

On the Intersection of Signal Processing and Machine Learning: A Use Case-Driven Analysis Approach

Sulaiman Aburakhia, *Member, IEEE*, Abdallah Shami, *Senior Member, IEEE* and George K. Karagiannidis, *Fellow, IEEE*

Abstract—Recent advancements in sensing, measurement, and computing technologies have significantly expanded the potential for signal-based applications, leveraging the synergy between signal processing and Machine Learning (ML) to improve both performance and reliability. This fusion represents a critical point in the evolution of signal-based systems, highlighting the need to bridge the existing knowledge gap between these two interdisciplinary fields. Despite many attempts in the existing literature to bridge this gap, most are limited to specific applications and focus mainly on feature extraction, often assuming extensive prior knowledge in signal processing. This assumption creates a significant obstacle for a wide range of readers. To address these challenges, this paper takes an integrated article approach. It begins with a detailed tutorial on the fundamentals of signal processing, providing the reader with the necessary background knowledge. Following this, it explores the key stages of a standard signal processing-based ML pipeline, offering an in-depth review of feature extraction techniques, their inherent challenges, and solutions. Differing from existing literature, this work offers an application-independent review and introduces a novel classification taxonomy for feature extraction techniques. Furthermore, it aims at linking theoretical concepts with practical applications, and demonstrates this through two specific use cases: a spectral-based method for condition monitoring of rolling bearings and a wavelet energy analysis for epilepsy detection using EEG signals. In addition to theoretical contributions, this work promotes a collaborative research culture by providing a public repository of relevant Python and MATLAB signal processing codes. This effort is intended to support collaborative research efforts and ensure the reproducibility of the results presented.

Index Terms—Signal processing, signal preprocessing, signal denoising, feature extraction, signal transforms

I. INTRODUCTION

The rapid advancements in sensing and measurement represent a paradigm shift in how data is collected, processed, and interpreted. This opens the door for a wide range of signal-based applications, marking a transformative phase across various fields. Moreover, the development of computing technologies and the rise of the Internet of Things (IoT) have paved the way to leverage machine learning (ML) within signal-based applications, offering new insights and achieving

unprecedented levels of accuracy and efficiency. This merge between signal processing and ML is expected to play a major role in the next generations of sensor-enabled systems across various areas [1]. The integration of signal processing pipelines into ML models forms the fundamental core of these systems. Further, it represents a critical intersection in their advancement, motivating the research community to address the role of signal processing in ML. However, the diverse landscape of signal types and application requirements shapes the scope of the existing body of work to be application-centric, limiting their scopes to specific applications. For instance, The role of feature extraction in ML has been extensively reviewed within the context of vibration-based predictive maintenance (PdM). The work in [2] presents a review that focuses on transforming traditional methods to ML techniques in applying vibration-based damage detection in civil structures. The article highlights traditional methods and presents a comprehensive review of the latest applications of ML algorithms used for this purpose. In [3], a systematic review is conducted on adopting machine learning for failure prediction in industrial maintenance. The review covers the used datasets, preprocessing, and the training and evaluation of prediction models. In [4], a comprehensive review is presented on signal processing techniques for vibration-based feature extraction in structural health monitoring (SHM). The work in [5] addresses the application of vibration-based condition monitoring techniques for the PdM of rotating Machinery. It provides a comprehensive review of vibration data acquisition and analysis, as well as the methods used for fault interpretation and diagnosis, including data acquisition, data transmission, signal processing, and fault detection. The work in [6] introduces a tutorial on the same topic that describes relevant signal processing methods in this field. Furthermore, the tutorial provides Python and MATLAB code examples to demonstrate these methods alongside explanatory videos.

In the biomedical field, the topic of signal processing in ML is an active area of research. For instance, the work in [7] provides an end-to-end review of biomedical signal processing for health monitoring applications. It introduces a flow for developing biomedical signal processing systems. Further, it covers recent applications, types of low-cost, non-invasive biomedical sensors, signal processing techniques, and future perspectives for building reliable systems. The application of MI-based brain-computer interfaces (BCIs) in controlling external devices through EEG signal processing

Sulaiman Aburakhia and Abdallah Shami are with the Department of Electrical and Computer Engineering, Western University, N6A 3K7, Canada (e-mail: saburakh@uwo.ca; abdallah.shami@uwo.ca).

George K. Karagiannidis is with Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Greece and also with Artificial Intelligence and Cyber Systems Research Center, Lebanese American University (LAU), Lebanon (e-mail: geokarag@auth.gr).

is addressed in [8]. The article reviews recent ML models, identifies major challenges, and suggests potential solutions by focusing on feature extraction and classification methods. The work in [9] reviews various feature extraction techniques for electrocardiogram (ECG) Signal analysis. In [10], the main steps in detecting and classifying EEG epileptic seizure activities are addressed, along with a review of related feature extraction techniques. The studies in [11] and [12] deal with the application of emotion recognition using EEG signals, where different existing feature extraction methods are analyzed and compared in terms of classification performance. A comprehensive review of methods and techniques that covers the entire process of EEG signal processing is presented in [13]. The study analyzed numerous articles related to EEG signal processing, identified limitations, and analyzed future development trends. Besides biomedical and BdM fields, the role of signal processing in ML has been addressed in various fields such as audio analysis and recognition [14]–[20], seismic signal analysis [20]–[22], and telecommunications [23]–[30].

Despite this diverse spectrum of articles addressing the role of signal processing within ML, there remains a noticeable gap in the literature, characterized by the following four main limitations in the existing studies:

- **Application-Specific Focus:** The analysis are often confined to specific applications, which limits the coverage and depth in which the topic is addressed.
- **Limited Audience Reach:** Many articles assume a substantial background in signal processing, restricting accessibility to a limited audience.
- **Task-Centric Approach:** The exploration of signal processing in ML tends to focus on particular tasks, such as signal preprocessing or feature extraction. This neglects other critical tasks such as signal segmentation, smoothing, and denoising, which are essential components of a typical signal processing ML pipeline.
- **Theory-to-Practice Gap:** There is a notable lack of practical demonstrations through use-case applications. Such demonstrations are critical to connecting theoretical concepts with real-world applications, and thus bridging the gap between theory and practice.

This paper attempts to bridge this gap by implementing an integrated-article approach that addresses the aforementioned shortcomings through the following contributions:

- **Comprehensive Tutorial for Diverse Readership:** The paper starts with a comprehensive tutorial on the fundamentals of signal processing that is aimed at readers from different domains, allowing the interested reader to develop a proper background before delving into the review.
- **Application-Independent Approach:** The paper adopts a broad, application-independent review, providing a comprehensive overview of signal processing in ML that is not shaped to a specific use case.
- **End-to-End Overview of Signal Processing Workflow:** The article thoroughly discusses the key tasks in a typical signal processing pipeline, grouping them under three

main categories: preprocessing, processing, and application.

- **Exhaustive Review with Novel Taxonomy:** The paper conducts a detailed review of feature extraction techniques, presented and categorized through a new taxonomy that presents new insights and enriches the reader's understanding.
- **Addressing Major Challenges and Potential Solutions:** The paper identifies the primary challenges faced in implementing signal processing-based ML applications and discusses both existing and potential solutions.
- **Bridging Theory with Practice:** The paper addresses the practical application of signal processing in ML through two use cases. In the first use case, a spectral-based method is introduced for vibration-based condition monitoring of rolling bearings. In the second case, wavelet-energy analysis is utilized for epilepsy detection using EEG signals.
- **Public Repository of Signal Processing:** The paper introduces a public repository of the Python and MATLAB codes used throughout the article, along with additional codes relevant to signal processing, thereby fostering a collaborative research environment and ensuring reproducibility of work.

With these contributions, the paper aims to become a foundational reference on the topic of signal processing in ML, serving a wide range of readers and offering new perspectives on the intersection of signal processing and ML. Each of these contributions has its own section in the paper. The sections are logically grouped under four parts. For reader convenience, a visual representation that shows the various sections grouped under each part is depicted in Fig. 1. Although these parts are organized to build knowledge progressively, they are written in a self-contained manner, allowing for selective reading based on interest or need.

The article is structured as follows: The tutorial is carried out in Section II and Section III. A typical signal-processing pipeline for signal-based ML applications is presented in Section IV. The review of feature extraction techniques is introduced in Section V. Section VI discusses the main challenges faced in implementing signal processing-based ML applications and addresses potential solutions. The practical application use cases are introduced in Section VII. The paper is finally concluded in Section VIII.

II. INTRODUCTION TO SIGNALS

This tutorial provides an in-depth introduction to signal processing, highlighting concepts, mathematical formulation, applications, advantages, and limitations of common signal processing tools. Additionally, the tutorial offers insights into implementation considerations and highlights programming libraries that offer functionalities for implementing these tools. Throughout the tutorial, illustrative examples are generated using Python and MATLAB codes. These codes are publicly available on the Github site of the Optimized Computing and Communications (OC2) Laboratory¹

¹<https://github.com/Western-OC2-Lab/Signal-Processing-for-Machine-Learning>



Fig. 1. Visual representation of the paper's contents.

A. Signal Characteristics

A signal is a function of time; it represents the value of a physical entity or phenomenon as it evolves over time, such as voltage, current, acceleration... etc. Depending on the field, there are more definite definitions of the term "signal". For instance, within a signal processing context, a signal can be defined as a function that conveys information about the behavior of a system or attributes of some phenomenon [31]. In manufacturing, the term "signal" refers to a physical quantity that carries a certain type of information and serves as a means for communication [32]. Examples of such signals include a vibration signal generated by an accelerometer attached to the rolling-bearing element of a rotating machinery and a torque signal generated by a torque Sensor in a computer numerical control (CNC) milling machine. The changes in these signals are directly related to the operation of the machine and can therefore be used to communicate the operating status of the machine to the machine operator [32].

Signals are typically represented by their time waveforms; Fig. 2. displays the waveform of a given $x(t)$ as it evolves over time t . Amplitude, frequency, and phase are the main characteristics of a signal. Amplitude measures the amount and direction of change in the signal with respect to a reference value, as shown in Fig. 2. Frequency is the number of cycles or oscillations that occur in a given unit of time. It represents the rate at which the signal oscillates. Frequency is measured in *Hertz* (Hz) where $1 Hz$ is equal to 1 cycle per second. The phase of a signal refers to the position of a point (time instant) on the signal's waveform cycle. The phase of a wave refers to its position within its cycle and is measured in either

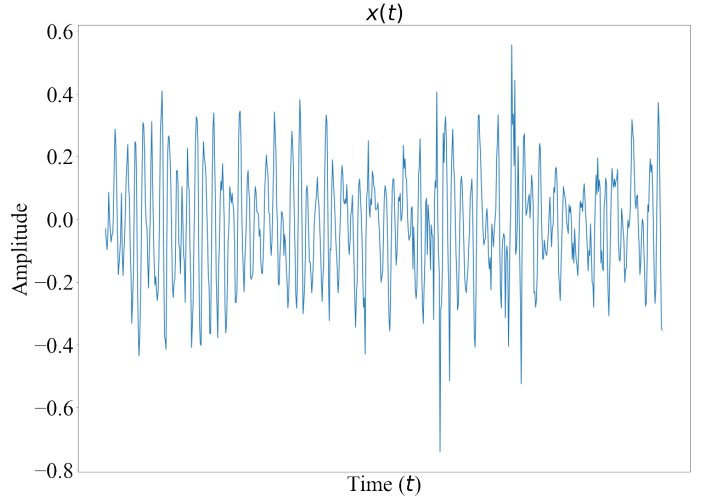


Fig. 2. Waveform of a signal, $x(t)$, as it evolves over time t .

degrees ($0 - 360$) or radians ($0 - 2\pi$). It helps describe the relative position and timing of two signals at a given moment. The concept of frequency of a signal is closely related to the rate at which its phase changes. Specifically, the frequency at a particular moment in time indicates the speed at which the phase is changing at that moment. A higher frequency indicates a faster rate of phase change. Therefore, frequency and phase are mathematically linked through the concept of the derivative. By knowing the phase information of a signal, the frequency can be calculated as the first derivative of the phase with respect to time. The period of a waveform refers

to the time it takes to complete one cycle. A periodic signal is a signal that repeats its pattern or the sequence of values exactly after a fixed duration time, known as the period T . This can be expressed mathematically as:

$$x(t) = x(t + T) \quad (1)$$

Since the period is the time duration a signal takes to complete one cycle, it is related to the signal's frequency, which represents the number of cycles per second, through the following relation:

$$T = \frac{1}{f} \text{ (seconds)} \quad (2)$$

While purely periodic signals do not exist in practice [33], they represent an essential theoretical concept in signals theory [34]. A sinusoidal wave, $s(t)$, represents the basic form of a periodic signal; it is expressed mathematically as:

$$s(t) = A \sin(2\pi ft + \phi) \quad (3)$$

Where A is the signal's peak amplitude, f is frequency, and ϕ is the phase. With respect to zero, signal amplitude can be either positive or negative, as shown in Fig.3. On the other

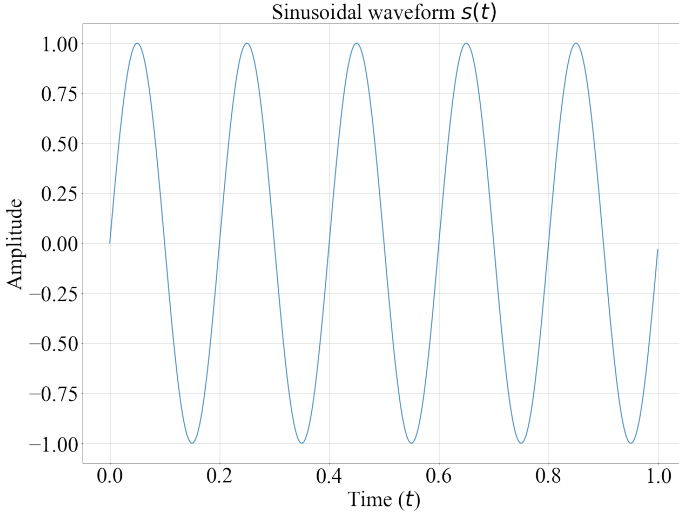


Fig. 3. Sinusoidal waveform $s(t)$.

hand, the magnitude of the signal $|s(t)|$, shown in Fig. 4, is the absolute value of the amplitude, it shows the change in the signal regardless of its direction since it is always a positive quantity. Amplitude values indicate the strength of the signal, reflecting the intensity of the measured physical phenomenon.

As explained earlier, a signal represents a physical entity or phenomenon that changes over time. A signal that remains unchanged over time when measured again under the same conditions is referred to as a deterministic signal. Such a signal has no uncertainty about its value at any given point in time and can be modeled using mathematical expressions. The sinusoidal function is an example of a deterministic function. However, signals that are completely free of unknown and uncontrollable factors and have true deterministic properties are extremely rare [35]. In practical scenarios, signals are non-deterministic as they are random in nature and exhibit various

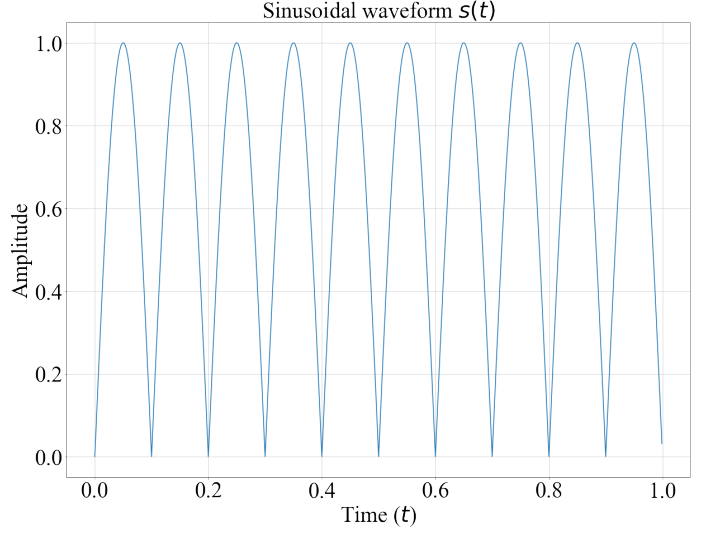


Fig. 4. Magnitude of the sinusoidal waveform $s(t)$.

degrees of uncertainty. Since non-deterministic signals are stochastic, they are described in terms of their statistical properties. From this perspective, random signals can be classified into two categories: stationary and nonstationary. Stationary signals generally maintain constant statistical properties over time, while nonstationary signals exhibit changing statistical properties over time.

From the preceding discussion, it is apparent that signal variation describes the changes in the measured physical entity or phenomenon as it evolves over time. This variation is subject to the influence of uncontrollable factors during the measurement period, which manifest themselves as noise or outliers in the signal. Other influencing factors include subsequent processing and operations imposed on the signal, commonly referred to as "systems" in the context of signal processing. Specifically, a system in signal processing is any process or operation that produces an output signal in response to an input signal. In practical situations, real-world systems exhibit mostly nonlinear behaviors² and operate under transient, nonstationary conditions [36]. Therefore, real-world signals in practical situations exhibit nonlinear, time-varying, and nonstationary characteristics. Understanding the unique characteristics of a signal holds significant importance, as it directly influences the selection of the proper signal processing tool, ensuring precise and effective analysis of the signal and, consequently, extracting discriminative features.

