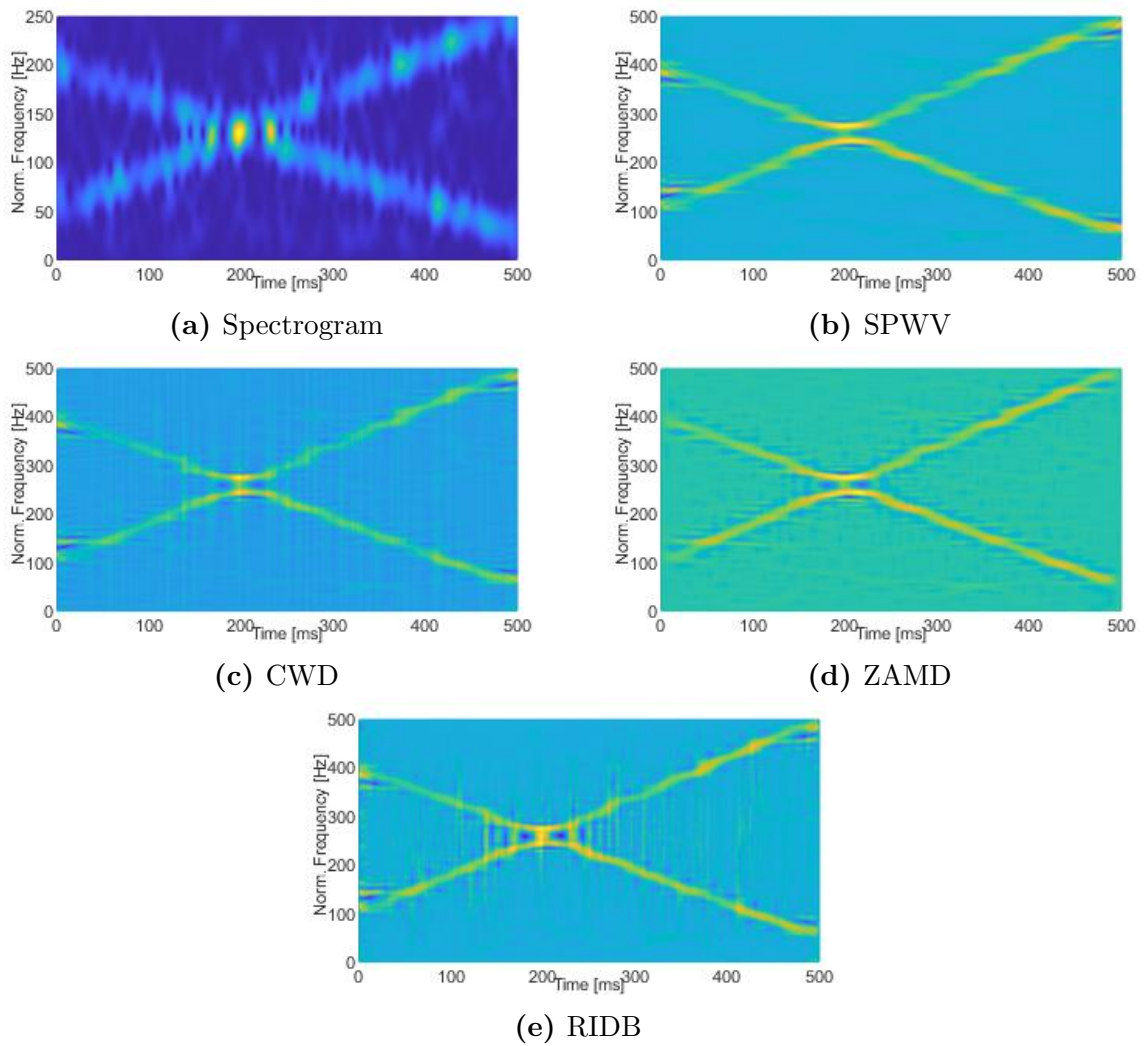


Figure 7.5: Noisy TFDs of the signal B with  $\text{SNR} = 0$



**Figure 7.6:** Noisy TFDs of the signal B with  $\text{SNR} = 5$

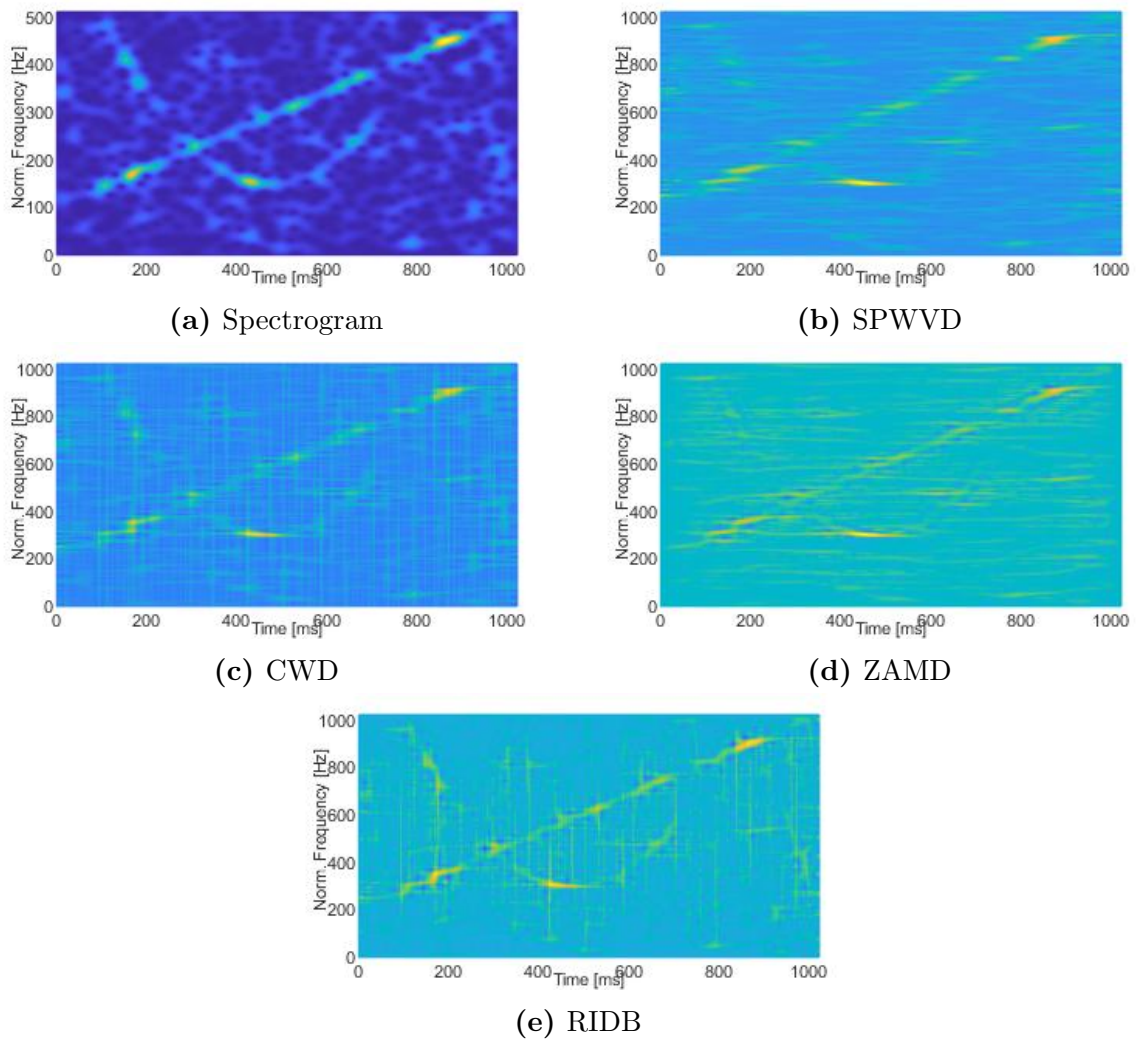
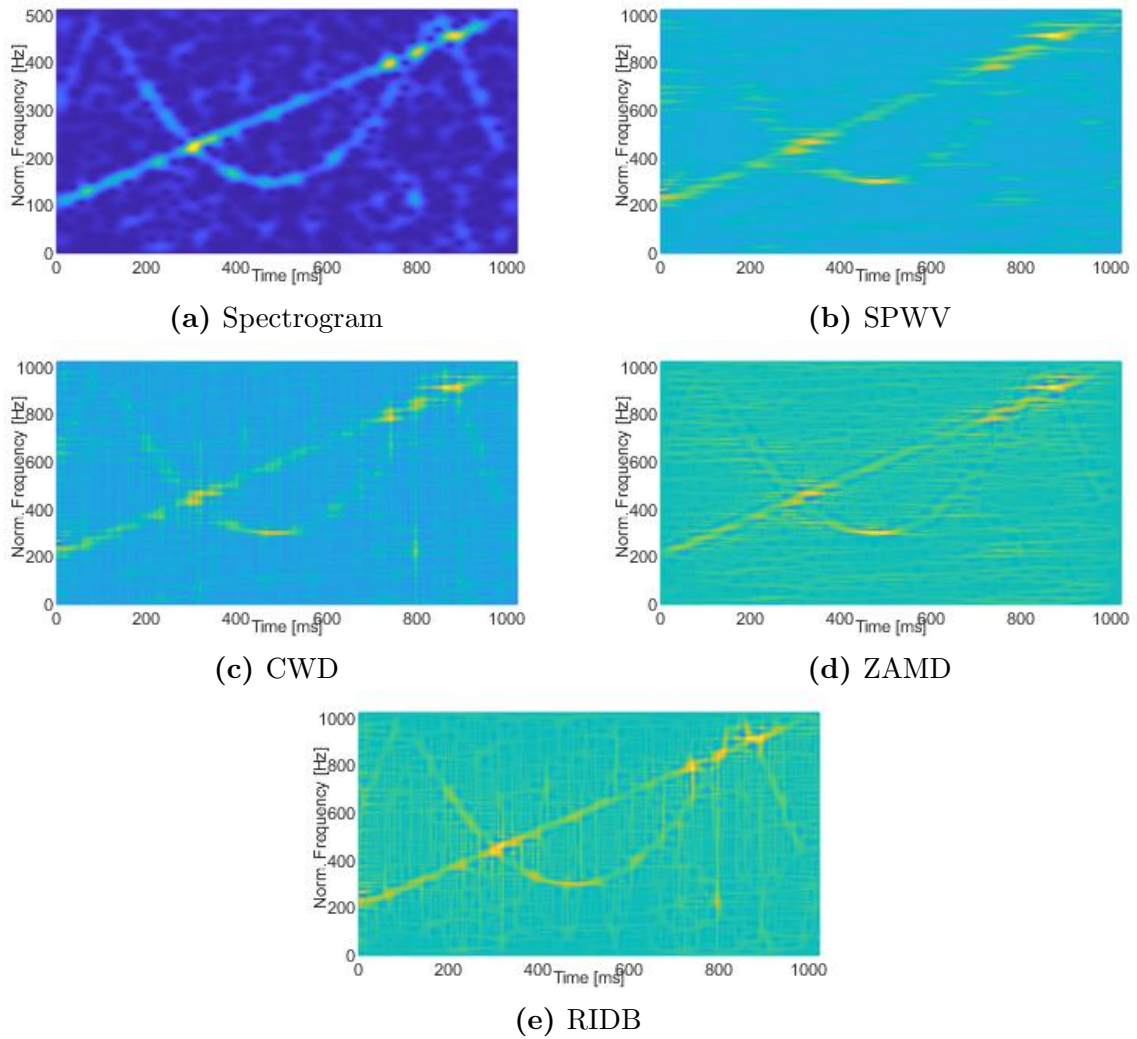


Figure 7.7: Noisy TFDs of the signal C with SNR = -3



**Figure 7.8:** Noisy TFDs of the signal C with  $\text{SNR} = 0$

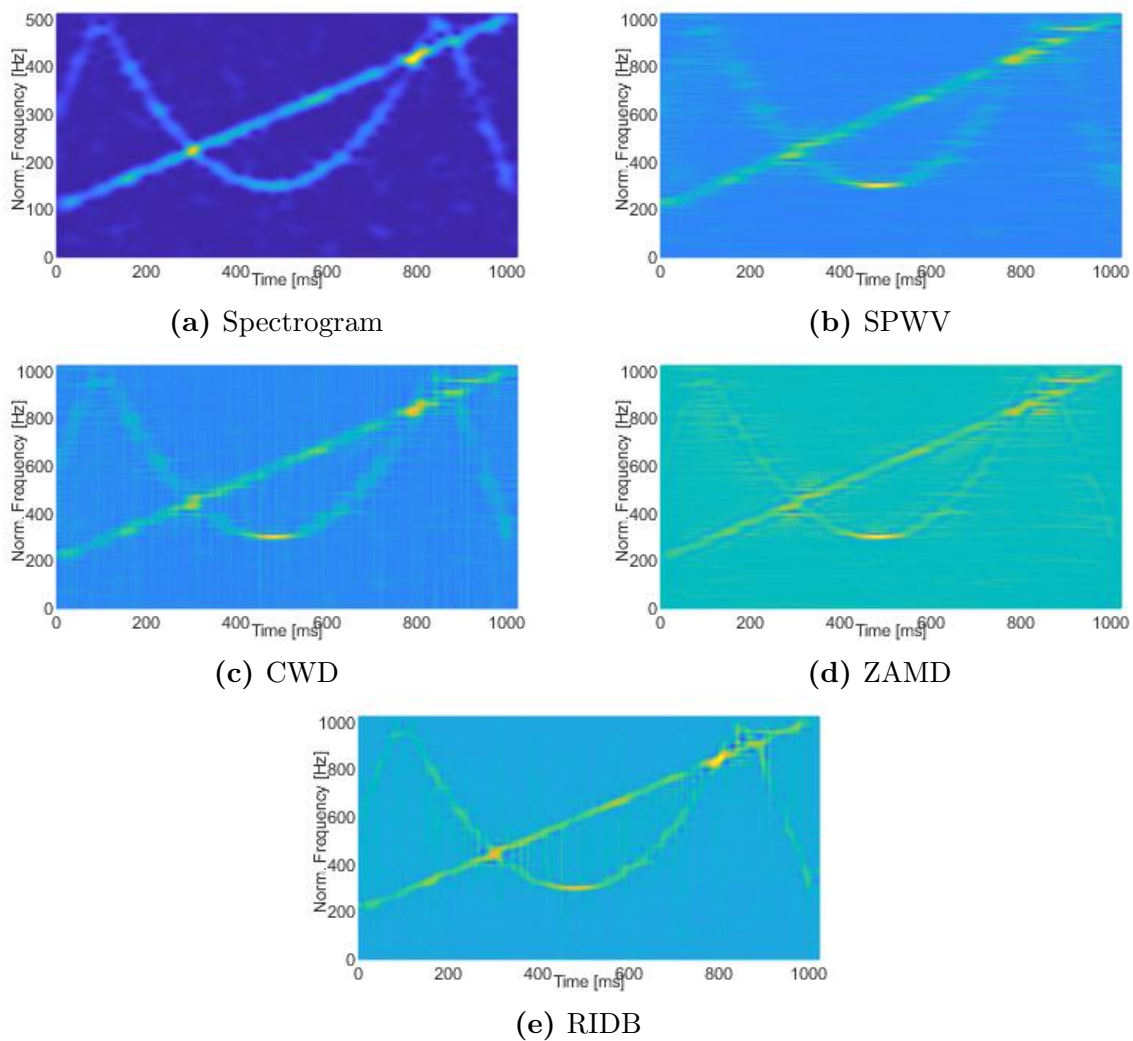


Figure 7.9: Noisy TFDs of the signal C with SNR = 5

Signals shown in Figures 7.1, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 7.8 and 7.9 were used as input to the proposed method with different entropy measures and the results are described in subsections 7.1.2, 7.1.3, 7.1.4 and 7.1.5. In section 7.1.6 comparison of the obtained results is given.

### 7.1.2. 2DLEM with Shannon entropy

Initially, Shannon entropy will be used in combination with a variety of signals, TFDs, and SNRs. Table 7.1 shows the classification results obtained with the Shannon entropy for signal A under different SNR and different time-frequency representations. Metrics include the accuracy and F1 score for each combination of SNR and time-frequency representation, illustrating the performance of the method. The highest values for accuracy and F1 score in each SNR category are shown in bold.

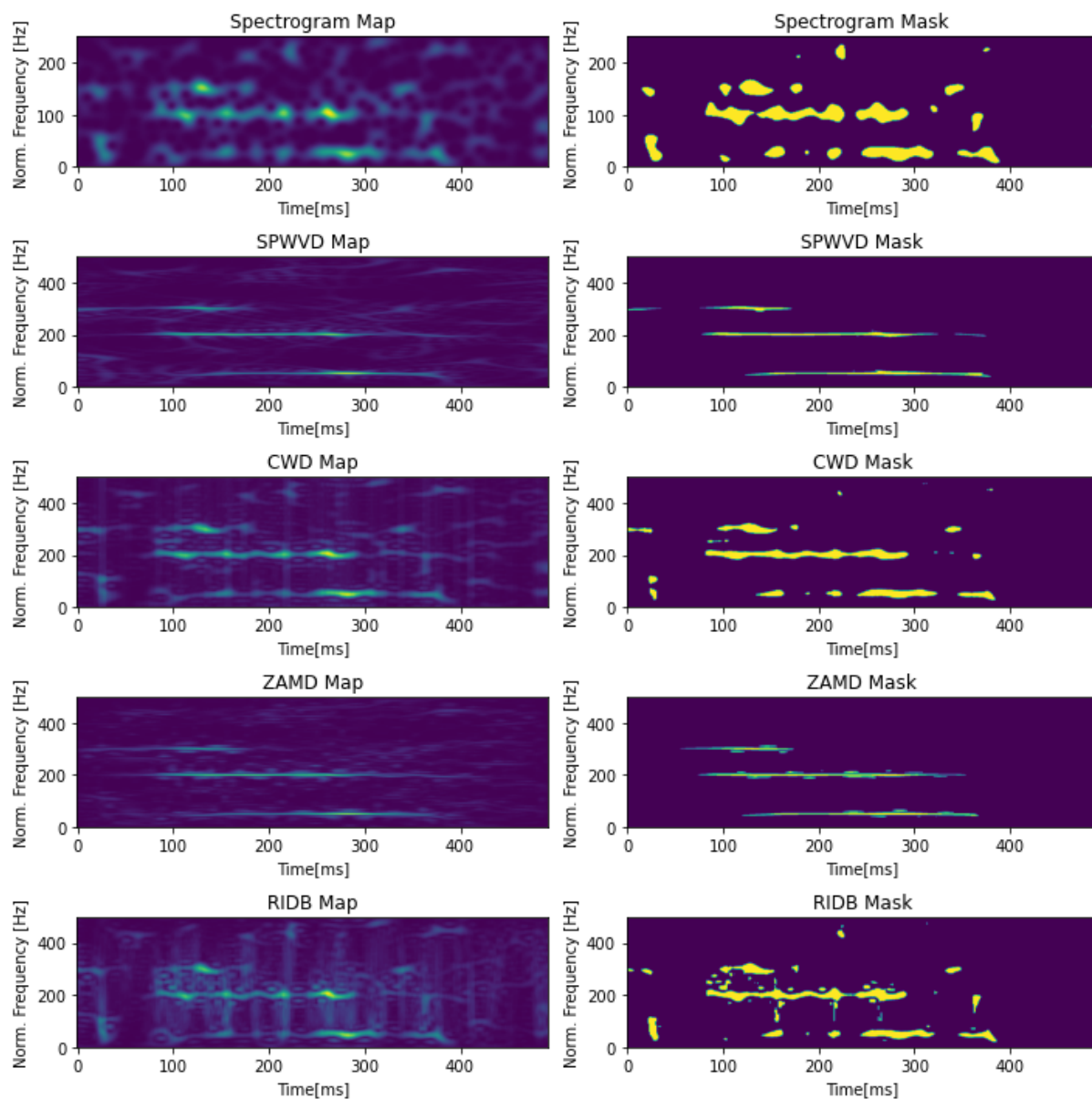
For high noise (SNR=-3), the ZAMD had the highest accuracy value of 97.3%, while the spectrogram had the highest F1 score at 98.7%. As the noise decreased (SNR=0), RIDB outperformed the other distributions, achieving the highest accuracy of 99.4% and an F1 score of 88.9%. In the least noisy scenario (SNR=5), RIDB still had the highest values with an accuracy of 99.8% and an F1 score of 96.1%.

A visual representation of both the entropy map and the entropy mask for each distribution can be found in Figure 7.10.

Signal A	SNR=-3, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.973	0.971	0.968	<b>0.987</b>	0.972
F1 Score	<b>0.828</b>	0.779	0.767	0.827	0.809
	SNR=0, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.976	0.985	0.984	0.991	<b>0.994</b>
F1 Score	0.856	0.820	0.808	0.874	<b>0.889</b>
	SNR=5, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.987	0.997	0.997	0.997	<b>0.998</b>
F1 Score	0.927	0.938	0.954	0.937	<b>0.961</b>

**Table 7.1:** Results obtained with Shannon entropy from signal A

The results for signal B are detailed in Table 7.2 and shown in Figure 7.11. The spectrogram showed remarkable consistency and secured the highest accuracy and F1



**Figure 7.10:** Entropy maps and masks showing signal components for different TFDs for Shannon entropy and SNR = -3 in case of signal A

values across different SNRs, except for SNR = 0, where CWD had a slightly higher F1 value of 85.3%. Its robust performance at different noise levels underlines the reliability of the spectrogram for Signal B. A parallel trend can be observed in the results for signal C,

<b>Signal B</b>	SNR=-3, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.938</b>	0.937	0.919	0.925	0.928
F1 Score	<b>0.795</b>	0.775	0.785	0.673	0.721
	SNR=0, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.963</b>	<b>0.963</b>	0.953	0.943	0.938
F1 Score	0.831	0.848	<b>0.853</b>	0.738	0.794
	SNR=5, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.979</b>	0.959	0.966	0.948	0.963
F1 Score	<b>0.946</b>	0.871	0.922	0.813	0.883

**Table 7.2:** Results obtained with Shannon entropy from signal B

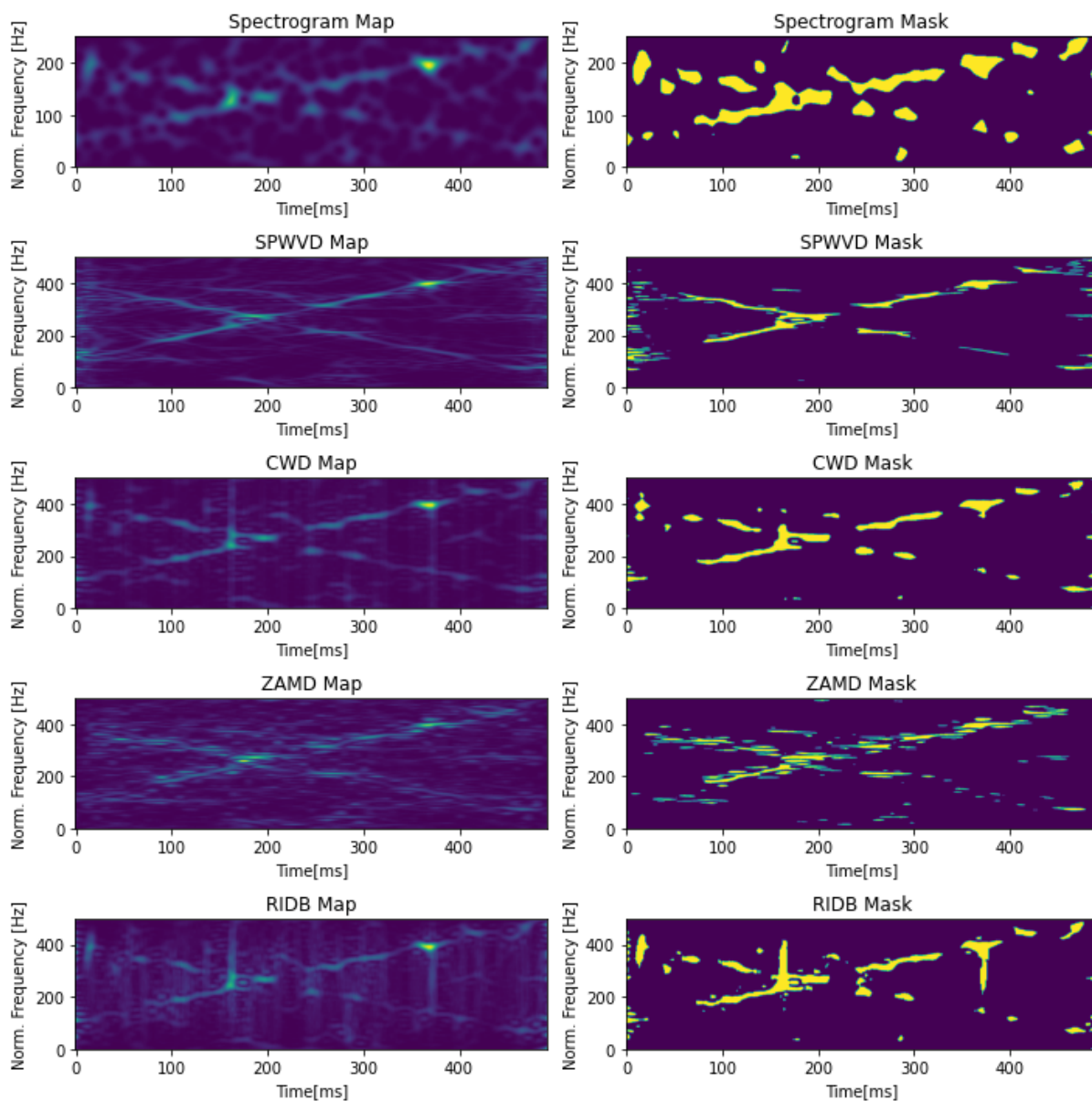
which are shown in Table 7.3 and Figure 7.12. The spectrogram outperformed the other distributions in both accuracy and F1 results, except for accuracy in the highest noise scenario, where SPWV outperformed with 96.6%. An Analysis of variance(ANOVA) was

<b>Signal C</b>	SNR=-3, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.946	<b>0.966</b>	0.952	0.921	0.941
F1 Score	<b>0.783</b>	0.742	0.750	0.625	0.711
	SNR=0, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.952</b>	0.942	0.939	0.924	0.950
F1 Score	<b>0.863</b>	0.754	0.792	0.674	0.771
	SNR=5, Shannon entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.976</b>	0.959	0.967	0.929	0.964
F1 Score	<b>0.918</b>	0.830	0.896	0.739	0.861

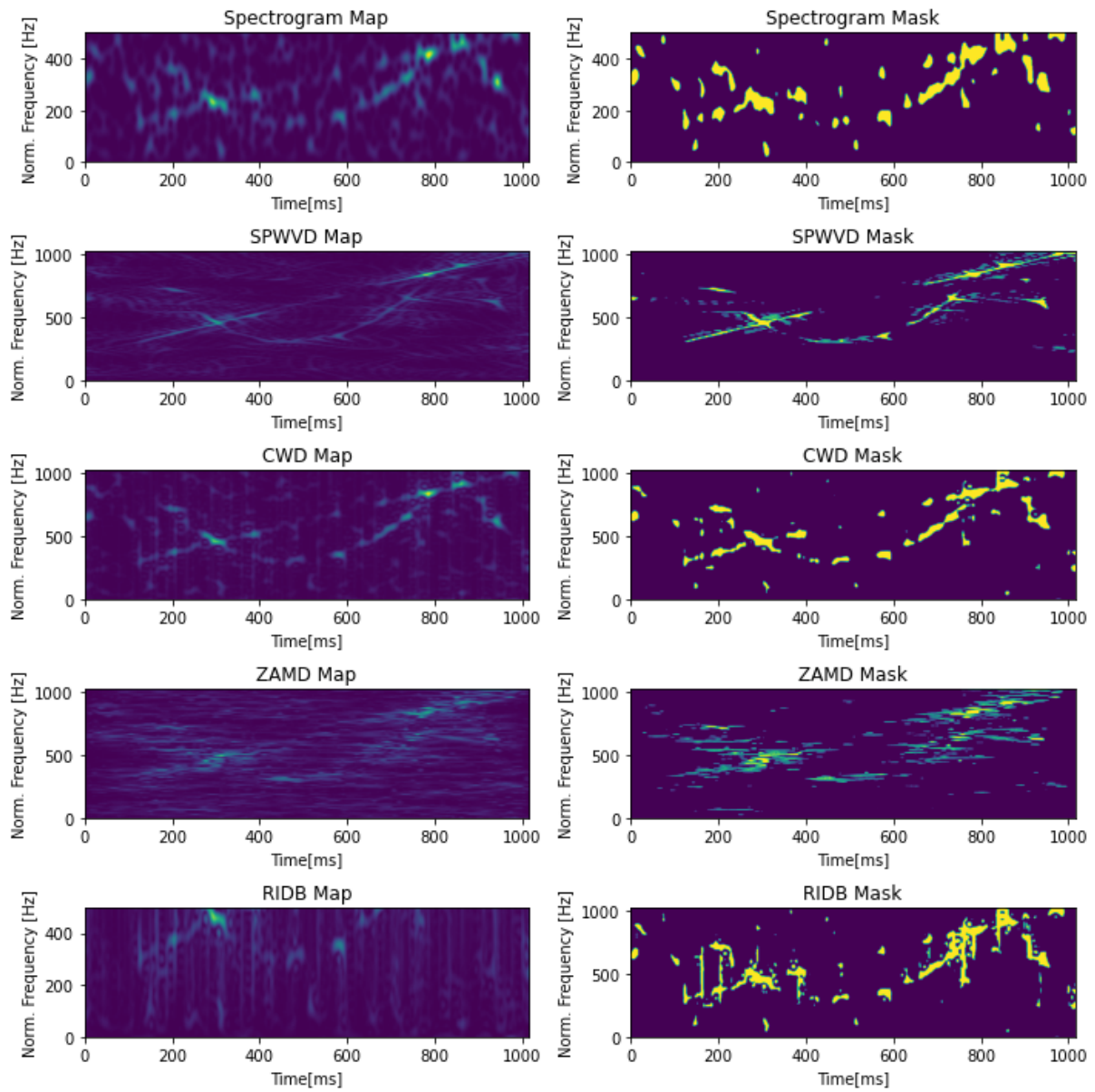
**Table 7.3:** Results obtained with Shannon entropy from signal C

performed to determine whether there was a statistically significant difference between the different TFDs. Prior to this, the Shapiro test( $p > 0.05$ ) was performed to ensure the normality of the results obtained. The p-value obtained in the ANOVA analysis was lower than 0.05, so the conclusion was that there was at least one significant difference between the TFDs. From the ANOVA analysis, we know that the differences are statistically





**Figure 7.11:** Entropy maps and masks showing signal components for different TFDs for Shannon entropy and SNR = -3 in case of signal B



**Figure 7.12:** Entropy maps and masks showing signal components for different TFDs for Shannon entropy and  $\text{SNR} = -3$  in case of signal C

significant, but it does not tell us which TFDs are significantly different from each other. To identify the pairs with significantly different TFDs, multiple pairwise comparison (post hoc comparison) analysis was performed for all pairs using Tukey’s honestly significant difference (HSD) test. Tukey’s honestly significant difference test performs a pairwise comparison of means for a set of samples. While ANOVA tests whether the true means underlying each sample are identical, Tukey’s HSD is a post-hoc test used to compare the mean of each sample with the mean of every other sample. The p-value of the pairwise comparison is shown in Table 7.4 for signal A in the case of SNR=-3. The null hypothesis is that each group has the same mean value. The p-value for the comparisons between most TFDs does not exceed 0.05, so the null hypothesis that they have the same mean is rejected. The p-value of the comparison between SPWV and RIDB exceeds 0.05, so we accept the null hypothesis that there is no significant difference between their means.