Another fundamental aspect of signal processing is signal acquisition and the time representation of the signal. From a time perspective, signals are classified as continuous or discrete-time signals. The continuous-time signal $x(t)$ is the signal that has a continuous value over the observation time range, as depicted in Fig. 2. A discrete-time version of the signal $x(t)$ is displayed in Fig. 5, denoted by $x[n]$. In the

²A linear system in signal theory is defined by two key properties: homogeneity and additivity. Homogeneity means if an input signal $x(t)$ results in an output $y(t)$, then any scaled input $ax(t)$ leads to a scaled output $ay(t)$. Additivity implies that the response to a sum of input signals $x_1(t) + x_2(t)$ is the sum of their responses, $y_1(t) + y_2(t)$.

discrete-time representation, the independent variable n is discrete in type and takes integer values only. That's it; in contrast to a continuous-time signal, the discrete signal is defined only over discrete time intervals. The discrete-time

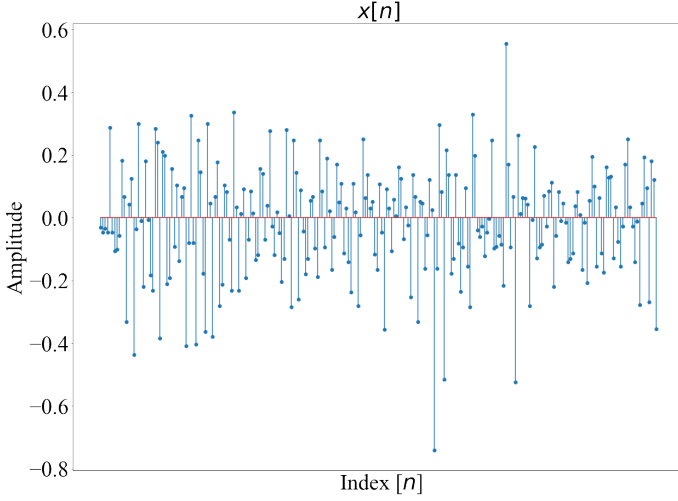


Fig. 5. $x[n]$, a discrete-time version of signal $x(t)$.

representation $x[n]$ is obtained by sampling the continuous-time signal $x(t)$ at time instants separated by T_s , *i.e.*

$$x[n] = x(nT_s), \quad n = 0, 1, 2, \dots \quad (4)$$

where T_s represents the sampling interval or sampling period in seconds. Accordingly, the sampling frequency or sampling rate f_s is the number of samples obtained in 1 second; it is given by:

$$f_s = \frac{1}{T_s} \quad (5)$$

The unit of f_s is samples per second or hertz. In practice, a measured signal is a sampled version of the actual physical signal since it is acquired by recording a specific number of measurements (samples) every second. The sampling rate plays a crucial role in signal measurement and processing. A higher sampling rate means more measured samples per second and, hence, a higher signal resolution. However, processing time and memory requirements increase as the sampling rate increases. A fundamental concept in signal sampling is the Nyquist theorem [37], which states that a signal can be correctly reconstructed from its sampled discrete-time sequence provided that the sampling rate is equal to or greater than twice the highest frequency (effective bandwidth) of the signal. This implies that the highest frequency, represented as f_{max} , that can be represented accurately in an acquired signal is one-half of its sampling rate f_s ; *i.e.*

$$f_s = 2f_{max} \quad (6)$$

This formula is called the Nyquist rate; it provides a lower bound on the sampling rate that is required to reconstruct a signal from its sampled version accurately. Accordingly, a sampled signal is categorized as either undersampled, critically sampled, or oversampled:

- Undersampled signal: is one where the sampling rate is below the Nyquist sampling rate. Undersampling leads to a distortion in the signal known as aliasing, causing signal components at higher frequencies than the sampling frequency to appear at a lower, aliased frequency [38].
- Critically sampled signal: is one where the sampling rate equals the Nyquist sampling rate. In this case, the signal is sampled at the lowest rate, which still allows for a complete reconstruction of the continuous-time signal, assuming the signal bandwidth is limited to half of the sampling frequency. While perfect reconstruction is theoretically still possible, it is less common in practical applications.
- Oversampled signal: is one where the sampling rate exceeds the Nyquist rate. Oversampling is widely used in signal measurements because it improves the resolution of the acquired signal. Additionally, with oversampling, the quantization noise (an error that represents the difference between the value of the measured sample and the closest mapped digital value) is spread over a broader frequency range compared to the effective bandwidth of the signal. Accordingly, filtering out high frequencies outside the effective bandwidth effectively reduces noise in the signal, improving the signal-to-noise ratio (SNR).

B. Signal Energy and Power

As noted above, signal amplitude reflects the intensity of the measured physical phenomenon, making it significant in identifying critical features of the signal, such as energy and power. Energy quantifies the total "work" done by the signal; mathematically, the energy of a signal is the area under the squared magnitude of the signal over a given time interval. For a continuous-time signal $x(t)$, its energy E is defined as:

$$E = \int_{-\infty}^{\infty} |x(t)|^2 dt \quad (7)$$

For a discrete-time signal $x[n]$, the energy is given by:

$$E = \sum_{n=-\infty}^{\infty} |x[n]|^2 \quad (8)$$

This definition assumes the signal is energy-limited, meaning that its energy is finite over the entire time domain. Signal power measures the average rate at which the signal transmits energy. It is calculated as the average power over a period of time. For a continuous-time signal, its average power P is defined as:

$$P = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T |x(t)|^2 dt \quad (9)$$

Similarly, for discrete signals:

$$P = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N |x[n]|^2 \quad (10)$$

If the signals' energy E converges to a finite but non-zero value, the signal is classified as an energy signal. Examples include non-zero signals over a limited period but zero elsewhere, such as pulse signals. The average power of an

energy signal is zero because the calculation of power involves averaging the finite energy over an infinite period of time. Power signals, on the other hand, are signals that have finite, non-zero average power. A common example is a periodic sinusoidal signal that continues indefinitely; such signals have infinite energy. For signals of finite duration, energy and power can be evaluated over their active time intervals. For periodic signals, due to the repetitive nature of the signal, the computation is simplified by considering one single period T as follows:

$$P = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} |x(t)|^2 dt \quad (11)$$

For a discrete-time periodic signal $x[n]$ with period N , the power is:

$$P = \frac{1}{N} \sum_{n=0}^{N-1} |x[n]|^2 \quad (12)$$

Instantaneous power refers to the power of a signal at a specific moment in time. Unlike average power, which is calculated over a period of time, instantaneous power gives a moment-to-moment view of the power level of the signal. For a continuous-time signal $x(t)$, the instantaneous power $P(t)$ at time t is defined as the square of the signal's magnitude at that particular time. Mathematically, it is expressed as:

$$P(t) = |x(t)|^2 \quad (13)$$

For a discrete signal $x[n]$, where n represents discrete time indices, the instantaneous power $P[n]$ at point n is defined as the square of the magnitude of the signal at that point. Mathematically, this is expressed as:

$$P[n] = |x[n]|^2 \quad (14)$$

For a given discrete time signal $x[n]$, represented by N samples, the total energy E is the sum of its instantaneous power values over its discrete time interval:

$$E = \sum_{n=0}^{N-1} |x[n]|^2 \quad (15)$$

Accordingly, the average power P is obtained by dividing the energy E by the number of samples N :

$$P = \frac{\sum_{n=0}^{N-1} |x[n]|^2}{N} \quad (16)$$

along with power and energy, statistical properties of the signal, such as mean and variance, represent important aspects of signal characteristics and play a major role in signal processing. For the discrete time signal $x[n]$, its mean μ is equal to:

$$\mu = \frac{1}{N} \sum_{n=0}^{N-1} x[n] \quad (17)$$

The mean represents the central tendency or the average value of the signal over its time period. The signal's variance σ^2 is given by:

$$\sigma^2 = \frac{1}{N} \sum_{n=0}^{N-1} (x[n] - \mu)^2 \quad (18)$$

The variance measures the spread of the signal's amplitude around its mean value, quantifying the deviation of the signal's samples from its mean. Note that for a zero-mean signal, The variance equals the signal's average power. For such signals, the root-mean-square (RMS) is a useful measure that represents the effective value of the signal because it provides a meaningful measure of the average magnitude of the signal, even though its mean is zero. The RMS is expressed as:

$$x_{RMS} = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} [x[n]]^2} \quad (19)$$

C. Signal Correlation

Correlation can be viewed as a similarity measure; it quantifies the similarity or the degree of dependence between two signals, providing insights into their common characteristics and patterns. Auto-correlation refers to the correlation of a signal with a delayed version of itself. It represents the degree of similarity between a signal and a time-shifted (lagged) version of itself over a given time interval. Auto-correlation helps to detect repeating patterns in the signal and assess its stability or predictability over time. For a continuous-time signal $x(t)$, the auto-correlation function is expressed as:

$$R_{xx}(\tau) = \int_{-\infty}^{\infty} x(t)x(t-\tau)dt \quad (20)$$

where τ represents time shift. For a discrete-time signal $x[n]$, autocorrelation $R_{xx}[k]$, at a lag k is given by defined as:

$$R_{xx}[k] = \sum_{n=-\infty}^{+\infty} x[n]x[n-k], \quad k = 0, \pm 1, \pm 2, \dots \quad (21)$$

Note that for zero lag, there is no time shift; hence, the autocorrelation R_{xx} will be at its maximum value,— which equals the signal's total energy— representing the maximum similarity as depicted in Fig. 6 which shows the autocorrelation of the signal $x(t)$.

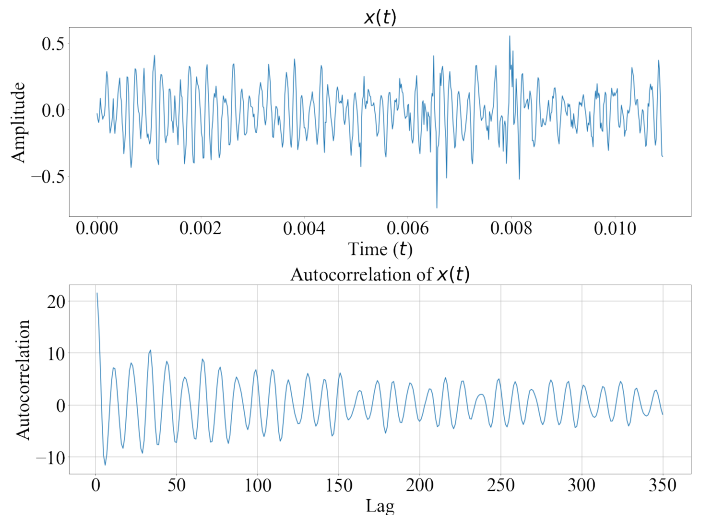


Fig. 6. Signal $x(t)$ and its autocorrelation.

Cross-correlation extends the concept of auto-correlation to two different signals, measuring the degree of similarity or dependence between them. It quantifies the degree of similarity between one signal and a time-shifted (lagged) version of another signal over a given time interval. For two continuous-time signals $x(t)$ and $y(t)$, their cross-correlation is expressed as:

$$R_{xy}(\tau) = \int_{-\infty}^{\infty} x(t)y(t-\tau)dt \quad (22)$$

For discrete-time signals $x[n]$ and $y[n]$, the cross-correlation R_{xy} at a given time lag k is defined as:

$$R_{xy}[k] = \sum_{n=-\infty}^{+\infty} x[n]y[n-k], \quad k = 0, \pm 1, \pm 2, \dots \quad (23)$$

The value of $R_{xy}[k]$ indicates how much of one signal is present in the other signal at a given delay k . Hence, it describes the mutual relationship between the two signals as a function of the displacement of one relative to the other. Cross-correlation is often used to search for desired features or patterns within a signal.

The correlation coefficient between two signals is a statistical measure that quantifies the degree to which the two signals are linearly related. It is a normalized form of the cross-correlation function and is widely used in signal processing and statistics to assess the strength of the linear relationship between two signals [39]. For discrete-time signals $x[n]$ and $y[n]$, each of length N , the correlation coefficient ρ_{xy} is defined as:

$$\rho_{xy} = \frac{\sum_{n=0}^{N-1} (x[n] - \mu_x)(y[n] - \mu_y)}{\sqrt{\sum_{n=0}^{N-1} (x[n] - \mu_x)^2 \sum_{n=0}^{N-1} (y[n] - \mu_y)^2}} \quad (24)$$

where μ_x and μ_y are the means of the signals $x[n]$ and $y[n]$, respectively.

The correlation coefficient lies between -1 and $+1$. Values close to $+1$ indicate a strong positive linear relationship, i.e. as one signal increases, so does the other. Values close to -1 indicate a strong negative linear relationship, where one signal increases as the other decreases. Values close to 0 suggest a weak or no linear relationship between the signals.

D. Signal Convolution

Convolution is an integral operation on two signals that produces a third signal; it expresses the amount of overlap of one signal as it is shifted over another signal. The convolution of two continuous-time signals $x(t)$ and $h(t)$ is given by:

$$x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau \quad (25)$$

where $*$ denotes the convolution operator, which is defined as the integral of the product of $x(t)$ with a time-reversed and shifted version of $h(t)$. The time reversal of $h(\tau)$, represented by $h(-\tau)$, involves reflecting the signal about the vertical axis that represents time origin ($\tau = 0$). Accordingly, the integral is evaluated for all shift values t , producing the convolution

signal $x(t) * h(t)$. For discrete-time signals $x[n]$ and $h[n]$ the convolution is expressed as:

$$x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] \quad (26)$$

The convolution process can be visualized as one signal sliding over the other signal, computing the area of overlap as a function of the degree of overlap. In contrast to correlation, which measures the similarity between two signals as a function of the displacement of one relative to the other, the convolution represents the amount of overlap between two signals as one is inverted and shifted over the other signal. Therefore, while correlation measures the similarity between two signals, convolution can be viewed as a measure of the effect of one signal on the other.

Convolution is particularly useful for describing how a linear time-invariant (LTI) system or filter responds over time to an input signal since the output of an LTI system or a filter for any arbitrary input can be determined by convolving the input signal with the system's impulse response. LTI Systems are systems that are characterized by linearity and time-invariance. Linearity, as explained earlier, implies that the response (output) to a weighted sum of inputs is the weighted sum of the responses to each input. Time-invariance means the system's response to a given input does not change over time. In signal theory, the impulse response of an LTI system or a filter, denoted as $h(t)$ for continuous-time representation or $h[n]$ in discrete-time case, represents the system's output when subjected to an impulse function. In continuous-time representation, the impulse function is denoted as $\delta(t)$ and is defined as a function that is zero at all times except at $t = 0$, where it is infinitely high in such a way that its integral over time is 1. In discrete time, the impulse function is denoted as $\delta[n]$ and is defined as a sequence that is zero at all samples except at $n = 0$, where its value is 1. The impulse response fully characterizes the system's or filter's behavior. Hence, given the impulse response of an LTI system or a filter for any arbitrary input, its output can be determined by convolving the input signal with the impulse response.

E. Signal Entropy

Entropy is an important concept that originated in information theory as a measure of "information" in a random variable. In signal processing, entropy quantifies the degree of unpredictability or randomness in a signal, thereby reflecting its structure and offering a quantitative measure of various signal characteristics. The fundamental concept of entropy-based applications in signal processing is to treat the signal as a random stochastic process. In this context, the signal samples represent a collection of possible outcomes of the process; each outcome is associated with a probability of occurrence. Hence, for a discrete signal where its samples are represented as a set of possible outcomes x_1, x_2, \dots, x_n with associated probabilities $p(x_1), p(x_2), \dots, p(x_n)$, the Shannon entropy is calculated as follows:

$$H(X) = - \sum_{i=1}^n p(x_i) \log p(x_i) \quad (27)$$

This formula calculates the average "information," "surprise," or "uncertainty" inherent in the signal's possible outcomes. In information theory, high entropy suggests that a signal contains a wide range of information. In signal processing, high entropy values are associated with high degrees of unpredictability, uncertainty, irregularity, and randomness, indicating a complex signal with more varying structures compared to predictable or repetitive patterns within the signal. On the other hand, low entropy values often imply a high degree of uniformity and predictability within a signal. This is illustrated in Fig. 7, which compares the entropy values of two signals: the random vibration signal $v(t)$ and a square wave. Each signal contains 12800 samples. The signals are segmented into 128 segments so that each segment contains 100 samples. Accordingly, the Shannon entropy (H) is calculated for each segment to quantify uncertainty within each segment and to assess signal irregularity in terms of the variation in entropy values between signal segments. The comparison shows that segments of the

signal can be assessed using entropy. High entropy signals have a high degree of randomness, whereas low entropy signals involve minimal randomness.

The Shannon entropy formula in (27), introduced by Claude Shannon in 1948 [40], is the fundamental form of entropy that is widely used in signal processing. Spectral Entropy is another type of entropy that applies Shannon entropy to the power spectrum of a signal, providing insight into its spectral characteristics. Other types of entropy include approximate entropy and sample entropy, commonly used in time-series analysis to measure the regularity and unpredictability of fluctuations in data series. More information on the different types of entropy can be found in [41].

III. SIGNAL TRANSFORMATION AND ANALYSIS

Signal transformation is a fundamental concept in signal theory; it plays an important role in many signal processing applications, such as signal analysis, filtering, denoising, compression, and feature extraction. Transforming signals from their time-domain representations into other-domain representations allows the signal to be analyzed in a new paradigm, revealing new aspects of the signal that are not apparent in the time domain. The process of signal transformation involves mapping a signal from its original domain (usually time) to a new domain. This is achieved by computing the inner product of the signal with a set of basic functions (kernels) that serve as the building blocks for representing the signal in the new domain. These functions are chosen based on the properties desired in the transformed domain, for example, sinusoids in Fourier transforms and wavelets in wavelet transforms. The inner product measures how much of the base function is present in the signal. A large value of the inner product indicates a high degree of similarity or a strong presence of the characteristics represented by the base function in the signal. This section reviews common frequency-domain and time-frequency domain transformations widely used in signal processing.

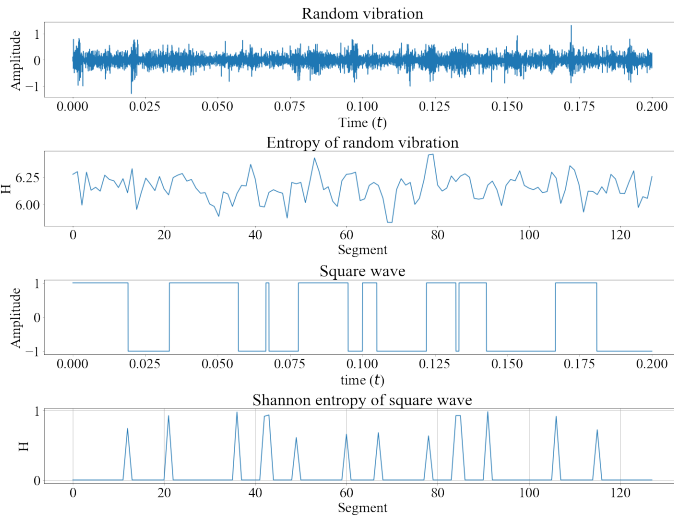


Fig. 7. A comparison between Shannon entropy values of a random vibration $v(t)$, and a square wave.

random vibration have higher entropy values than those of the square wave, indicating a higher degree of uncertainty. Moreover, the high variation of entropy values among different segments of the random vibration reflects the complex and irregular patterns in the signal. These concepts find useful applications in signal processing, including but not limited to:

- **Measurement of complexity:** Entropy serves as a metric for signal complexity, with higher entropy values indicating more complex signals. Such signals are typically characterized by their low compressibility.
- **Uncertainty Quantification:** Entropy is a direct measure of the uncertainty in a signal. A signal with high entropy suggests less predictability and greater uncertainty about its current state or future values.
- **Regularity Analysis:** Lower entropy values indicate a higher degree of regularity in the signal, which could indicate a fault or abnormal behavior in applications such as condition monitoring and fault detection.
- **Randomness Evaluation:** The degree of randomness in a

A. Fourier Transform

The Fourier transform (FT) is one of the most widely used transforms in signal processing. It provides a frequency-domain representation of the signal, revealing its spectral contents and enabling frequency-based analysis. The basic principle underlying the FT is that any waveform can be expressed as a combination of sinusoidal components with different amplitudes, frequencies, and phases. Accordingly, FT attempts to decompose signals into their constituent sinusoidal components. FT is mathematically expressed as:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt \quad (28)$$

where,

$F(\omega)$ represents the FT the function $f(t)$,

$f(t)$ is the original time-domain signal,

and $e^{-j\omega t}$ is the complex exponential function used in the transform.

The discrete Fourier transform (DFT) is the discrete version of the continuous FT, which is used for analyzing discrete signals; it is defined as:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn} \quad (29)$$

where,

$X[k]$ represents the DFT of the discrete time-domain signal, $x[n]$,

N is the total number of samples in $x[n]$,

$k = 0, 1, \dots, N-1$, is the index in the frequency domain,

and $n = 0, 1, \dots, N-1$, is the index in the time domain.

Both the FT and the DFT are reversible operations where the frequency domain representation of a signal can be transformed back into its original time-domain form by applying the inverse transform operation. This property is fundamental in signal processing and has various applications, such as signal compression, filtering, and noise reduction.

The fast Fourier transform (FFT) and its inverse, IFFT, are computational algorithms that efficiently implement the DFT and its inverse, IDFT. The key advantage of FFT and IFFT is their high computational efficiency. NumPy and SciPy Python libraries offer reliable functions for FFT and IFFT computation. When computing the FFT of a signal, two critical parameters need to be considered: the sampling rate, f_s , and the FFT length, n . The values of these parameters determine the frequency range and frequency resolution of the resulting spectrum. It is essential to choose an appropriate FFT length, usually a power of two (e.g., 256, 512, 1024), as the FFT algorithm performs efficiently when the number of points is a power of two. The FFT length also determines the number of frequency bins in the spectrum, where each bin corresponds to a specific range of frequencies within the total frequency spectrum analyzed by the FFT. The number of unique frequency bins in the FFT output is:

$$\frac{n}{2} + 1 \quad (30)$$

The bin width defines the frequency resolution, Δf , of the FFT representing the smallest distinguishable frequency in the spectrum. It is defined as:

$$\Delta f = \frac{f_s}{N} \quad (31)$$

Frequency resolution is critical in frequency analysis because it affects the accuracy with which closely spaced frequency components can be resolved in the resulting spectrum. If the FFT length is set equal to the number of samples N in the input signal, the FFT operates directly on the signal as it is. If the FFT length is set greater than N , the signal is typically zero-padded to the desired length. Zero padding does not alter the actual frequency content of the signal but increases the number of frequency bins in the FFT result, leading to a finer frequency resolution in the spectrum. On the other hand, setting the FFT length to less than N truncates the signal, resulting in loss of information and reduced frequency resolution. The maximum frequency that can be accurately represented in the spectrum is the

Nyquist frequency of the signal ($\frac{f_s}{2}$, which equals half of the sampling rate). Frequencies higher than the Nyquist frequency will not be correctly resolved in the spectrum. The output of the FFT function is an array that contains complex Fourier coefficients representing the amplitude and phase information of each frequency bin. The function `fftfreq` in Python can be used to obtain the frequency bins; it takes two inputs, the FFT length n and the sampling period, $T_s = \frac{1}{f_s}$, and returns an array containing frequency bins. Accordingly, the frequency spectrum is constructed by mapping the coefficients to the x -axis and the corresponding frequency bins to the y -axis. Since Fourier coefficients are complex numbers, each bin is expressed either by the magnitude or power of the corresponding Fourier coefficients, resulting in amplitude spectrum or power spectrum, respectively. This gives a two-sided frequency spectrum symmetric around the y axis, showing positive and "negative" frequencies. The negative frequency spectrum is inherent to the Fourier analysis of signals; it is a mirror of the positive "real" spectrum around the y -axis. Thus, it represents redundant frequency information. Reconstructing a positive spectrum involves discarding the second half of coefficients and frequency bins arrays. Thus, to accurately represent the signal's energy in the frequency domain, it's necessary to multiply the magnitudes of Fourier coefficients by 2 to restore the energy distribution in the positive frequency since the signal's energy was equally divided between the positive and negative frequencies. It is worth noting that the number of Fourier coefficients is directly proportional to the length of the input signal, as stated in (30). Consequently, the magnitude of the FFT coefficients increases with the length of the signal. Therefore, when longer signal segments are used, the frequency spectrum displays larger values as more components contribute to the sum. Therefore, to reflect the true amplitudes of the signal components, the magnitudes of the Fourier coefficients are usually normalised by the signal length.

To demonstrate the concept of FT in resolving various frequency components, a sinusoidal signal, denoted as $s_3(t)$ composed of three components, is created as shown in Fig. 8. The used sampling rate is 1600 Hz, and the length of the signal is 1536 data points; accordingly, this corresponds to 0.96 seconds of the signal's time duration. The signal consists of three sinusoidal components with different durations, amplitudes and frequencies. The first component consists of 960 data points and oscillates at 100 Hz with an amplitude of 10. The second component contains 64 data points with a frequency of 400 Hz and an amplitude of 5. The third component consists of 512 data points oscillating at a dominant frequency of 200 Hz and an amplitude of 20. Consequently, the time duration which corresponds to each of the components are 0.6 seconds, 0.04 seconds, and 0.32 seconds, respectively. Compared to the first and third components, the second component represents weak, rapid oscillations of very short duration, simulating a transient component of particular importance in the composite sinusoidal. The frequency spectrum of the signal is displayed in Fig. 9, which clearly shows that the three components in the composite signal at 100 HZ, 200 Hz, and 400 Hz are accurately resolved and represented in the spectrum. The

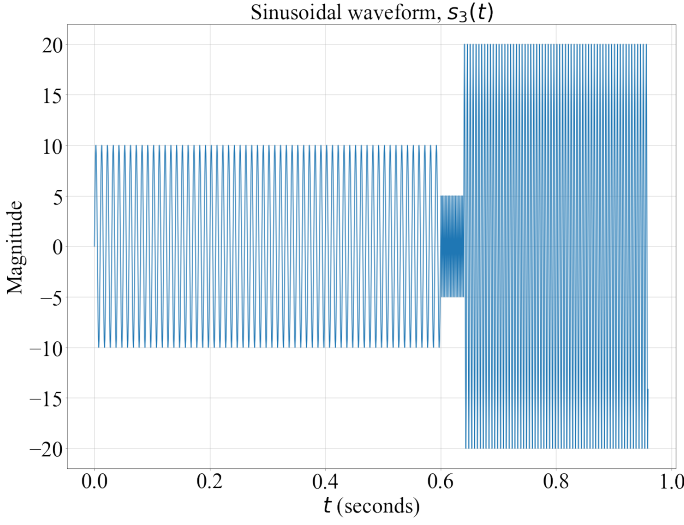


Fig. 8. Composite sinusoidal signal $s_3(t)$.

peaks correspond to the amplitudes of the components and therefore convey how much energy is being dissipated within each component. In practical terms, the presence of a high peak in the spectrum indicates the presence of a dominant (energetic) oscillatory component in the time domain signal. The frequency value associated with this peak reflects the oscillation rate of the component. While the FT can iden-

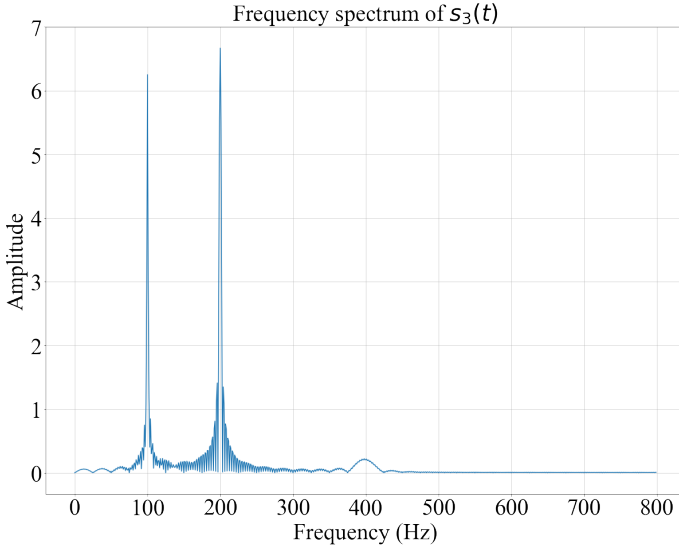


Fig. 9. Frequency spectrum of $s_3(t)$.

tify the three frequency components within the signal, its analysis spans the signal's entire duration, which is evident from the infinite integral in the transform equation (28). Thus, although it accurately distinguishes between different frequency components, it cannot provide temporal information about the timing of the occurrence of these frequencies within the signal. This lack of time resolution in the FT limits its applications to pure-frequency analysis only. Additionally, the FT is not ideal for analyzing signals with highly time-localized components, like short bursts with high energy concentration.

Such components produce a broad range of frequencies in the frequency spectrum of the signal due to the inherent uncertainty principle associated with Fourier analysis [42], which states that a signal cannot simultaneously have an arbitrarily small duration in time and an arbitrarily narrow bandwidth in frequency. This means that if a signal is highly localized in time, it must spread widely in the frequency domain and vice versa. This concept is demonstrated in Fig. 10 and Fig. 11, which show a composite sinusoidal signal, denoted as $s_{m3}(t)$, and its frequency spectrum, respectively. The signal $s_{m3}(t)$ is a modified version of the sinusoidal signal $s_3(t)$ where the duration of the second component is reduced to 0.05 (The duration of the first component is increased accordingly), and its amplitude is increased to 30 to simulate localized oscillations of high energy concentration. As depicted in Fig.

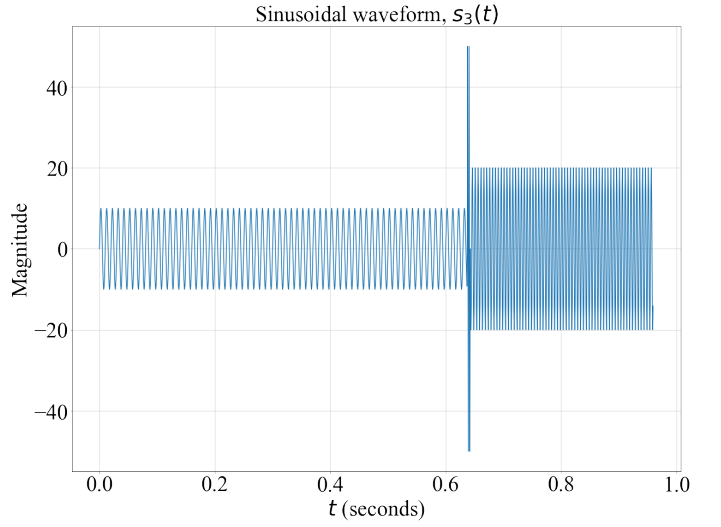


Fig. 10. Composite sinusoidal signal $s_{m3}(t)$.

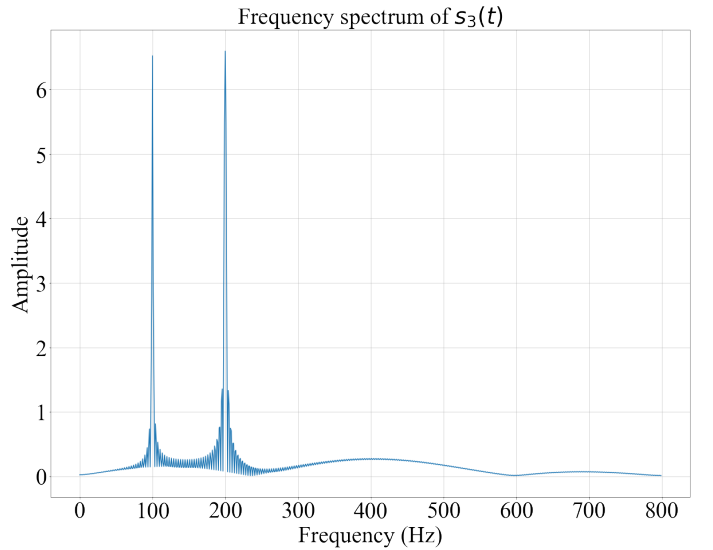


Fig. 11. Frequency spectrum of $s_{m3}(t)$.

11, the frequency of the localized component in $s_{m3}(t)$ is broadly spread across the frequency spectrum, which poses

a challenge in distinguishing subtle frequency components in practical applications.

B. Short-Time Fourier Transform

The short-time Fourier transform (STFT) bridges the gap between time-domain and frequency-domain analysis, making it particularly useful for analyzing nonstationary signals whose spectral characteristics evolve over time. Unlike FT, which spans the signal's entire duration for frequency analysis, STFT analyzes the signal over finite, overlapping time windows, thereby preserving temporal information. The computation of the STFT involves dividing the time-domain signal into segments of equal time length, which is achieved by multiplying the signal with a sliding window function. The FFT is then computed for each segment, providing frequency bins and corresponding Fourier coefficients for each segment. For a continuous signal, $x(t)$, the STFT is mathematically defined as:

$$STFT\{x(t)\}(\tau, \omega) = \int x(t)w(t - \tau)e^{-j\omega t} dt \quad (32)$$

where $w(t - \tau)$ is the window function centered at time τ , and ω is the angular frequency. For discrete signal $x[n]$, the discrete STFT is defined as:

$$DSTFT[n, k] = \sum_{m=0}^{N-1} x[n + m]w[m]e^{-j2\pi \frac{km}{N}} \quad (33)$$

where,

N is the size of the window, defining the number of signal samples included in each signal's segment for analysis,

n is the time index around which the window is centered,

k is the frequency bin index,

$w[n + m]$ represents the signal samples within the window,

$w[m]$ is the window function of size N ,

and $e^{-j2\pi \frac{km}{N}}$ is the DFT kernel.

The SciPy Python library provides an efficient implementation of STFT and its inverse transformation. The output of the STFT function contains three sets:

- The temporal set contains the duration of each segment in seconds. Its size equals the number of segments.
- The frequency set that contains groups of frequency bins corresponding to each time segment. The size of the set is equal to $\frac{\pi}{2} + 1$.
- Coefficients set: A matrix containing Fourier coefficients in each frequency bin for every segment. Thus, its size is $d_1 \times d_2$, where d_1 and d_2 are sizes of frequency and temporal sets, respectively. STFT features are usually obtained by examining the coefficients matrix. However, since Fourier coefficients are complex numbers, extracting features in the STFT commonly involves analyzing the magnitudes of these coefficients.

Heatmaps serve as an effective tool for visualizing the STFT output, providing a clear representation of the variations in the frequency content of the signal over time. These heatmaps, commonly referred to as spectrograms, are generated by mapping the temporal set to the x -axis and frequency set to the y -axis. The "intensity" of each time-frequency pair in the $x-y$

plane is represented by the magnitude of the corresponding Fourier coefficients. Fig. 12 shows the STFT of the composite signal $s_3(t)$ introduced earlier. The signal is segmented into ten segments using a Hamming window with an overlap of 50% of the segment length. The Fourier coefficients are computed for each segment using an FFT length of 2048. Compared to the signal's frequency spectrum depicted in 9, the STFT spectrogram shows the spectral contents and reveals their temporal characteristics simultaneously, providing time-frequency analysis of various frequency components in the signal. Specifically, The duration of each frequency component in the signal is clearly shown over the timeline (X -axis). Furthermore, the temporal edges, represented by the start and end times of each component, are well-defined, thereby reflecting high temporal resolution that allows the identification of the transient component at 400 Hz. In real-world scenarios, the ef-

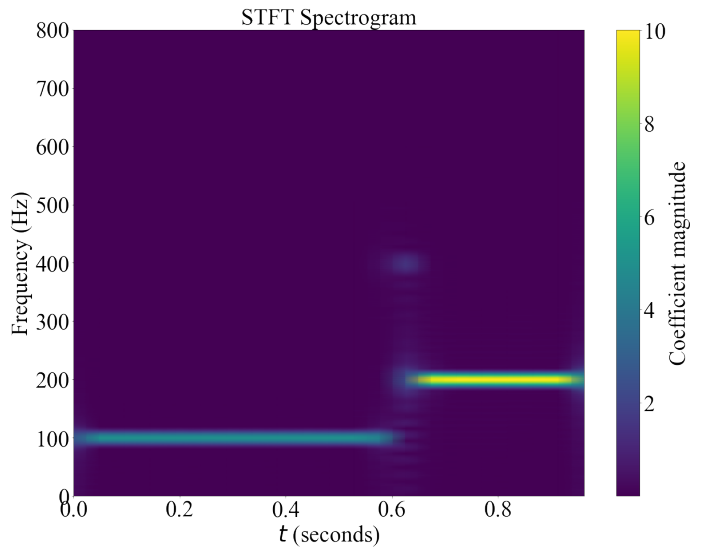


Fig. 12. STFT spectrogram of the composite sinusoidal signal $s_3(t)$.

ficiency of the STFT in time-frequency analysis is significantly influenced by the choice of its parameters, particularly the segment length and the percentage of overlap between adjacent segments. Segment overlap provides continuity and reduces information loss between adjacent segments. A high degree of overlap introduces more redundancy into the segments and improves time resolution, which is particularly useful when a large segment length is used. The segment length determines the resolution of the time-frequency analysis. A smaller segment length enables better time resolution, which helps identify rapid transient patterns in the signal, as demonstrated in Figure 12. On the other hand, a larger segment length enhances frequency resolution by allowing more frequency cycles to fit within the segment. However, this comes at the cost of reduced temporal resolution, as transient changes in the signal over short time durations may be overlooked. Therefore, there is a trade-off between time and frequency resolutions in the STFT analysis. This concept is demonstrated in Fig. 13, which shows the STFT spectrogram of $s_3(t)$ re-computed by dividing the signal into three segments only instead of ten segments. The frequency resolution is represented by the

“thickness” of the timeline. A thinner timeline—reflecting a finer frequency resolution—allows precise identification of the corresponding frequency bin on the y -axis compared to a thick timeline that spans several frequency bins. A comparison of the two spectrograms shows that the use of a longer segment improves the frequency resolution but reduces the temporal resolution, resulting in overlapping temporal edges that obscure the identification of the transient component. The

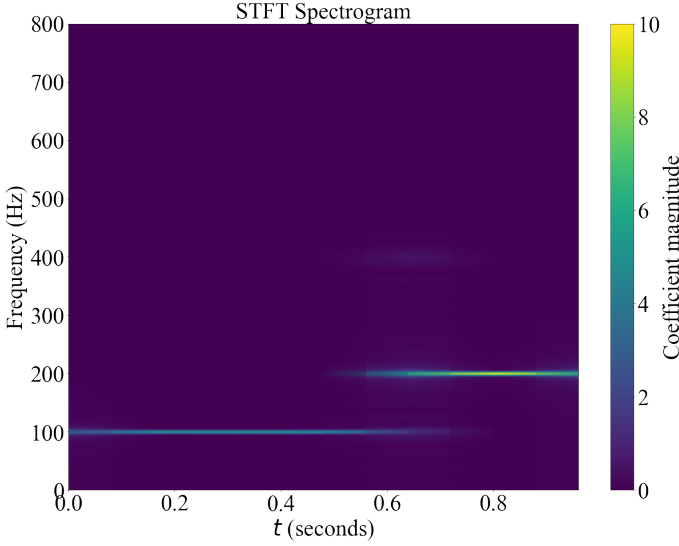


Fig. 13. STFT spectrogram of the composite sinusoidal signal $s_3(t)$.

fixed segment length is a major limitation in STFT that results in a uniform resolution analysis of the signal and, hence, leads to an inherent compromise between time and frequency resolutions when applying STFT for time-frequency analysis.