Tukey’s HSD test	SigA (SNR=-3)				
p-value	Spec	SPWV	CW	ZAM	RIDB
Spec	/	< .001	< .001	< .001	< .001
SPWV	< .001	/	.01	< .001	.812
CW	< .001	.01	/	< .001	.050
ZAM	< .001	< .001	< .001	/	< .001
RIDB	< .001	.812	.050	< .001	/

**Table 7.4:** Results of pair-wise comparison between different TFDs for signal A when SNR=-3

The p-value of the pairwise comparison is shown in table 7.5 for signal A in the case of SNR=0, and table 7.6 for SNR=5. In the case of SNR=0, there was only no difference between PSWVD and CWD, with a p-value of 0.149.

Tukey’s HSD test	SigA (SNR=0)				
p-value	Spec	SPWV	CW	ZAM	RIDB
Spec	/	< .001	< .001	< .001	< .001
SPWV	< .001	/	.149	< .001	< .001
CW	< .001	.149	/	< .001	< .001
ZAM	< .001	< .001	< .001	/	< .001
RIDB	< .001	< .001	< .001	< .001	/

**Table 7.5:** Results of pair-wise comparison between different TFDs for signal A when SNR=0

For SNR=5, results were different; there was only a statistically significant difference

between the spectrogram and all the other distributions.

Tukey's HSD test	SigA (SNR=5)				
p-value	Spec	SPWV	CW	ZAM	RIDB
Spec	/	< .001	< .001	< .001	< .001
SPWV	< .001	/	.977	.816	.328
CW	< .001	.977	/	.988	.694
ZAM	< .001	.816	.988	/	.930
RIDB	< .001	.328	.694	.930	/

**Table 7.6:** Results of pair-wise comparison between different TFDs for signal A when SNR=5

For signal B, there was no difference between CW and RIDB (p-value = .670) and between RIDB and ZAM (p-value= .249) when SNR = -3. For SNR = 0, there was a difference between all TFDs except spectrogram and SPWVD(p-value = .552), and for SNR=5 there was a difference between all tested distributions.

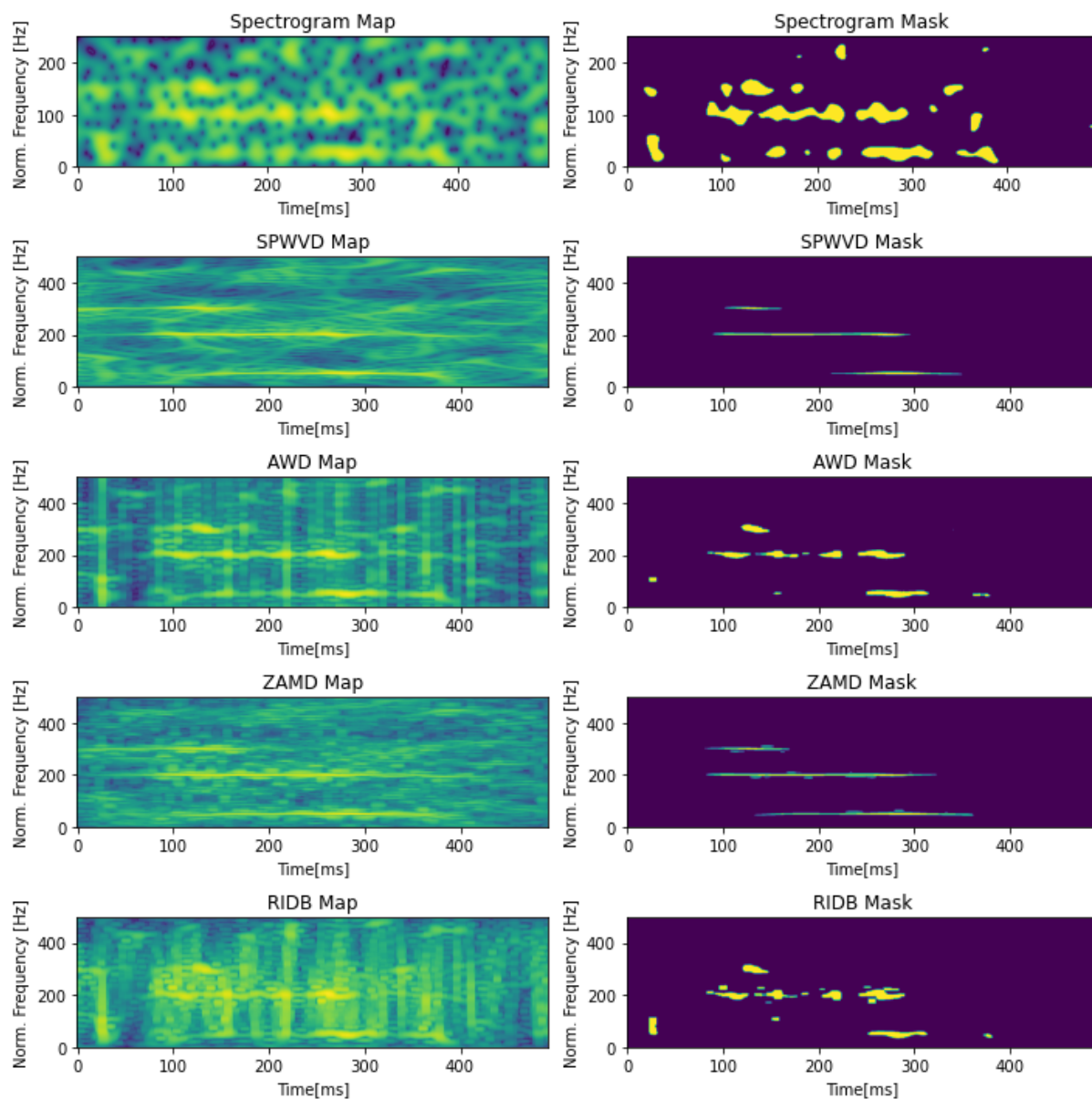
For signal C, there was a significant difference in SNR for all TFDs tested.

### 7.1.3. 2DLEM with Rényi entropy

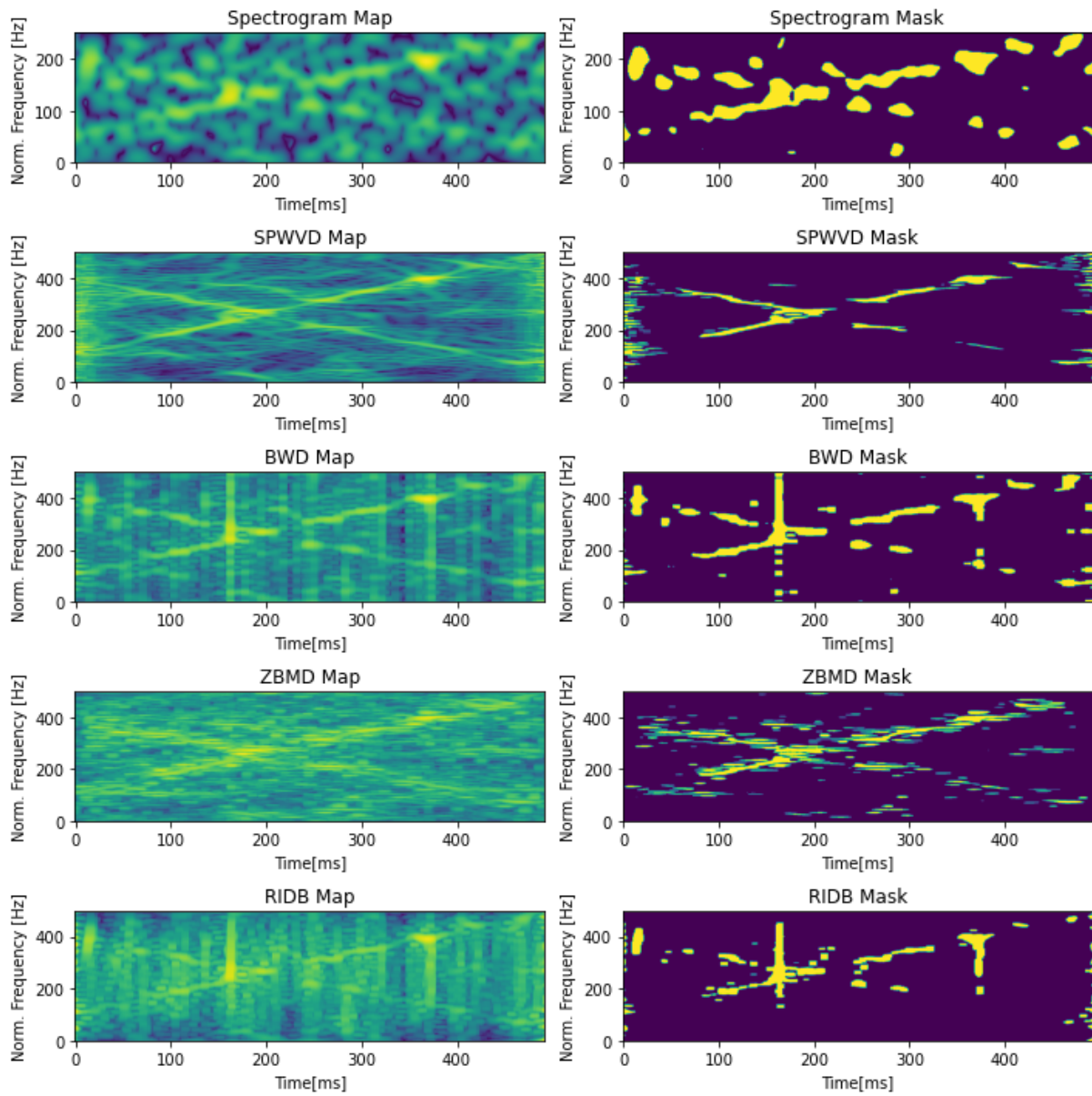
For the same synthetic signals with different TFDs and different SNRs, the 2DLEM method was used again, but in combination with Rényi entropy instead of Shannon entropy.

Table 7.7 gives an overview of the results for the first signal, Signal A, while a visual example can be found in Figure 7.13. The results showed considerable variability. In contrast to the dominance of RIDB in the Shannon entropy case, SPWVD provided the highest values for SNR = -3, ZAMD outperformed the others for SNR = 0, and RIDB outperformed for SNR = 5.

An interesting and important observation is that Rényi entropy consistently outperformed the results obtained with Shannon entropy in all scenarios. For signal B in Table 7.8, the results are again different, but with a marginal difference between the spectrogram and the distribution with the highest metrics. In the case of SNR = -3, this is SPWVD, and in the case of the F1 score when SNR = 5, the distribution is CWD. For SNR = 0, the spectrogram has both the highest accuracy and the highest F1 score. An example can be seen in Figure 7.14.



**Figure 7.13:** Entropy maps and masks for different TFDs for Rényi entropy and SNR = -3 in case of signal A



**Figure 7.14:** Entropy maps and masks for different TFDs for Rényi entropy and  $\text{SNR} = -3$  in case of signal B

<b>Signal A</b>	SNR=-3, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.980	<b>0.989</b>	0.981	<b>0.989</b>	0.975
F1 Score	0.826	<b>0.890</b>	0.850	0.857	0.761
	SNR=0, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.971	0.992	0.980	<b>0.997</b>	0.996
F1 Score	0.863	0.910	0.837	<b>0.942</b>	0.927
	SNR=5, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.990	0.997	0.996	<b>0.998</b>	<b>0.998</b>
F1 Score	0.923	0.931	0.937	0.953	<b>0.964</b>

**Table 7.7:** Results obtained with Rényi entropy from signal A

<b>Signal B</b>	SNR=-3, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.938	<b>0.940</b>	0.926	0.912	0.927
F1 Score	0.785	<b>0.786</b>	0.756	0.686	0.687
	SNR=0, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.956</b>	0.953	0.938	0.912	0.928
F1 Score	<b>0.879</b>	0.850	0.845	0.762	0.747
	SNR=5, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.976</b>	0.972	0.967	0.933	0.956
F1 Score	0.895	0.894	<b>0.906</b>	0.817	0.861

**Table 7.8:** Results obtained with Rényi entropy from signal B

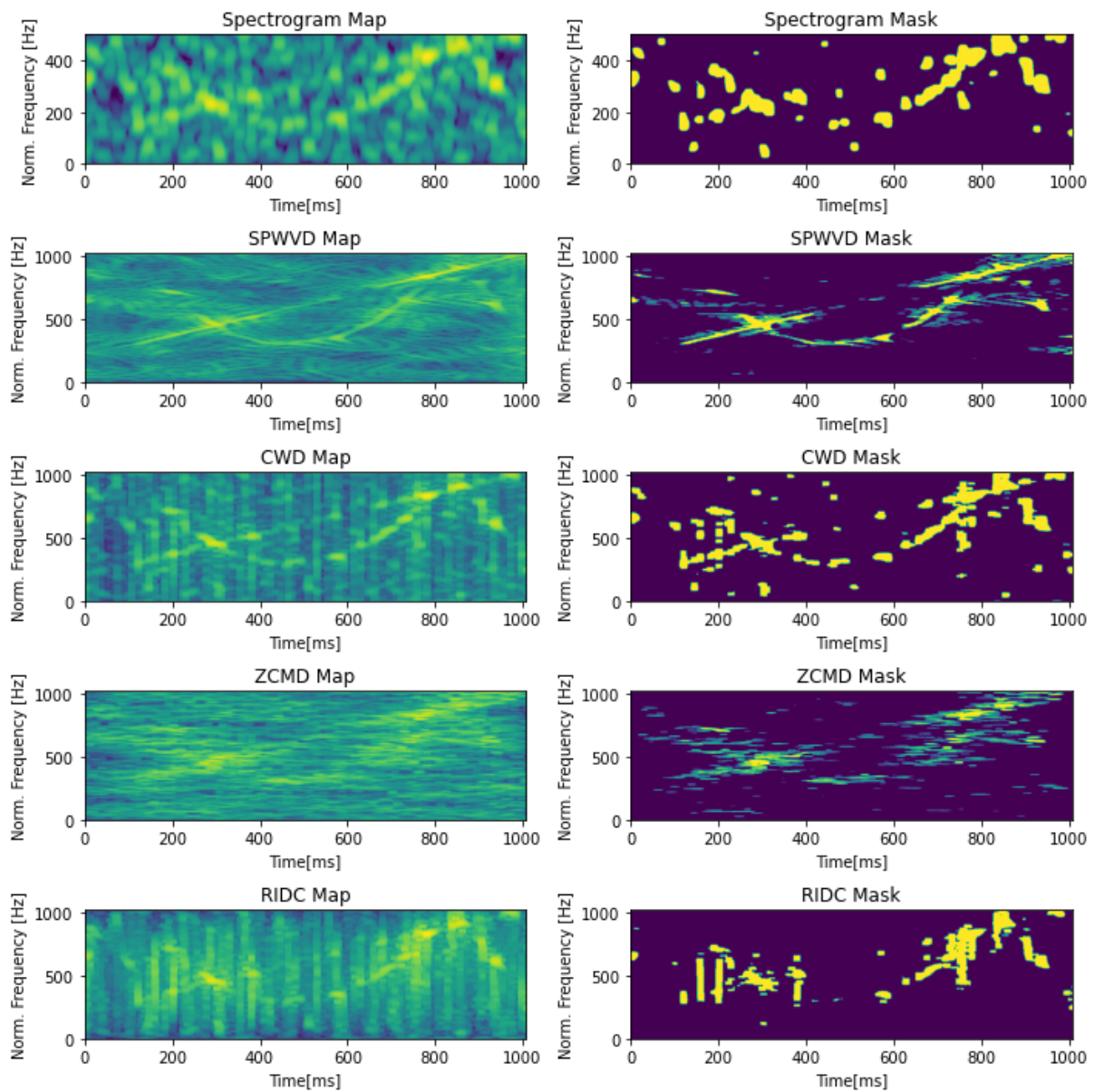
Table 7.9 describes the results for signal C. An example can be seen in Figure 7.15. The spectrogram performs significantly better than the other distributions, regardless of the SNR. The results in this case differ slightly from those obtained with Shannon entropy.

ANOVA was also performed in the case of Rényi entropy. The test showed that there was a significant difference in all cases.

#### 7.1.4. 2DLEM with Tsallis entropy

Finally, the Tsallis entropy was used as part of the 2DLEM method for comparison with other classical entropy measures, Shannon’s and Rényi’s.

The results for the first signal are shown in the table 7.10. A visual illustration can be found in Figure 7.16. For SNR = -3, ZAMD had the highest values for both metrics, 99.4%



**Figure 7.15:** Entropy maps and masks for different TFDs for Rényi entropy and SNR = -3 in case of signal C



<b>Signal C</b>	SNR=-3, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.944</b>	0.931	0.917	0.888	0.937
F1 Score	<b>0.783</b>	0.725	0.738	0.612	0.673
	SNR=0, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.953</b>	0.921	0.937	0.903	0.934
F1 Score	<b>0.866</b>	0.753	0.780	0.678	0.729
	SNR=5, Rényi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.969</b>	0.928	0.950	0.904	0.935
F1 Score	<b>0.919</b>	0.803	0.859	0.726	0.815

**Table 7.9:** Results obtained with Rényi entropy from signal C

accuracy and 89.9% F1 score. ZAMD for SNR = 0 also had the highest accuracy (99.7%), but SPWVD had the highest F1 score (94.3%). For SNR = 5, SPWVD outperformed the other distributions with an accuracy of 99.9% and an F1 score of 96.6%. Results

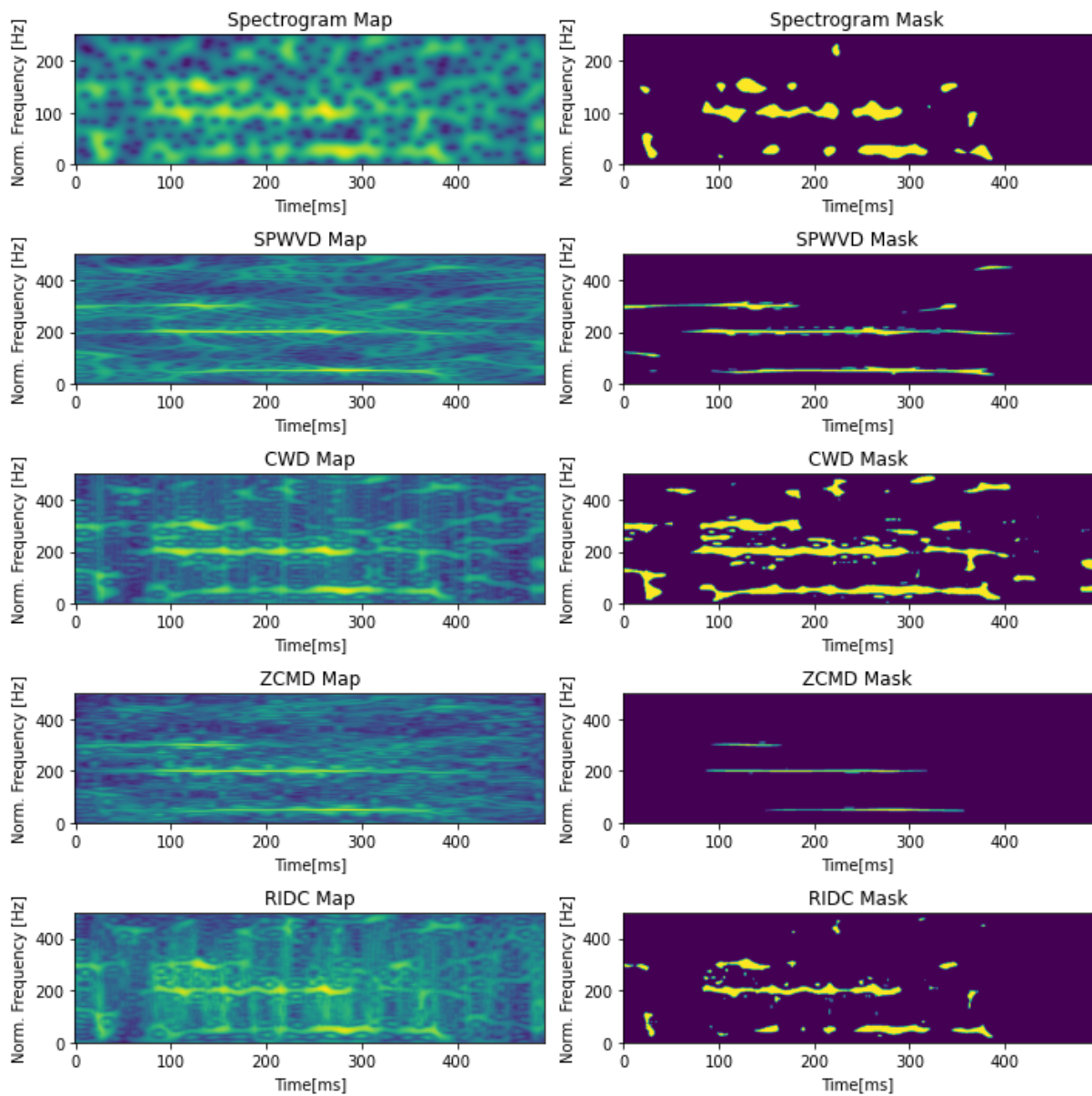
<b>Signal A</b>	SNR=-3, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.967	0.988	0.972	<b>0.994</b>	0.985
F1 Score	0.815	0.882	0.837	<b>0.899</b>	0.810
	SNR=0, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.973	0.995	0.975	<b>0.997</b>	0.988
F1 Score	0.844	<b>0.943</b>	0.830	0.938	0.828
	SNR=5, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.990	<b>0.999</b>	0.987	0.996	0.995
F1 Score	0.921	<b>0.966</b>	0.830	0.949	0.938

**Table 7.10:** Results obtained with Tsallis entropy from signal A

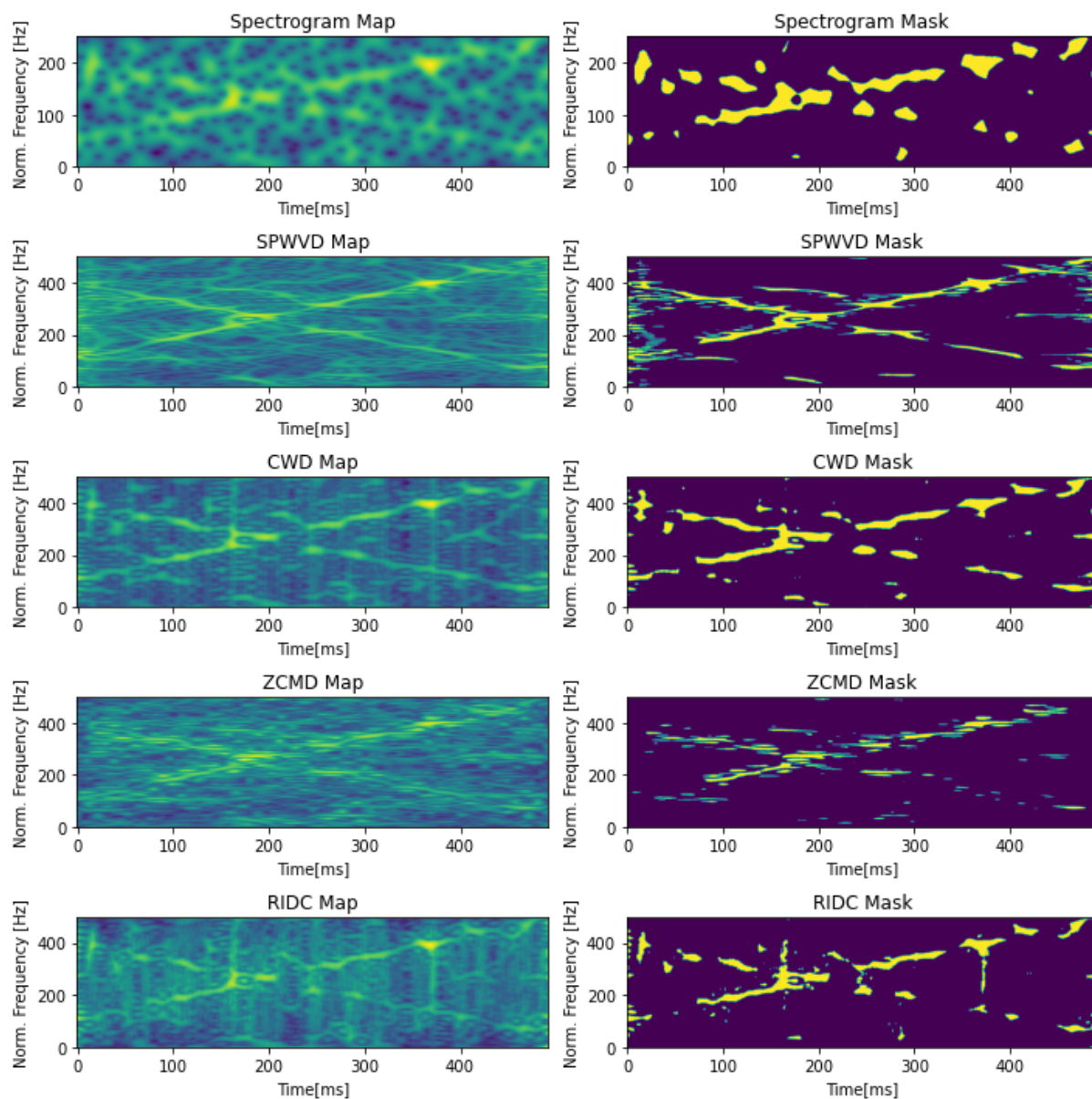
for signal B are given in table 7.11 with a visual example in Figure 7.17. Results are once again diverse, but in most cases, the spectrogram has the highest score (in 4 out of 6). Results in all cases are lower than in the case of both Shannon and Rényi's entropy measure.

For signal C (Table 7.12 and Figure 20) the Spectrogram again clearly outperforms all the other distributions regardless of SNR value with great difference in metrics.