C. Power Spectral Density

Power spectral density (PSD) describes how the power of a signal is distributed across different frequencies, providing a measure of the power per unit frequency. This is achieved through the normalization of the power spectrum by frequency resolution (frequency bin width), converting the power spectrum into a power spectral density that shows the power distribution per unit frequency. Mathematically, the PSD $S_{xx}(f)$ of a continuous-time signal $x(t)$ is defined as the FT of the autocorrelation function $R_{xx}(\tau)$; it is expressed as:

$$S_{xx}(f) = \int_{-\infty}^{\infty} R_{xx}(\tau) e^{-j2\pi f\tau} d\tau \quad (34)$$

where,

$$R_{xx}(\tau) = E[x(t)x(t+\tau)] \quad (35)$$

and $E[\cdot]$ denotes the expected value. For discrete signals, the discrete PSD (DPSD) is utilized, involving the DFT of the autocorrelation sequence. The DPSD $S_{xx}[k]$ of a discrete-time signal $x[n]$ is given by:

$$S_{xx}[k] = \sum_{n=-\infty}^{\infty} r_{xx}[n] e^{-j\frac{2\pi}{N}kn} \quad (36)$$

where $r_{xx}[n]$ is the autocorrelation sequence of $x[n]$.

The SciPy Python library provides two methods for PSD estimation: the periodogram method (`scipy.signal.periodogram`) and the Welch method `scipy.signal.welch` [43]. The periodogram estimation involves obtaining the spectral power of the signal by applying FFT and squaring the magnitude of the resultant FFT coefficients. The PSD estimate is obtained by normalizing the spectral power by the frequency resolution of the FFT. One of the limitations of the periodogram method is its high variance, especially when the number of samples in the signal is limited. The Welch method overcomes this limitation by segmenting the signal into overlapping segments using a sliding window, computing the periodogram for each segment, and averaging the periodograms to obtain the PSD estimate. This effectively reduces noise and provides a smoother estimate than the periodogram method. Fig. 14 compares the two methods in estimating PSD of the random vibration signal $v(t)$ introduced earlier. In the Welch method, the signal is divided into four segments with 50% of overlap. It is evident that averaging the estimate over several segments greatly reduces the variance of the estimate, resulting in a smoother PSD. The smoothness of the PSD is of significant importance in power spectral analysis and feature extraction since high estimation variance can hinder major characteristics in the PSD, such as peaks. A smooth PSD allows precise identification of such characteristics, improving the sensitivity of the extracted features.

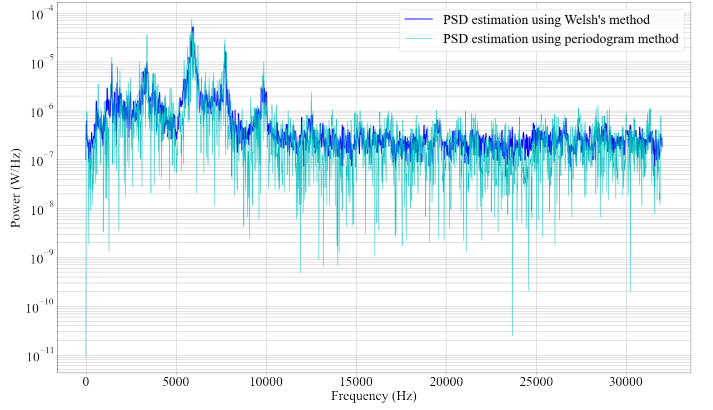


Fig. 14. Periodogram and Welch's PSD estimations of vibration signal $v(t)$.

Similar to the STFT, the choice of parameters in the Welch method, such as window type, segment length and overlap, has a significant effect on the PSD estimate. While STFT and the PSD possess similar aspects, they serve different purposes and are based on distinct theoretical foundations. Understanding their differences is crucial, especially in machine learning and feature extraction applications. Table I compares STFT and PSD and summarises their main characteristics.

D. Wavelet Transform

The accuracy of time-frequency analysis depends heavily on the localization of the base function in time and frequency domains. In Fourier analysis, the basis functions are sinusoids (complex exponentials) that are perfectly localized in the frequency domain since each corresponds to a single

TABLE I
COMPARISON BETWEEN STFT AND PSD

| Properties | STFT | PSD |
|----------------------------|---|--|
| <i>Purpose and Concept</i> | STFT is used for time-frequency analysis of signals, especially nonstationary signals where frequency components vary over time | PSD is used to describe the power distribution of a signal across frequency. It is particularly useful in understanding the energy content of different frequency components of a signal |
| <i>Representation</i> | The result of STFT is a two-dimensional representation of the signal, showing both time and frequency dimensions. This is crucial in analyzing signals whose spectral properties evolve over time | PSD provides a one-dimensional frequency domain representation, indicating how power is distributed across frequencies. Unlike STFT, it does not offer time-domain information |
| <i>Characteristics</i> | STFT involves a trade-off between time and frequency resolution controlled by the window size. A larger window offers better frequency resolution but poorer time resolution, and vice versa. | PSD assumes signal stationary, making it less suitable for analyzing signals with time-varying characteristics |
| <i>Application</i> | STFT is used for analyzing signals with time-varying spectral properties. | PSD is more suited for stationary signals or for assessing overall frequency content |

frequency component with no frequency spread. In the time domain, these sinusoidal functions extend infinitely, lacking time localization due to their infinite duration. In the context of wavelets, signal analysis is based on a family of wavelets that are localized in both time and frequency domains. These wavelets are generated from a single mother wavelet through scaling and translation, facilitating signal analysis at different frequencies and time intervals. A wavelet is a mathematical function that satisfies certain conditions, such as finite energy and a zero mean. The term "wavelet" itself means a small wave, which captures the essence of these functions and how they behave. The main characteristics of a wavelet function include the following:

- **Zero Mean:** This ensures the wavelet oscillates around the zero level; this is crucial for detecting changes in the signal since it allows it to capture both positive and negative variations in the signal effectively.
- **Finite Energy:** This implies that the total energy of a wavelet is finite, ensuring integrability and existence of the inner product with the signal. The energy of a given wavelet function is calculated by integrating the squared amplitude of the wavelet over its entire duration. Mathematically, the total energy of a wavelet function, $\psi(t)$, is expressed as:

$$\int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty \quad (37)$$

A wavelet with finite energy is inherently localized in time and has compact support, meaning it is non-zero only over a limited range, and its oscillations are confined to a small region. This time localization allows wavelets to effectively represent signals that have nonstationary or transient components. A wavelet's integrability implies that it won't produce infinite results when integrated over its domain. This is crucial for signal analysis since it is essentially carried out by computing the inner product of the signal with scaled and transformed versions of the wavelet.

Wavelet analysis is conducted through two operations, scaling and translation:

- **Scaling (Dilation):** This involves stretching or compressing the wavelet. Scaling changes the frequency content of the wavelet, allowing the signal to be analysed in different

frequency bands. Specifically, the oscillation frequency of a wavelet is directly affected by the scale. At lower scales, the wavelet compresses and oscillates more rapidly, which allows it to detect high-frequency components or rapid changes in the signal. On the other hand, at higher scales, the wavelet is stretched, resulting in fewer oscillations for the same amount of time, making it more sensitive to low-frequency components or slow-changing features in the signal.

- **Translation (Shifting):** This refers to moving the wavelet along the time axis, enabling the wavelet to analyze different signal parts at a given scale.

These aspects allow high time-frequency localization and provide a multi-resolution analysis through varying scales and translations. The choice of the mother wavelet and scale values is derived by signal characteristics and application-specific requirements. Attributes such as orthogonality, compact support, vanishing moments, and the ability to capture signal discontinuities and singularities play a crucial role in this choice. The wavelet family includes various mother wavelets; each with unique properties that cater to various signal types and application requirements. Fig. 15 displays the time-domain waveforms of some of the commonly used wavelet functions.

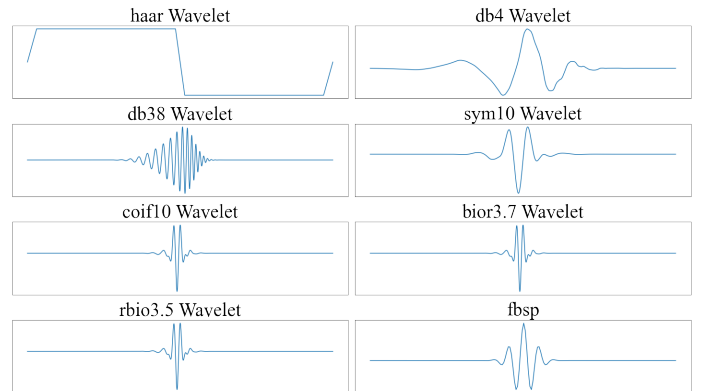


Fig. 15. Some of the commonly used wavelet functions.

1) *The Continuous Wavelet Transform (CWT)*: The CWT of a signal $x(t)$ can be expressed mathematically as follows:

$$CWT_x(a, b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t) \psi^* \left(\frac{t-b}{a} \right) dt \quad (38)$$

where,

a is the scale parameter, b is the translation parameter, $CWT_x(a, b)$ represents the wavelet coefficient at scale a and translation b ,

$\psi(t)$ is the mother wavelet, and $\psi^*(t)$ is its complex conjugate.

The output of CWT is a matrix of wavelet coefficients, where each row contains the coefficients of the corresponding scale. These coefficients represent the correlation between the signal and a scaled and translated wavelet, indicating how closely the signal matches the wavelet at those specific scales and translations. This emphasizes the importance of selecting a proper wavelet function for signal analysis. The CWT output is commonly visualized through 2D heatmaps, known as scalograms, where the x -axis and y -axis represent time and scale, respectively. The intensity of the heatmap is represented by the magnitude of the wavelet coefficients, which reflect the signal wavelet energy at different scales and translations. The scale parameter inversely relates to the signal's frequency content. High scales correspond to low frequencies, capturing the slower-varying components of the signal. On the other hand, low scales, corresponding to high frequencies, capture rapidly changing components within the signals. Hence, scalograms show how the spectral content varies over time, enabling the identification of transient or nonstationary components within the signal. Fig.16 displays the scalogram of the composite signal $s_3(t)$ using a scale range from 1 to 135. The fb_{sp} , displayed in the bottom right corner of Fig. 15, is employed in the analysis; fb_{sp} is a spline wavelet constructed using a spline function. The selection of this wavelet is based on its sinusoidal-like waveform, resembling the waveform of the composite signal $s_3(t)$. The scalogram demonstrates a multi-resolution signal analysis at various scale values. In particular, scales greater than 120 effectively capture the slowly varying components (components 1 and 3) of the signal. On the other hand, the rapid change in the signal, represented by component 2, can only be detected at small-scale values (scales between 30 and 40).

2) *Discrete Wavelet Transform (DWT)*: DWT is a wavelet-based signal decomposition tool that uses discrete values for scale and translation, reducing redundancy and computational complexity. In DWT, wavelet multi-resolution analysis is achieved by decomposing the signal into high and low-frequency components, commonly known as elementary modes, using digital filter banks and down-sampling operations. The Key parameters include the type of mother wavelet and the degree of decomposition. Mathematically, the DWT of a discrete signal $x[n]$ can be expressed as:

$$DWT_x[j, k] = \sum_{n=-\infty}^{\infty} x[n] \psi_{jk}[n] \quad (39)$$

where,

$DWT_x[j, k]$ represents the wavelet coefficient at the j -th level

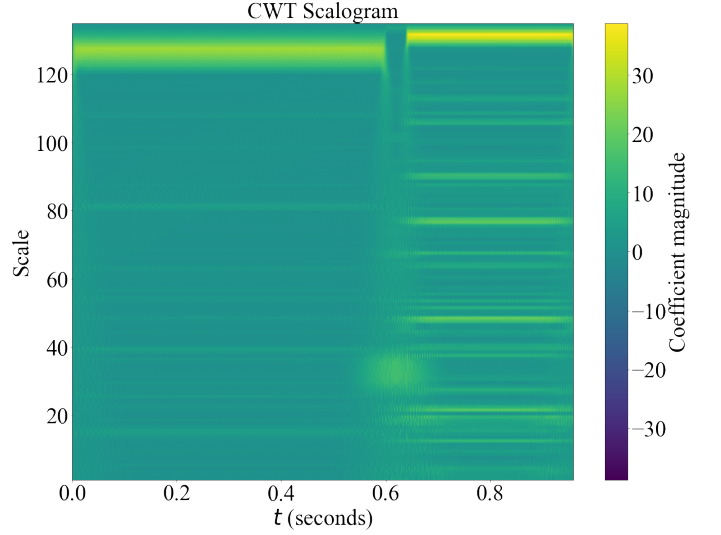


Fig. 16. CWT (scalogram) of the composite signal $s_3(t)$.

of decomposition and the k -th position,

$\psi_{jk}[n]$ are the discrete wavelets, defined as scaled and shifted versions of the mother wavelet.

In the decomposition process, the signal undergoes convolution with consecutive low-pass and high-pass filters followed by down-sampling. The main aspects and steps of the decomposition process include:

Mother Wavelet: The first step involves selecting an appropriate wavelet base function, often called the "mother wavelet."
Scaling and Translation: The decomposition process starts by scaling and translating the mother wavelet to generate a family of wavelet functions that serve as bases to decompose the signal. In the DWT, the mother wavelet is used in a discretized form where scaling and translation are done in discrete steps, typically in powers of two for scaling and integer multiples of that scale for translations. For each level of decomposition, the wavelet is scaled by a factor of 2^j , where j indicates the level of decomposition. This scaling effectively compresses the wavelet to capture signal features at different frequencies. Consequently, the wavelet is translated along the signal in steps that are multiples of the current scale factor 2^j . This stepwise translation allows the wavelet to cover the entire signal, ensuring that all signal parts are analyzed.

High-Pass Filter: This filter represents the mother wavelet since its coefficients are essentially a discrete representation of the mother wavelet. As a high-pass filter, it extracts high-frequency *detail* components from the signal, capturing rapid changes within the signal.

Low-Pass Filter: This filter represents the scaling function, as its coefficients are derived from the scaled wavelets. As a low-pass frequency filter, it extracts low-frequency *approximate* components that vary slowly over time.

Down-Sampling: After separating the signal into high and low-frequency components, each component is down-sampled by a factor of two. Down-sampling involves keeping every second sample of the filtered signals. This process reduces the sample rate by half, effectively decreasing the data size and ensuring

the transform is computationally efficient.

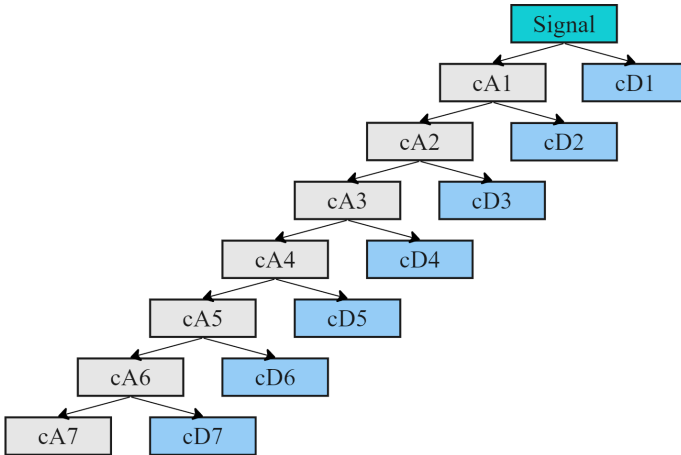
Approximate and Detail Coefficients: Approximation and detail coefficients are outputs of down-sampling operations performed on approximate and detail components, respectively. The approximate coefficients represent a smoothed and down-scaled version of the original signal, while the detail coefficients describe high-frequency content in the signal.

Iterative Analysis: The process of high and low pass filtering followed by down-sampling is performed iteratively on the resultant *approximation coefficients only* until reaching the specified number of levels (j) or until reaching maximum decomposition constrained by signal length. This process creates a j -level tree structure of wavelet coefficients, shown in Fig.17.

Signal reconstruction: The original signal can be perfectly reconstructed by applying the inverse DWT, which processes the resultant coefficients through reconstruction filters in an inverse manner to the decomposition process.

Elementary modes: These modes are obtained through the individual reconstruction of the resultant coefficients (coefficients $cD1$ to cDj and cA_j as shown in Fig. 17), allowing the processing of the composite signal at the level of its constituent modes. Since these modes form the underlying structure that constitutes the original signal, perfect reconstruction of the original signal is possible through direct summation of these modes.

Fig. 18 displays elementary modes of the random vibration



cA_n : Approximation coefficients of level n
 cD_n : Detail coefficients of level n

Fig. 17. 7-level DWT decomposition tree.

signal $v(t)$ along with their corresponding spectra. These modes are obtained through 7-level DWT decomposition of the signal using Daubechies $db4$ wavelet (shown in the top right corner of Fig. 15). The Daubechies family of wavelets, especially those with higher orders (e.g., $db4$, $db5$), are compactly supported and well-localized in time, making them highly effective for analyzing signals that contain transient patterns or nonstationary components. This is particularly useful in vibration analysis [44], where detecting sudden changes or faults is essential for condition monitoring and diagnostics. The DWT decomposition of $v(t)$ iteratively extracts slowly

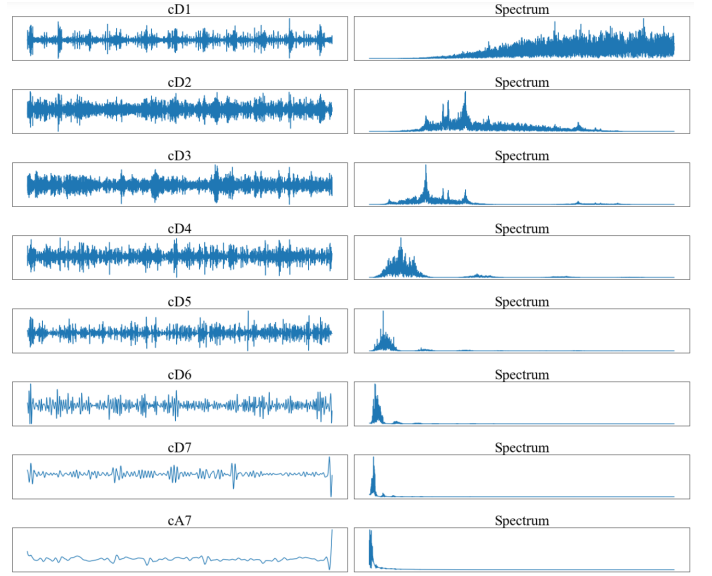


Fig. 18. 7-level DWT decomposition of random vibration signal $v(t)$: Elementary modes and their corresponding frequency spectra.

varying modes of the signal, corresponding to lower frequencies in the signal at each level. This progressively reduces the resolution of high-frequency details with each level of decomposition, making DWT well-suited for signals where the information of interest is concentrated in lower frequencies.

3) Stationary Wavelet Transform (SWT): A major limitation of the DWT is the lack of time invariance, which stems primarily from the down-sampling process. The down-sampling performed at each level makes the output of DWT sensitive to shifts in the input signal. Even small shifts can cause significant changes in the resulting wavelet coefficients, which is problematic in applications where the precise timing of signal features is essential. Furthermore, down-sampling reduces the signal length by half at each level of decomposition, which may lead to the loss of specific signal details, especially at higher decomposition levels. SWT is an extension of the DWT that addresses these limitations by omitting the down-sampling step, thus maintaining the original size of the signal throughout the decomposition levels. Unlike DWT, which provides a critically sampled (non-redundant) representation, SWT provides a more detailed but redundant representation of the signal. SWT is implemented by applying high-pass and low-pass filters without downsampling the filtered signals. Instead, the filters are up-sampled at each level to match the size of the original signal. The up-sampling of filters is done by inserting zeros between the filter coefficients, effectively increasing the size of the filters as the decomposition level increases. This leads to an overcomplete signal representation and increases computational burden compared to DWT.

4) Wavelet Packet Transform (WPT): The WPT extends the capabilities of classical wavelet-based decomposition. Unlike the DWT, which limits its decomposition to approximation coefficients, the WPT performs a full decomposition at each level by processing detail and approximation coefficients, as depicted in Fig. 19, thereby capturing both low-frequency

and high-frequency components. While this process is more computationally intensive, it enhances frequency resolution in the analysis, enabling the detection of subtle signal features that DWT may overlook. Fig. 20 displays elementary modes

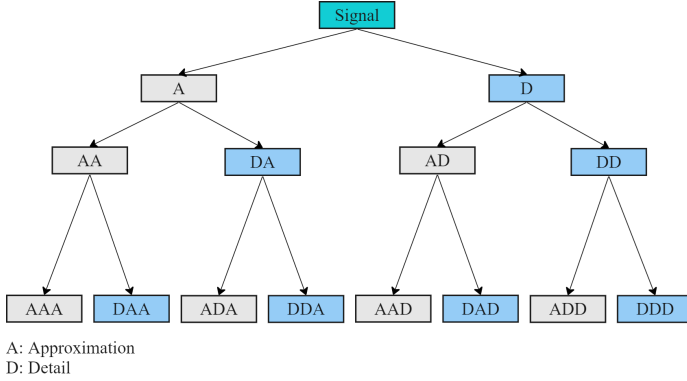


Fig. 19. 3-level WPT decomposition tree.

of the vibration signal $v(t)$, obtained through 3-level WPT decomposition using the $db4$ wavelet, along with their corresponding spectral contents. In contrast to DWT, which offers

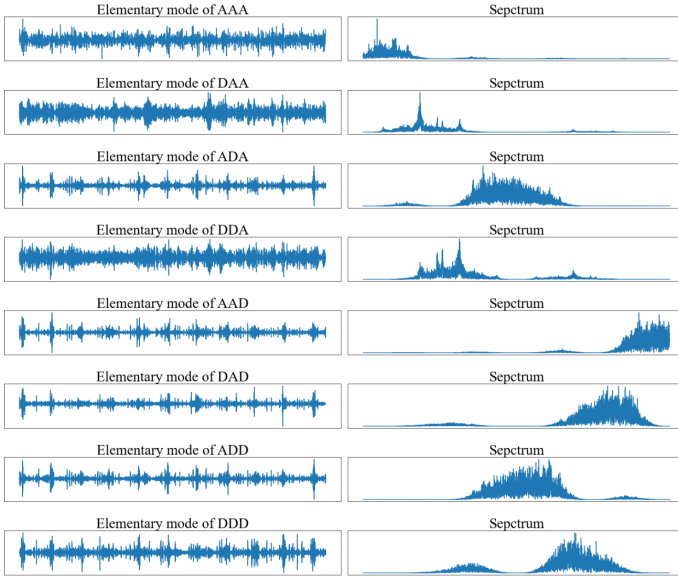


Fig. 20. 3-level WPT decomposition of random vibration signal $v(t)$: Elementary modes and their corresponding frequency spectra.

a hierarchical frequency decomposition with good frequency resolution at lower frequencies, WPT provides a more uniform time-frequency localization across the spectrum due to its symmetric decomposition of detail and approximation coefficients at each level. Hence, it maintains a balance between time and frequency resolution throughout the frequency spectrum, allowing for a more precise localization across a wider range of frequencies. Additionally, the spectrum in DWT is skewed towards lower frequencies with each successive level of decomposition. While this can be advantageous for analyzing signals characterized by low-frequency components, it inadequately represents higher-frequency details. WPT, on the other

hand, provides a balanced spectrum with a more equitable representation of both high and low-frequency components.

Wavelet decomposition offers an effective tool for analyzing complex real-world signals that involve transit and nonstationary components, opening the door for a wide range of applications such as:

- **Multi-resolution analysis:** Wavelet decomposition methods attempt to decompose the signal into elementary modes characterized by their high time-frequency localization, offering the capacity for conducting a multi-resolution analysis of the original signal.
- **Feature Extraction:** The obtained modes encompass meaningful information as they exhibit distinct amplitude and frequency characteristics. Accordingly, distinctive features directly related to the inherent structure of the signal can be extracted from these modes or their corresponding coefficients.
- **Noise Reduction:** Noise and signal components often have different characteristics in the wavelet domain. By thresholding certain wavelet coefficients before signal reconstruction [45], noise can be effectively reduced or eliminated.
- **Signal Enhancement:** Wavelet reconstruction can be utilized to enhance signals, where certain characteristics can be amplified or smoothed out as required, improving the quality of the reconstructed signal.
- **Signal Compression:** By keeping only the most significant wavelet coefficients for reconstruction, signals can be compactly represented with minimal loss of quality.

As previously stated, selecting the appropriate mother wavelet and decomposition level are crucial steps in wavelet analysis. Different wavelets have different shapes and properties, such as smoothness, symmetry, and the number of vanishing moments, making the selection mainly dependent on application requirements and the shape similarity between the wavelet and the signal. The decomposition level determines the resolution at which the signal is analyzed. Higher decomposition levels allow for the analysis of lower-frequency components but also increase computational complexity and could dilute the significance of higher-frequency components. The maximum level of decomposition in DWT and WPT is typically limited by the length of the signal, N . Generally, each level of decomposition reduces the number of data points by half. Thus, the number of maximum possible decomposition levels equals $\log_2(N)$. In SWT, the maximum decomposition level is determined by the signal's length and the filters' length at each decomposition level, which increases with each level due to the up-sampling operation. Further decomposition becomes impractical when the filters become too long relative to the signal.

The Python PyWavelets package provides comprehensive modules for wavelet analysis. It supports custom wavelets and provides various wavelet functions such as CWT, DWT, SWT, WPT and over 100 built-in wavelet filters. Other libraries supporting wavelet analysis include PyCWT, a Python module for continuous wavelet spectral analysis, and scipy.signal, which provides CWT and other wavelet functions, particularly

for signal filtering.

E. Hilbert Transform

The Hilbert transform (HT) is a fundamental operator in signal theory; it is particularly useful in obtaining the analytic signal representation of real-valued signals. An analytic signal is a complex-valued representation of the signal that provides a comprehensive way to describe both its amplitude and phase characteristics. The HT of a signal $x(t)$ is defined as:

$$H\{x(t)\} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau \quad (40)$$

The transform essentially modifies the phase of each frequency component of the signal by $\pm 90^\circ$. The $X_a(t)$ of $x(t)$ is formed by augmenting the signal with its HT $H\{x(t)\}$ as the imaginary part. Mathematically, it is expressed as:

$$X_a(t) = x(t) + jH\{x(t)\} \quad (41)$$

Represented in its polar form, the analytical is expressed as:

$$x_a(t) = A(t)e^{j\theta(t)} \quad (42)$$

where, $A(t)$ is the instantaneous amplitude (also known as the amplitude envelope), it is given by:

$$A(t) = |x_a(t)| = \sqrt{x(t)^2 + H\{x(t)\}^2}, \quad (43)$$

and $\theta(t)$ is the instantaneous phase; given by:

$$\theta(t) = \arctan\left(\frac{H\{x(t)\}}{x(t)}\right) \quad (44)$$

Accordingly, the instantaneous frequency $f(t)$ can be obtained by taking the derivative of the instantaneous phase $\theta(t)$:

$$f(t) = \frac{1}{2\pi} \frac{d\theta(t)}{dt} \quad (45)$$

The instantaneous amplitude of a signal shows how the signal strength varies with time. The instantaneous phase provides the phase angle of the signal as a function of time. The instantaneous frequency reveals how the frequency content of a signal evolves over time, providing a dynamic view of the signal's spectral properties. By obtaining instantaneous amplitude, phase, and frequency information, the HT serves as an effective tool to identify distinct characteristics, such as common patterns and sudden changes in phase and frequency where relevant features can be extracted accordingly. For instance, HT can be used to obtain instantaneous amplitudes of amplitude-modulated signals such as fault vibration signals. In these signals, vibrations of damaged bearings are manifested as modulations in the amplitude of the generated vibration signal. Hence, obtaining the signal envelope provides an efficient approach for extracting fault signature frequencies from the envelope's spectrum. Fig. 21 displays a composite signal, $c(t)$, that has a time duration of 1 second. The signal is composed of various components so that it exhibits four sudden changes as follows:

- Rapid oscillations of 50 Hz from 0.3 time instant to 0.5 time instant.

- Rapid oscillations of 100 Hz from 0.7 time instant to 0.9 time instant.
- Abrupt phase change of 180 degrees at 0.4 time instant.
- Abrupt phase change of 90 degrees at 0.6 time instant.

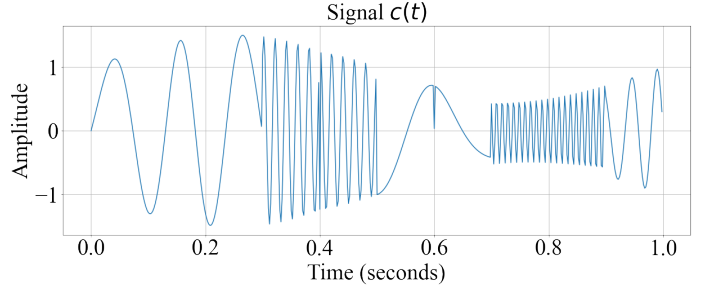


Fig. 21. Composite signal $c(t)$.

The function `scipy.signal.hilbert` in the `scipy.signal` library is used to compute the HT of $c(t)$. The function returns the analytical signal $c_a(t)$ as an output. The instantaneous amplitude, phase, and frequency are calculated accordingly and displayed in Fig. 22. As shown in the plots, the obtained

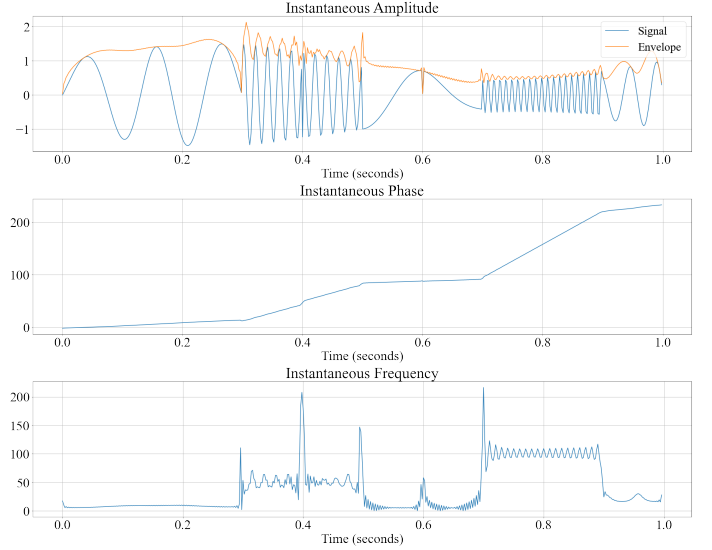


Fig. 22. Hilbert Transform of the signal $c(t)$.

instantaneous information provides critical insights into the signal's behavior, highlighting common patterns and sudden changes. Specifically, the envelope shows the main pattern of the signal, reflecting its energy content over time. The abrupt phase changes at the 0.4 and 0.6 time instants are manifested as small step changes at these time instants in the instantaneous phase. Moreover, the phase is directly impacted by temporal changes in the signal frequency, as shown in the plot. The instantaneous frequency provides vital information about frequency variations and sudden changes in the signal. Particularly, the instantaneous frequency plot shows the rapid oscillations of 50 Hz and 100 Hz between 0.3 – 0.5 and 0.7 – 0.9 time instants, respectively. Additionally, the sudden changes in frequency and phase are manifested as a spike at the corresponding time instant in the plot. In real-world

applications, such spikes represent significant features in the signal that would be of particular interest.

F. Hilbert–Huang transform

The Hilbert–Huang transform (HHT) [46] is a well-known adaptive method for analyzing nonlinear and nonstationary signals. Unlike traditional transforms, HHT does not impose a fixed basis function in signal analysis. Instead, it utilizes an adaptive approach to analyze the signal, making it highly responsive to variations in signal. The HHT is a two-stage process that involves empirical mode decomposition (EMD) of the signal followed by the Hilbert Spectral Analysis (HSA).

1) *Empirical Mode Decomposition (EMD)*: EMD is an adaptive decomposition method that decomposes the signal into a set of simpler functions known as intrinsic mode functions (IMFs). EMD forms the basis of adaptive mode decomposition (AMD) methods introduced over the last three decades. In contrast to wavelet-based decomposition, AMD uses adaptive approaches rather than priori basis functions to decompose the signal. Hence, the obtained IMFs are not influenced by a priori basis function [47]. This makes AMD particularly effective when dealing with nonlinear and nonstationary signals due to its ability to adapt to varying signal characteristics. However, this adaptability comes at the cost of higher computational complexity. The EMD algorithm, introduced in 1998 as part of the HHT, represents the core of all AMD methods. The process of EMD, commonly known as sifting, involves the following steps:

- 1) Identification of Extrema: Start by identifying all the local maxima and minima of the original signal.
- 2) Envelope Creation: Construct the upper and lower envelopes of the signal by interpolation between the local maxima and minima, respectively. These envelopes essentially outline the signal's oscillatory amplitude.
- 3) Mean Envelope Calculation: Compute the mean of the upper and lower envelopes.
- 4) Extraction of Detail: Subtract the mean envelope from the original signal. This step isolates a component of the signal that potentially qualifies as an IMF.
- 5) IMF Check: Verify if the extracted component meets the criteria for being an IMF:
 - The number of extrema (local maxima and minima) and the number of zero crossings in the component must either be equal or differ at most by one. This condition ensures that the IMF captures a well-defined oscillatory mode without bias toward upward or downward trends.
 - The mean value of the envelope defined by the local maxima and the envelope defined by the local minima must be zero at any point in the component. This condition guarantees that the IMFs have well-balanced oscillations around zero, reflecting true oscillatory modes rather than trends or biases in the signal.

If the component doesn't qualify for an IMF, return to step 2 and use this component as the new signal.

- 6) Completion of one IMF: Once an IMF is identified, subtract it from the original signal. This leaves a residue signal.
- 7) Repetition: Repeat steps 1-6 on the residue signal. This process is iterated, yielding a new IMF at each iteration, and the residual becomes the input for the next iteration.
- 8) Termination (Stopping Criterion): The process terminates when the residual signal becomes a monotonic function from which no more IMFs can be extracted, or it becomes sufficiently small in amplitude (based on a predefined threshold)

The output of the EMD process of the random vibration signal $v(t)$ is displayed in Fig. 23, including IMFs, residual, and their respective spectra. As shown, the EMD adaptively

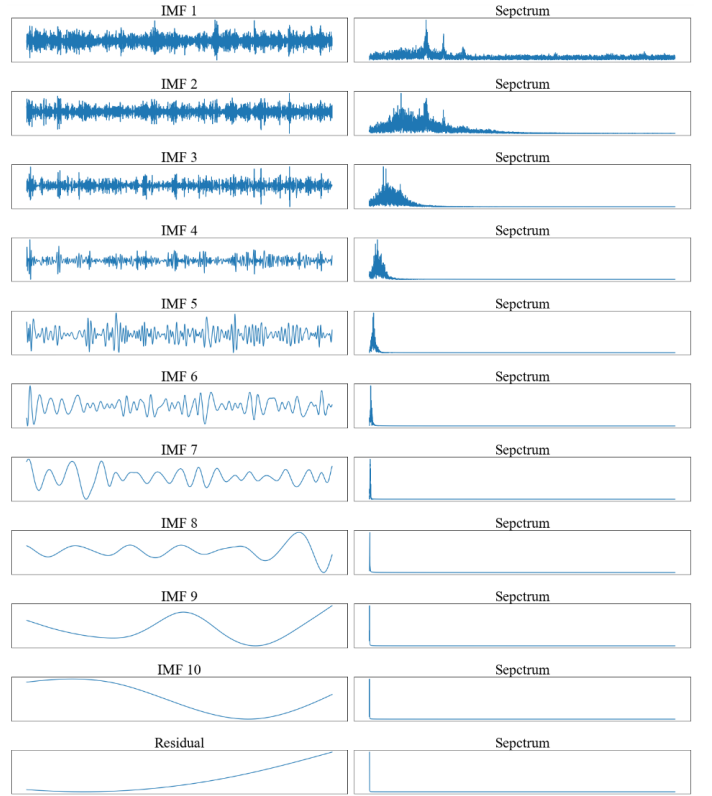


Fig. 23. EMD of the random vibration signal $v(t)$: IMFs, residual, and their respective frequency spectra.

decomposed the signal into ten IMFs of distinct frequency spectra where the frequency contents of IMFs decrease with each successive IMF. While EMD is well-known to be effective in analyzing complex signals, the decomposition process is sensitive to noise since it works directly on the signal's extrema and minima. Additionally, the process requires a complete envelope to decompose the signal into IMFs accurately, which can be difficult to define at the boundaries due to the absence of neighboring data points. This can lead to distortions of IMFs near the ends of the signal, commonly known as the end effect [48], impacting the accuracy and reliability of EMD for signal analysis, particularly when the signal's behavior at the boundaries is critical. Another challenge in EMD is mode mixing [49]; ideally, each IMF should represent a unique frequency component. However, in signals with

close spectral proximity or exhibit intermittency characteristics such as abrupt amplitude and frequency changes, the standard EMD algorithm may not be able to separate these features accurately. As a result, a single IMF may contain information from multiple modes, which can lead to a loss of significant physical interpretation of the IMFs. Further improvements to the standard EMD algorithm have been introduced to address these limitations. For example, the ensemble empirical mode decomposition (EEMD) algorithm [50] was developed by incorporating an ensemble approach and adding finite amplitude white noise to alleviate problems of mode mixing and end effects. Accordingly, the natural oscillatory modes are obtained by averaging the corresponding IMFs obtained from an ensemble of the signal and the added noise. Other common AMD algorithms include variational mode decomposition (VMD) [51], local mean decomposition [52], and empirical wavelet transform (EWT) [53]. More details on AMD techniques and their applications can be found in [54] [55] [56] [57].