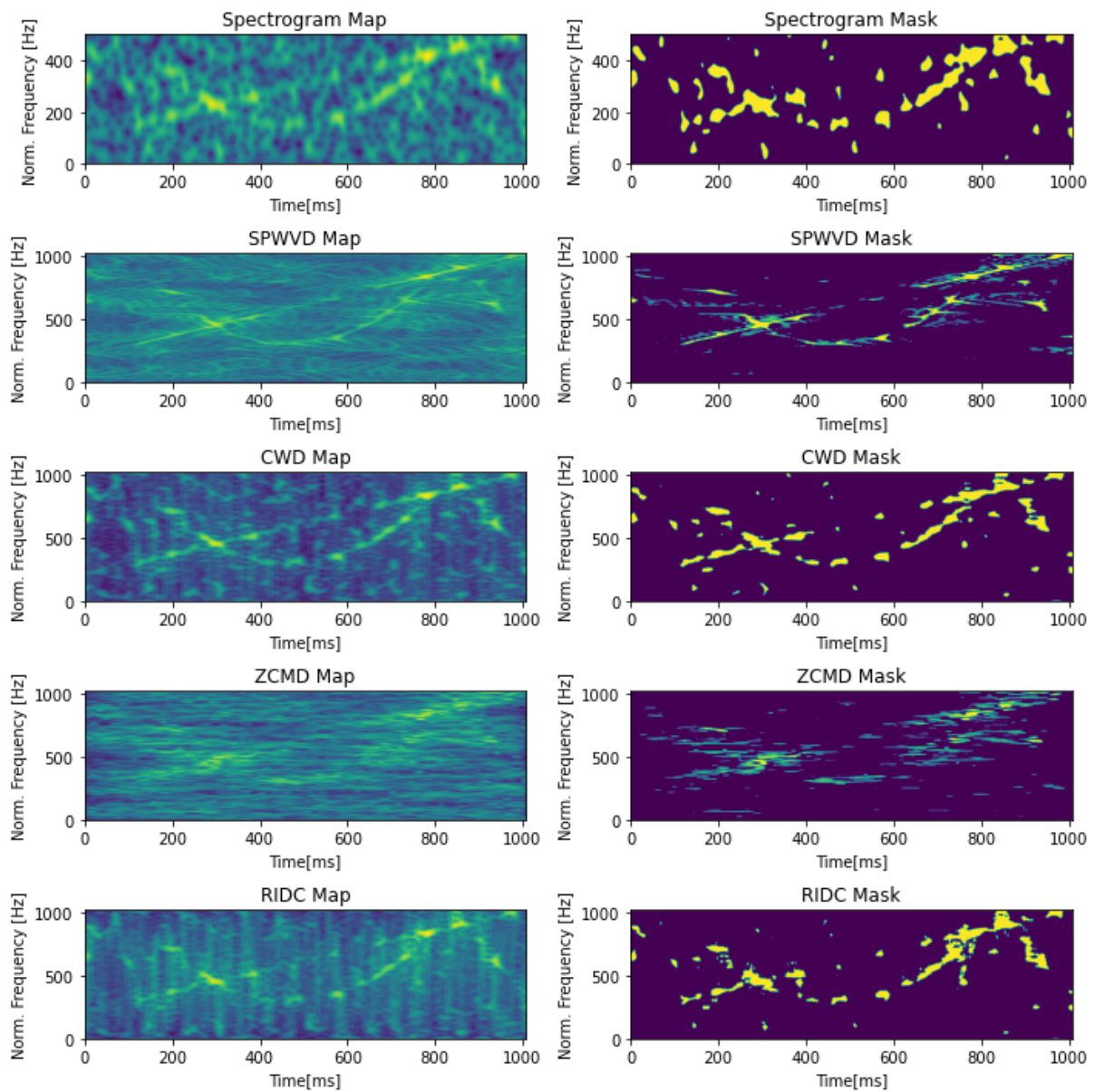
Statistical tests were also performed in the case of Tsallis entropy. The test showed that there was a significant difference in all cases ( $p < .001$ ).



**Figure 7.16:** Entropy maps and masks for different TFDs for Tsallis entropy and SNR = -3 in case of signal A



**Figure 7.17:** Entropy maps and masks for different TFDs for Tsallis entropy and SNR = -3 in case of signal B



**Figure 7.18:** Entropy maps and masks for different TFDs for Tsallis entropy and SNR = -3 in case of signal C

<b>Signal B</b>	SNR=-3, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.940	<b>0.944</b>	0.930	0.920	0.921
F1 Score	<b>0.788</b>	0.770	0.790	0.676	0.743
	SNR=0, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.963</b>	0.953	0.952	0.923	0.936
F1 Score	0.859	0.833	<b>0.862</b>	0.813	0.742
	SNR=5, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.979</b>	0.972	0.973	0.956	0.963
F1 Score	<b>0.929</b>	0.896	0.925	0.815	0.895

**Table 7.11:** Results obtained with Tsallis entropy from signal B

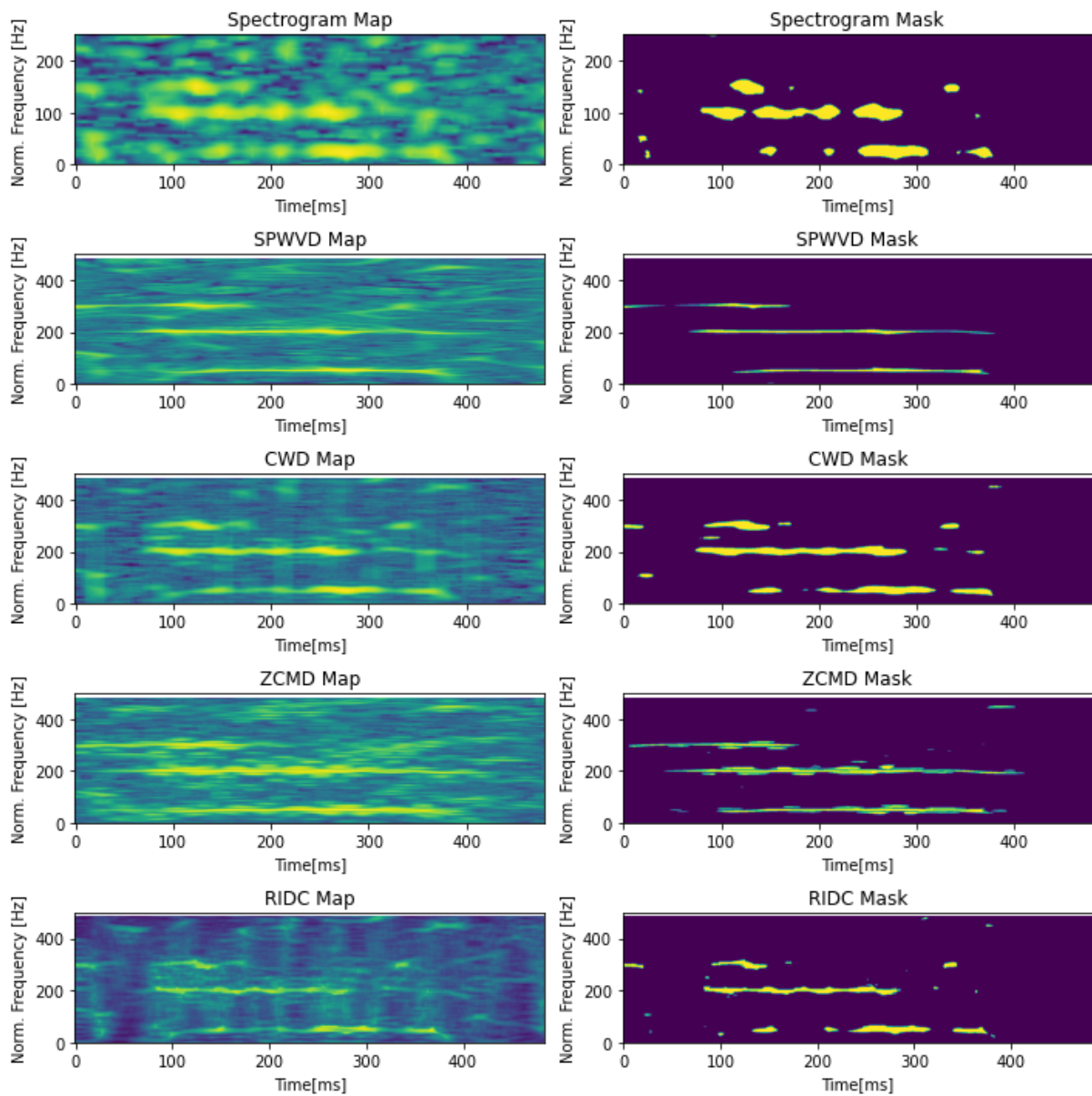
<b>Signal C</b>	SNR=-3, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.946</b>	0.910	0.933	0.905	0.923
F1 Score	<b>0.770</b>	0.676	0.760	0.612	0.713
	SNR=0, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.961</b>	0.940	0.932	0.901	0.932
F1 Score	<b>0.841</b>	0.743	0.811	0.656	0.777
	SNR=5, Tsallis entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.970</b>	0.934	0.945	0.897	0.911
F1 Score	<b>0.900</b>	0.782	0.863	0.713	0.804

**Table 7.12:** Results obtained with Tsallis entropy from signal C

### 7.1.5. 2DLEM with Dombi entropy

After combining the 2DLEM method with various TFDs and classical entropy measures, it was combined with the newly proposed Dombi entropy measure. The parameters for the Dombi entropy were selected empirically based on tests of different values and optimized based on the results. The results for  $\alpha = 2$  and  $\beta = 0.5$  are presented here. Table 7.13 shows the performance metrics obtained with the Dombi entropy for signal A at different SNRs (-3, 0 and 5) and different TFDs. The accuracy and F1 score metrics are shown for each combination of SNR and TFD. The highest accuracy and F1 scores were achieved with the ZAMD for all SNRs tested. A visual example for SNR = -3 is shown in Figure 7.19.

Table 7.14 showcases the performance metrics on signal B using Dombi entropy. A



**Figure 7.19:** Entropy maps and masks for different TFDs for Dombi entropy and SNR = -3 in case of signal A

<b>Signal A</b>	SNR = -3, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.972	0.991	0.987	<b>0.995</b>	0.981
F1 Score	0.860	0.901	0.866	<b>0.908</b>	0.786
	SNR=0, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.978	0.994	0.992	<b>0.997</b>	0.996
F1 Score	0.881	0.933	0.913	<b>0.937</b>	0.933
	SNR=5, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.988	0.999	0.997	<b>0.999</b>	0.997
F1 Score	0.922	0.957	0.958	<b>0.973</b>	0.955

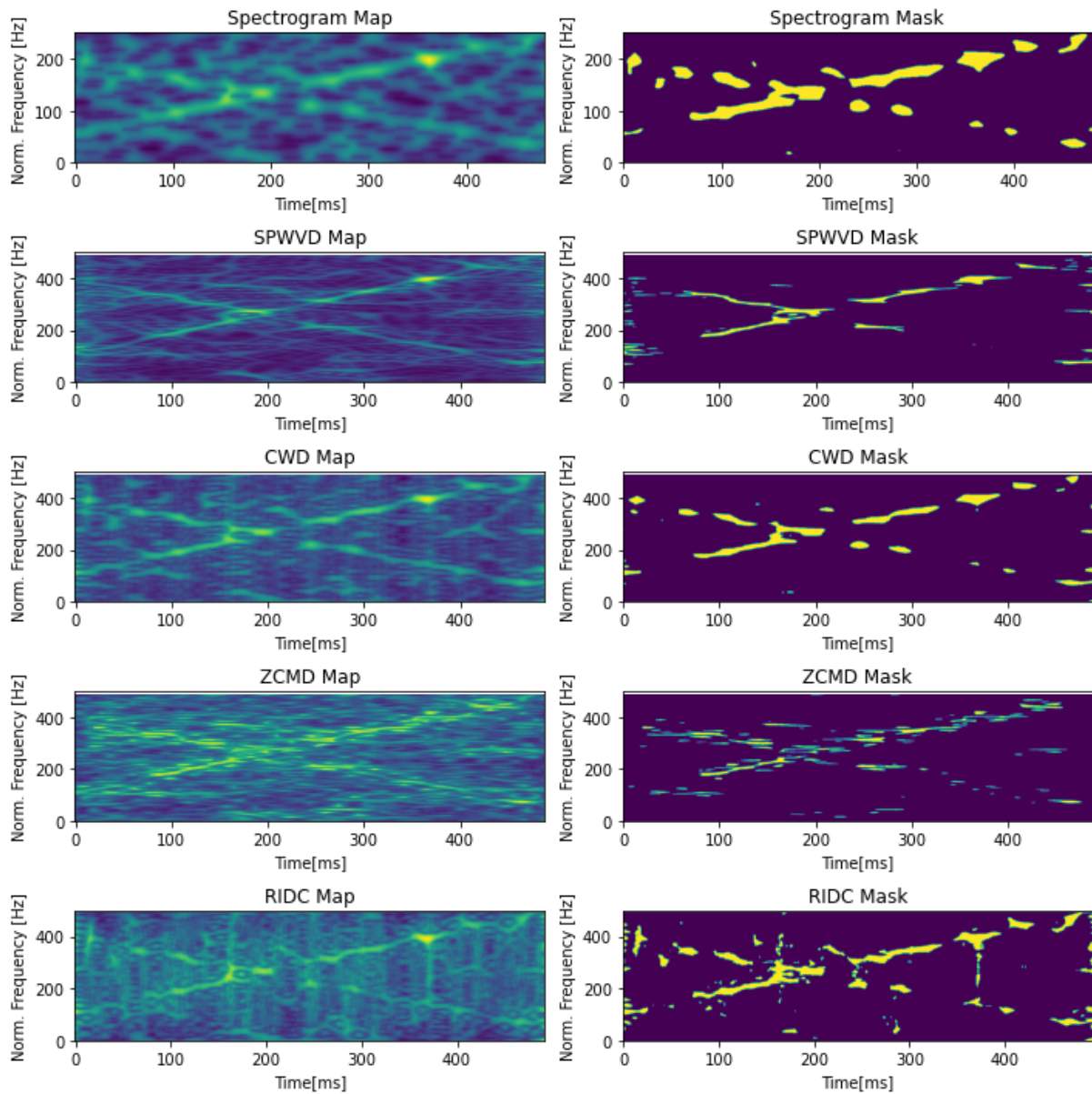
**Table 7.13:** Results obtained with Dombi entropy from signal A

visual example for SNR = -3 can be seen in Figure 7.20. For SNR = -3, the highest accuracy score (0.945) was achieved using SPWVD, while the highest F1 score (0.812) was attained with the spectrogram. For SNR = 0, the highest accuracy (0.960) and F1 score (0.883) were obtained using the spectrogram. For SNR = 5, the highest accuracy (0.976) and the highest F1 score (0.934) was also achieved with the spectrogram.

<b>Signal B</b>	SNR=-3, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.933	<b>0.945</b>	0.925	0.903	0.922
F1 Score	<b>0.812</b>	0.749	0.786	0.666	0.743
	SNR=0, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.960</b>	0.958	0.951	0.905	0.936
F1 Score	<b>0.883</b>	0.811	0.847	0.734	0.815
	SNR=5, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.976</b>	0.967	0.965	0.954	0.964
F1 Score	<b>0.934</b>	0.854	0.911	0.809	0.896

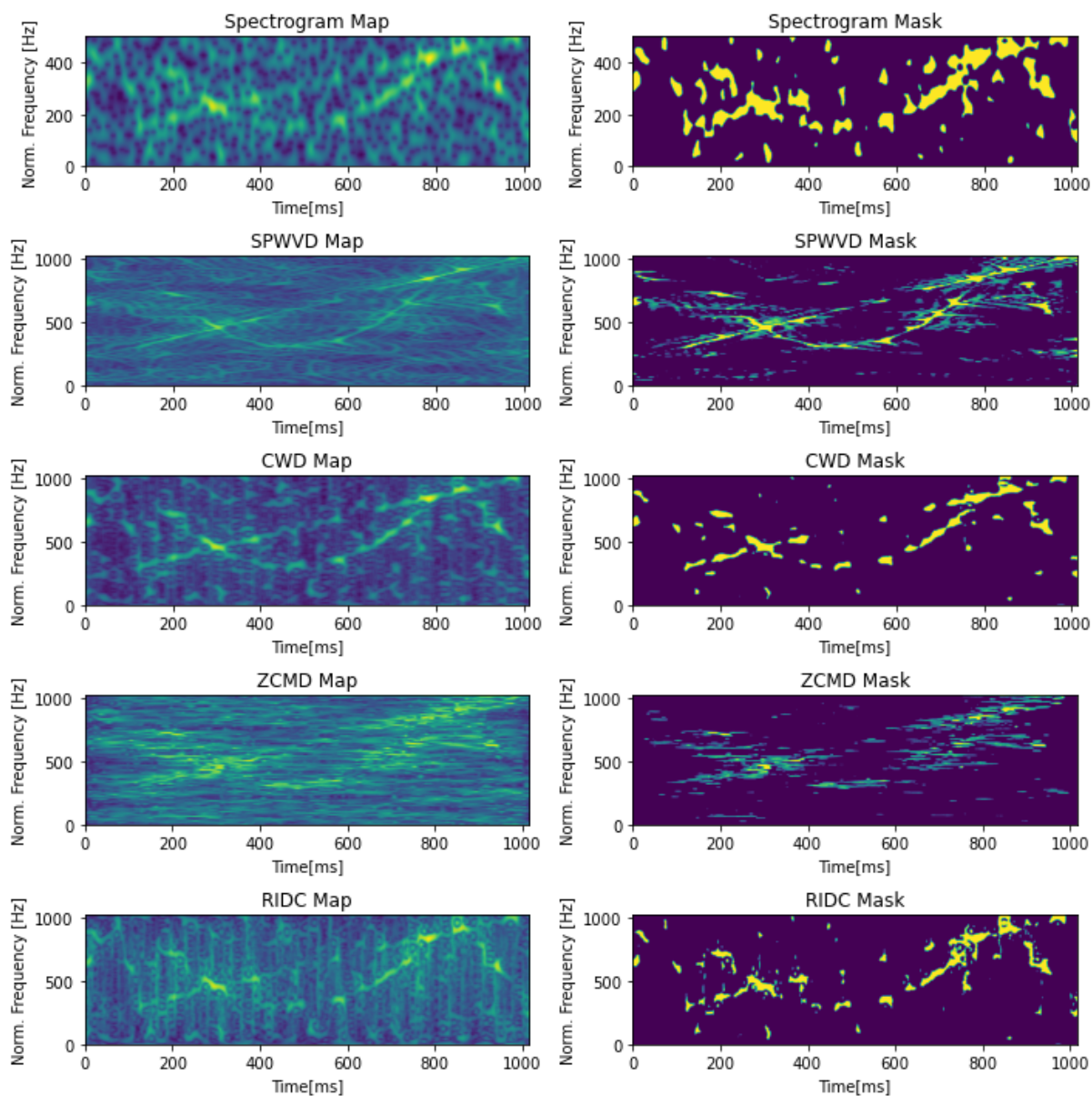
**Table 7.14:** Results obtained with Dombi entropy from signal B

Results for signal C are presented in Table 7.15. A visual example for SNR = -3 can be seen in Figure 7.21. For SNR = -3, the highest accuracy (0.965) was achieved using the SPWVD, whereas the highest F1 score (0.782) was obtained with the spectrogram. For SNR = 0, the SPWVD obtained the highest accuracy (**0.977**), while the highest F1 score (0.855) was again attained with the spectrogram. For SNR = 5, the highest accuracy (0.976) and the highest F1 score (0.923) were obtained with the spectrogram.



**Figure 7.20:** Entropy maps and masks for different TFDs for Dombi entropy and SNR = -3 in case of signal B





**Figure 7.21:** Entropy maps and masks for different TFDs for Dombi entropy and SNR = -3 in case of signal C

<b>Signal C</b>	SNR=-3, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.946	<b>0.965</b>	0.935	0.898	0.935
F1 Score	<b>0.782</b>	0.740	0.756	0.621	0.706
	SNR=0, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	0.958	<b>0.977</b>	0.947	0.879	0.950
F1 Score	<b>0.855</b>	0.783	0.831	0.633	0.785
	SNR=5, Dombi entropy				
	Spectrogram	SPWVD	CWD	ZAMD	RIDB
Accuracy	<b>0.976</b>	0.972	0.955	0.902	0.956
F1 Score	<b>0.923</b>	0.812	0.860	0.648	0.828

**Table 7.15:** Results obtained with Dombi entropy from signal C

### 7.1.6. Comparison of 2DLEM with different entropy measures on synthetic data

At first glance, it can be seen that the results of the 2DLEM method were obtained with different entropy measures, which influenced the results. To determine whether the difference is statistically significant, another ANOVA with post-hoc tests was performed.

For Signal A, the tests showed no statistically significant difference between Rényi and Tsali's entropy in the case of spectrogram at SNR=0 and SNR=5. There was also no statistically significant difference between Rényi and Shannon for SNR=5. For the ZAM distribution, there were no statistically significant differences between Rényi and Tsali's entropy for SNR=0. For RIDB, there were no statistically significant differences between Shannon and Tsallis for both SNR=0 and SNR=5.

For SNR=-3, Tsallis entropy with ZAM distribution achieved the best results in both metrics. The difference in accuracy between Shannon and Rényi entropy was 0.002, with Rényi also having a higher F1 value of 0.062. Tsallis entropy outperforms Shannon entropy with a difference of 0.007 in accuracy and a 0.071 higher F1 score. The difference in accuracy between the Rényi and Tsallis entropies is 0.005, with Tsallis having a slightly higher F1 score of 0.009. For SNR=0, Rényi entropy also achieved the best results in both metrics with the ZAM distribution, but there was no statistical difference between Rényi's and Tsallis's entropy. The difference in accuracy between Shannon and Rényi entropy was 0.023, with Rényi also having a higher F1 score of 0.086. Shannon entropy outperforms

Tsallis entropy with a difference of 0.023 in accuracy and a 0.114 higher F1 score. The difference in accuracy between Rényi and Tsallis entropy is 0.0005, with Rényi having a slightly higher F1 score of 0.005.

For SNR=5, Tsallis entropy with SPWV distribution provides the highest score. The difference in accuracy between the Shannon entropy and the Rényi entropy is 0.011, with the Rényi entropy having a 0.036 higher F1 score. The Shannon entropy outperforms the Tsallis entropy with a difference of 0.003 in accuracy and a 0.001 higher F1 score. The difference in accuracy between Rényi and Tsallis entropy is 0.009, with Rényi having a slightly higher F1 score of 0.027.

For signal B, there was no difference between Shannon and Rényi when SNR=-3 and the distribution was SPWV. There was no difference between Shannon and Tsallis when SNR =-3 for both the ZAM and RIDB distributions and for SNR=0 and the CW and RDIB distributions. Rényi and Tsallis were the same for the ZAM distribution when SNR=0.

For signal B at SNR=0, Tsallis entropy with SPWVD had the highest accuracy score and Shannon entropy with spectrogram was had the highest F1 score, but there was no statistically significant difference between Shannon and Tsallis when it came to accuracy scores. At SNR=0, Rényi entropy with spectrogram had the highest F1 score, while Shannon scored the same for accuracy, but there was no significant difference between Shannon and Rényi in this case. At SNR=5, Shannon entropy with spectrogram outperformed all other combinations, but there was no statistical difference between Shannon and Rényi entropy.

For signal B, the entropies that achieve the highest accuracy and F1 score vary between the different SNRs. Tsallis has the highest accuracy for SNR=-3, Rényi for SNR=0, and Tsallis for SNR=5. For F1 score, Rényi consistently performs best at all SNRs.

Signal C had the least varied results, where the spectrogram outperformed the other distributions in all cases, regardless of SNR or entropy measure. In this case, there were no differences, only between Shannon and Tsallis when the spectrogram was used for SNR=-3, and between Rényi and Tsallis when SPWVD was used. For SNR=-3, Shannon entropy had the highest accuracy, while Tsallis had the highest F1 value. For SNR=0, Tsallis had the highest accuracy, while Rényi had the highest F1 score. For SNR=5, Shannon had the highest accuracy, and Rényi had the highest F1 score, but there was no

statistically significant difference between the accuracy values. For signal C, the entropies achieving the highest accuracy and F1 score varied between the different SNRs. Shannon has the highest accuracy for SNR=-3 and SNR=5, Rényi for SNR=0 and SNR=5, and Tsallis for SNR=0. For the F1 score, Rényi performs best across all SNRs.

There is considerable diversity in the results, indicating that all classical entropies perform effectively with the proposed 2DLEM method. Although each entropy achieved the highest score in different cases, the differences between them, although often statistically significant, were not particularly large.

The highest scores obtained with the classical entropies will now be compared to the proposed Dombi entropy. For signal A, from the classical entropies, Tsallis had the highest score for all SNRs. At SNR = -3, Dombi had a slight edge in both accuracy and F1 scores and showed statistically significant differences. However, at SNR = 0, there was no discernible difference between the Tsallis and Dombi entropies. Even at SNR = 5, there was no clear difference between the best-performing TFDs for all entropies.

For signal B, Tsallis was most accurate at SNR=-3, Rényi at SNR=0, and again Tsallis at SNR=5. In terms of F1 score, Rényi performed consistently better at all SNRs. At SNR=-3, Dombi showed the highest accuracy and F1 score, with no significant difference in accuracy between Dombi and Tsallis. However, at the highest TFD score in relation to the F1 score, there were significant differences between all entropies. At an SNR of 0, Dombi scored best on the F1 score but showed similar accuracy values to Shannon and Tsallis, while Dombi and Rényi had the best F1 scores and differed only minimally from each other. No significant statistical differences in accuracy were found between Dombi and Tsallis. At SNR=5, Shannon scored the highest, but no significant differences in metrics were found between Shannon and Dombi.

For Signal C, the spectrogram distribution consistently produced the best results. At SNR = -3, Shannon showed slightly higher scores than Dombi, but the difference was not statistically significant. At SNR = 0, Dombi achieved the highest accuracy, but no significant difference in F1 scores was observed between Dombi and Shannon. Rényi had the highest F1 score. At SNR = 5, Dombi's accuracy was the same as Shannon's but significantly better, with the highest F1 score.

Compared to classical entropy measures such as Tsallis, Shannon, and Rényi, Dombi showed remarkable performance differences at different SNRs and signals. At certain

SNRs, Dombi showed either comparable or superior accuracy and F1 scores.

## **7.2. 2DLEM applied to real-world data**

### **7.2.1. 2DLEM for noisy speech signals**

The application of the method has been extended by applying it to recorded speech signals corrupted by noise from different sources. The database used is commonly used in speech recognition.

The speech signals were extracted from the AURORA database [138], which includes 30 different recordings. In this database, 30 IEEE sentences were articulated by three male and three female speakers, which were corrupted by eight different real sounds with different SNRs of 0 dB, 5 dB, and 10 dB. The selected sound signals aim to simulate likely usage scenarios in telecommunication terminals and include sounds from sources such as trains, crowds (babble), cars, exhibition halls, restaurants, streets, airports, and train stations.

To simulate real conditions, sounds are intentionally inserted into the speech signal. A sound segment is randomly extracted from the sound recordings that correspond in length to the speech signal. It is scaled in amplitude accordingly to achieve the desired SNR level and then added to the filtered clean speech signal. This process culminates in the creation of a noisy speech corpus, referred to as NOIZEUS, as described in [139].

The AURORA database is widely used in speech analysis, especially in speech recognition, as highlighted in the original paper [138]. The proposed method was tested with both clean and noisy signals and showed higher accuracy with clean signals. Similar investigations have been conducted in several studies [140, 141, 142] presenting different methods for speech recognition and enhancement. In addition, the NOIZEUS corpus was used in [143, 144] to evaluate the correlations of widely used objective measures in speech enhancement.

Here, the 2DLEM method was used in combination with the spectrogram and different entropy measures motivated by previous tests with synthetic signals. The results for different noise sources are shown in Tables 7.16, 7.17, 7.18 and 7.19 for Shannon, Rényi, Tsallis and Dombi entropy retrospectively.

<b>Train Noise</b>	<b>Accuracy</b>	<b>Precision</b>	<b>Recall</b>
SNR=0	98.39%	99.35%	99.34%
SNR=5	98.82%	99.39%	99.39%
SNR=10	98.75%	99.49%	99.49%
<b>Babbel Noise</b>			
SNR=0	98.52%	98.99%	99.52%
SNR=5	98.69%	99.33%	99.35%
SNR=10	99.12%	99.54%	99.56%
<b>Car Noise</b>			
SNR=0	97.82%	98.94%	98.94%
SNR=5	98.38%	99.22%	99.22%
SNR=10	98.87%	99.54%	99.59%
<b>Train Station Noise</b>			
SNR=0	98.81%	99.49%	99.49%
SNR=5	98.87%	99.5%	99.5%
SNR=10	98.75%	99.3%	99.3%
<b>Restaurant Noise</b>			
SNR=0	98.49%	99.4%	99.4%
SNR=5	98.59%	99.35%	99.35%
SNR=10	98.96%	99.58%	99.58%
<b>Airport Noise</b>			
SNR=0	98.49%	99.4%	99.4%
SNR=5	98.59%	99.35%	99.35%
SNR=10	98.96%	99.58%	99.58%
<b>Street noise</b>			
SNR=0	98.71%	99.57%	99.57%
SNR=5	98.94%	99.62%	99.62%
SNR=10	99.03%	99.52%	99.52%
<b>Exhibition Hall Noise</b>			
SNR=0	98.83%	99.41%	99.41%
SNR=5	98.94%	99.49%	99.49%
SNR=10	98.98%	99.49%	99.49%

**Table 7.16:** Results obtained for speech signals with added noise for different sources using Shannon entropy

The accuracy of the signal spectrogram ranges from 97.78% to 98.87% at an SNR of 0 dB, whereby the precision and recall are slightly higher at 98.93% to 99.6%. The best results are achieved in the case of the artificially introduced noise from the train station, although the difference to other sources is not significant. At an SNR of 5 dB, the improvement in accuracy is between 0.06 and 0.43. Precision and recall remain mostly consistent, with minimal improvements in some cases. However, the results for restaurant

<b>Train Noise</b>	<b>Accuracy</b>	<b>Precision</b>	<b>Recall</b>
SNR=0	98.37%	99.55%	99.24%
SNR=5	98.82%	99.38%	99.38%
SNR=10	98.77%	99.39%	99.47%
<b>Babbel Noise</b>			
SNR=0	98.51%	98.97%	99.52%
SNR=5	98.66%	99.34%	99.35%
SNR=10	99.15%	99.53%	99.52%
<b>Car Noise</b>			
SNR=0	97.78%	98.94%	98.93%
SNR=5	97.99%	99.02%	99.02%
SNR=10	98.86%	99.44%	99.54%
<b>Train Station Noise</b>			
SNR=0	98.83%	99.48%	99.48%
SNR=5	98.85%	99.51%	99.49%
SNR=10	98.89%	99.31%	99.3%
<b>Restaurant Noise</b>			
SNR=0	98.45%	99.41%	99.41%
SNR=5	98.53%	99.35%	99.35%
SNR=10	98.94%	99.57%	99.57%
<b>Airport Noise</b>			
SNR=0	98.52%	99.42%	99.42%
SNR=5	98.59%	99.34%	99.34%
SNR=10	98.95%	99.54%	99.54%
<b>Street noise</b>			
SNR=0	98.69%	99.57%	99.57%
SNR=5	98.95%	99.62%	99.62%
SNR=10	99.02%	99.51%	99.51%
<b>Exhibition Hall Noise</b>			
SNR=0	98.82%	99.41%	99.41%
SNR=5	98.93%	99.48%	99.48%
SNR=10	98.99%	99.49%	99.49%

**Table 7.17:** Results obtained for speech signals with added noise for different sources using Rényi entropy

and airport noise do not reach the same high values as with an SNR of 0 dB. In most scenarios, the highest accuracy is achieved at an SNR of 10 dB. In the case of train noise, however, the accuracy drops to 98.75%. It is noteworthy that all scenarios show the highest accuracy and recognition value at an SNR of 10 dB, while the train station sacrifices the improvement in accuracy for a gain in accuracy and recall value. The least favourable results were observed when car noise was introduced, with an average accuracy

<b>Train Noise</b>	<b>Accuracy</b>	<b>Precision</b>	<b>Recall</b>
SNR=0	98.38%	99.58%	99.27%
SNR=5	98.84%	99.35%	99.35%
SNR=10	98.77%	99.42%	99.51%
<b>Babble Noise</b>			
SNR=0	98.52%	99.0%	99.55%
SNR=5	98.69%	99.31%	99.32%
SNR=10	99.1%	99.56%	99.55%
<b>Car Noise</b>			
SNR=0	97.81%	98.99%	98.96%
SNR=5	97.97%	99.05%	99.06%
SNR=10	98.85%	99.46%	99.57%
<b>Train Station Noise</b>			
SNR=0	98.81%	98.99%	98.99%
SNR=5	98.88%	99.51%	99.51%
SNR=10	98.91%	99.34%	99.33%
<b>Restaurant Noise</b>			
SNR=0	98.46%	99.43%	99.44%
SNR=5	98.56%	99.38%	99.38%
SNR=10	98.93%	99.55%	99.55%
<b>Airport Noise</b>			
SNR=0	98.5%	99.4%	99.4%
SNR=5	98.62%	99.33%	99.33%
SNR=10	98.96%	99.57%	99.57%
<b>Street noise</b>			
SNR=0	98.7%	99.51%	99.51%
SNR=5	98.98%	99.55%	99.55%
SNR=10	98.97%	99.47%	99.47%
<b>Exhibition Hall Noise</b>			
SNR=0	98.83%	99.43%	99.43%
SNR=5	98.96%	99.48%	99.48%
SNR=10	99.02%	99.47%	99.47%

**Table 7.18:** Results obtained for speech signals with added noise for different sources using Tsallis entropy

of 97.82% at an SNR of 0 dB.

Differences between different entropy measures are minimal, ranging to 0.03. Based on the results it can not be said that one entropy measure has an advantage over others in this particular test case.

Although the method was applied to different speech signals with different SNRs and noise sources, the results remained consistent in all scenarios tested. Accuracy ranged from



<b>Train Noise</b>	<b>Accuracy</b>	<b>Precision</b>	<b>Recall</b>
SNR=0	98.39%	99.38%	99.37%
SNR=5	98.84%	99.37%	99.37%
SNR=10	98.77%	99.45%	99.45%
<b>Babble Noise</b>			
SNR=0	98.52%	99.0%	99.0%
SNR=5	98.69%	99.31%	99.32%
SNR=10	99.1%	99.56%	99.55%
<b>Car Noise</b>			
SNR=0	97.81%	98.99%	98.96%
SNR=5	97.97%	99.05%	99.06%
SNR=10	98.85%	99.46%	99.57%
<b>Train Station Noise</b>			
SNR=0	98.87%	99.51%	99.52%
SNR=5	98.88%	99.55%	99.53%
SNR=10	98.91%	99.34%	99.33%
<b>Restaurant Noise</b>			
SNR=0	98.46%	99.43%	99.44%
SNR=5	98.56%	99.38%	99.38%
SNR=10	98.96%	99.6%	99.6%
<b>Airport Noise</b>			
SNR=0	98.55%	99.46%	99.45%
SNR=5	98.62%	99.37%	99.37%
SNR=10	98.98%	99.57%	99.57%
<b>Street noise</b>			
SNR=0	98.7%	99.59%	99.6%
SNR=5	98.98%	99.65%	99.65%
SNR=10	99.03%	99.54%	99.54%
<b>Exhibition Hall Noise</b>			
SNR=0	98.83%	99.43%	99.43%
SNR=5	98.96%	99.51%	99.51%
SNR=10	99.02%	99.52%	99.52%

**Table 7.19:** Results obtained for speech signals with added noise for different sources using Dombi entropy

97.78% to 99.03%, precision from 98.94% to 99.65%, and recall from 98.96% to 99.65%. This indicates the method has robust performance regardless of signal type, noise source, or SNR. The SNR value appears to have the greatest influence on the results. It can also be seen that where SNR values are high entropy choice does not impact the results.

### 7.2.2. 2DLEM with CNN for seismic signal classification

In order to test the proposed method on real-world scenarios, it is here applied to the problem of classification of seismological signals. As mentioned in chapter 5.2.2., TFDs have recently been applied to this problem with great success. Spectrograms provide a graphical representation of the frequency content of seismic signals over time, while entropy maps quantify the complexity or uncertainty in these spectrograms. The goal of comparing the classification results derived from these representations is to identify the differences between the two in categorizing seismic data. For 2DLEM, motivated by the findings on the synthetic signals and the speech dataset, the spectrogram with Dombi entropy was used. For this problem, the last step was omitted in 2DLEM. Instead of the entropy mask, the entropy map was used for the classification problem.

#### Dataset - seismic signals

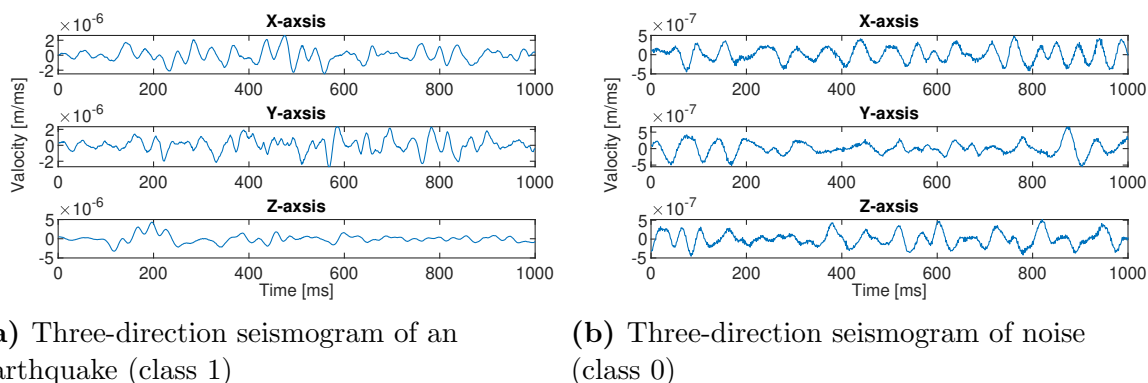
The data used for this study comes from the seismological recording system for events of the Istituto Nazionale di Geofisica e Vulcanologia. This dataset included 35,055 samples, each corresponding to a specific seismic event. These samples were recorded in three different directions (Z, N, and E) by the sensor for each event.

The identification of seismic events in waveform time series is similar to object recognition in 2D photos. Here, the three elements of a 3-component seismogram are comparable to the RGB color channels that form a 1D image. Similarly, in the case of seismic signals, TFDs can be generated for each component of the three-component seismograms, and in this way, three-channel images can be compiled using the same principle. Images can be assembled from the entropy maps in the same way.

The dataset showed a slight imbalance in the classes, with about 60% of the samples categorized as earthquake events (class 1 in Figure 7.22a), while the remaining 40% were categorized as non-earthquake events (class 0 in Figure 7.22a).

The training subset, which included 70% of the instances, maintained the balance of the original dataset between positive (earthquake) and negative (non-earthquake) samples. This segment was used to train the models effectively.

At 15% of instances, the validation subset had the same proportion of samples as the training set. It was used to refine the models and adjust the hyperparameters during the



**Figure 7.22:** Two randomly sampled seismogram instances from the dataset

training phase.

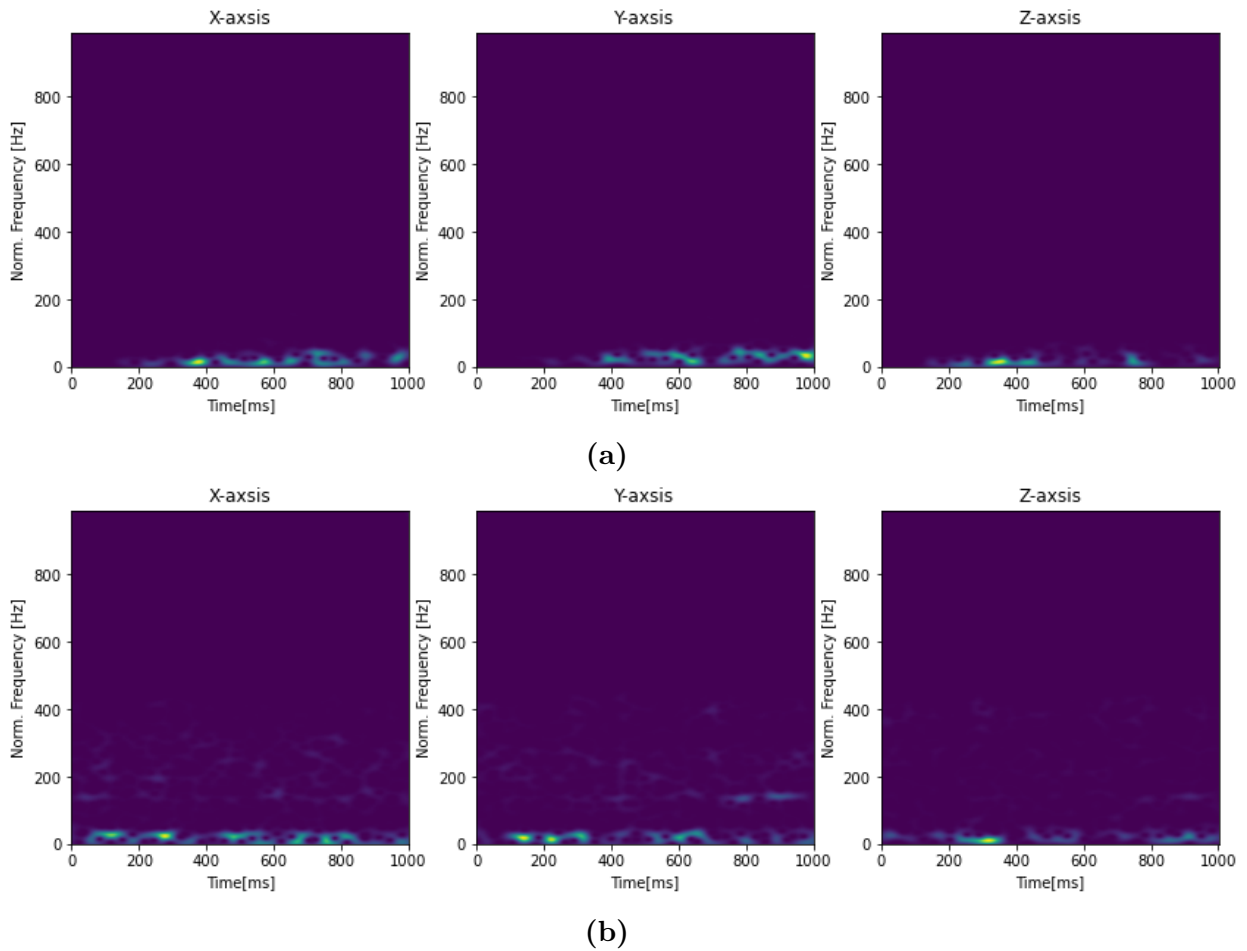
The test subset, also accounting for 15% of the instances, also contained the same proportion of samples as the training set. This split was used to evaluate the final performance and generalization capabilities of the trained models.

Care was taken to ensure that no single data point appeared in more than one subset to ensure a clear distinction between the subsets and to avoid overlap or bias. In this way, the exclusivity of the subsets representing different phases of model training and evaluation was maintained.

Three deep CNN architectures, VGG19, ResNet50, and DenseNet121, were used in this study, as mentioned in chapter 4.. Each model followed its original specifications, although some adjustments were made to the output layer. In addition, dropout layers (with  $p=0.5$ ) were inserted in the fully linked (dense) layers before the output layer. The final prediction layer was adjusted to allow binary classification between two classes. This adaptation involved a single neuron with a sigmoid activation function and used a binary cross-entropy loss function for all models. Experimental evaluations were performed on the validation dataset to determine the optimal learning rate.

For optimization purposes, the Adam optimizer was applied with a learning rate of  $\alpha = 3 \times 10^{-5}$ , and a stack size of 64 was used in training. The models were trained for 50 epochs. The time series were first transformed into spectrograms, as shown in Figure 7.23. From there, entropy maps were obtained for the spectrograms( Figure 7.24).

Both spectrograms and entropy maps were then combined into RGB pictures using three seismographs as three channels, as shown in Figures 7.25 and 7.26.



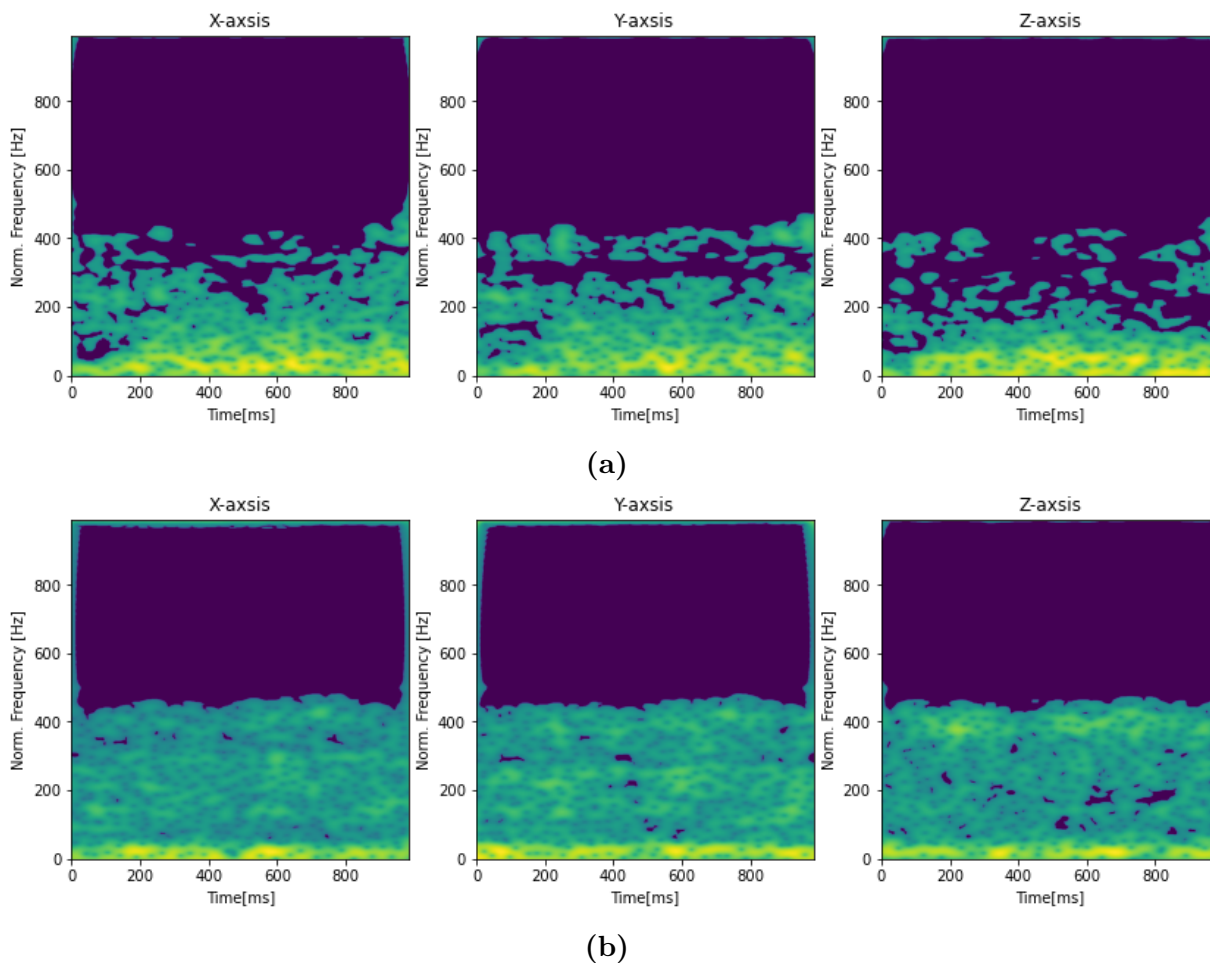
**Figure 7.23:** Spectrograms obtained from earthquake sample(a) and noise sample(b)

### Results of seismic signals classification using the 2DLEM with CNN

The provided table (7.20) shows the classification results obtained using two different methods: 2DLEM with Dombi entropy and spectrogram analysis. The table shows the evaluation metrics for the different CNN models Resnet50, VGG19, and DN121 used for both methods.

For 2DLEM, the three CNN models — Resnet50, VGG19, and DN121 — achieved high-performance metrics in all categories, with accuracy between 0.9753 and 0.9755, AU-ROC between 0.9763 and 0.9772, F1 score between 0.9802 and 0.9807 and MCC between 0.9478 and 0.9488.

In contrast, the spectrogram analysis, which used the same CNN models, had slightly lower metrics compared to 2DLEM. The accuracy ranged from 0.9491 to 0.9584, the AUROC from 0.9531 to 0.9627, the F1 score from 0.9586 to 0.9652, and the MCC from 0.8940 to 0.9095.



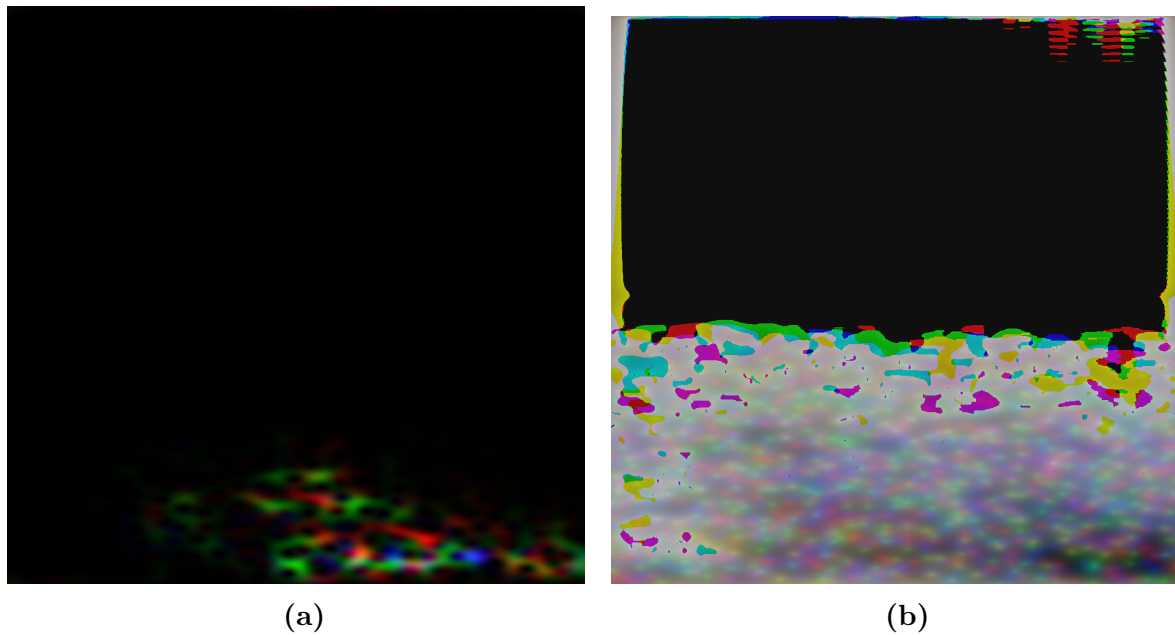
**Figure 7.24:** Entropy maps obtained from earthquake sample(a) and noise sample(b)

The table illustrates the comparative performance of the CNN models between the 2DLEM and spectrogram methods. It shows that the 2DLEM method achieves better results for the evaluation metrics considered.

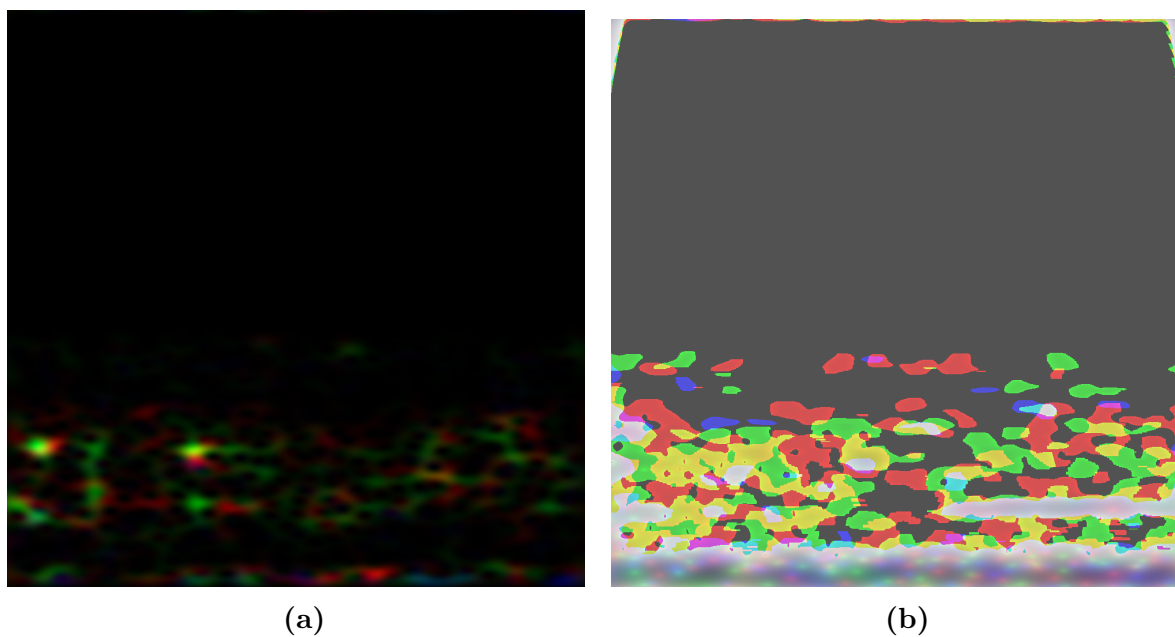
Additionally, the confusion matrix for each model can be seen in Figure 7.27.

The Cochran’s Q test was used to determine whether there was a significant difference in model performance. The purpose of Cochran’s Q test is to determine whether the proportions of a categorical variable are the same for many dependent groups. Data with repeated measures or matched samples are often analyzed in research and experimental contexts. Each group studied has a binary variable that makes up the data for the Cochran Q test. Each set consists of the same subjects or things, and each subject in each group has the variable of interest measured.

Cochran’s Q-test assumes that the observations within each group are dependent and that the binary variable follows a Bernoulli distribution. It is insensitive to deviations



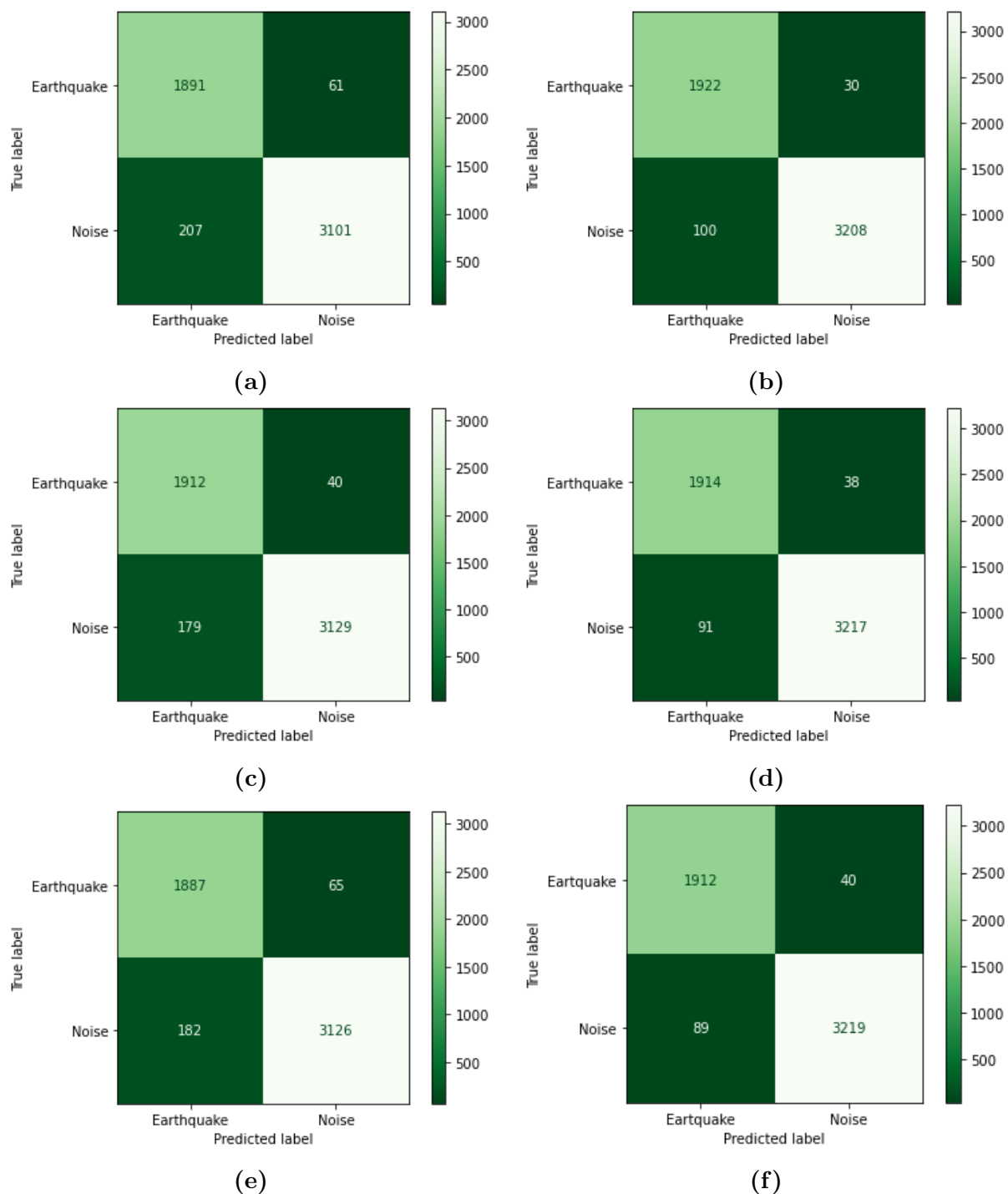
**Figure 7.25:** RGB images obtained from earthquake sample spectrogram(a) and earthquake sample entropy map(b)



**Figure 7.26:** RGB images obtained from noise sample spectrogram(a) and noise sample entropy map(b)

from the assumption of equal variance or normality.

The null hypothesis is rejected if  $p$  is greater than the critical value resulting from the chi-squared distribution at a certain significance level ( $\alpha = 0.05$ ). This means that there are no significant differences in the proportions of the categorical variables between the groups. If the null hypothesis was rejected, post-hoc analyses with pairwise com-



**Figure 7.27:** Confusion matrices for evaluating the CNN models. (a) Spectrogram distributions as input to ResNet50, (b) Entropy maps as input to ResNet50, (c) Spectrogram distributions as input to VGG19, (d) Entropy maps as input to VGG19, (e) Spectrogram distributions as input to DenseNet121, (f) Entropy maps as input to DenseNet121

	2DLEM			Spectrogram		
	Resnet50	VGG19	DN121	Resnet50	VGG19	DN121
Accuracy	<b>0.9755</b>	<b>0.9755</b>	0.9753	0.9530	0.9491	0.9584
AUROC	0.9763	0.9765	<b>0.9772</b>	0.9558	0.9531	0.9627
F1 Score	0.9804	0.9802	<b>0.9807</b>	0.9620	0.9586	0.9652
MCC	0.9479	0.9478	<b>0.9488</b>	0.9016	0.8940	0.9095

**Table 7.20:** Classification results on the test set

parisons were conducted to determine whether there was a significant difference between specific groups. McNemar tests with Bonferroni adjustment were used for the post-hoc analysis. Analysis of the classification results using the 2DLEM approach revealed no statistically significant differences between the three CNNs used – Resnet50, VGG19, and DN121. Using this method, performance metrics such as accuracy, AUROC, F1 score, and MCC showed similar values for all three CNN models, indicating a consistent level of performance throughout.

The spectrogram analysis, however, showed a significant statistical difference between the CNN models Resnet50 and VGG19. This difference was observed in various evaluation metrics and indicates a performance divergence between these two specific CNN architectures within the spectrogram methodology. The metrics showed different values between Resnet50 and VGG19, indicating a statistically significant disparity in their classification performance.

In the analysis comparing the performance of a particular CNN model with two different methods, 2DLEM and spectrogram, there was a statistically significant difference in the classification results. When the CNN was applied to the 2DLEM approach and then to the spectrogram method, the evaluation metrics showed differences.