2) *Hilbert Spectral Analysis*: The second step of HHT involves applying the HT to each IMF to obtain its analytic signal and, accordingly, its instantaneous amplitude, phase, and frequency. An IMF, due to its defining conditions, ensures that its HT is well-behaved and meaningful for signal analysis [58]. Specifically, due to the equal number of extrema and zero crossings, an IMF exhibits a consistent oscillatory pattern, which is essential for a coherent HT. Further, the symmetry of the IMF's envelope about the zero line guarantees that the IMF doesn't have a bias towards positive or negative values, facilitating a more accurate HT.

Mathematically, the HSA can be expressed as follows: Given a signal $x(t)$, decomposed into n IMFs $c_i(t)$, where $i = 1, 2, \dots, n$, the analytic signal $Z_i(t)$ for each IMF is formed as follows:

$$Z_i(t) = c_i(t) + jH\{c_i(t)\} \quad (46)$$

where $H\{c_i(t)\}$ is the HT of $c_i(t)$. The analytical signal in the polar form is expressed as:

$$Z_i(t) = A_i(t)e^{j\theta_i(t)} \quad (47)$$

Accordingly, the instantaneous amplitude $A_i(t)$ is:

$$A_i(t) = |Z_i(t)| = \sqrt{c_i(t)^2 + H\{c_i(t)\}^2}, \quad (48)$$

and the instantaneous phase $\theta_i(t)$ is:

$$\theta_i(t) = \arctan\left(\frac{H\{c_i(t)\}}{c_i(t)}\right) \quad (49)$$

Accordingly, the instantaneous frequency $f_i(t)$ is obtained by taking the derivative of $\theta_i(t)$:

$$f_i(t) = \frac{1}{2\pi} \frac{d\theta_i(t)}{dt} \quad (50)$$

The instantaneous frequencies and amplitudes obtained from the HT, can be used to construct a time-frequency distribution of the signal. This is accomplished by plotting the instantaneous amplitude or energy against the instantaneous frequencies of the IMFs for each point in time. The result is a distribution that shows how the signal's frequencies evolve over time, highlighting the signal's nonlinear and

nonstationary characteristics. The time-frequency-energy distribution is commonly visualized as a 2D heatmap, where the x -axis represents time, the y -axis represents frequency, and the color intensity represents the amplitude or energy level of the signal at each time-frequency point. Considering the composite sinusoidal signal $s_3(t)$, Fig. 24 displays the resulting IMFs and residuals from the EMD process, as well as their spectral contents. Consequently, Fig. 25 shows the 2D heatmap visualization of the output from the HHT. These results are obtained using `textitemd`, `vmd`, and `hht` built-in MATLAB functions, which, in contrast to Python, provide a mature and user-friendly environment for EMD, VMD, and HHT computations and visualization. It is obvious that EMD

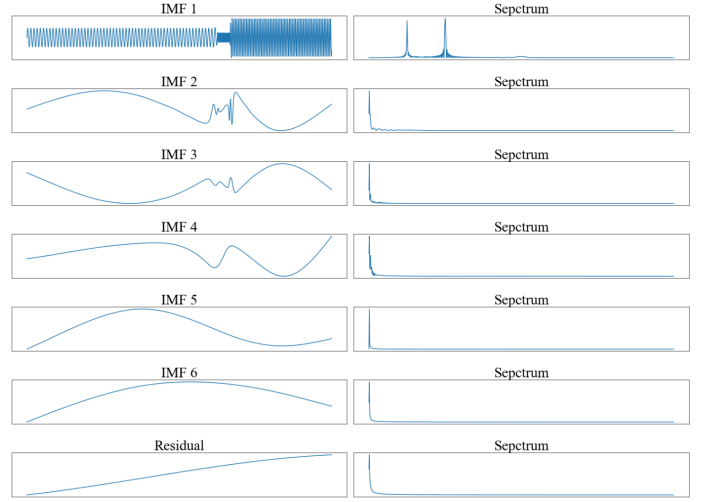


Fig. 24. EMD of the composite sinusoidal signal $s_3(t)$: IMFs and residual alongside with their respective frequency spectra.

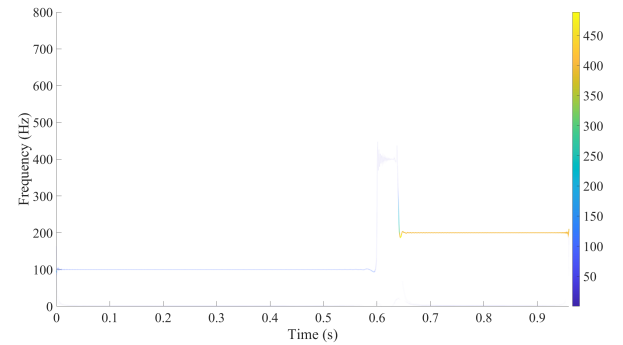


Fig. 25. 2D heatmap visualization of the HHT of $s_3(t)$.

and, consequently, HHT effectively capture all components in the signal, including the transit oscillations of 400 Hz at 0.6 seconds, and effectively shows the temporal energy distribution of the signal at different frequencies. This demonstrates the advantage of the HHT that lies in its adaptability to nonlinear and nonstationary signals.

However, the performance of the HHT is tied to the choice of AMD method, which is fundamental to the accurate decomposition of the signal into its IMFs. Furthermore, the employed stopping criterion for the sifting process [59] and

the used interpolation method for envelope estimation [60], have direct impacts on the effectiveness of the HHT. A proper stopping criterion ensures that the decomposition process neither overfits nor underfits the signal, thereby preserving the signal's essential characteristics without introducing artifacts. The interpolation method for envelope estimation influences the HHT's performance since it determines how well the upper and lower envelopes capture the true oscillatory modes of the signal.

G. Wigner–Ville Distribution

The Wigner–Ville distribution (WVD), as a time-frequency analysis tool is characterized by its ability to provide highly resolved energy-time-frequency representations of signals. The WVD of a continuous-time signal $x(t)$ is expressed as:

$$WVD(t, f) = \int_{-\infty}^{\infty} x(t + \frac{\tau}{2}) x^*(t - \frac{\tau}{2}) e^{-j2\pi f\tau} d\tau \quad (51)$$

where, $x^*(t)$ is the complex conjugate of $x(t)$, τ and f are variables representing time-shift and frequency, respectively. According to this expression, the kernel function of WVD is given by:

$$x(t + \frac{\tau}{2}) x^*(t - \frac{\tau}{2}), \quad (52)$$

which is essentially the instantaneous autocorrelation function (IACF) of $x(t)$, commonly known as Wigner auto-correlation function [61]. The use of the complex conjugate of the signal in the autocorrelation accounts for the magnitude and phase parts of the signal. Note that the traditional autocorrelation function of (20) integrates the autocorrelation over time t , providing a global measure of the correlation. In contrast, the IACF provides a local measure of the correlation, which is better suited to reflect local and time-varying features of the signal. Hence, facilitating a more accurate representation of the signal's characteristics [62]. The Fourier kernel $e^{-j2\pi f\tau}$ in the WVD expression transforms the IACF into the frequency domain with respect to the time-shift variable τ , thereby providing a joint time-frequency representation of the signal. For a discrete signal $x[n]$ with N samples, The discrete WVD is given by:

$$WVD[n, k] = \sum_{m=-N}^N x[n + \frac{m}{2}] x^*[n - \frac{m}{2}] e^{-j2\pi \frac{km}{N}} \quad (53)$$

Here, $x^*[n]$ is the complex conjugate of $x[n]$, and k represents the discrete frequency index. The term,

$$x[n + \frac{m}{2}] x^*[n - \frac{m}{2}], \quad (54)$$

captures the auto-correlation of the signal at different lags.

The output of WVD, $WVD(t, f)$, is essentially a 2D function of time and frequency that is commonly visualized as a 2D heatmap where the value of $WVD(t, f)$ at any point (t, f) reflects signal's energy at that particular time and frequency. The high energy concentration of WVD results from its inherent quadratic (nonlinear) structure since the energy itself is a quadratic representation of the signal [63]. In

contrast to FT, STFT, and CWT, which are linear transforms³, the WVD is nonlinear since it involves a product of the signal with a time-shifted version of itself (correlation). This energetic and correlative nature of the WVD [63] makes it a unique tool for energy-time-frequency analysis compared to other transforms, however the quadratic nature of the WVD presents a major challenge when analysing multi-component signals due to the high level of cross-term interference [64] that appears in the time-frequency representations. Cross-terms are caused by undesired cross-correlation between various signal components, leading to a false indication of signal components between the desired auto-correlation terms in the WVD representation of the signal. To overcome this problem, window-based approaches [65], [66] and kernel-based approaches [67], [68], such as the pseudo-Wigner distribution (PWVD), are commonly utilized to suppress cross-terms. PWVD is a windowed version of the WVD that applies a smoothing kernel to WVD in either time-domain or frequency domain to reduce the cross-terms at the cost of a certain loss in resolution. The PWVD of a signal $x(t)$ can be expressed as:

$$PWVD(t, f) = \int_{-\infty}^{\infty} h(\tau) x(t + \frac{\tau}{2}) x^*(t - \frac{\tau}{2}) e^{-j2\pi f\tau} d\tau \quad (55)$$

where $h(\tau)$ is the smoothing kernel in the time domain. In the frequency domain, the smoothing operation becomes the convolution operation between the smoothing kernel and the WVD. The smoothing operation essentially averages the signal's energy over the time and/or frequency domains, reducing the presence of cross-terms arising from the quadratic nature of the WVD. However, this comes at the cost of blurring the signal's time-frequency representation. The limitation of WVD resulting from undesired cross-correlation and the advantage of PWVD are demonstrated by comparing the WVD and PWVD of the composite signal, $c(t)$, which, as illustrated earlier, comprises multiple time-varying components. The MATLAB function `wvd` is used to compute WVD and PWVD and visualize the outputs as 2D heatmaps. This built-in function provides straightforward computations and visualizations of various WVD types. The WVD and PWVD results are displayed in Fig.26 and Fig.27, respectively. The comparison reveals that the WVD representation is impaired by false indications of non-existent signal components due to the cross-correlation between the actual signal components. In contrast, the PWVD, through its smoothing operation, eliminated the interference of cross-terms and accurately resolved all the time-varying frequency components. However, it can be seen that the improvement achieved with the PWVD led to a reduction in frequency resolution, manifested as wider lines in the PWVD's heatmap compared to the WVD's heatmap, thereby affecting the precise estimation of the corresponding frequency values.

³In signal theory, a linear transform satisfies the superposition or linearity condition, which states that if the input signal $x(t)$ is a linear combination of some signal components, then, the transform of $x(t)$ is also a linear combination of the transforms of each signal component.

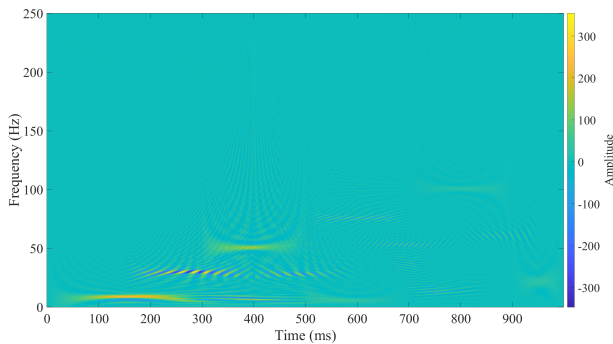


Fig. 26. WVD of the composite sinusoidal signal $c(t)$.

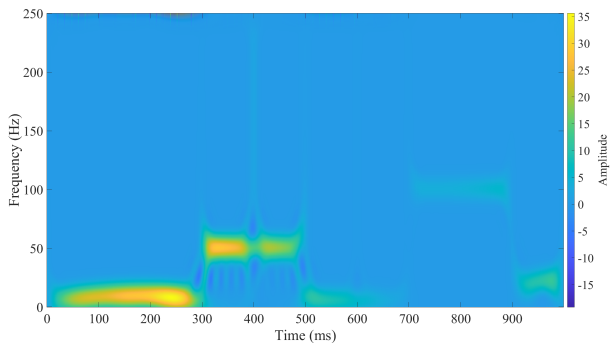


Fig. 27. PWVD of the composite sinusoidal signal $c(t)$.

H. Conclusive Comparison

Aimed at providing interested readers a thorough understanding of signal processing fundamentals, this tutorial offered a comprehensive introduction to signal characteristics with an in-depth discussion of prevalent signal transformation and analysis tools. The discussion focused on elucidating the main concepts, mathematical foundations, essential characteristics, advantages, and limitations. Additionally, the tutorial addressed implementation considerations by underscoring the use of programming libraries and built-in functions to facilitate efficient implementations of these tools. Furthermore, the codes employed for generating the illustrative plots throughout this section have been made publicly accessible as previously mentioned.

Throughout the section, it became evident that selecting the proper tool depends entirely on signal nature, application-specific requirements, and available computational resources. For stationary signals, where the primary interest lies in frequency analysis, The FT efficiently obtains the frequency spectrum, thereby enabling the identification of predominant frequency components in the signal. The PSD is particularly advantageous in scenarios involving comparative spectral analysis of multiple signals varying in length and/or bandwidth. It provides a normalised spectral density measure that facilitates effective comparison. The HT is particularly effective in applications where the signal envelope is of primary importance. Moreover, instantaneous phase and frequency information can be efficiently obtained from the analytical signal of the HT.

For signals with time-varying spectral characteristics, the STFT provides a computationally efficient tool for the time-frequency representation of the signal. However, its fixed window size results in a uniform resolution across all frequencies, reducing its effectiveness for applications requiring variable-resolution analysis. In such contexts, Wavelet analysis presents a viable solution due to its ability to perform multi-resolution analysis through scaled and shifted versions of a wavelet base function.

For the analysis of nonstationary and nonlinear signals characterized by rapidly changing components, the HHT provides an adaptive and more robust approach than CWT but with more computational requirements. The WVD offers a powerful method for providing a joint energy-time-frequency representation of the signal, although it is more computationally intensive than the other tools. It is particularly useful in high-resolution analysis scenarios or where energy concentration is a focal requirement.

Signal decomposition techniques allow the analysis of the elementary components of a signal. This makes it possible to extract unique manageable-sized features closely related to the signal's inherent structure. Regarding wavelet decomposition, its performance is highly dependent on the base wavelet function and the level of decomposition. In contrast, AMD methods use an adaptive mechanism rather than an a priori basis function to decompose the signal, thereby generating elementary modes that are not influenced by an a priori basis function. Moreover, AMD methods dynamically adapt to varying signal characteristics, making them effective for analyzing nonlinear and nonstationary signals. However, AMD methods are more computationally intensive than wavelet decomposition. Additionally, AMD methods are sensitive to noise and their efficiency highly depends on the employed stopping criterion and the used interpolation method for envelope estimation. Table II summarizes the comparison, highlighting the main aspects of each tool regarding signal nature, application requirements, and computational complexity.

IV. A TYPICAL SIGNAL-BASED ML PIPELINE

In signal-based ML applications, the pipeline of signal processing is typically divided into three core stages: preprocessing, processing, and application, as illustrated in Fig. 28. This section addresses each stage, highlighting its purpose and main aspects.

A. Preprocessing

The first stage encompasses the preprocessing of acquired signals. Here, it is essential to distinguish between data preprocessing techniques [69], commonly used in ML models, and signal preprocessing, which involves the tasks of signal smoothing, signal denoising, and signal segmentation. Signal preprocessing is performed on the raw signals, whereas data preprocessing techniques, such as normalization and scaling, are applied to the extracted features.

TABLE II
COMPARISON BETWEEN COMMON SIGNAL ANALYSIS AND TRANSFORMATION TOOLS

| Tool | Signal Nature | Application | Computational Complexity |
|---------------------------------------|-----------------------------|---|--------------------------|
| Fourier Transform (FT) | Stationary | Frequency analysis | Low |
| Power Spectral Density (PSD) | Stationary | Spectral analysis | low |
| Hilbert Transform (HT) | Stationary/nonstationary | Envelope and instantaneous phase/frequency analysis | low |
| Short-Time-Frequency Transform (STFT) | nonstationary | Fixed-resolution time-frequency analysis | moderate |
| Wavelet Transform (WT) | nonstationary | Multi-resolution time-frequency analysis | moderate |
| Hilbert-Huang Transform (HHT) | nonstationary and nonlinear | Adaptive time-frequency analysis | High |
| Wigner-Ville Distribution (WVD) | nonstationary | High-resolution energy-time-frequency analysis | High |
| Wavelet Decomposition (WD) | nonstationary | Signal decomposition | moderate |
| Adaptive Mode Decomposition (AMD) | nonstationary and nonlinear | Signal decomposition | High |

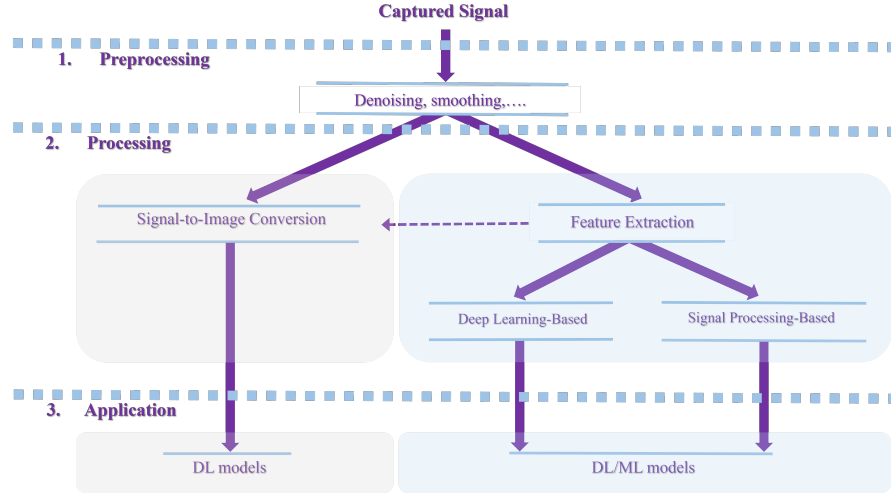


Fig. 28. Overview of a typical signal-based ML framework.

1) *Signal Smoothing*: The acquired signal may exhibit rapid amplitude changes between successive samples. Such random changes in amplitude can negatively impact application-level performance. Furthermore, during measurements, the signal is susceptible to outlier samples caused by external factors such as instrument malfunction [70]. Signal smoothing attempts to remove such impairments in the signal by adjusting the amplitudes of individual samples with respect to the amplitudes of adjacent samples. Smoothing acts as an approximation filter that works on N -successive samples at a time and outputs N smoothed samples. Common smoothing methods involve the well-known moving average (MA) filter and the Savitzky–Golay (SG) filter [71]. MA filter offers a simple method to smooth the signal by averaging neighboring data points while preserving the signal’s shape. It operates by averaging a set of signal data points (window) and sliding this window across the signal to smooth out short-term fluctuations. The SG filter uses a different approach to smoothing the signal by performing a local polynomial regression on the signal data points present in the moving window to provide more nuanced smoothing that preserves the original shape and characteristics of the signal. The design of an SG filter involves properly selecting the window size and the order of the polynomial [72]–[75]. Signal smoothing helps refine the signal’s waveform shape and captures its main patterns, particularly useful in many signals-based applications. However, as a lossy operation, smoothing could cause a

notable distortion in applications where peak-related features are of main interest. More information on signal smoothing can be found in [76]–[79].

2) *Signal Denoising*: Denoising refers to the process of removing noise from the acquired signal or, in other words, reconstructing the desired signal from its acquired noisy representation. This can be expressed mathematically as follows:

$$w(t) = s(t) + n(t) \quad (56)$$

where $w(t)$ denotes the noisy signal, $s(t)$ is the desired signal, and $n(t)$ is the noise component. Noise can be defined as any unwanted signal present in the measurement process other than the desired signal [80]. Signal denoising and signal smoothing are often treated synonymously in the literature [81], as both aim to refine the signal’s waveform while preserving meaningful patterns within the signal. However, they have different impacts on the noise and frequency contents of the signal. Smoothing is typically applied in the time domain on a sample-by-sample basis while denoising attempts to remove the whole noise component, $n(t)$.

Furthermore, in smoothing, the rapid fluctuations can be viewed as highly oscillating components of small amplitudes compared to the whole signal over a relatively long duration. From a frequency point of view, such oscillations represent high-frequency components of low amplitudes in the signal’s spectrum. Smoothing, therefore, acts as a low-pass filter that removes these high-frequency components from the signal. On

the other hand, signal denoising typically involves the use of various signal processing methods to develop more advanced noise filtering techniques. Signal denoising is addressed with more details in Section VI-A.

3) *Signal Segmentation*: Depending on the length of the acquired signal, expressed in terms of the number of samples, it might be necessary to divide the signal into smaller segments for the following practical considerations:

- Online deployment: Online applications require working on a pre-defined length of the input signal.
- Real-time and delay-sensitive applications: Given the sampling frequency, f_s in samples per second, the time duration, t_s , of the segment, expressed in *seconds* equals:

$$t_s = \frac{N_o}{f_s} \quad (57)$$

Hence, in real-time and delay-sensitive applications, the number of samples, N_o , in the input segment is of paramount importance since time delay increases as a function of N_o [44].

- Computation burden: The computational complexity of signal-based ML systems increases as a function of N_o [44].
- Effectiveness of extracted features: The number of samples, N_o , in the input segment significantly shapes the discriminative characteristics of the extracted features. Shorter segments may not provide an adequate representation of the signal's characteristics, reducing the reliability of extracted features. Conversely, very long segments could incorporate a high redundancy level, weakening the discriminative capability of the extracted features.

Constant segmentation is commonly used in various applications where the segment length, N_o , is kept constant. Consequently, signals are divided into smaller segments of equal lengths. Determining the value of N_o is application-specific and typically relies on achieving good trade-offs among the aforementioned practical considerations. Constant segmentation is commonly implemented using a sliding window of size N_o . Another aspect of constant segmentation is the percentage of overlap between successive segments. Introducing overlap between segments increases the number of segments extracted from the signal. The more overlap, the more segments are extracted. However, increasing the overlap increases correlation or dependence among adjunct segments. In general, the extent to which overlap can be used and its percentage depends on the nature of the application and the signal processing techniques used [82]–[84].

In some applications, the acquired signal exhibits random intervals of activity and inactivity, as observed in various sound and biomedical signals. In such scenarios, constant segmentation becomes unreliable, as it does not count for these inherent intervals. In such situations, adaptive segmentation [85]–[91] is commonly utilized to dynamically adjust the segment length, N_o . Adaptive segmentation exploits the underlying characteristics of the signal and adapts N_o accordingly. Unlike fixed length segments, adaptive length segments are more discriminative because the signal's characteristics and

inherent patterns change from segment to segment, allowing more discriminative features to be extracted. However, adaptive segmentation involves more computational burden compared to constant segmentation.

B. Processing

The processing stage involves processing the resultant segments to extract appropriate features that serve as inputs to ML models. The methods and algorithms available in the literature can be broadly categorized into two main approaches: feature extraction and signal-to-image conversion. In feature extraction, signal processing or deep learning (DL) can be used to extract relevant features from the input segment. The next section comprehensively covers the topic of feature extraction. Signal-to-image conversion approaches [92]–[101] transform 1-dimensional (1D) signal segments into 2-dimensional (2D) spatial representations, allowing to leverage DL models in signal-based applications. Gramian Angular Field (GAF) and Markov Transition Field (MTF) [92] are two well-known techniques that are commonly used to transform time series into images. Advanced techniques involve the use of signal processing to generate 2D time-frequency representations of the signal segments. The essence of these techniques is to represent time-frequency-energy characteristics of the signal and features as width-height-color intensity, mimicking image features. However, to serve training and testing purposes of DL models, a large number of transformed images should be available. Nevertheless, signal-to-image conversion is especially feasible in integrated systems such as unmanned aerial vehicles (UAVs) and autonomous vehicles (AVs) [102]. In these systems, computer vision applications, such as object detection and navigation, and signal-based applications are deployed within the same environment. Therefore, it is deemed computationally efficient to convert signals into images and use transfer learning approaches that utilize the trained deep-learning infrastructure for inference tasks of signal-based applications. Moreover, signal-to-image conversion and transfer learning offer a practical solution when the size and type of available signal data are insufficient to fulfill training and testing requirements.

C. Application

The application is the final stage of a typical signal-based ML pipeline. As a task-specific stage, it uses appropriate ML models [103]–[105] to address tasks, including classification, clustering, and anomaly detection. Feature preprocessing, hyperparameter tuning, and relevant metrics [106], [107] are often used to improve and evaluate model performance. Table III serves as a high-level overview of a typical signal-based ML framework, highlighting each stage's purpose, inputs, functions, and outputs.

V. FEATURE EXTRACTION METHODS

This section provides a comprehensive review of existing feature extraction methods. The review utilizes a new hierarchical taxonomy, depicted in Fig. 29, to categorize and

TABLE III
INPUTS, FUNCTIONS, AND OUTPUTS OF EACH STAGE OF TYPICAL SIGNAL-BASED ML FRAMEWORK

| Stage | Inputs | Functions | Outputs |
|----------------------|--|---|------------------|
| <i>Preprocessing</i> | Acquired signal+impairments (noise / outliers) | Smoothing / Denoising / Segmentation | Segmented signal |
| <i>Processing</i> | Segmented signal | Feature extraction / Signal-to-image conversion | Features/ Images |
| <i>Application</i> | Features/ Images | Classification / Clustering / Anomaly detection | Results |

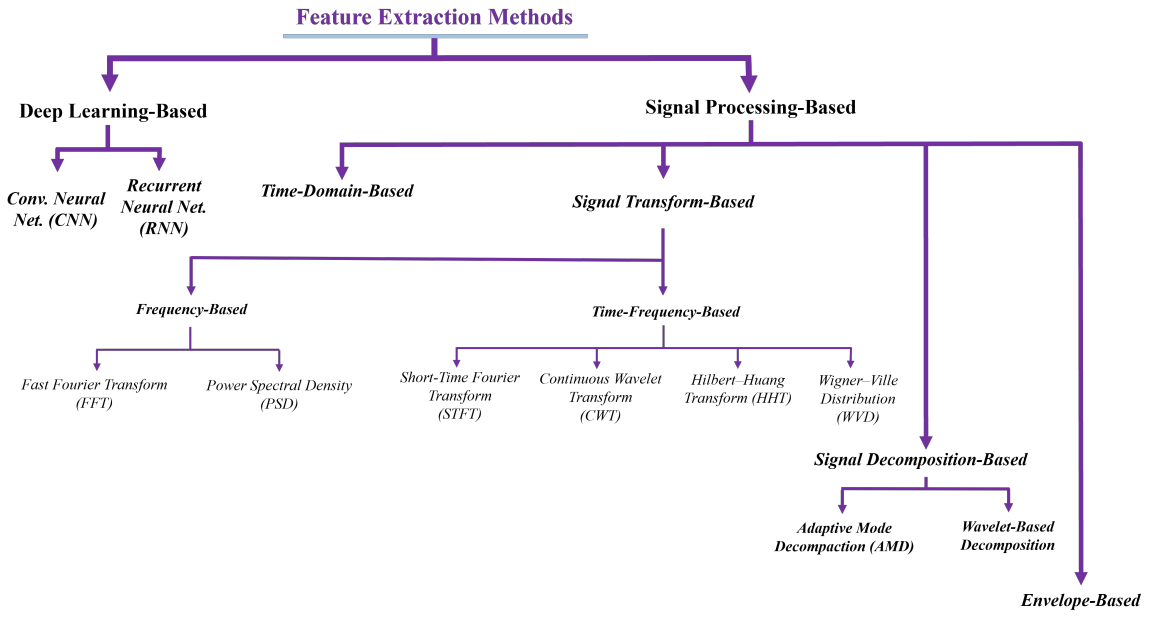


Fig. 29. Taxonomy of feature extraction techniques.

structure various methods that exist in the literature. The taxonomy starts by grouping the existing methods under two main categories: DL-based and signal processing-based. This categorization distinguishes between two mechanisms that exist for feature extraction: feature *learning* and feature *engineering*. The first mechanism relies on DL models—such as convolutions neural networks (CNN), recurrent neural networks (RNN), and auto-encoders—to *learn* high-level feature representations from signal data [108]–[117]. The second mechanism, on the other hand, employs signal processing to extract distinctive features from the signal, deliberately *engineered* to emphasize specific attributes and reveal certain characteristics. This deliberate engineering allows meaningful connections to be made between the resulting features and various conditions or classes that exist within the signals. In contrast, DL features are not interpretable since DL models are black-box models, making establishing such meaningful connections impossible. In addition, features extracted by DL models are model-dependent, whereas features based on signal processing are solely signal-dependent. This implies that variations in the model’s architecture, parameters, training, or tuning lead to different feature representations and, consequently, inconsistent performance results. Thus, it can be deduced that in DL-based approaches, performance is model-dependent, while in signal processing-based approaches, performance is mainly feature-dependent. Moreover, DL features would exhibit higher dimensionality and redundancy than signal processing-based features. This requires the use of appropri-

ate dimensionality reduction and feature selection techniques [111], [118], [119], thereby increasing the online processing time and computational burden. Table IV summarizes the aforementioned comparison between signal processing-based and DL-based approaches. A considerable amount of work in the literature uses hybrid approaches where signal processing-based features are fed into DL models to further extract highly distinctive representations from these features [120]–[133]. Compared to DL-based and signal processing-based features, hybrid features have higher discriminative power, which generally leads to better performance. However, employing hybrid approaches involves increased online processing time and higher computational complexity, making practical deployment subject to trade-offs between system complexity and performance requirements. In general, hybrid approaches prove feasible in applications that involve complex signals, such as radio modulation recognition [128]–[133]. In these applications, signals are randomly modulated, corrupted with high noise levels, and subject to various types of interference.

Focusing on signal processing-based feature extraction, the taxonomy of Fig. 29, from a computational point-of-view, divides the existing methods into four main categories: time domain-based methods, transform-based methods, decomposition-based methods, and envelope-based methods. Time domain-based methods extract features directly from time domain waveforms of the signal without further processing. On the other hand, transform-based, decomposition-based, and envelope-based methods use additional steps to

TABLE IV
A COMPARISON BETWEEN DEEP LEARNING-BASED AND SIGNAL PROCESSING-BASED FEATURE EXTRACTION

| <i>Aspect</i> | Deep Learning Approaches | Signal Processing Approaches |
|---|---------------------------------|-------------------------------------|
| <i>Feature Extraction Mechanism</i> | Feature learning | Feature engineering |
| <i>Interpretability of Extracted Features</i> | Not interpretable | Interpretable |
| <i>Size of Required Dataset</i> | Large | Small-to-medium |
| <i>Data Dimensionality and Redundancy in Extracted Features</i> | High, depends on DL model | Low-to-medium |
| <i>Computational Cost</i> | High | Moderate |

process the signal and, consequently, additional computations to extract features. The remainder of this section reviews common methods that exist within each category.

A. Time Domain-Based Methods

In time-domain-based methods, features are calculated from the signal's amplitude, representing specific aspects of the signal's dynamics over its time period. This allows for the quantification of amplitude variations from one period to another. Conventional methods [134]–[149] involve extricating shape characteristics—such as maximum, peak-to-peak value, and crest factor—and statistical properties of the input signal. Statistical features describe characteristics of the probability distribution of the input signal; common statistical features involve mean, root-mean-square (RMS) value, variance, and high-order statistics (HOS), such as skewness and kurtosis. Compared to first and second-order statistics (mean and variance), HOS features have more distinguishable characteristics as they can identify more complex aspects of the input signal, such as the shape of its distribution. Furthermore, HOS are less sensitive to noise. Besides shape-based and statistical methods, time-domain entropy methods [41], [150]–[152] are commonly utilized for feature extraction. Entropy methods rely on the observation that signals belonging to different classes or conditions typically exhibit different levels of irregularity or dynamic changes depending on the underlying state. Hence, different entropy measures can be utilized to quantify the irregularity of the input signal. In contrast to statistical features, entropy-based features are more sensitive to changes and complexities that arise within the signal. Thus, they can capture variations and irregularities in the signal that may be difficult to detect using traditional statistical measures. Additionally, entropy-based features are resilient to noise and outliers. In general, time-domain feature extraction approaches work directly on input signals, making them conceptually straightforward and relatively easy to implement. Further, they are computationally efficient since no further processing steps are required, making them advantageous in applications where real-time processing is crucial, especially with limited computational resources. However, time-domain analysis is highly susceptible to noise since noise in the signal would alter its amplitude and mask the dynamic characteristics of the signal. Furthermore, to achieve reliable performance, input segments of relatively long duration are usually required to capture the changes and complexities that evolve within the signal over time.

B. Transform-Based Methods

Transform-based approaches use advanced techniques to convert signals from the time domain to the frequency domain or time-frequency domain. This allows for the identification of frequency and frequency-temporal information, facilitating robust signal analysis and feature extraction.

1) *Frequency-Based Methods:* Frequency-based approaches represent signals in terms of their frequency content, revealing details that are not apparent in the time-domain waveform. Further, in contrast to time-domain analysis, frequency-based analysis allows the isolation and removal of noise or unwanted components by applying appropriate frequency filtering mechanisms to improve the signal-to-noise ratio and improve reliability of the extracted features. Frequency-based approaches involve the application of the Fourier transform (FT) to the input signal to compute its frequency content, which can be broadly categorized into two main categories: spectrum-based [12], [135], [153]–[166] and spectral density-based [163], [167]–[171]. Spectrum-based approaches use the frequency spectrum of the signal for feature extraction. Specifically, Some approaches directly use resultant Fourier coefficients of the input segment as input features [153]–[157] since these coefficients describe the distribution of the signal's energy over the range of frequencies contained in the signal. Here, dimensionality reduction and feature selection techniques are often used to reduce the size of coefficients and select the most discriminative coefficients.

Other approaches involve the extraction of various frequency-domain features from the frequency spectrum, such as statistical properties, energy, entropy, and correlation coefficients [12], [135], [158]–[166]. Spectral density-based approaches, on the other hand, utilize the power spectral density (PSD) of the input signal to extract the features [163], [167]–[171] since it provides a normalized measure of the power per unit frequency compared to the frequency spectrum. Entropy is frequently utilized in PSD analysis to calculate Spectral Entropy (SE) [168], [169]. SE is a very useful spectral feature that measures the irregularity or randomness of power distribution across signal frequencies and, hence, can be used to quantify the spectral complexity of the input signal. Other methods utilize Mel-frequency cepstral coefficients (MFCCs) as input features [172]–[181] because they effectively capture the shape of the power spectrum. MFCCs are originally developed for audio signals [182] and are gaining attraction in other various applications as well. A thorough explanation of MFCCs, including computational steps and usage scenarios, can be found in [182]. Higher-order spectra [183]–[186] are

commonly used in frequency-domain analysis to conduct a more comprehensive spectral analysis compared to second-order spectral analysis. Common higher-order spectra include bispectrum and trispectrum [184], [187]–[196], which correspond to the FT of the signal’s third-order cumulant and the fourth-order cumulant, respectively.

A key advantage of higher-order spectra over power spectrum is their ability to retain phase information [197]. Specifically, the power spectrum, representing the second-order spectrum or the FT of the autocorrelation function, captures the magnitude of frequencies present in a signal but discards phase information. This limitation makes it challenging to identify phase-related properties of the signal, such as frequency coupling or interactions between different frequency components. In contrast, higher-order spectra provide insight into the phase relationships among frequency components by examining moments or cumulants of the signal beyond the second order, enabling the detection of nonlinearities, phase coupling, and other characteristics that are invisible in the power spectrum. However, higher order spectra are more complex to calculate and interpret than traditional power spectral analysis because they require more data for reliable estimation and involve more sophisticated mathematical and computational methods. Furthermore, due to their multi-dimensional nature, visualisation and interpretation of higher order spectra can be challenging. Because of this complexity, specific cross sections or slices of the bispectrum or trispectrum are often analysed to extract meaningful information. This approach simplifies the analysis by reducing the dimensionality of the data, making it easier to visualise and interpret.