The comparison between the performance of the CNN on the 2DLEM method and the performance on the spectrogram method showed statistically significant differences (Table 7.21) between models output resulting in differences in all metrics. These differences in performance metrics indicate that the model running with the 2DLEM method produced different classification results than its performance within the spectrogram approach. This suggests that the same CNN model had significantly different levels of performance when applied to these different signal representation methods.

In summary, the observed differences in the performance of the CNN models, partic-



	Spectrogram			
	p-value	Resnet50	VGG19	DN121
	Resnet50	<0.001	<0.001	<0.001
2DLEM	VGG19	<0.001	<0.001	<0.001
	DN121	<0.001	<0.001	<0.001

**Table 7.21:** P-values for all models comparing both representations

ularly the improved performance on the 2DLEM representation compared to the spectrogram, suggest that the 2DLEM methodology provides a more effective signal representation for the classification task at hand. The models performed better when using the 2DLEM signal representation compared to the spectrogram representation, indicating the potential usefulness of 2DLEM as a more discriminative and informative signal representation for this particular classification problem.

## 8. Chapter

# CONCLUSION

In this thesis, the main objective of developing an effective method for extracting content from the time-frequency domain of signals utilizing information entropy measures was achieved. The first objective was the development of a specialized 2D entropy calculation method. This method was further improved by integrating adaptive windows that allow dynamic adaptation to data.

The extraction of masks from the derived entropy maps provided additional analytical insight into the signal data. The research also carefully compared the performance of different entropy measures and introduced a novel entropy measure tailored to improve the separation between noise and useful signal content.

The presented method was used as a preprocessing step for signal classification tasks and has proven its usefulness in extracting relevant information from signals. The entropy map derived from the method provided new and valuable information that proved to be crucial for improving the accuracy of signal classification. Comparison between spectrogram representation, which is the most commonly used TFD representation, and entropy map representation showed statistical differences in the model output. Models with entropy maps achieved a higher accuracy of 1.69% to 2.64%, an AUROC of 1.45% to 2.34%, an F1 score of 1.55% to 2.16%, and an MCC of 3.93% to 5.38%. Examining other datasets could help assess the generalizability of these results. Different datasets may have unique characteristics that influence model performance and the best choice of representations. Testing these models and representations on different datasets could reveal their robustness and suitability for different domains and data sources.

This work contributes to a novel method for separating signal components from noisy data. The adaptation and application of a fuzzy entropy measure further emphasize the versatility of the proposed method. The research shows that the developed method can serve as a robust preprocessing step in classification tasks, facilitating the extraction of useful content from signals.

Exploring alternative time-frequency distributions, such as the quadratic Cohen distribution or the continuous wavelet transform, could represent the next research direction aiming to provide high-resolution insights into the time-frequency properties of the data and their impact on classification accuracy. Exploring the influence of various noise types on the method's performance could be a valuable avenue for future investigation. While the current study primarily focused on the efficacy of the method with white noise, assessing its robustness against different types of generated noise—such as coloured or impulse noise—would offer insights into its adaptability across diverse signal environments. Understanding how the method responds to distinct noise profiles could further improve its applicability in real-world scenarios. Comparing model performance with different entropy measures and quadratic, high-resolution TFD representations with reduced cross-terms could reveal new ways to improve classification and deepen our understanding of the hidden data properties.

## BIBLIOGRAPHY

- [1] B. Boashash, *Time-frequency signal analysis and processing: a comprehensive reference*. Australia: Elsevier Academic Press, 2016.
- [2] A. Belouchrani and M. Amin, “Blind source separation based on time-frequency signal representations,” *IEEE Transactions on Signal Processing*, vol. 46, no. 11, pp. 2888–2897, 1998.
- [3] M. Zibulevsky and B. A. Pearlmutter, “Blind Source Separation by Sparse Decomposition in a Signal Dictionary,” *Neural Computation*, vol. 13, no. 4, pp. 863–882, 04 2001. [Online]. Available: <https://doi.org/10.1162/089976601300014385>
- [4] K. Hild, D. Pinto, D. Erdogmus, and J. Principe, “Convolutional blind source separation by minimizing mutual information between segments of signals,” *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I-REGULAR PAPERS*, vol. 52, no. 10, pp. 2188–2196, OCT 2005.
- [5] K. Hild, D. Erdogmus, and J. Principe, “An analysis of entropy estimators for blind source separation,” *SIGNAL PROCESSING*, vol. 86, no. 1, pp. 182–194, JAN 2006.
- [6] X.-Y. Zhang, W.-R. Wang, C.-Y. Shen, Y. Sun, and L.-X. Huang, “Extraction of eeg components based on time - frequency blind source separation,” in *Advances in Intelligent Information Hiding and Multimedia Signal Processing*. Springer International Publishing, 2018, pp. 3–10.
- [7] N. Saulig, Ž. Tomasović, and C. Ioana, “A local entropy-based algorithm for information content extraction from time-frequency distributions of noisy signals,” *Digital Signal Processing*, vol. 70, 08 2017.

- [8] L. Zadeh, "Probability measures of fuzzy events," *Journal of Mathematical Analysis and Applications*, vol. 23, no. 2, pp. 421–427, 1968.
- [9] A. De Luca and S. Termini, "A definition of a nonprobabilistic entropy in the setting of fuzzy sets theory," *Information and Control*, vol. 20, no. 4, pp. 301–312, 1972.
- [10] S.-F. Liang, C.-E. Kuo, F. Shaw, Y.-H. Chen, C.-H. Hsu, and J.-Y. Chen, "Combination of expert knowledge and a genetic fuzzy inference system for automatic sleep staging," *IEEE Transactions on Biomedical Engineering*, vol. 63, pp. 1–1, 12 2015.
- [11] C. Yang, Z. Deng, K.-S. Choi, and S. Wang, "Takagi-sugeno-kang transfer learning fuzzy logic system for the adaptive recognition of epileptic electroencephalogram signals," *IEEE Transactions on Fuzzy Systems*, vol. 24, pp. 1–1, 01 2015.
- [12] Y. T. Liu, Y.-Y. Lin, S.-L. Wu, C.-H. Chuang, and C.-T. Lin, "Brain dynamics in predicting driving fatigue using a recurrent self-evolving fuzzy neural network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, pp. 1–14, 11 2015.
- [13] L. Cohen, "Time-frequency distributions-a review," *Proceedings of the IEEE*, vol. 77, no. 7, pp. 941–981, July 1989.
- [14] F. Hlawatsch and G. F. Boudreaux-Bartels, "Linear and quadratic time-frequency signal representations," *IEEE Signal Processing Magazine*, vol. 9, no. 2, pp. 21–67, 1992.
- [15] D. Gabor, "Theory of communication," *Part 1, The Journal of the Institution of Electrical Engineers, Part III, Radio and Communication*, vol. 93, pp. 429–457, 1946.
- [16] M. H. Ackroyd, "Short-Time Spectra and Time-Frequency Energy Distributions," *The Journal of the Acoustical Society of America*, vol. 50, no. 5A, pp. 1229–1231, 08 2005. [Online]. Available: <https://doi.org/10.1121/1.1912761>
- [17] F. Hlawatsch and F. Auger, *Time-frequency Analysis: Concepts and Methods*, ser. Digital signal and image processing series. ISTE, 2008. [Online]. Available: <https://books.google.hr/books?id=W0DpAAAACAAJ>

- [18] E. Wigner, “On the quantum correction for thermodynamic equilibrium,” *Phys. Rev.*, vol. 40, pp. 749–759, Jun 1932. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRev.40.749>
- [19] S. Kadambe and G. Boudreaux-Bartels, “A comparison of the existence of ‘cross terms’ in the wigner distribution and the squared magnitude of the wavelet transform and the short-time fourier transform,” *IEEE Transactions on Signal Processing*, vol. 40, no. 10, pp. 2498–2517, 1992.
- [20] L. Cohen, “Generalized Phase-Space Distribution Functions,” *Journal of Mathematical Physics*, vol. 7, no. 5, pp. 781–786, 05 1996. [Online]. Available: <https://doi.org/10.1063/1.1931206>
- [21] P. Loughlin, J. Pitton, and L. Atlas, “Bilinear time-frequency representations: new insights and properties,” *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 750–767, 1993.
- [22] P. Flandrin, “Some features of time-frequency representations of multicomponent signals,” in *ICASSP ’84. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 9, 1984, pp. 266–269.
- [23] H.-I. Choi and W. Williams, “Improved time-frequency representation of multicomponent signals using exponential kernels,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 6, pp. 862–871, 1989.
- [24] Y. Zhao, L. Atlas, and R. Marks, “The use of cone-shaped kernels for generalized time-frequency representations of nonstationary signals,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 7, pp. 1084–1091, 1990.
- [25] S. Oh and R. Marks, “Some properties of the generalized time frequency representation with cone-shaped kernel,” *IEEE Transactions on Signal Processing*, vol. 40, no. 7, pp. 1735–1745, 1992.
- [26] Zhenyu Guo, L. . Durand, and H. C. Lee, “The time-frequency distributions of nonstationary signals based on a bessel kernel,” *IEEE Transactions on Signal Processing*, vol. 42, no. 7, pp. 1700–1707, July 1994.

- [27] C. E. Shannon, “A mathematical theory of communication,” *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1538-7305.1948.tb01338.x>
- [28] A. Rényi, “On measures of entropy and information,” in *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*. Berkeley, Calif.: University of California Press, 1961, pp. 547–561. [Online]. Available: <https://projecteuclid.org/euclid.bsmsp/1200512181>
- [29] C. Tsallis, “Possible generalization of boltzmann-gibbs statistics,” *Journal of Statistical Physics*, vol. 52, pp. 479–487, 07 1988.
- [30] R. J. V. dos Santos, “Generalization of shannon’s theorem for tsallis entropy,” *J. Math. Phys.*, vol. 38, pp. 4104—4107, 1997.
- [31] S. Abe, “Axioms and uniqueness theorem for tsallis entropy,” *Physics Letters A*, vol. 271, no. 1, pp. 74–79, 2000. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0375960100003376>
- [32] S. Martinez, F. Nicolas, F. Pennini, and A. Plastino, “Tsallis’ entropy maximization procedure revisited,” *Physica A: Statistical Mechanics and its Applications*, vol. 286, no. 3, pp. 489–502, 2000. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378437100003599>
- [33] J. Havrda and F. Charvát, “Quantification method of classification processes. concept of structural  $\alpha$ -entropy,” *Kybernetika*, vol. 03, no. 1, pp. (30)–35, 1967. [Online]. Available: <http://eudml.org/doc/28681>
- [34] G. P. Patil and C. Taillie, “Diversity as a concept and its measurement,” *Journal of the American Statistical Association*, vol. 77, no. 379, pp. 548–561, 1982. [Online]. Available: <http://www.jstor.org/stable/2287709>
- [35] L. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
- [36] J. Aczel, *Lectures on Functional Equations and Their Applications*. Academic Press, 1966.

- [37] B. Sharma and I. Taneja, “Entropy of type  $(\alpha, \beta)$  and other generalized measures in information theory,” *Metrika*, vol. 22, pp. 205–215, 1975.
- [38] T. Koski and L. Persson, “Some properties of generalized exponential entropies with applications to data compression,” *Information Sciences*, vol. 62, no. 1, pp. 103–132, 1992.
- [39] J. Fan and Y. Ma, “Some new fuzzy entropy formulas,” *Fuzzy Sets and Systems*, vol. 128, no. 2, pp. 277–284, 2002. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0165011401001270>
- [40] D. S. Hooda, “On generalized measures of fuzzy entropy,” *Mathematica Slovaca*, vol. 54, no. 3, pp. 315–325, 2004.
- [41] B. D. Sharma and D. P. Mittal, “New non-additive measures of entropy for discrete probability distributions,” *Journal of Mathematical Sciences*, vol. 10, pp. 28–40, 1975.
- [42] R. Joshi and S. A. Kumar, “New exponential fuzzy entropy of order- $(\alpha, \beta)$  and its application in multiple attribute decision-making problems,” *Communications in Mathematics and Statistics*, pp. 213–229, 2017.
- [43] R. S. Verma and B. Sharma, “On generalized exponential fuzzy entropy,” *World Academy of Science, Engineering and Technology*, vol. 5, no. 12, pp. 1895–1898, 2011.
- [44] N. Pal and S. Pal, “Object-background segmentation using new definitions of entropy,” in *Proceedings of the Institution of Electrical Engineers*, 1989, pp. 284–295.
- [45] V. M. Ilić, J. Korbelt, S. Gupta, and A. M. Scarfone, “An overview of generalized entropic forms,” *Europhysics Letters*, vol. 133, no. 5, p. 50005, mar 2021. [Online]. Available: <https://doi.org/10.1209%2F0295-5075%2F133%2F50005>
- [46] M. I. Jordan and T. M. Mitchell, “Machine learning: Trends, perspectives, and prospects,” *Science*, vol. 349, no. 6245, pp. 255–260, 2015. [Online]. Available: <https://www.science.org/doi/abs/10.1126/science.aaa8415>



- [47] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [48] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [49] A. Rosebrock, *Deep Learning for Computer Vision with Python: Starter Bundle*. PyImageSearch, 2017. [Online]. Available: <https://books.google.hr/books?id=9Ul-tgEACAAJ>
- [50] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv 1409.1556*, 09 2014.
- [51] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, jun 2016, pp. 770–778.
- [52] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
- [53] I. Santamaria, “Handbook of blind source separation: Independent component analysis and applications (common, p. and juttén, ; 2010 [book review],” *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 133–134, 2013.
- [54] P. Comon, “Independent component analysis, a new concept?” *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994, higher Order Statistics. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0165168494900299>
- [55] I. Yamane, F. Yger, M. Berar, and M. Sugiyama, “Multitask principal component analysis,” 01 2016.
- [56] F. Castells, P. Laguna, L. Sörnmo, A. Bollmann, and J. M. Roig, “Principal component analysis in ecg signal processing,” *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1–21, 2007.

- [57] K. I. Diamantaras and T. Papadimitriou, “Applying pca neural models for the blind separation of signals,” *NEUROCOMPUTING*, vol. 73, no. 1-3, SI, pp. 3–9, DEC 2009, 10th Engineering Applications of Neural Networks, Thessaloniki, GREECE, 2007.
- [58] L. Sarmiento, A. Gonzalez, and J. Millet, “Fetal eeg extraction using hybrid bss techniques,” in *2015 COMPUTING IN CARDIOLOGY CONFERENCE (CINC)*, ser. Computing in Cardiology Conference, A. Murray, Ed., vol. 42, 2015, pp. 141–144, 2015 Comp in Cardiol Conference (CinC), Comp in Cardiol 2015, nice, FRANCE, SEP 06-09, 2015.
- [59] A. Hyvärinen and E. Oja, “Independent component analysis: algorithms and applications,” *Neural networks : the official journal of the International Neural Network Society*, vol. 13, no. 4-5, p. 411—430, 2000.
- [60] G. M. Davis, S. Mallat, and M. Avellaneda, “Adaptive greedy approximations,” *Constructive Approximation*, vol. 13, pp. 57–98, 1997. [Online]. Available: <https://api.semanticscholar.org/CorpusID:121066701>
- [61] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, “Algorithms and applications for approximate nonnegative matrix factorization,” *Computational Statistics and Data Analysis*, vol. 52, no. 1, pp. 155–173, 2007. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167947306004191>
- [62] I. F. Gorodnitsky and B. D. Rao, “Sparse signal reconstruction from limited data using focuss: a re-weighted minimum norm algorithm,” *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.
- [63] S. Chen, D. Donoho, and M. Saunders, “Atomic decomposition by basis pursuit,” *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.
- [64] A. S. Janani, S. A. Rezaeieh, A. Darvazehban, S. E. Keating, and A. M. Abbosh, “Portable electromagnetic device for steatotic liver detection using blind source separation and shannon wavelet entropy,” *IEEE JOURNAL OF ELECTROMAG-*

- NETICS RF AND MICROWAVES IN MEDICINE AND BIOLOGY*, vol. 6, no. 4, pp. 546–554, DEC 2022.
- [65] D. Erdogmus, K. Hild, and J. Principe, “Blind source separation using renyi’s alpha-marginal entropies,” *NEUROCOMPUTING*, vol. 49, pp. 25–38, DEC 2002.
- [66] A. Aissa-El-Bey, N. Linh-Trung, K. abed meraim, A. Belouchrani, and Y. Grenier, “Underdetermined blind separation of nondisjoint sources in the time-frequency domain,” *Signal Processing, IEEE Transactions on*, vol. 55, pp. 897 – 907, 04 2007.
- [67] F. Abrard and Y. Deville, “A time–frequency blind signal separation method applicable to underdetermined mixtures of dependent sources,” *Signal Processing*, vol. 85, no. 7, pp. 1389–1403, 2005. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S016516840500054X>
- [68] M. Puigt and Y. Deville, “Time–frequency ratio-based blind separation methods for attenuated and time-delayed sources,” *Mechanical Systems and Signal Processing*, vol. 19, no. 6, pp. 1348–1379, 2005, special Issue: Blind Source Separation. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0888327005001123>
- [69] A. Belouchrani and M. Amin, “Blind source separation using time-frequency distributions: Algorithm and asymptotic performance.” *Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on*, 05 1997.
- [70] C. Févotte and C. Doncarli, “Two contributions to blind source separation using time-frequency distributions,” *Signal Processing Letters, IEEE*, vol. 11, pp. 386 – 389, 04 2004.
- [71] K. Rahbar and J. Reilly, “A frequency domain method for blind source separation of convolutive audio mixtures,” *Speech and Audio Processing, IEEE Transactions on*, vol. 13, pp. 832 – 844, 10 2005.
- [72] B. Albouy and Y. Deville, “A time-frequency blind source separation method based on segmented coherence function.” vol. 2687, 06 2003, pp. 289–296.
- [73] M. Puigt and Y. Deville, “Time–frequency ratio-based blind separation methods for

- attenuated and time-delayed sources,” *Mechanical Systems and Signal Processing*, vol. 19, pp. 1348–1379, 11 2005.
- [74] —, “Two time-frequency ratio-based blind source separation methods for time-delayed mixtures,” in *International Conference on Agents*, 2006. [Online]. Available: <https://api.semanticscholar.org/CorpusID:1766496>
- [75] Y. Chao, Y. Guo-ping, L. Li-wei, Z. Yao-xian, and G. Li, “A novel algorithm for bss of frequency-hopping signals based on time frequency ratio,” 12 2009.
- [76] Y. Deville and P. White, “A new source separation approach for instantaneous mixtures based on time-frequency analysis,” 01 2001.
- [77] F. Abrard and Y. Deville, “Blind separation of dependent sources using the “time-frequency ratio of mixtures” approach,” in *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings.*, vol. 2, 2003, pp. 81–84 vol.2.
- [78] X. Yao, T. Yi, C. Qu, and H.-N. Li, “Blind modal identification using limited sensors through modified sparse component analysis by time-frequency method,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, 04 2018.
- [79] F. Ye, J. Chen, L. Gao, W. Nie, and Q. Sun, “A mixing matrix estimation algorithm for the time-delayed mixing model of the underdetermined blind source separation problem,” *Circuits, Systems, and Signal Processing*, pp. 1–18, 2018.
- [80] Q. Guo, G. Ruan, and L. Qi, “A complex-valued mixing matrix estimation algorithm for underdetermined blind source separation,” *Circuits, Systems, and Signal Processing*, vol. 37, no. 8, pp. 3206–3226, 2018.
- [81] F. Ye, J. Chen, L. Gao, W. Nie, and Q. Sun, “A mixing matrix estimation algorithm for the time-delayed mixing model of the underdetermined blind source separation problem,” *Circuits, Systems, and Signal Processing*, vol. 38, pp. 1–18, 08 2018.
- [82] N. Saulig, V. Sucic, B. Boashash, and D. Sersic, “A comparison of quadratic tfds for entropy based detection of components time supports in multicomponent non-stationary signal mixtures,” in *2013 8TH INTERNATIONAL WORKSHOP ON*

- SYSTEMS, SIGNAL PROCESSING AND THEIR APPLICATIONS (WOSSPA)*. Ctr Dev Technologies Avancees, 2013, pp. 435–441, 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA), Zeralda, ALGERIA, MAY 12-15, 2013.
- [83] R. V. Allen, “Automatic earthquake recognition and timing from single traces,” *Bulletin of the Seismological Society of America*, vol. 68, no. 5, pp. 1521–1532, 10 1978.
- [84] G. Kwiatek and Y. Ben-Zion, “Theoretical limits on detection and analysis of small earthquakes,” *Journal of Geophysical Research: Solid Earth*, vol. 121, 08 2016.
- [85] A. Trnkoczy, “Understanding and parameter setting of sta/lta trigger algorithm,” 2009. [Online]. Available: <https://api.semanticscholar.org/CorpusID:145040868>
- [86] M. Withers, R. C. Aster, C. J. Young, J. I. Beiriger, M. Harris, S. G. Moore, and J. R. Trujillo, “A comparison of select trigger algorithms for automated global seismic phase and event detection,” *Bulletin of the Seismological Society of America*, 1998. [Online]. Available: <https://api.semanticscholar.org/CorpusID:14977843>
- [87] R. Allen, “Automatic phase pickers: Their present use and future prospects,” *Bulletin of the Seismological Society of America*, vol. 72, no. 6B, pp. S225–S242, 12 1982.
- [88] J. Brown, G. Beroza, and D. Shelly, “An autocorrelation method to detect low frequency earthquakes within tremor,” *Geophysical Research Letters - GEOPHYS RES LETT*, vol. 35, 08 2008.
- [89] D. Shelly, G. Beroza, and S. Ide, “Low frequency earthquake swarms and non-volcanic tremor under shikoku, japan,” *AGU Fall Meeting Abstracts*, 01 2006.
- [90] M. Hearst, S. Dumais, E. Osman, J. Platt, and B. Scholkopf, “Support vector machines,” *Intelligent Systems and their Applications, IEEE*, vol. 13, pp. 18 – 28, 08 1998.
- [91] M. Buscema, G. Massini, and G. Maurelli, “Artificial adaptive systems to predict

- the magnitude of earthquakes,” *Bollettino di Geofisica Teorica ed Applicata*, vol. 56, pp. 227–256, 06 2015.
- [92] G. Asencio-Cortés, F. Martínez-Álvarez, A. Troncoso, and A. Morales-Esteban, “Medium—large earthquake magnitude prediction in tokyo with artificial neural networks,” *Neural Comput. Appl.*, vol. 28, no. 5, p. 1043–1055, may 2017. [Online]. Available: <https://doi.org/10.1007/s00521-015-2121-7>
- [93] A. Bhatia, S. Pasari, and A. Mehta, “Earthquake forecasting using artificial neural networks,” *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-5, pp. 823–827, 2018. [Online]. Available: <https://isprs-archives.copernicus.org/articles/XLII-5/823/2018/>
- [94] T. Bhandarkar, V. K. N. Satish, S. Sridhar, R. Sivakumar, and S. Ghosh, “Earthquake trend prediction using long short-term memory rnn,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, p. 1304, 04 2019.
- [95] J. González, W. Yu, and L. Telesca, “Earthquake magnitude prediction using recurrent neural networks,” *Proceedings*, vol. 24, no. 1, 2019. [Online]. Available: <https://www.mdpi.com/2504-3900/24/1/22>
- [96] Q. Wang, Y. Guo, L. Yu, and P. Li, “Earthquake prediction based on spatio-temporal data mining: An lstm network approach,” *IEEE Transactions on Emerging Topics in Computing*, vol. 8, no. 1, pp. 148–158, 2020.
- [97] T. Perol, M. Gharbi, and M. Denolle, “Convolutional neural network for earthquake detection and location,” *Science Advances*, vol. 4, 02 2017.
- [98] S. Zhang, B. Ku, and H. Ko, “Learnable maximum amplitude structure for earthquake event classification,” *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.
- [99] B. Ku, G. Kim, J.-K. Ahn, J. Lee, and H. Ko, “Attention-based convolutional neural network for earthquake event classification,” *IEEE Geoscience and Remote Sensing Letters*, vol. 18, no. 12, pp. 2057–2061, 2021.

- [100] D. Jozinović, A. Lomax, I. Štajduhar, and A. Michellini, “Rapid prediction of earthquake ground shaking intensity using raw waveform data and a convolutional neural network,” *Geophysical Journal International*, vol. 222, no. 2, pp. 1379–1389, 05 2020.
- [101] G. Li, S. Zhang, F. Fioranelli, and H. Griffiths, “Effect of sparsity-aware time-frequency analysis on dynamic hand gesture classification with radar micro-doppler signatures,” *IET RADAR SONAR AND NAVIGATION*, vol. 12, no. 8, pp. 815–820, AUG 2018.
- [102] G. Kim, D. K. Han, and H. Ko, “Specmix : A mixed sample data augmentation method for training with time-frequency domain features,” in *INTERSPEECH 2021*, ser. Interspeech, 2021, pp. 546–550, interspeech Conference, Brno, CZECH REPUBLIC, AUG 30-SEP 03, 2021.
- [103] S. Zhang, L. Zhang, T. Zhao, and M. M. Selim, “Fault diagnosis of rotating machinery based on time-frequency image feature extraction,” *JOURNAL OF INTELLIGENT & FUZZY SYSTEMS*, vol. 39, no. 4, pp. 5193–5200, 2020.
- [104] C. Xu, J. Ren, W. Yu, Y. Jin, Z. Cao, X. Wu, and W. Jiang, “Time-frequency analysis-based deep interference classification for frequency hopping system,” *EURASIP Journal on Advances in Signal Processing*, vol. 2022, 09 2022.
- [105] R. V. Sharan and T. J. Moir, “Time-frequency image resizing using interpolation for acoustic event recognition with convolutional neural networks,” in *2019 IEEE INTERNATIONAL CONFERENCE ON SIGNALS AND SYSTEMS (ICSIGSYS)*. IEEE; IEEE Signal Proc Soc Indonesia Chapter; IEEE Commun Soc Indonesia Chapter; Telkom Univ, Sch Elect Engn, Fakultas Teknik Elektro, 2019, pp. 8–11, iEEE International Conference on Signals and Systems (ICSigSys), Bandung, INDONESIA, JUL 16-18, 2019.
- [106] Y. Miao, Y. Zakharov, V. H. Sun, J. Li, and J. Wang, “Underwater acoustic signal classification based on sparse time-frequency representation and deep learning,” *IEEE JOURNAL OF OCEANIC ENGINEERING*, vol. 46, no. 3, pp. 952–962, JUL 2021.

- [107] N. Lopac, F. Hrzic, I. P. Vuksanovic, and J. Lerga, “Detection of non-stationary gw signals in high noise from cohen’s class of time-frequency representations using deep learning,” *IEEE ACCESS*, vol. 10, pp. 2408–2428, 2022.
- [108] L. Batistić, J. Lerga, and I. Stanković, “Detection of motor imagery based on short-term entropy of time–frequency representations,” *BioMedical Engineering OnLine*, vol. 22, 05 2023.
- [109] M. Njirjak, E. Otović, D. Jozinović, J. Lerga, G. Mauša, A. Michelini, and I. Štajduhar, “The choice of time-frequency representations of non-stationary signals affects machine learning model accuracy: A case study on earthquake detection from len-db data,” *Mathematics*, vol. 10, no. 6, 2022.
- [110] S. Mousavi, Z. Weiqiang, Y. Sheng, and G. Beroza, “Cred: A deep residual network of convolutional and recurrent units for earthquake signal detection,” *Scientific Reports*, vol. 9, 07 2019.
- [111] R. Dokht, H. Kao, R. Visser, and B. Smith, “Seismic event and phase detection using time-frequency representation and convolutional neural networks,” *Seismological Research Letters*, 01 2019.
- [112] A. Z. Sha’ameri, W. A. W. Aris, S. Sadiyah, and T. A. Musa, “Reliability of seismic signal analysis for earthquake epicenter location estimation using 1 hz gps kinematic solution,” *MEASUREMENT*, vol. 182, SEP 2021.
- [113] X. Lu, Y. Xu, Y. Tian, B. Cetiner, and E. Taciroglu, “A deep learning approach to rapid regional post-event seismic damage assessment using time-frequency distributions of ground motions,” *EARTHQUAKE ENGINEERING & STRUCTURAL DYNAMICS*, vol. 50, no. 6, pp. 1612–1627, MAY 2021.
- [114] N. Arunkumar, K. Kumar, and V. Venkataraman, “Automatic detection of epileptic seizures using new entropy measures,” *Journal of Medical Imaging and Health Informatics*, vol. 6, pp. 724–730, 06 2016.
- [115] C. Zhou, F. Xu, C. Qiao, and W. Dai, “Chaos and wavelet based colonic motility analysis for subjects with constipation,” in *2016 12th International Conference*



- on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, 2016, pp. 1374–1380.
- [116] Q. Zhao, B. Hu, Y. Li, H. Peng, L. Li, Q. Liu, Y. Li, Q. Shi, and J. Feng, “An alpha resting eeg study on nonlinear dynamic analysis for schizophrenia,” 11 2013, pp. 484–488.
- [117] Y. Li, X. Wang, S. Si, and S. Huang, “Entropy based fault classification using the case western reserve university data: A benchmark study,” *IEEE TRANSACTIONS ON RELIABILITY*, vol. 69, no. 2, pp. 754–767, JUN 2020.
- [118] J. Xiang, Q. Li, A. Wang, and Y. Zhang, “Mathematical analysis of characteristic pressure fluctuations in a bubbling fluidized bed,” *Powder Technology*, vol. 333, 04 2018.
- [119] C. Wang, Y. Li, and M. Gao, “Target classification using renyi entropy features of cyclic bispectrum,” *JOURNAL OF ENGINEERING-JOE*, vol. 2019, no. 21, pp. 7717–7720, NOV 2019.
- [120] A. Asgharzadeh-Bonab, M. C. Amirani, and A. Mehri, “Spectral entropy and deep convolutional neural network for eeg beat classification,” *Biocybernetics and Biomedical Engineering*, vol. 40, no. 2, pp. 691–700, 2020.
- [121] A. R. See, I. S. Cabili, and Y.-J. Chen, “Heart sound classification based on non-linear time-frequency features,” *SENSORS AND MATERIALS*, vol. 34, no. 1, pp. 217–223, 2022.
- [122] M. Li, X. Sun, W. Chen, Y. Jiang, and T. Zhang, “Classification epileptic seizures in eeg using time-frequency image and block texture features,” *IEEE ACCESS*, vol. 8, pp. 9770–9781, 2020.
- [123] C. Leracitano, N. Mammone, A. Bramanti, S. Marino, A. Hussain, and F. C. Morabito, “A time-frequency based machine learning system for brain states classification via eeg signal processing,” in *2019 INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN)*, ser. IEEE International Joint Conference on Neural Networks (IJCNN), 2019, international Joint Conference on Neural Networks (IJCNN), Budapest, HUNGARY, JUL 14-19, 2019.

- [124] A. Asgharzadeh-Bonab, M. C. Amirani, and A. Mehri, “Spectral entropy and deep convolutional neural network for ecg beat classification,” *BIOCYBERNETICS AND BIOMEDICAL ENGINEERING*, vol. 40, no. 2, pp. 691–700, APR-JUN 2020.
- [125] C. Uyulan, T. T. Erguzel, and N. Tarhan, “Entropy-based feature extraction technique in conjunction with wavelet packet transform for multi-mental task classification,” *BIOMEDICAL ENGINEERING-BIOMEDIZINISCHE TECHNIK*, vol. 64, no. 5, pp. 529–542, OCT 2019.
- [126] J. Lerga, M. Vrankic, and V. Sucic, “A signal denoising method based on the improved ici rule,” *IEEE Signal Processing Letters*, vol. 15, pp. 601–604, 2008.
- [127] G. Segon, J. Lerga, and V. Sucic, “Improved lpa-ici-based estimators embedded in a signal denoising virtual instrument,” *Signal, Image and Video Processing*, vol. 11, 06 2016.
- [128] J. Lerga, M. Franušić, and V. Sucic, “Parameters analysis for the time-varying automatically adjusted lpa based estimators,” *Journal of Automation and Control Engineering*, vol. 2, pp. 203–208, 01 2014.
- [129] G. Blanco, A. Traina, C. Jr, P. Azevedo-Marques, A. E. Jorge, D. de Oliveira, and M. Bedo, “A superpixel-driven deep learning approach for the analysis of dermatological wounds,” 09 2019.
- [130] H. Li, H. Li, J. Kang, Y. Feng, and J. Xu, “Automatic detection of parapapillary atrophy and its association with children myopia,” *Computer Methods and Programs in Biomedicine*, vol. 183, p. 105090, 09 2019.
- [131] F. Bayer, A. Kozakevicius, and R. Cintra, “An iterative wavelet threshold for signal denoising,” *Signal Processing*, 04 2019.
- [132] M. Sharma, S. Singh, A. Kumar, R. S. Tan, and U. R. Acharya, “Automated detection of shockable and non-shockable arrhythmia using novel wavelet-based ecg features,” *Computers in Biology and Medicine*, vol. 115, 09 2019.
- [133] J. Lee, S.-J. Lee, M. Choi, M. Seo, and S. W. Kim, “Qrs detection method based

- on fully convolutional networks for capacitive electrocardiogram,” *Expert Systems with Applications*, vol. 134, 05 2019.
- [134] F. J. Valverde-Albacete, and C. Peláez-Moreno, “The case for shifting the Rényi entropy,” *Entropy*, vol. 21, no. 1, 2019.
- [135] A. M. Fink and M. Jodeit, “A generalization of the arithmetic-geometric means inequality,” *Proceedings of the American Mathematical Society*, vol. 61, pp. 255–261, 1976.
- [136] P. Landsberg, “A generalized mean,” *Journal of Mathematical Analysis and Applications*, vol. 76, no. 1, pp. 209–212, 1980. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0022247X80900736>
- [137] J. Dombi, “The generalized dombi operator family and the multiplicative utility function,” in *Soft Computing Based Modeling in Intelligent Systems*. Springer, 2009, pp. 115–131.
- [138] D. Pearce and H.-G. Hirsch, “The aurora experimental framework for the performance evaluations of speech recognition systems under noisy condition,” vol. 4, 01 2000, pp. 29–32.
- [139] Y. Hu and P. Loizou, “Evaluation of objective measures for speech enhancement,” vol. 3, 01 2006.
- [140] M. Bahrami and N. Faraji, “Minimum mean square error estimator for speech enhancement in additive noise assuming weibull speech priors and speech presence uncertainty,” *International Journal of Speech Technology*, vol. 24, pp. 1–12, 03 2021.
- [141] M. Madruga, Y. Campos-Roca, and C. J. Pérez, “Multicondition training for noise-robust detection of benign vocal fold lesions from recorded speech,” *IEEE Access*, vol. 9, pp. 1707–1722, 2021.
- [142] M. Strake, B. Defraene, K. Fluyt, W. Tirry, and T. Fingscheidt, “Speech enhancement by LSTM-based noise suppression followed by CNN-based speech restoration,” *EURASIP Journal on Advances in Signal Processing*, vol. 2020, no. 1, dec 2020. [Online]. Available: <https://doi.org/10.1186%2Fs13634-020-00707-1>

- [143] J. Ma, Y. Hu, and P. Loizou, “Objective measures for predicting speech intelligibility in noisy conditions based on new band-importance functions,” *The Journal of the Acoustical Society of America*, vol. 125, pp. 3387–405, 06 2009.
- [144] Y. Hu and P. C. Loizou, “Evaluation of objective quality measures for speech enhancement,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, 2008.



# LIST OF ABBREVIATIONS

**ANN** Artificial Neural Network.

**AUROC** Area under the receiver operating characteristic.

**BSS** Blind-source separation.

**CNN** Convolutional Neural Networks.

**CWD** Choi-William distribution.

**EEG** Electroencephalogram.

**FC** Fully connected.

**FFNN** Feedforward Neural Network.

**FN** False negatives.

**FP** False positives.

**FT** Fourier transform.

**GPS** Global Positioning System.

**IAF** Instantaneous autocorrelation function.

**ICA** Independent component analysis.

**ILSVR** ImageNet large scale visual recognition challenge.

**LSTM** Long Short-Term Memory.

**LTA** Long-Term Averaging.

**MCC** Matthews correlation coefficient.

**PCA** Principal Component Analysis.

**PWVD** Pseudo-Wigner-Ville distribution.

**ReLU** Rectified Linear Unit.

**RICI** Relative intersection of confidence intervals.

**RID** Reduced interference distribution.

**RIDB** Reduced interference distribution with Bessel function.

**SNR** Signal-to-noise ration.

**SPWVD)** Smoothed pseudo-Wigner-Ville distribution.

**STA** Short-Term Averaging.

**STFT** Short-time Fourier transform.

**SVD** Singular Value Decomposition.

**SVM** Support Vector Machines.

**TFD** Time frequency distribution.

**TN** True Negatives.

**TP** True Positive.

**WVD** Wigner-Vile distribution.

**ZAMD** Zhao-Atlas-Mark distribution.

# LIST OF SYMBOLS

$t$	continuous time
$f$	continuous frequency
$\tau$	continuous lag
$\omega(t)$	smoothing window
$x(t)$	signal
$g$	time smoothing window
$h$	frequency smoothing window
$p_i$	probability value
$H$	entropy
$H_d$	Dombi entropy
$H_r$	Rényi entropy
$H_s$	Shannon entropy
$H_t$	Tsallis entropy
$\alpha$	entropy order
$\delta$	window size
$H_{(t,f)}^\delta$	Rényi entropy for window size
$H_{(t,f)}^{\text{RICI}}$	optimal Rényi entropy
$M$	frequency bins
$N$	number of samples
$\sigma$	estimation variance
$\beta$	constant
$D$	confidence interval
$L$	lower limit of D
$U$	upper limit of D



$\bar{L}$	the largest lower limit of $D$
$\underline{U}$	the smallest upper limit of $D$
$\Gamma$	threshold parameter
$O$	overlapping confidence intervals
$R(n, \delta)$	ratio of finite interval and confidence interval
$R_c$	threshold
$\delta^+$	optimal window size
$E$	energy
$M_p$	distribution
$\chi$	entropy mask
$\tau^+$	optimal threshold
$S_z(t, f)$	Spectrogram representation
$R_s(t, \tau)$	instantaneous autocorrelation function
$WVD_s(t, f)$	Wigner-Vile distribution
$PWVD_s(t, f)$	pseudo-Wigner-Ville distribution
$SPWVD_s(t, f)$	smoothed pseudo-Wigner-Ville distribution
$CWD_s(t, f)$	Choi-Williams distribution
$ZAMD_s(t, f)$	Zhao-Atlas-Mark distribution
$RIDB_s(t, f)$	reduced interference distribution
$R_s(t, \tau)$	kernel
$p(x)$	probability
$X$	random variable
$I(x)$	information
$I$	input matrix
$K$	filter matrix
$S$	output feature map
$q$	constant
$A$	fuzzy set
$\mu_A(x)$	membership function
$H_f(A)$	entropy of a fuzzy set
$H_\alpha^\beta(A)$	entropy of type $(\alpha, \beta)$ on fuzzy set

$Y$  sample vector

$w$	set of parameters
$f(X; w)$	model
$L(y, f(X; w))$	inaccuracy
$A^{(\alpha)}(\mathbf{x}, \mathbf{w})$	generalized arithmetic mean
$G(\mathbf{x}, \mathbf{w})$	generalized geometric mean
$F(x)$	transformation
$D$	Dombi measure



# LIST OF FIGURES

2.1	Example of a two-component non-stationary signal . . . . .	9
2.2	Example of a two-component non-stationary signal with added noise(SNR=5 dB) . . . . .	9
2.3	Spectrogram of the example of a two-component non-stationary signal $s(t)$ with a parabolic and a linear frequency modulated component with and without added noise . . . . .	14
2.4	SPWVD of the example of a two-component non-stationary signal $s(t)$ with a parabolic and a linear frequency modulated component with and without added noise . . . . .	14
2.5	CWD of the example of a two-component non-stationary signal $s(t)$ with a parabolic and a linear frequency modulated component with and without added noise . . . . .	15
2.6	ZAMD of the example of a two-component non-stationary signal $s(t)$ with a parabolic and a linear frequency modulated component with and without added noise . . . . .	15
2.7	RIDB of the example of a two-component non-stationary signal $s(t)$ with a parabolic and a linear frequency modulated component with and without added noise . . . . .	15
3.1	Rényi entropy( $H_R^{(\alpha)}$ ) for different $\alpha$ values when $n = 2$ ; for $\alpha = 1$ then $H_R^{(\alpha)} = H_S$ . . . . .	21
3.2	Tsallis entropy for different $q$ values when $n = 2$ ; when $q \rightarrow 1$ then $H_T = H_S$	23
6.1	TFDs for signal A . . . . .	47
6.2	TFDs for signal B . . . . .	48

6.3	TFDs for signal C . . . . .	49
6.4	Example of entropy map with different fixed window sizes for signal A . . .	50
6.5	Example of entropy map with different fixed window sizes for signal B . . .	50
6.6	Example of entropy map with different fixed window sizes for signal C . . .	51
6.7	Entropy values in one point of TFD for different window sizes . . . . .	51
6.8	Number of occurrences in one example of each window size . . . . .	53
6.9	Entropy map and mask of an example signals TFD . . . . .	54
6.10	Energy calculations for different thresholds . . . . .	54
6.11	2DLEM method steps . . . . .	55
6.12	Dombi entropy measure for different $\alpha$ and $\beta$ values for $n = 2$ . . . . .	61
6.13	Comparison of the Dombi entropy measure and the Shannon entropy . . .	62
7.1	Noisy TFDs of the signal A with SNR = -3 . . . . .	64
7.2	Noisy TFDs of the signal A with SNR = 0 . . . . .	65
7.3	Noisy TFDs of the signal A with SNR = 5 . . . . .	66
7.4	Noisy TFDs of the signal B with SNR = -3 . . . . .	67
7.5	Noisy TFDs of the signal B with SNR = 0 . . . . .	68
7.6	Noisy TFDs of the signal B with SNR = 5 . . . . .	69
7.7	Noisy TFDs of the signal C with SNR = -3 . . . . .	70
7.8	Noisy TFDs of the signal C with SNR = 0 . . . . .	71
7.9	Noisy TFDs of the signal C with SNR = 5 . . . . .	72
7.10	Entropy maps and masks showing signal components for different TFDs for Shannon entropy and SNR = -3 in case of signal A . . . . .	74
7.11	Entropy maps and masks showing signal components for different TFDs for Shannon entropy and SNR = -3 in case of signal B . . . . .	76
7.12	Entropy maps and masks showing signal components for different TFDs for Shannon entropy and SNR = -3 in case of signal C . . . . .	77
7.13	Entropy maps and masks for different TFDs for Rényi entropy and SNR = -3 in case of signal A . . . . .	80
7.14	Entropy maps and masks for different TFDs for Rényi entropy and SNR = -3 in case of signal B . . . . .	81

7.15	Entropy maps and masks for different TFDs for Rényi entropy and SNR = -3 in case of signal C . . . . .	83
7.16	Entropy maps and masks for different TFDs for Tsallis entropy and SNR = -3 in case of signal A . . . . .	85
7.17	Entropy maps and masks for different TFDs for Tsallis entropy and SNR = -3 in case of signal B . . . . .	86
7.18	Entropy maps and masks for different TFDs for Tsallis entropy and SNR = -3 in case of signal C . . . . .	87
7.19	Entropy maps and masks for different TFDs for Dombi entropy and SNR = -3 in case of signal A . . . . .	89
7.20	Entropy maps and masks for different TFDs for Dombi entropy and SNR = -3 in case of signal B . . . . .	91
7.21	Entropy maps and masks for different TFDs for Dombi entropy and SNR = -3 in case of signal C . . . . .	92
7.22	Two randomly sampled seismogram instances from the dataset . . . . .	102
7.23	Spectrograms obtained from earthquake sample(a) and noise sample(b) . .	103
7.24	Entropy maps obtained from earthquake sample(a) and noise sample(b) . .	104
7.25	RGB images obtained from earthquake sample spectrogram(a) and earthquake sample entropy map(b) . . . . .	105
7.26	RGB images obtained from noise sample spectrogram(a) and noise sample entropy map(b) . . . . .	105
7.27	Confusion matrices for evaluating the CNN models. (a) Spectrogram distributions as input to ResNet50, (b) Entropy maps as input to ResNet50,(c) Spectrogram distributions as input to VGG19, (d) Entropy maps as input to VGG19,(e) Spectrogram distributions as input to DenseNet121, (f) Entropy maps as input to DenseNet121) . . . . .	106
1	Entropy maps and masks for different TFDs for Shannon entropy and SNR = 0 in case of signal A . . . . .	145
2	Entropy maps and masks for different TFDs for Shannon entropy and SNR = 0 in case of signal B . . . . .	145

3	Entropy maps and masks for different TFDs for Shannon entropy and SNR = 0 in case of signal C . . . . .	146
4	Entropy maps and masks for different TFDs for Shannon entropy and SNR =5 in case of signal A . . . . .	146
5	Entropy maps and masks for different TFDs for Shannon entropy and SNR =5 in case of signal B . . . . .	147
6	Entropy maps and masks for different TFDs for Shannon entropy and SNR = 5 in case of signal C . . . . .	147
7	Entropy maps and masks for different TFDs for Rényi entropy and SNR = 0 in case of signal A . . . . .	148
8	Entropy maps and masks for different TFDs for Rényi entropy and SNR = 0 in case of signal B . . . . .	148
9	Entropy maps and masks for different TFDs for Rényi entropy and SNR = 0 in case of signal C . . . . .	149
10	Entropy maps and masks for different TFDs for Rényi entropy and SNR =5 in case of signal A . . . . .	149
11	Entropy maps and masks for different TFDs for Rényi entropy and SNR =5 in case of signal B . . . . .	150
12	Entropy maps and masks for different TFDs for Rényi entropy and SNR = 5 in case of signal C . . . . .	150
13	Entropy maps and masks for different TFDs for Tsallis entropy and SNR = 0 in case of signal A . . . . .	151
14	Entropy maps and masks for different TFDs for Tsallis entropy and SNR = 0 in case of signal B . . . . .	151
15	Entropy maps and masks for different TFDs for Tsallis entropy and SNR = 0 in case of signal C . . . . .	152
16	Entropy maps and masks for different TFDs for Tsallis entropy and SNR =5 in case of signal A . . . . .	152
17	Entropy maps and masks for different TFDs for Tsallis entropy and SNR =5 in case of signal B . . . . .	153
18	Entropy maps and masks for different TFDs for Tsallis entropy and SNR = 5 in case of signal C . . . . .	153

19	Entropy maps and masks for different TFDs for Dombi entropy and SNR = 0 in case of signal A . . . . .	154
20	Entropy maps and masks for different TFDs for Dombi entropy and SNR = 0 in case of signal B . . . . .	154
21	Entropy maps and masks for different TFDs for Dombi entropy and SNR = 0 in case of signal C . . . . .	155
22	Entropy maps and masks for different TFDs for Dombi entropy and SNR =5 in case of signal A . . . . .	155
23	Entropy maps and masks for different TFDs for Dombi entropy and SNR =5 in case of signal B . . . . .	156
24	Entropy maps and masks for different TFDs for Dombi entropy and SNR = 5 in case of signal C . . . . .	156





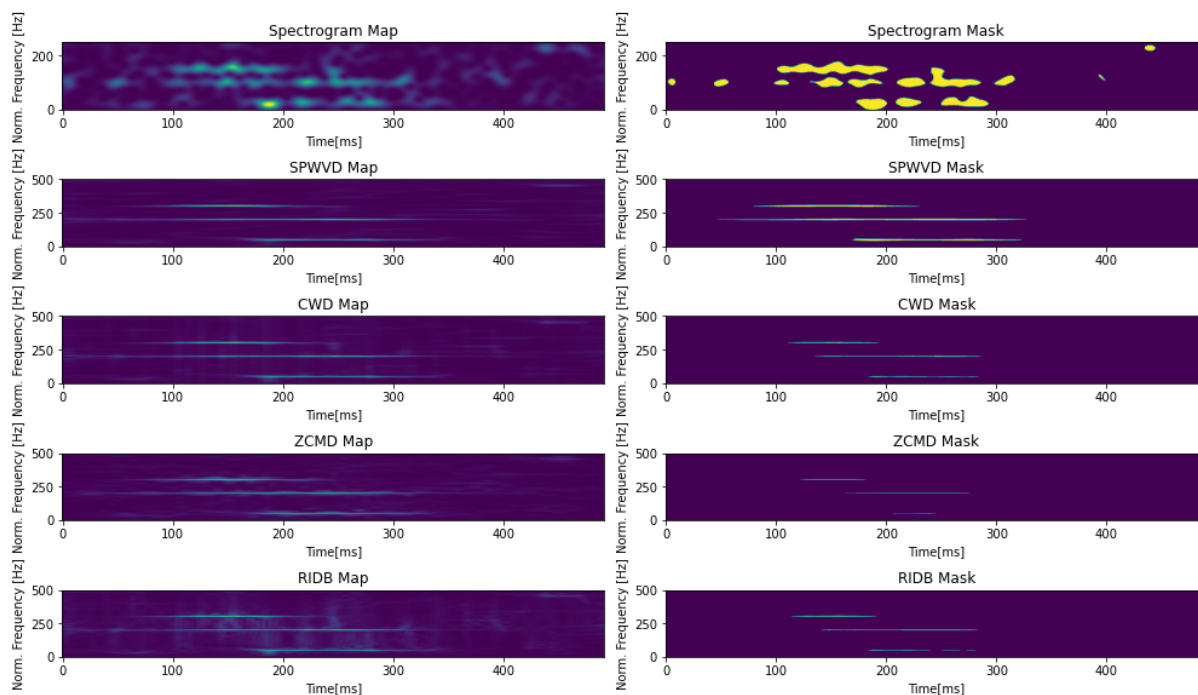
# LIST OF TABLES

4.1	Layers in the VGG16 Architecture . . . . .	33
4.2	Layers in the ResNet-50 Architecture . . . . .	34
4.3	Layers in the DenseNet-101 Architecture . . . . .	35
7.1	Results obtained with Shannon entropy from signal A . . . . .	73
7.2	Results obtained with Shannon entropy from signal B . . . . .	75
7.3	Results obtained with Shannon entropy from signal C . . . . .	75
7.4	Results of pair-wise comparison between different TFDs for signal A when SNR=-3 . . . . .	78
7.5	Results of pair-wise comparison between different TFDs for signal A when SNR=0 . . . . .	78
7.6	Results of pair-wise comparison between different TFDs for signal A when SNR=5 . . . . .	79
7.7	Results obtained with Rényi entropy from signal A . . . . .	82
7.8	Results obtained with Rényi entropy from signal B . . . . .	82
7.9	Results obtained with Rényi entropy from signal C . . . . .	84
7.10	Results obtained with Tsallis entropy from signal A . . . . .	84
7.11	Results obtained with Tsallis entropy from signal B . . . . .	88
7.12	Results obtained with Tsallis entropy from signal C . . . . .	88
7.13	Results obtained with Dombi entropy from signal A . . . . .	90
7.14	Results obtained with Dombi entropy from signal B . . . . .	90
7.15	Results obtained with Dombi entropy from signal C . . . . .	93
7.16	Results obtained for speech signals with added noise for different sources using Shannon entropy . . . . .	97

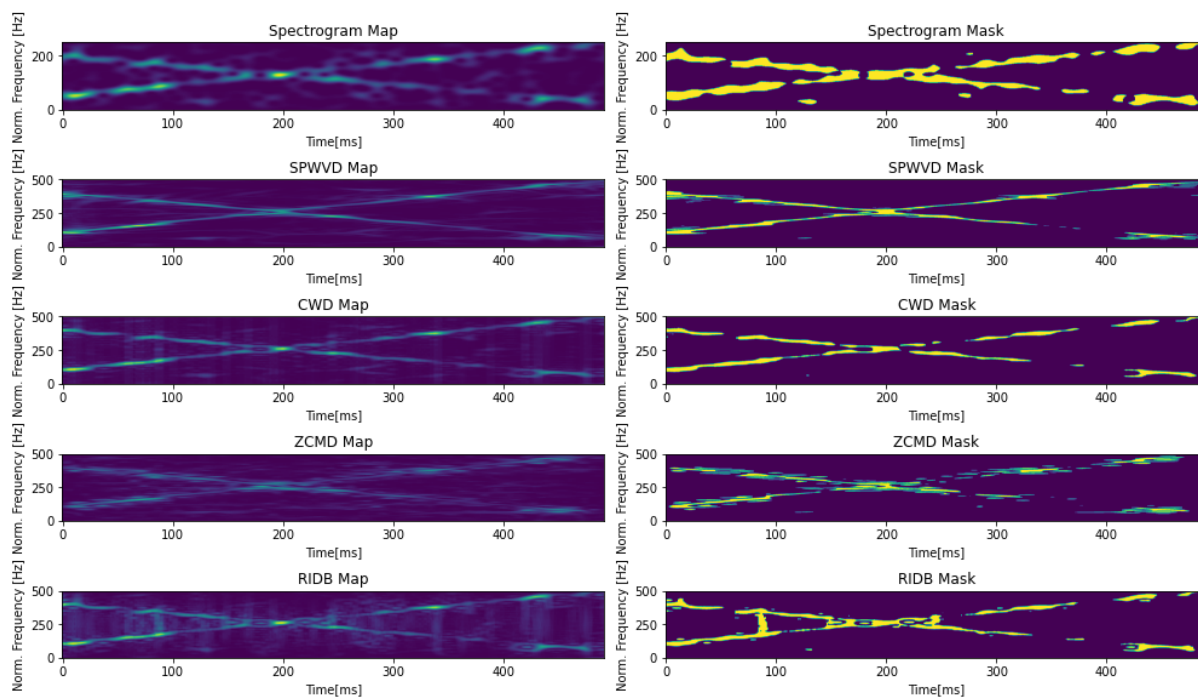
7.17 Results obtained for speech signals with added noise for different sources using Rényi entropy . . . . .	98
7.18 Results obtained for speech signals with added noise for different sources using Tsallis entropy . . . . .	99
7.19 Results obtained for speech signals with added noise for different sources using Dombi entropy . . . . .	100
7.20 Classification results on the test set . . . . .	107
7.21 P-values for all models comparing both representations . . . . .	108

# Appendices

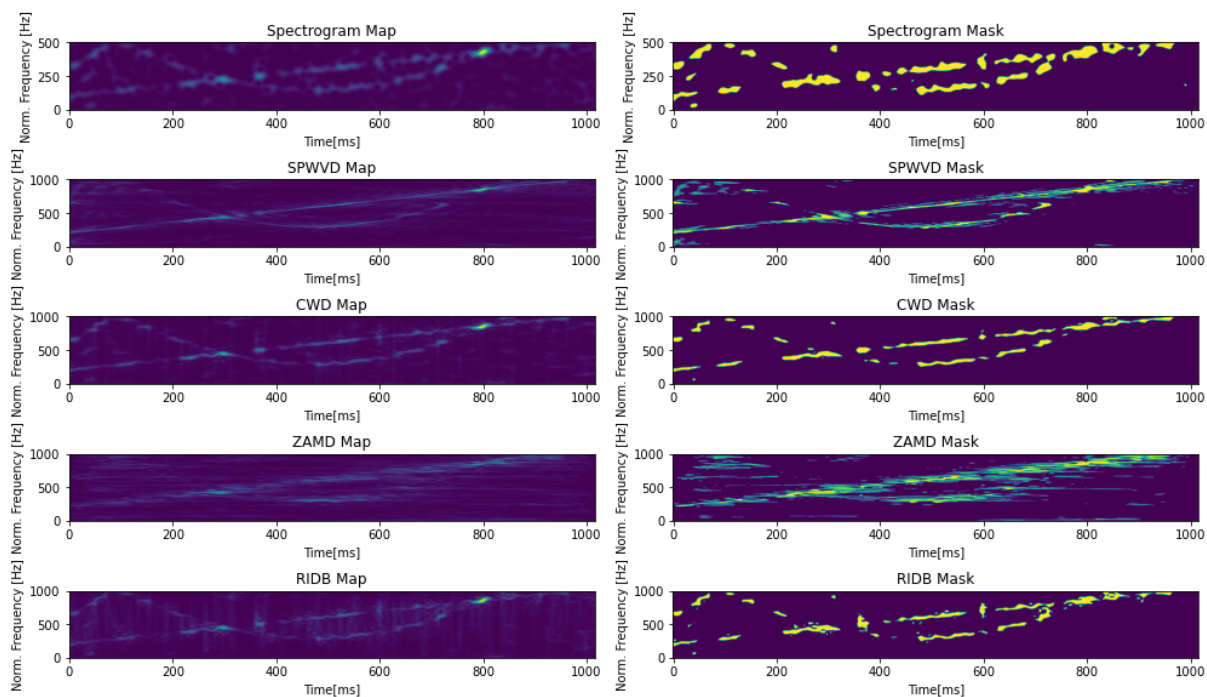




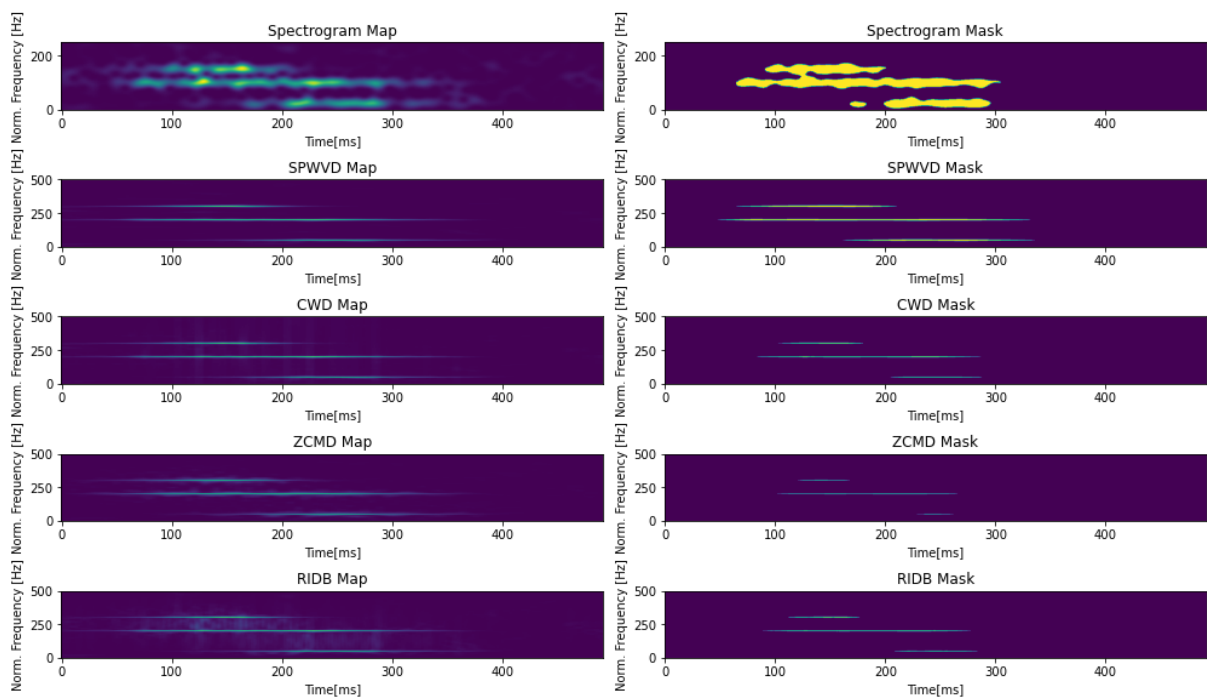
**Figure 1:** Entropy maps and masks for different TFDs for Shannon entropy and  $\text{SNR} = 0$  in case of signal A



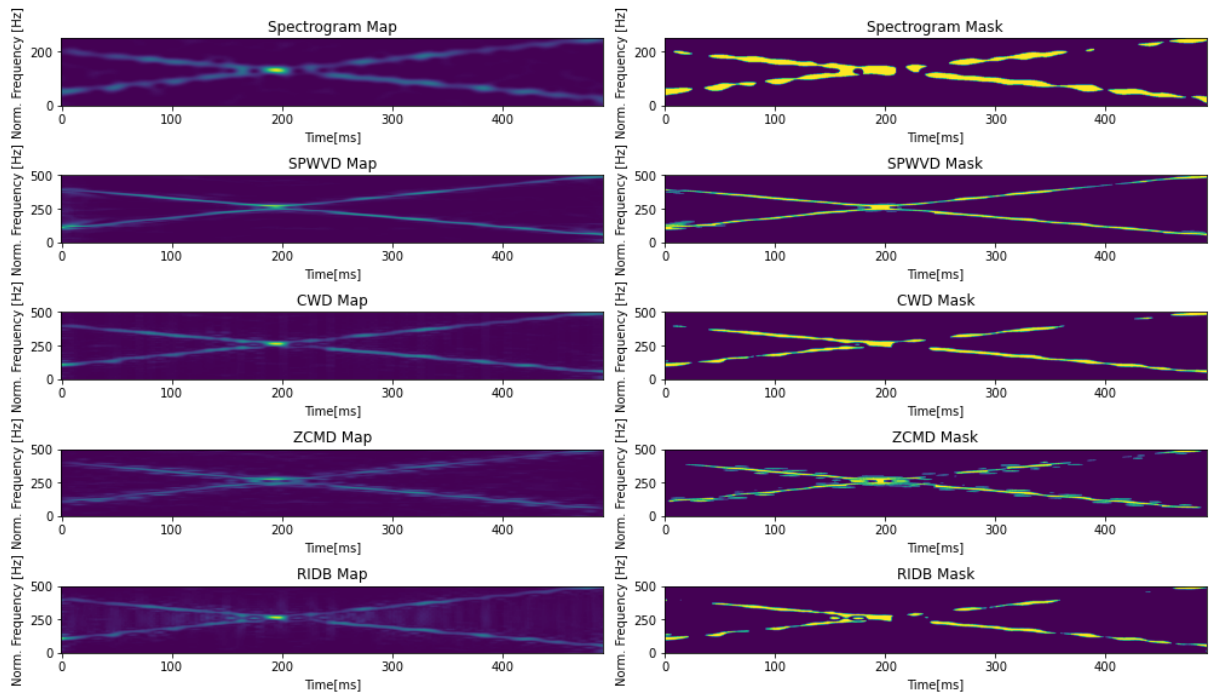
**Figure 2:** Entropy maps and masks for different TFDs for Shannon entropy and  $\text{SNR} = 0$  in case of signal B



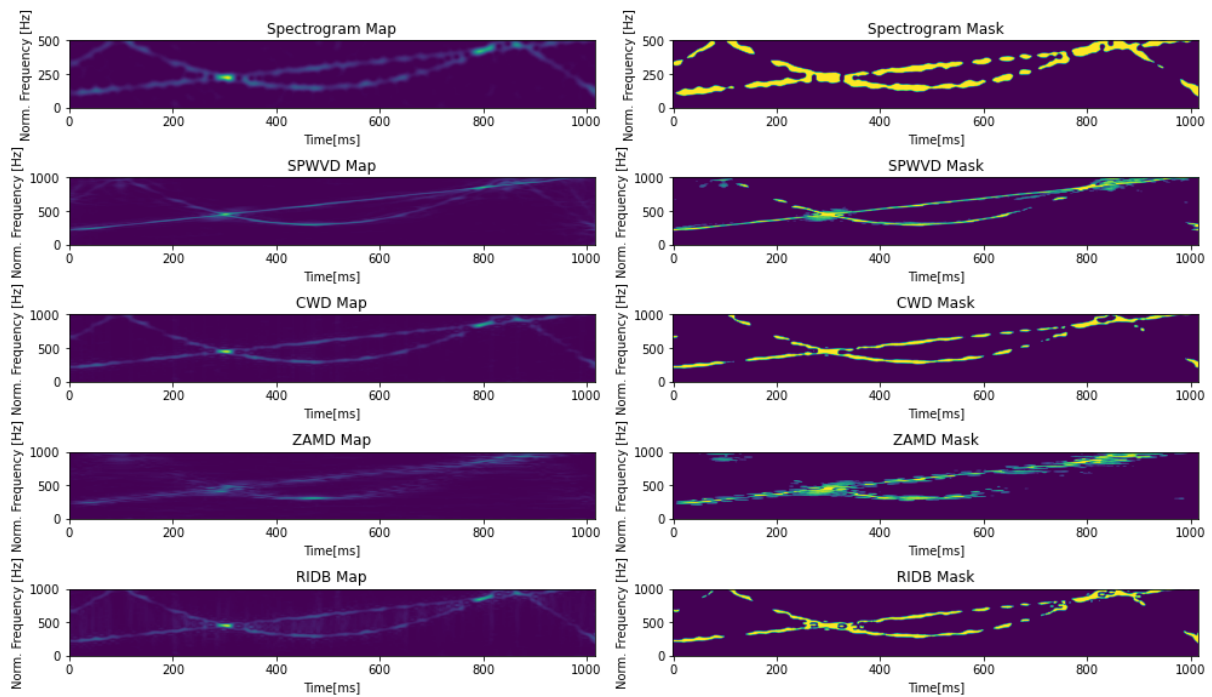
**Figure 3:** Entropy maps and masks for different TFDs for Shannon entropy and  $SNR = 0$  in case of signal C



**Figure 4:** Entropy maps and masks for different TFDs for Shannon entropy and  $SNR = 5$  in case of signal A

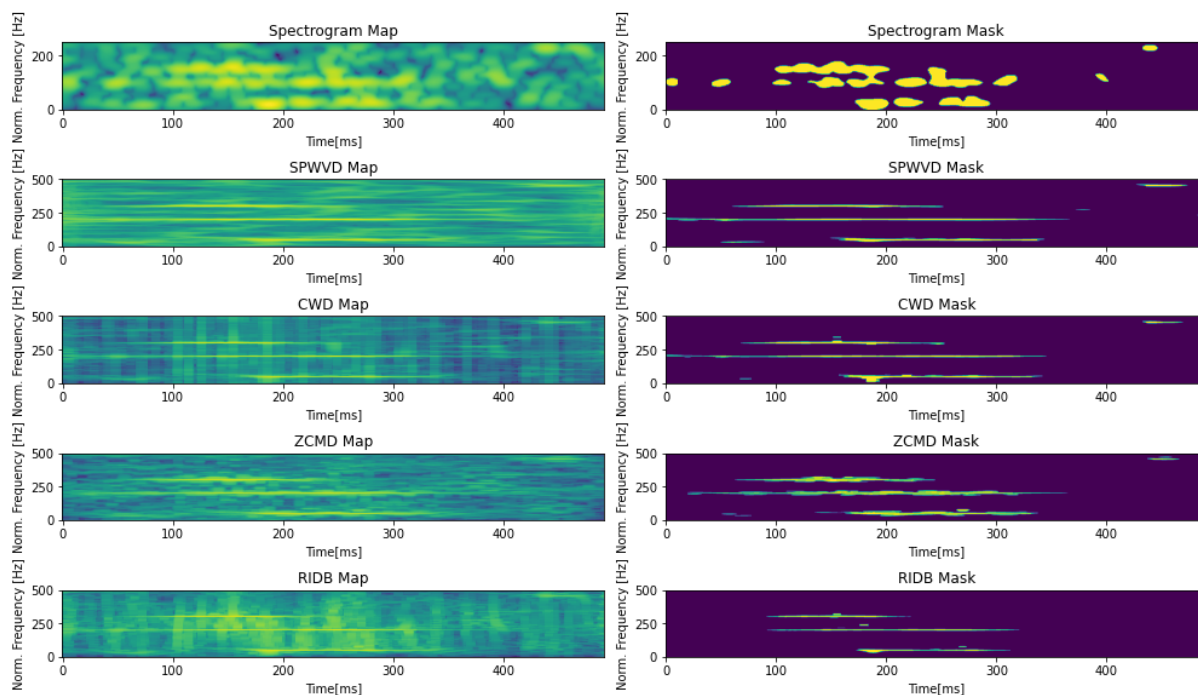


**Figure 5:** Entropy maps and masks for different TFDs for Shannon entropy and  $\text{SNR} = 5$  in case of signal B

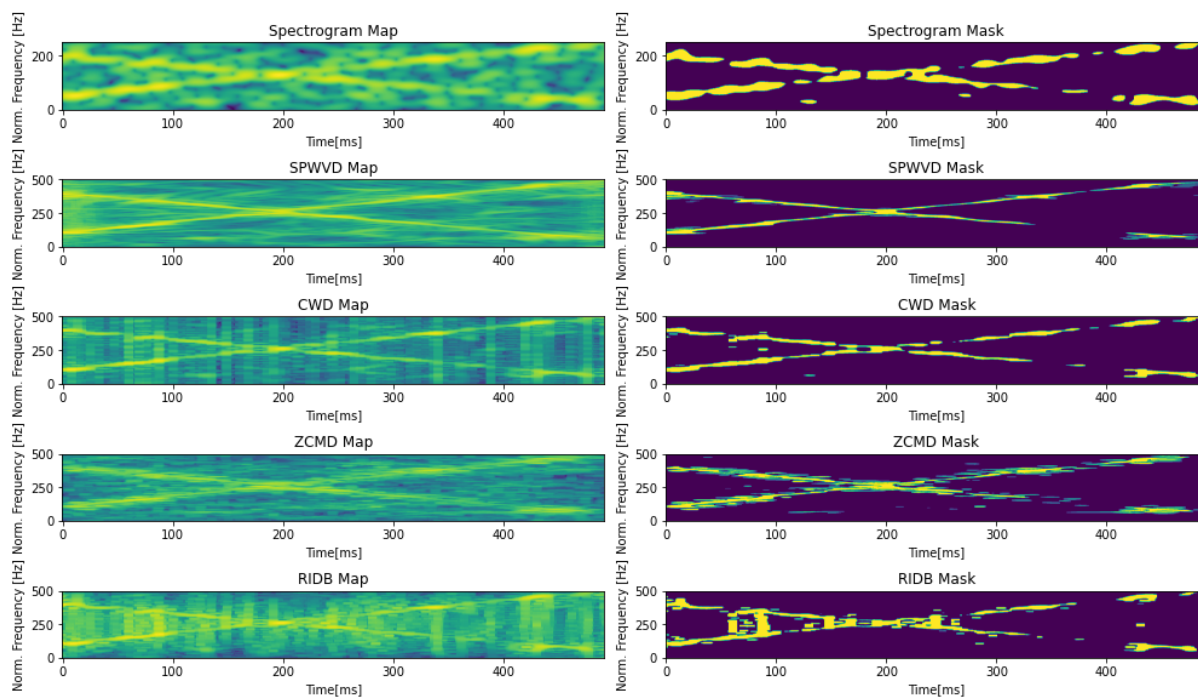


**Figure 6:** Entropy maps and masks for different TFDs for Shannon entropy and  $\text{SNR} = 5$  in case of signal C

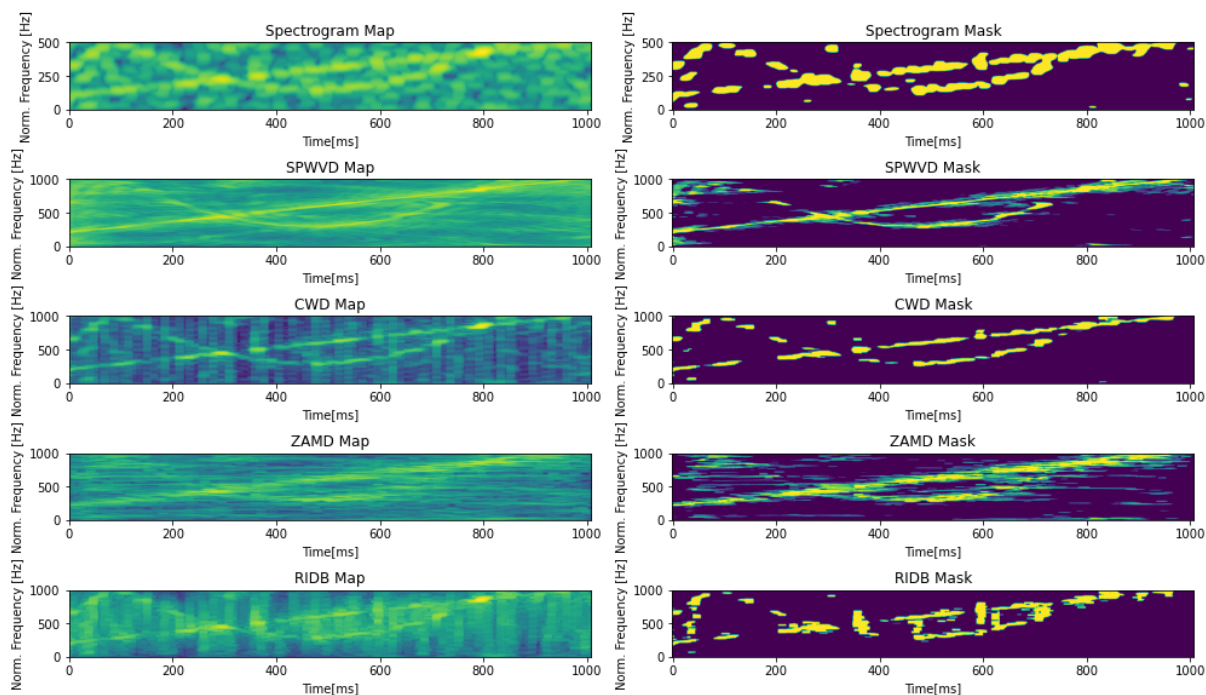




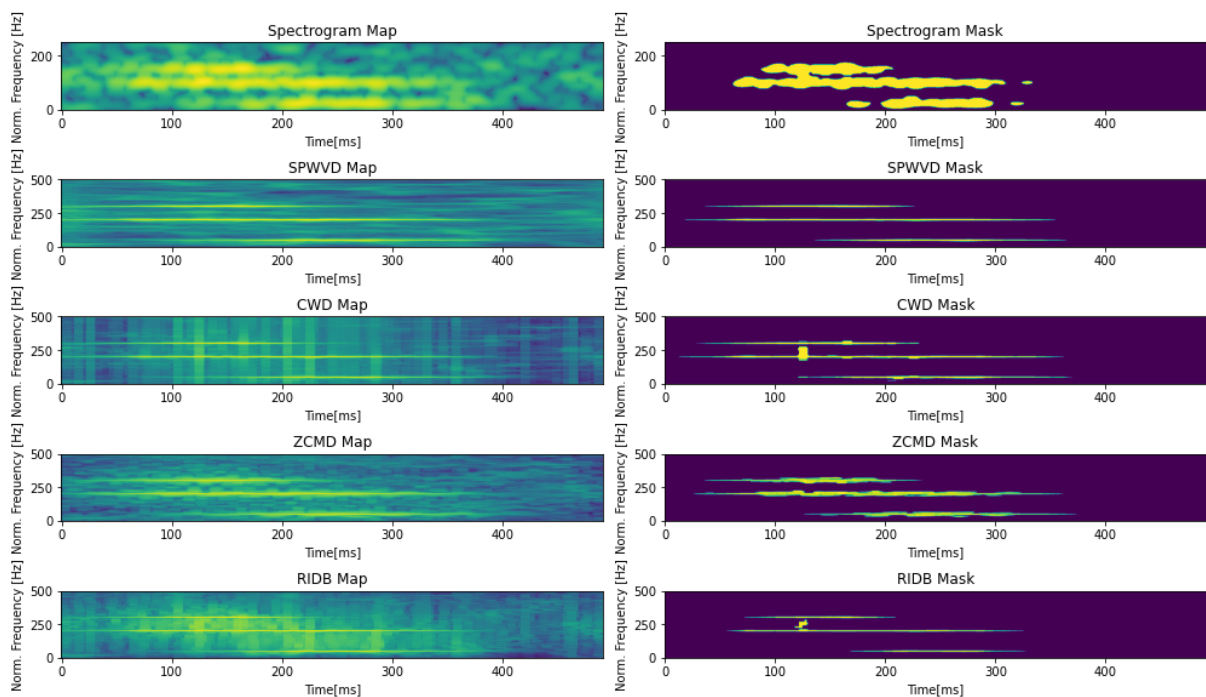
**Figure 7:** Entropy maps and masks for different TFDs for Rényi entropy and  $\text{SNR} = 0$  in case of signal A



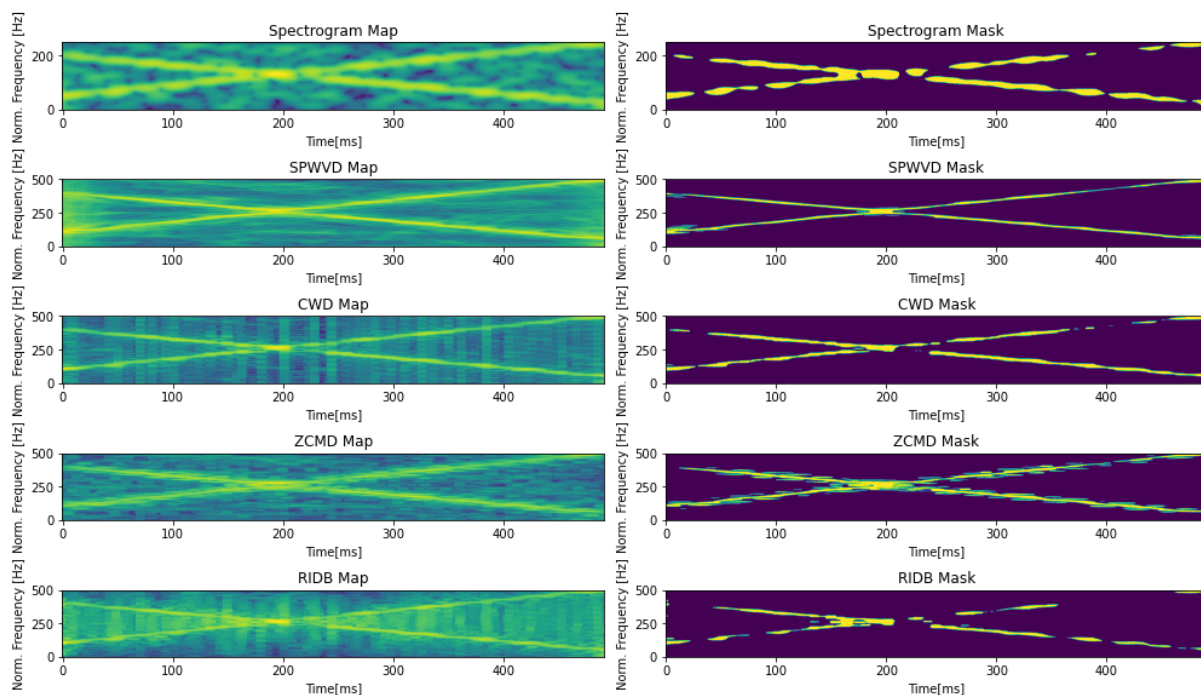
**Figure 8:** Entropy maps and masks for different TFDs for Rényi entropy and  $\text{SNR} = 0$  in case of signal B



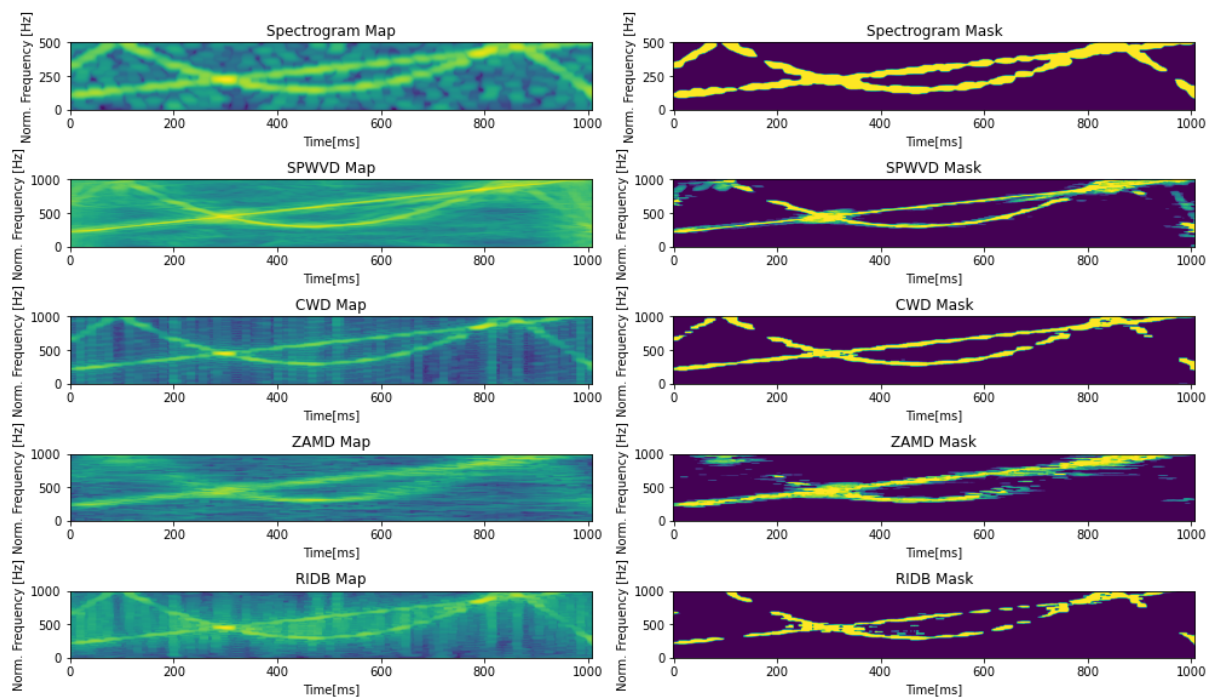
**Figure 9:** Entropy maps and masks for different TFDs for Rényi entropy and  $\text{SNR} = 0$  in case of signal C



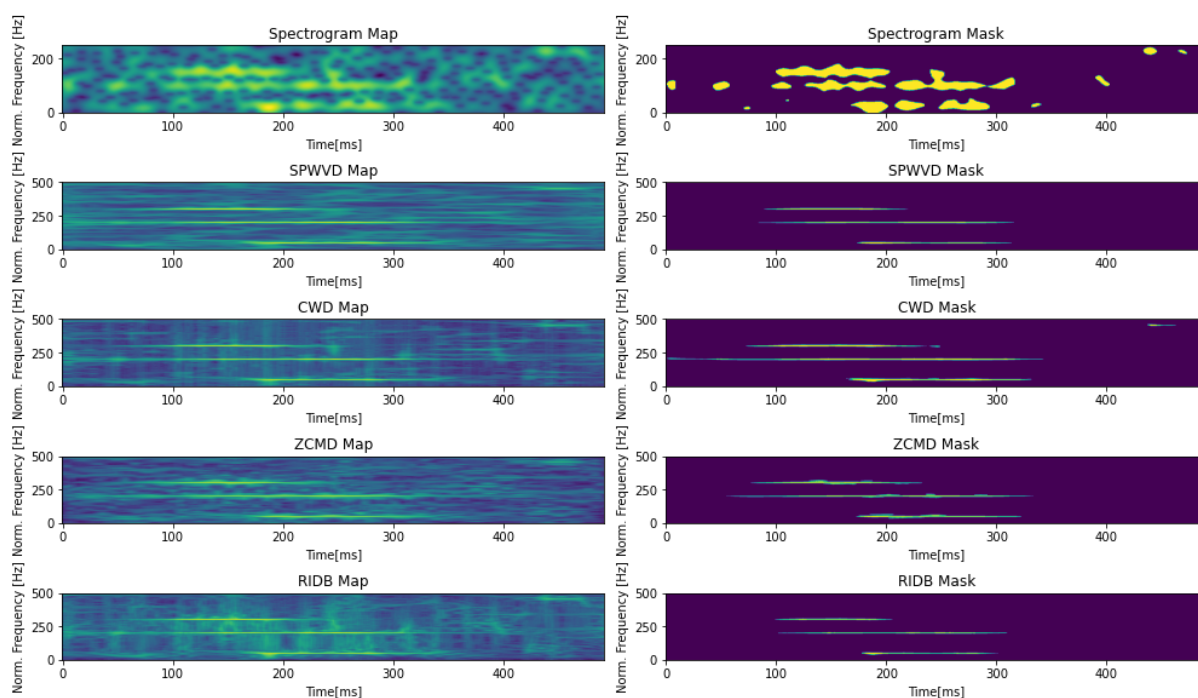
**Figure 10:** Entropy maps and masks for different TFDs for Rényi entropy and  $\text{SNR} = 5$  in case of signal A



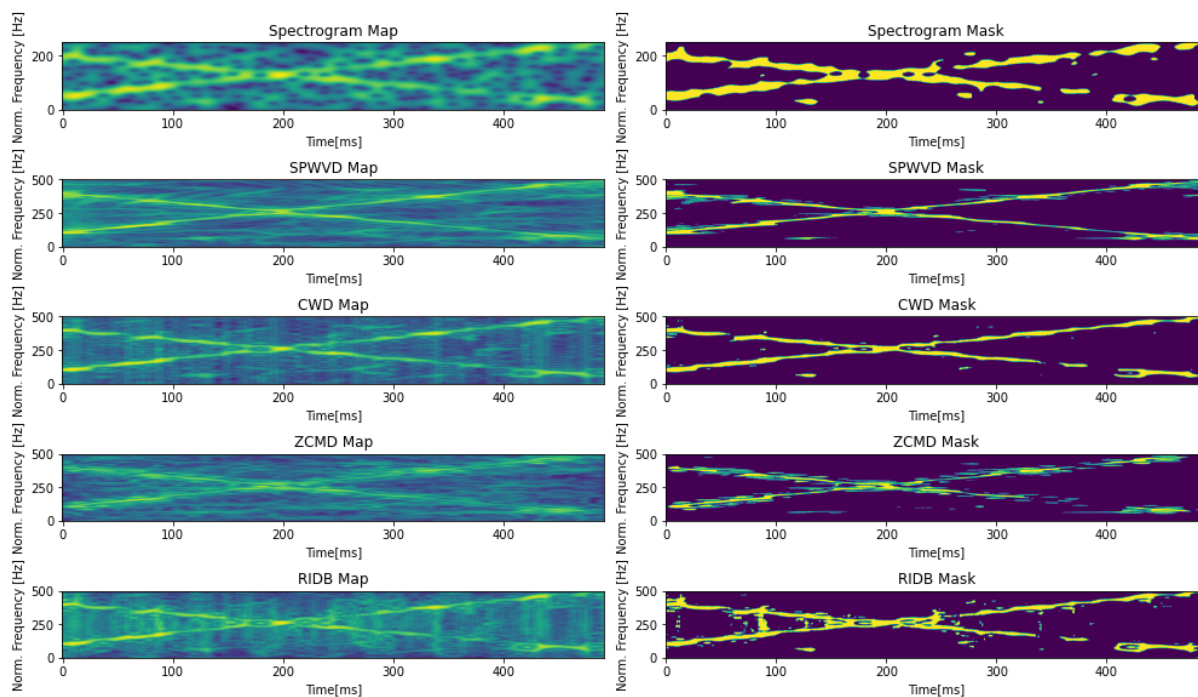
**Figure 11:** Entropy maps and masks for different TFDs for Rényi entropy and  $SNR = 5$  in case of signal B



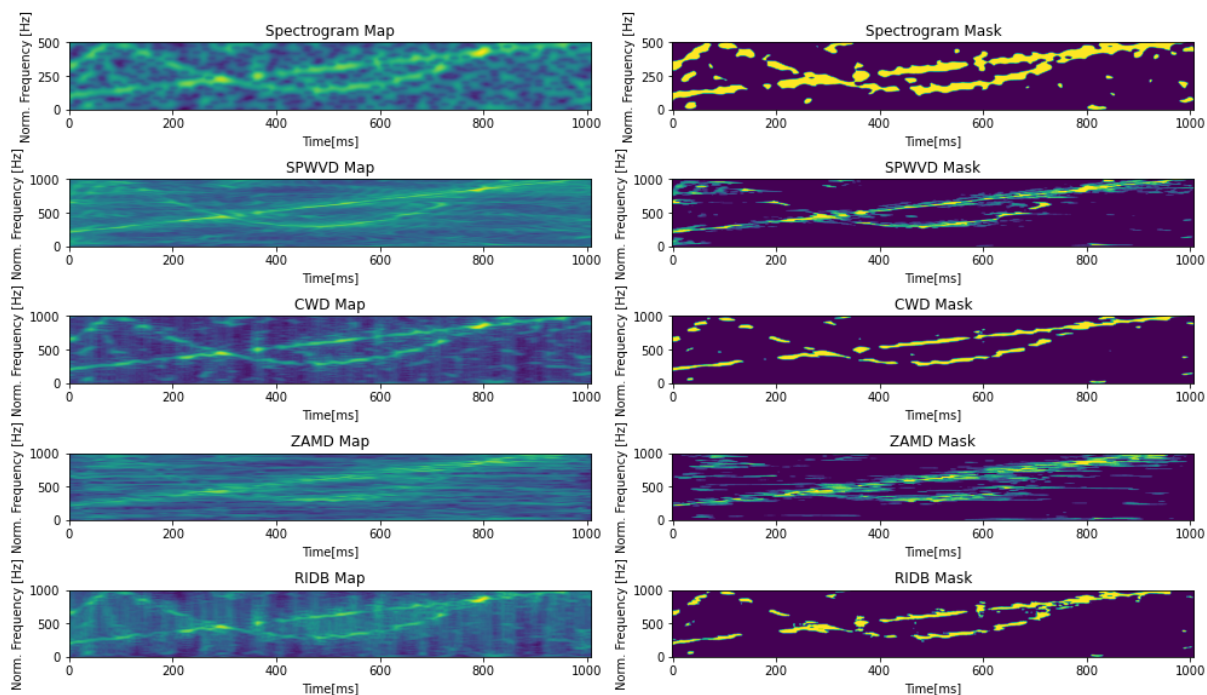
**Figure 12:** Entropy maps and masks for different TFDs for Rényi entropy and  $SNR = 5$  in case of signal C



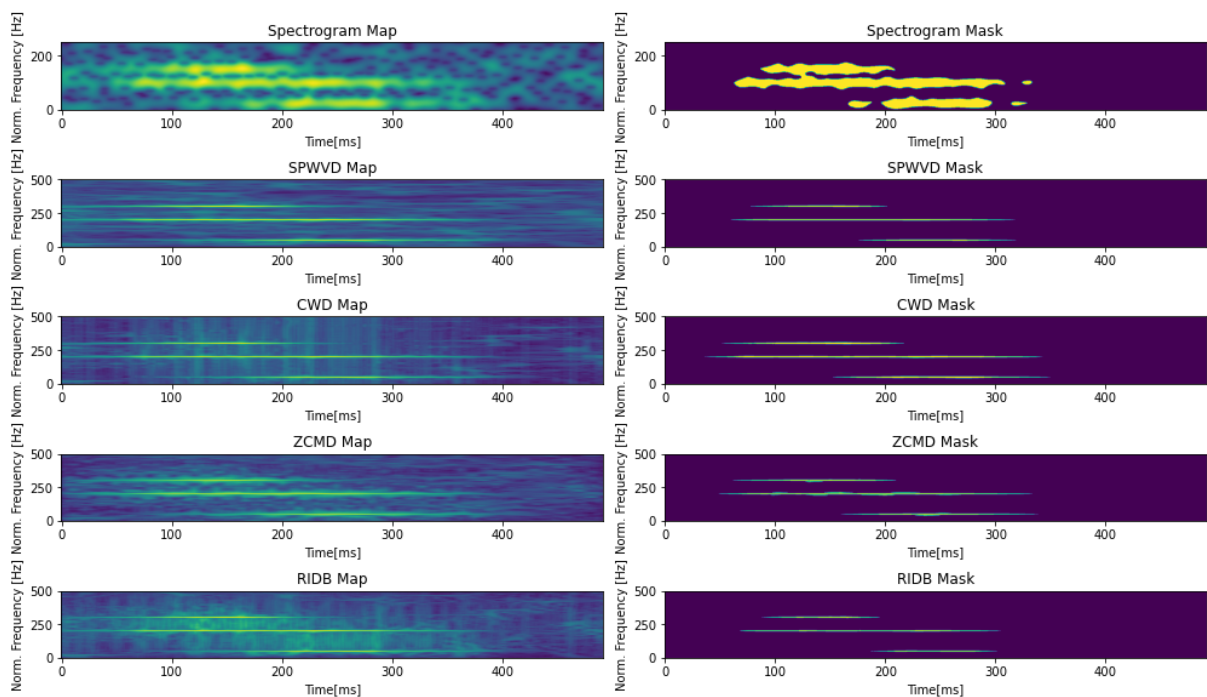
**Figure 13:** Entropy maps and masks for different TFDs for Tsallis entropy and  $\text{SNR} = 0$  in case of signal A



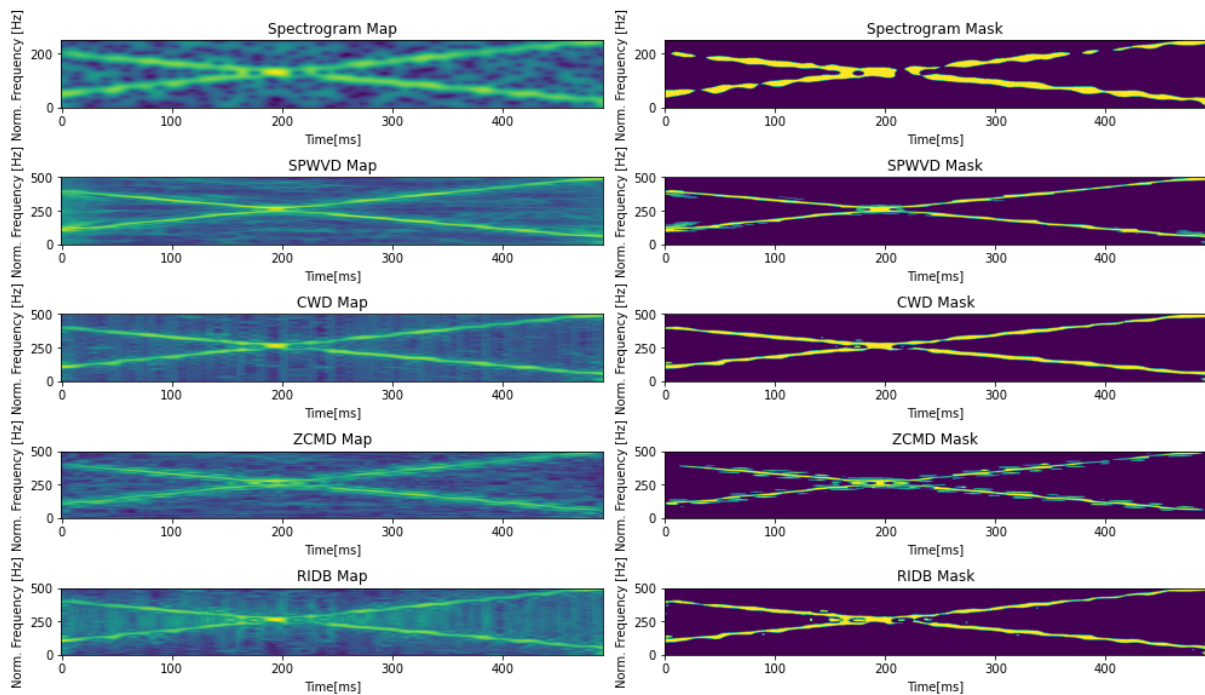
**Figure 14:** Entropy maps and masks for different TFDs for Tsallis entropy and  $\text{SNR} = 0$  in case of signal B



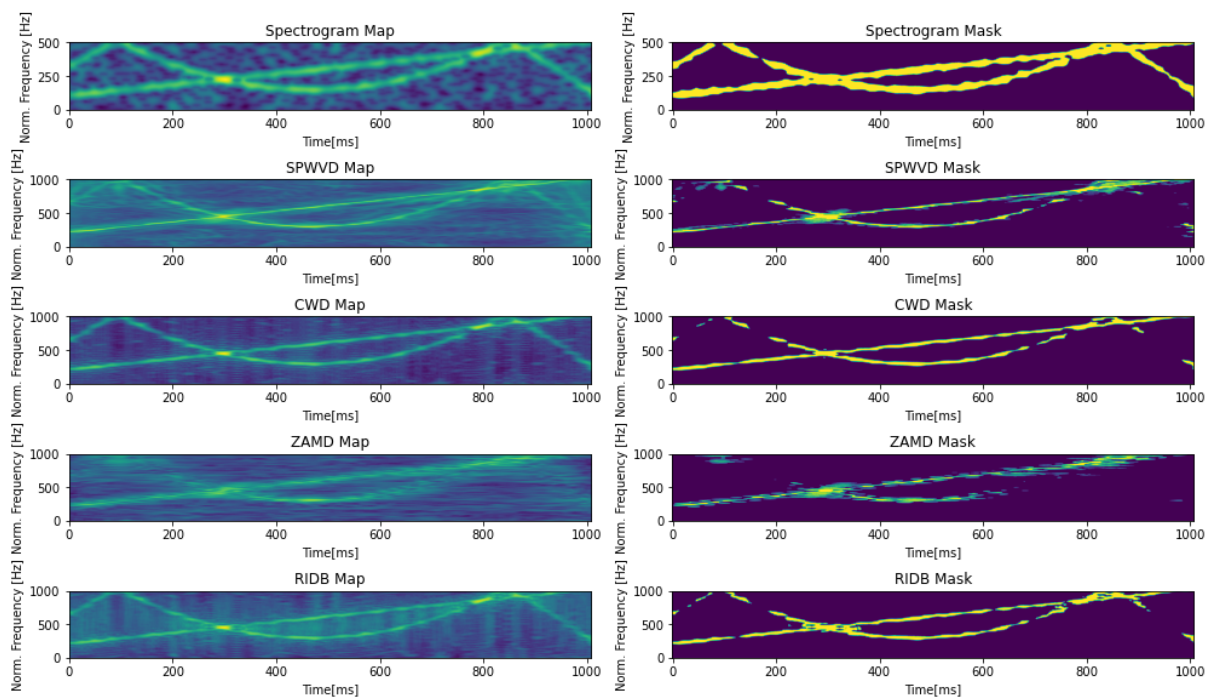
**Figure 15:** Entropy maps and masks for different TFDs for Tsallis entropy and  $SNR = 0$  in case of signal C



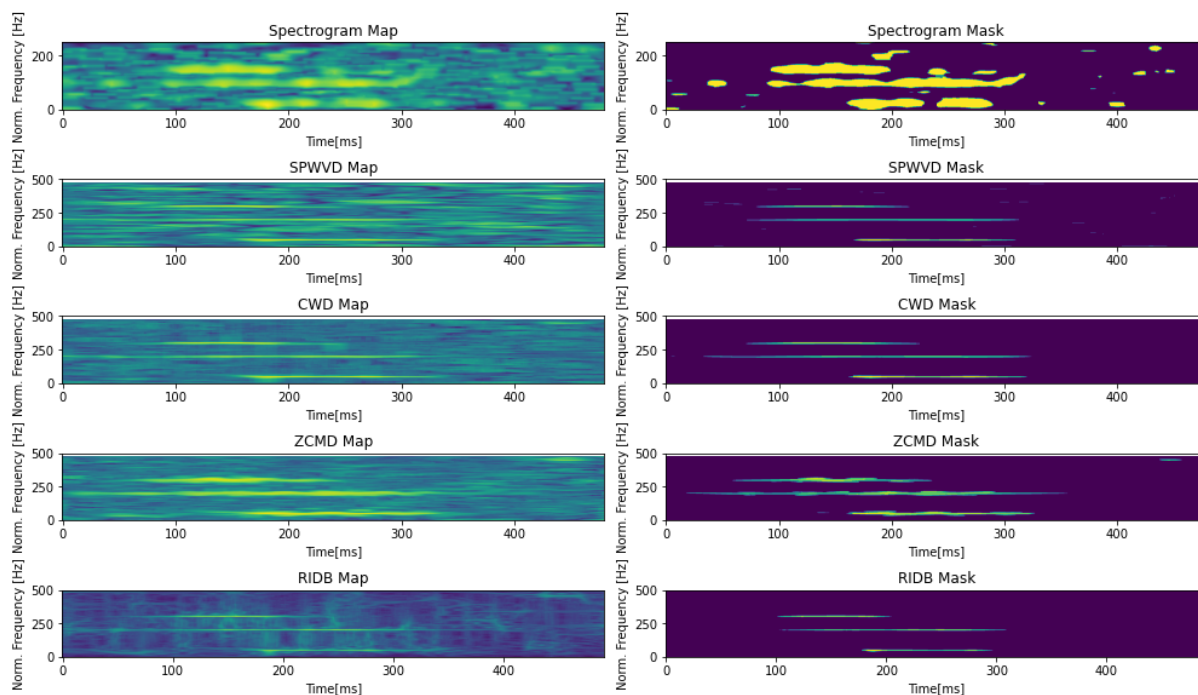
**Figure 16:** Entropy maps and masks for different TFDs for Tsallis entropy and  $SNR = 5$  in case of signal A



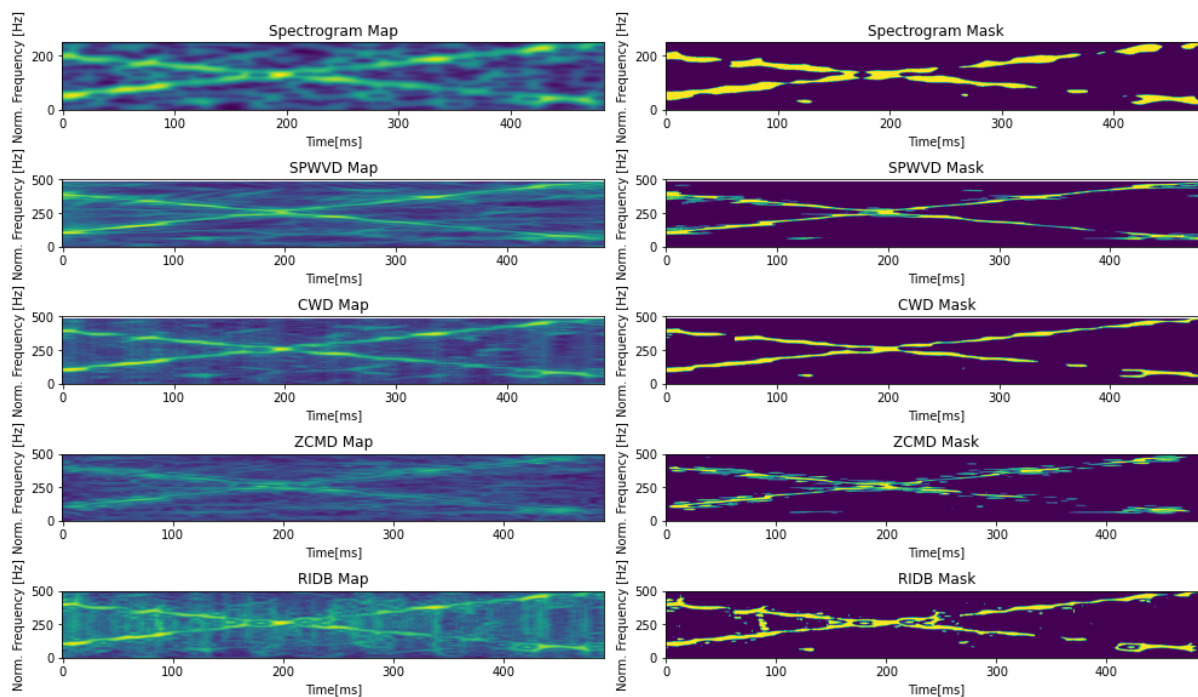
**Figure 17:** Entropy maps and masks for different TFDs for Tsallis entropy and  $\text{SNR} = 5$  in case of signal B



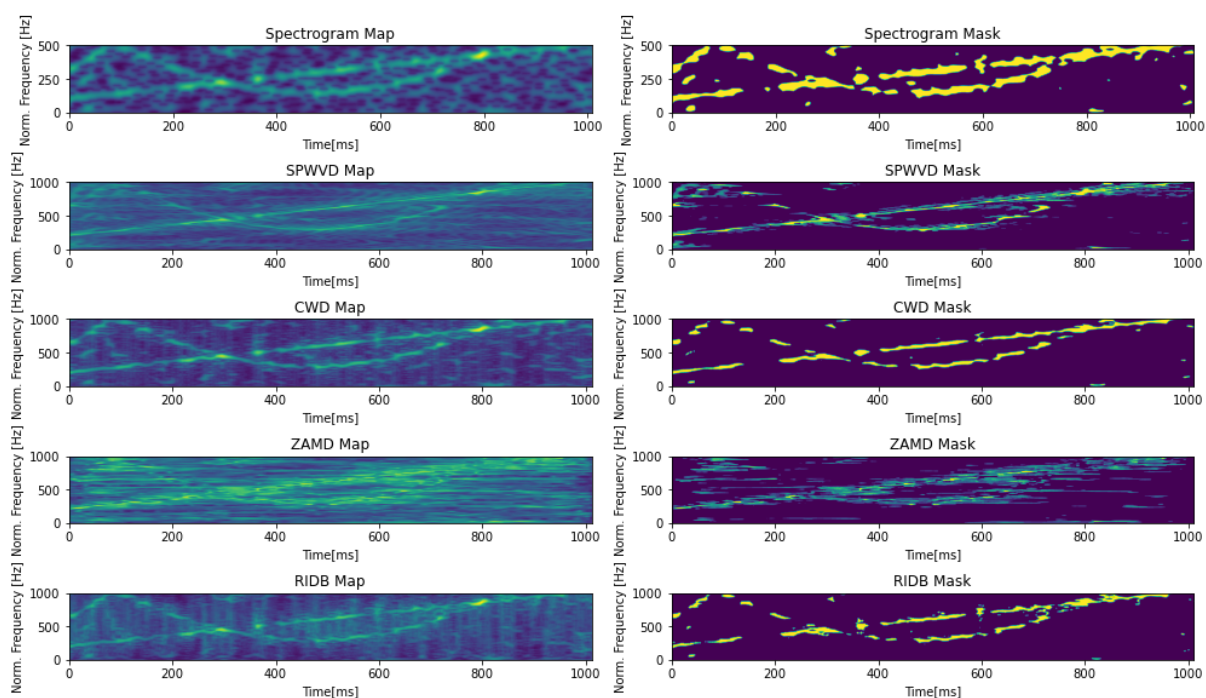
**Figure 18:** Entropy maps and masks for different TFDs for Tsallis entropy and  $\text{SNR} = 5$  in case of signal C



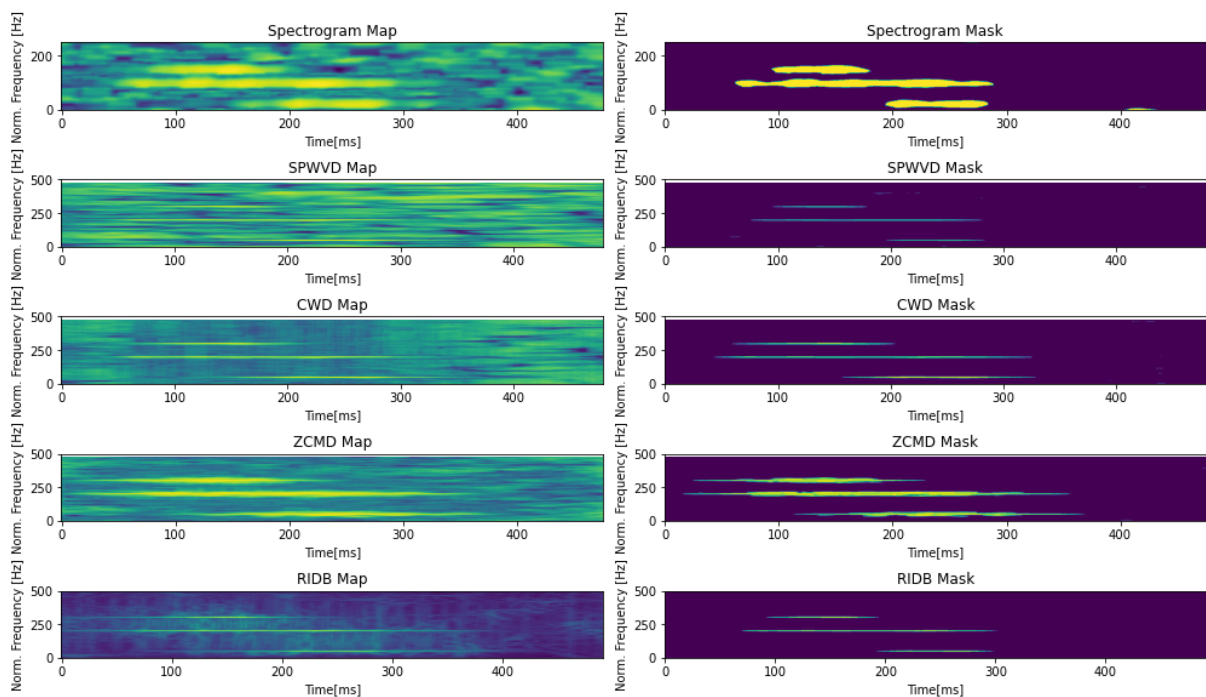
**Figure 19:** Entropy maps and masks for different TFDs for Dombi entropy and  $SNR = 0$  in case of signal A



**Figure 20:** Entropy maps and masks for different TFDs for Dombi entropy and  $SNR = 0$  in case of signal B

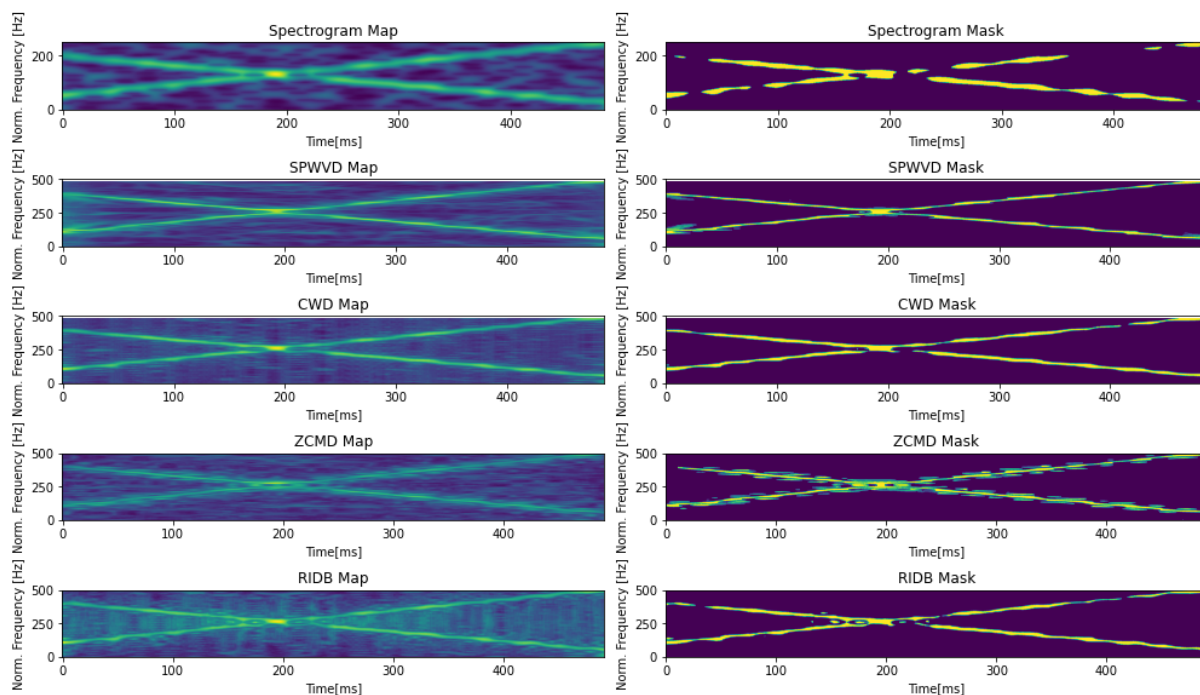


**Figure 21:** Entropy maps and masks for different TFDs for Dombi entropy and  $\text{SNR} = 0$  in case of signal C

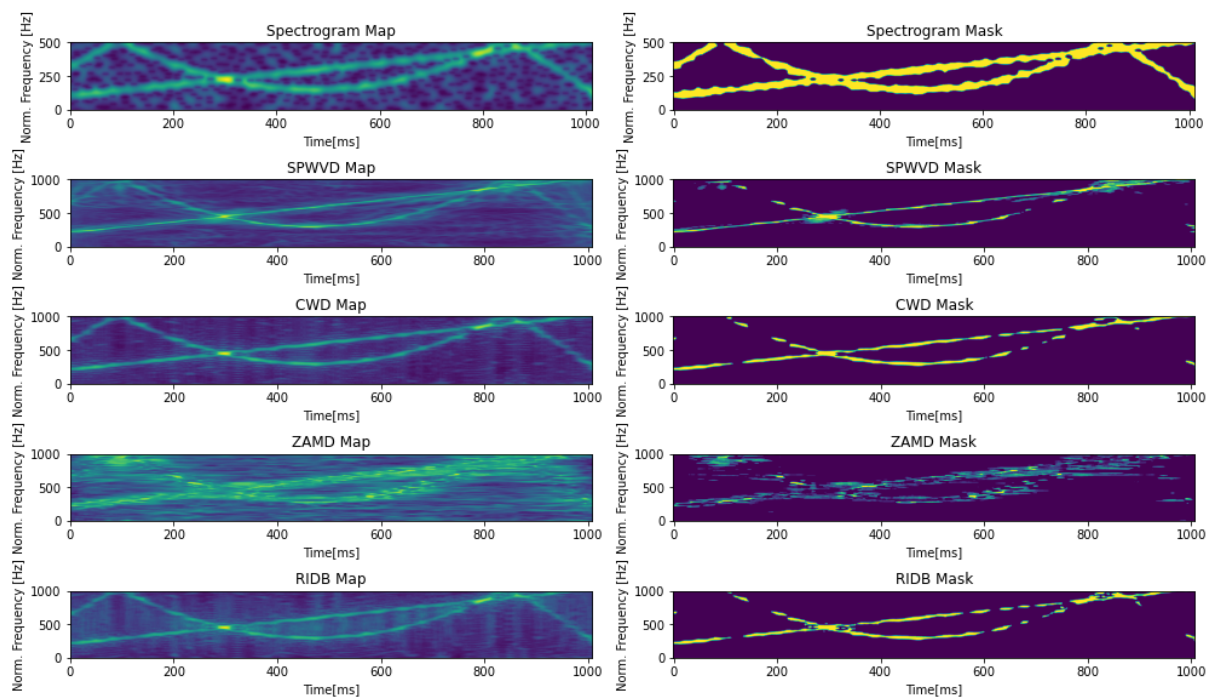


**Figure 22:** Entropy maps and masks for different TFDs for Dombi entropy and  $\text{SNR} = 5$  in case of signal A





**Figure 23:** Entropy maps and masks for different TFDs for Dombi entropy and  $SNR = 5$  in case of signal B



**Figure 24:** Entropy maps and masks for different TFDs for Dombi entropy and  $SNR = 5$  in case of signal C

# CURRICULUM VITAE

Ana Vranković Lacković was born on January 13, 1993, in Rijeka, Croatia. After completing her primary and secondary education, she enrolled in the University of Rijeka, Faculty of Engineering, in 2011. She completed her undergraduate and graduate university studies in computing at the same institution in 2016. She received annual Faculty of Engineering Dean's award for academic excellence for computing graduate studies. In 2016, she was employed at the Faculty of Engineering, University of Rijeka, as a research assistant on a scientific project. In 2019 she was employed as a teaching assistant in the Department of Computer Engineering at the Faculty of Engineering, University of Rijeka. In 2019, she also enrolled in the Faculty of Engineering for postgraduate university doctoral study in the area of engineering sciences in the field of computer science. She has participated in several scientific projects, such as "Hyperspectral Image Analysis Using Machine Learning and Adaptive Data-Driven Filtering", "A network for Gravitational Waves, Geophysics and Machine Learning" and the University of Rijeka research project "Computer-aided digital analysis and classification of signals", under the leadership of Prof. D. Sc. Jonatan Lerga.



# LIST OF PUBLICATIONS

József Dombi, Ana Vranković Lacković, and Jonatan Lerga. A new insight into entropy based on the fuzzy operators, applied to useful information extraction from noisy time-frequency distributions. *Mathematics*, 11(3):505, 2023.

Tihana Galinac Grbac and Ana Vranković. Software structure evolution and relation to subgraph defectiveness. In *SoftCOM 2016 PhD Forum*, 2017.

Ana Vranković Lacković. Preparation for replicating empirical study of a selection method for software reliability growth models. In *My First Conference*, pages 54–55, 2017.

Ana Vranković Lacković, Jonatan Lerga, and Marijana Tomić. Local shannon, rényi, and tsallis entropy for useful content extraction from choi-williams and zhao-atlas-marks time-frequency distributions. In *2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, pages 1–5. IEEE, 2022.

Ana Vranković, Tihana Galinac Grbac, and Željka Car. Software structure evolution and relation to subgraph defectiveness. *IET Software*, 13(5):355–367, 2019.

Ana Vranković, Tihana Galinac Grbac, and Bruno Blaskovic. A case study on reliable erlang communication in open platforms based on scribble. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1472–1475. IEEE, 2018.

Ana Vrankovic, Tihana Galinac Grbac, and Melinda Tóth. Comparison of software structures in java and erlang programming languages. In *SQAMIA*, 2017.

Ana Vranković, Ivo Ipšić, and Jonatan Lerga. Entropy-based extraction of useful content from spectrograms of noisy speech signals. In *2021 International Symposium ELMAR*, pages 83–86. IEEE, 2021.

Ana Vranković, Jonatan Lerga, and Nicoletta Saulig. A novel approach to extracting useful information from noisy tdfs using 2d local entropy measures. *EURASIP Journal on Advances in Signal Processing*, 2020:1–19, 2020.