While frequency analysis accurately distinguishes between different frequency components in the signal, the analysis spans the signal’s entire duration, lacking the ability to provide temporal information about the timing of occurrence of these frequencies within the signal. Additionally, the FT is not ideal for analyzing signals that contain highly time-localized components, such as short bursts with high energy concentrations, because the components produce a wide range of frequencies in the frequency spectrum due to the inherent uncertainty principle associated with Fourier analysis [42].

2) *Time-Frequency-Based Methods*: In contrast to frequency-domain analysis, which lacks time resolution, time-frequency transforms produce representations that map the signal’s energy across both time and frequency, allowing for a localized analysis in the time-frequency domain. This approach allows the identification of transient oscillatory components within the signal, providing a more complete understanding of its dynamics. Time-frequency transforms that are commonly used in the literature for feature extraction include short-time Fourier transform (STFT), [198]–[210] Hilbert-Huang transform (HHT) [211]–[221], Wigner-Ville distribution (WVD) [222]–[230], and continuous wavelet transform (CWT) [202], [228], [231]–[241]. STFT is efficient in analyzing nonstationary signals whose spectral properties vary over time. However, it uses a fixed segment length in analyzing the signal, imposing a trade-off between time-domain and frequency-domain resolutions. In CWT, on the other hand, time-frequency analysis is based on a family of

wavelets generated from a single mother wavelet through scaling and translation, offering a multi-resolution analysis of the signal at different frequencies (scales) and time intervals (translations). Moreover, in contrast to STFT, which uses complex exponentials as basis functions— that extend infinitely in the time domain—, wavelet base functions are localized in time and frequency, making them more suitable for signals with transient or highly localized components. The HHT uses an adaptive approach for time-frequency analysis. Unlike STF and CWT, HHT does not impose a fixed basis function on the signal analysis, but instead uses an adaptive approach to analyze the signal.. This adaptivity makes it highly responsive to variations in the signal, facilitating very effective analysis of nonlinear and nonstationary signals. However, as an adaptive analysis method, its performance depends heavily on two factors: the stopping criterion for the sifting process [59] and the choice of the interpolation method for envelope estimation [60].

A proper stopping criterion ensures that the decomposition process neither overfits nor underfits the signal, thereby preserving the signal’s essential characteristics without introducing artifacts. On the other hand, the interpolation method for envelope estimation determines how well the upper and lower envelopes capture the actual oscillatory modes of the signal, heavily influencing the HHT’s performance. The WVD offers a time-frequency representation with high energy concentration, facilitating detailed analysis of a signal’s energy distribution over both time and frequency domains. The high concentration of energy arises from the quadratic nonlinear structure of WVD compared to STFT and CWT. This energetic and correlative nature of the WVD [63] makes it very useful in analyzing single-component signals. However, the quadratic nature of the WVD represents a major challenge when analyzing multi-component signals due to high cross-term interference [64] caused by undesired cross-correlation between various signal components, leading to a false indication of non-existent signal components in the resultant time-frequency representations of the signal. Window-based WVD [65], [66] and kernel-based approaches [67], [68] are commonly used to reduce the cross-terms at the cost of some loss in resolution.

From a feature extraction perspective, time-frequency-based methods can be grouped under three main approaches in which the generated energy-time-frequency mappings are treated differently:

- 1) Transformation coefficients as input features: In these approaches, the generated transform energy coefficients, representing signals’ energy distribution across time-frequency instants, are used directly as input features for the ML model [198], [199], [211], [216], [219], [220], [231], [240]. Feature selection techniques are commonly used here to select the most distinctive representations, which helps reduce redundancy and relaxes computational requirements for training and inference.
- 2) Energy-time-frequency 2D heatmaps as imagery data: The generated energy-time-frequency representations are used to visualize 2D heatmaps that map the signal’s energy across time-frequency instants. Accordingly, computer vision and DL techniques are leveraged where

these heatmaps serve as input images [200]–[208], [212], [217], [222], [223], [227], [228], [230], [232], [233], [238], [239], [241].

- 3) Transformation coefficients as signal representations: In these approaches, the generated mappings are treated as transformed representations of the signal. Subsequently, these representations are subjected to further signal-processing techniques to extract features of interest, such as entropy, energy, and statistical properties [209], [210], [213]–[215], [218], [221], [224]–[226], [229], [234]–[237]. Compared to the other two approaches, these methods generally result in a smaller number of features, making them more suitable for applications with low computational requirements.

Spectral kurtosis (SK) [242], [243] is a popular time-frequency analysis tool in signal processing that is commonly used to locate harmonics, transients, and repetitive impulses in the frequency domain. SK is based on computing the normalized fourth-order moment (kurtosis) of the STFT of the signal at each frequency bin, providing a statistical measure of the signal's characteristics at each time-frequency instant. Essentially, SK quantifies the "peakedness" or impulsiveness of the distribution of the power at each frequency bin. One of its primary applications is the detection of transient components in signals. Transients, due to their short and sharp nature, create distinct, non-Gaussian patterns in the signal's frequency content. When these features are present, the kurtosis values in the affected frequency bins will be significantly higher than those of normal behavior. Accordingly, high SK values can indicate the presence of impulses or bursts, as they cause a deviation from the Gaussian distribution typically observed in stable operating conditions. This is particularly useful in condition monitoring and fault diagnosis of rotating machinery and mechanical systems [244]–[246], where transient vibrations often indicate faults. Additionally, SK is an effective tool for distinguishing between harmonic components—which often manifest as consistent, periodic components—and noise in the signal. These harmonics will often show higher kurtosis values, indicating a non-Gaussian distribution, possibly due to the periodic and repetitive nature of the harmonics. In contrast, Gaussian noise tends to have a kurtosis value close to 0, indicating a Gaussian distribution.

C. Signal Decomposition-Based Methods

Signals generated by physical systems typically consist of multiple components or modes of time-varying nature in amplitude, phase, and frequency. These components contain meaningful information about the underlying structure of the signal. Decomposition techniques attempt to decompose the signal into elementary modes, enabling signal analysis at the level of its constituent components. This, in turn, allows for the extraction of highly distinctive features, that are closely related to the inherent structure of the signal. Common decomposition approaches include wavelet-based decomposition and adaptive mode decomposition (ADM) methods. In wavelet-based decomposition methods [44], [240], [247]–[267], discrete wavelet transform (DWT), stationary wavelet

transform (SWT), and wavelet packet transform (WPT) are commonly used to decompose the signal into elementary modes of high and low-frequency components using digital filter banks. The performance of wavelet decomposition is highly dependent on the base wavelet function and the level of decomposition. AMD methods [249], [254], [268]–[283] utilize an empirical decomposition approach where the signal is adaptively decomposed into a set of simpler functions known as intrinsic mode functions (IMFs). Hence, in contrast to wavelet-based methods, AMD methods use an adaptive mechanism rather than a priori basis function to decompose the signal, leading to more "natural" modes that are not influenced by a priori basis function [47]. Moreover, the ability of AMD methods to adapt to varying signal characteristics makes them effective when dealing with nonlinear and non-stationary signals. However, this adaptability comes at the cost of higher computational complexity. Furthermore, AMD methods are generally sensitive to noise because they rely on the extrema and minima of the signal for signal decomposition. Additionally, they suffer from inherent limitations, such as end effect [48] and mode mixing [49]. Various methods have been introduced to address these limitations, such as ensemble empirical mode decomposition (EEMD) [50] and variational mode decomposition (VMD) [51], local mean decomposition [52], and empirical wavelet transform (EWT) [53].

In the existing literature, signal decomposition is utilized through two approaches for feature extraction; each approach serves a distinct objective:

- 1) Extraction of highly discriminative features: In these methods, features are extracted from elementary modes resulting from the decomposition process. Accordingly, feature extraction methods can be generally grouped under four main categories: entropy-based [262], [263], [267], [273], [274], [283], energy-based [251], [257], [264], [265], spectral-based [44], [252], [266], [272], [275], [280], [282], [283], and statistical-based [247], [249]–[251], [253], [254], [260], [261], [275], [283]. Entropy as a measure of uncertainty, is used to quantify the irregularity within the decomposed modes. Energy-based approaches utilize the changes in energy content of the decomposed modes to extract the features. Spectral-based approaches use spectral characteristics of the elementary modes to extract the features. Statistical-based approaches rely on time-domain properties of the modes such as skewness, kurtosis, Root-Mean-Square (RMS), and crest factor for feature extraction.
- 2) Obtaining a low-redundancy (highly informative) version of the composite signal [240], [248], [255], [256], [258], [259], [268]–[271], [276]–[279], [281]: This is accomplished by decomposing the signal into elementary modes and applying further processing to examine these modes based on predefined ranking criteria. The highest-ranked mode is thus selected and various feature extraction techniques can be applied to the selected mode accordingly. Here, the selected mode represents a low-redundancy and high-informativeness version of the original signal.

An important aspect of the first approach is that it provides a convenient way to construct a feature vector of controllable size and highly discriminative nature [44], which is particularly helpful in applications with limited computational resources. This is facilitated by decomposing the signal into a finite number of modes and extracting a few features from each mode. For instance, decomposing the input segment of a signal into 8 modes and taking the energy or entropy of each mode as a feature leads to a feature vector with a size of 1×8 distinctive features. It is also less computationally intensive than the second approach, making it more suitable for real-time processing and applications with limited processing capabilities. However, by focusing on the highest-ranked mode, the second method ensures that the extracted features are highly informative and less redundant. Moreover, the capability to rank and select modes based on predefined criteria allows for dynamic screening that can be automatically adapted according to the nature of the input signal, which becomes particularly useful in applications that involve multiple signals from various sources, such as advanced diagnostic systems and integrated condition monitoring applications. Additionally, selecting the most informative mode can result in more accurate and relevant feature extraction, potentially improving the performance.

D. Envelope-Based Methods

Envelope-based feature extraction methods [284]–[295] primarily focus on analyzing the envelope of the signal to reveal essential characteristics of its amplitude, such as statistical properties, peak frequency, energy, and entropy. In these methods, the HT is commonly used to obtain the signal's envelope, which is then subjected to various feature extraction techniques. Envelope-based feature extraction is particularly useful in applications where amplitude modulation (AM) is a key characteristic of the signal; that's it, the amplitude of the signal is varied or altered in response to specific factors that are of interest, for instance, impacts generated by defects in rolling bearings [296]; in such situations, envelope-based analysis can effectively extract meaningful features. Generally speaking, as an amplitude-focused approach, envelope analysis offers limited frequency information about the signal. Furthermore, its effectiveness can vary significantly depending on the nature of the signal, as not all signal types are amenable to envelope-based analysis. Additionally, it is highly susceptible to noise that contaminates the signal's amplitude such as impulsive noise, making it difficult to extract effective features from the envelope [295].

The selection of an appropriate feature extraction method for a given signal depends on several factors, including the nature of the signal, the specific requirements of the application, and the available computational resources. For stationary signals, where the primary interest lies in frequency analysis, Fourier analysis provides a highly efficient approach to identifying dominant frequency components within the signal. In scenarios involving comparative spectral analysis of multiple signals that vary in length and/or bandwidth, PSD is particularly advantageous since it provides a normalized

spectral density measure, facilitating effective comparison. On the other hand, signals exhibiting time-varying spectral characteristics necessitate the use of proper time-frequency analysis for effective feature extraction. The STFT offers a computationally efficient tool for time-frequency analysis of the signal. However, its fixed resolution across all frequencies reduces its effectiveness for applications requiring variable-resolution analysis. In such contexts, wavelet analysis presents a practical solution due to its ability to perform multi-resolution analysis through scaled and shifted versions of a wavelet base function. For analyzing nonstationary and non-linear signals characterized by rapidly changing components, the HHT provides an adaptive and more robust approach than Wavelet analysis but with more computational requirements. The WVD provides a practical approach for achieving a joint energy-time-frequency representation of the signal, although it is computationally intensive compared to the other transforms. It is particularly useful in high-resolution analysis scenarios or where energy concentration is a focal requirement. Signal decomposition techniques enable signal analysis at the level of its constituent components. This, in turn, allows for the extraction of highly distinctive features of a controllable size that are closely associated with the inherent structure of the signal. The envelope-based analysis is especially effective in applications where the envelope characteristics are of primary interest.

VI. CHALLENGES AND POTENTIAL SOLUTIONS

The practical deployment of signal-based ML systems faces significant challenges that can impede the efficiency and effectiveness of the system. Key among these challenges are the detrimental effects of noisy environments on signal integrity, the limited availability of abnormal samples, and computational complexity. These challenges require effective solutions to mitigate their effects and enhance signal processing techniques. This section delves into each of these challenges, exploring their implications and highlighting potential solutions.

A. Effects of Noise

In real-world systems, signals are nonstationary, nonlinear in nature, and subject to noise [297]. From a measurement perspective, noise can be defined as any unwanted signal present in the measurement process other than the desired signal [298]. In signal processing, noise generally refers to any unwanted or random variation in the signal that can obscure or distort the intended information. It is an inevitable part of signal processing, particularly in machine learning applications where it can hinder feature extraction and significantly reduce its discriminative power.

1) *Noise characteristics:* Noise, originating from various sources that are typically present in all measurement environments can be grouped into three main categories [298]:

- 1) Intrinsic noise that arises from random fluctuations within measurement physical systems such as thermal and shot noise.

- 2) Man-made noise sources such as motors, power lines, radio broadcasts, oscillators, laboratory equipment, and cellular phones.
- 3) Noise caused by natural disturbances such as lightning and sunspots.

Understanding the characteristics of noise and how it affects the desired signal is the first step toward an effective noise removal approach. Noise is typically modeled and characterized by various aspects such as spectrum, mode of occurrence, and statistical distribution.

Noise Spectrum (Noise Colors): Color of noise is a spectral-based description of noise where different “colors” of noise are classified according to their power spectral densities. The most important and common type in the context of signal processing is white noise. A defining characteristic of this type of noise is its flat power spectral density (PSD). Thus, it has a constant power at each frequency, affecting all frequencies in the desired signal equally. Pink noise, or $\frac{1}{f}$ noise, has a PSD that decreases proportionally to the inverse of its frequency, making it more dominant at lower frequencies in the desired signal. In contrast, blue noise mainly impacts high-frequency components in the signal since its PSD increases proportionally to the frequency. Purple noise has more energy at higher frequencies compared to blue noise since its PSD increases with frequency at a much faster rate than that of blue noise.

Occurrence Nature: The presence of noise in the signal depends on the nature of the occurrence of noise. Continuous noise is present continuously during signal acquisition, persistently contaminating the signal over its entire duration. The severity of continuous noise is characterized by its amplitude (power); it is related to the signal power by the signal-to-noise ratio (SNR), which provides a measure to evaluate the severity of the noise and the performance of the signal denoising technique employed. Intermittent noise, in contrast, occurs at irregular intervals. Burst noise, on the other hand, is characterized by sudden occurrences, often of short durations. Impulse noise consists of random noise pulses. Compared to burst noise, impulse noise has extremely short durations. Burst and impulse noises are characterized by their large amplitudes compared to the desired signal.

Statistical Distribution: Noise can be described in terms of amplitude distribution. Gaussian noise, commonly called additive white Gaussian noise (AWGN), is a common noise model in signal processing. Characterized by its Gaussian distribution, the amplitude of AWGN can be completely described by its mean and variance, which represent noise power. AWGN is commonly modeled as a zero-mean Gaussian process, maintaining the mean of the desired signal unchanged. As a white continuous noise, AWGN affects the spectral contents of the signal equally.

2) *Signal Denoising Techniques:* As mentioned earlier, the process of signal denoising involves eliminating noise while preserving the important features of the signal. The first step in signal processing-ML pipelines is to remove noise and outliers. Thus, the effectiveness of subsequent steps depends heavily on the efficiency of the employed signal denoising technique. For impulse and burst noise, Order-Statistic Filtering (OSF) [299] is particularly effective; it is based on replacing each sample

in the signal with a statistic (like the median, minimum, or maximum) calculated from the samples in its neighborhood, defined by a sliding window of a certain size that moves across the signal. The median filter [300]–[303], a specific type of OSF, is widely used to remove impulsive noise. OSF is highly robust to outliers such as impulse noise since order statistics are less sensitive to extreme values than linear operations such as averaging. Moreover, the median filter is highly effective in preserving signal edges (sudden changes in signal amplitude) while reducing noise, which is crucial for maintaining the integrity of the signal’s features. The efficiency of the median filter is significantly influenced by the selection of the window size. A larger window size will include more samples, improving the ability to remove impulsive noise. However, a larger window could over-smooth the signal, potentially erasing important details. Further, it will increase the computational burden. Conversely, a smaller window size preserves more detail since fewer samples are involved in determining the median, but it may be less effective in removing noise as the median value will be influenced by the presence of noise in the window. Additionally, the performance of median filters is limited by the proportion of impulsive noise in the window. Filtering performance decreases as the proportion of impulsive noise increases [304]. Specifically, with low levels of impulsive noise, the noise spikes are likely to be the minority within the window, and the median value computed within the window is likely to be a true signal value, effectively removing the noise. As the proportion of impulsive noise increases, there’s a higher likelihood that the median value computed within the window will itself be a noise value, especially if the noise spikes constitute a considerable part of the samples in the window. In this case, replacing the original signal sample with the median does not remove the noise. Instead, it might even introduce additional distortion, especially if the median value is a noise spike. To overcome these problems, adaptive median filtering is commonly applied where the window size changes adaptively based on impulsive noise content [304]–[306].

For complex noise patterns such as intermittent and continuous noise, several techniques can be used to denoise the signal in the time domain, frequency domain, time-frequency domain, or deep learning (DL) latent space. Time-domain methods involve techniques that directly manipulate the signal in the time domain, aiming to identify and reduce noise components while preserving signal characteristics. Moving average (MA) filter [307], [308] is a simple yet effective method that smooths the signal by averaging neighboring data points. It is particularly useful for reducing simple random noise while preserving the signal’s shape. Traditional MA filters operate by averaging a set of signal data points (referred to as the window or kernel) and sliding this window across the signal. This process smooths out short-term fluctuations, effectively reducing random noise. However, the primary limitation of a traditional moving average filter is its fixed window size, which may not be optimal for all signal conditions. Adaptive moving average (AMA) filters [309]–[311] address this limitation by dynamically adjusting the window size based on the characteristics of the input signal. Here, the window

size changes in response to the variability or other statistical properties of the signal. This adaptivity is critical when dealing with nonstationary signals or when the noise characteristics are inconsistent throughout the signal. The design of the adaptive mechanism involves selecting proper criteria for window size adjustment. Common approaches include the use of local signal variance, rate of change, or other statistical measures that reflect the signal's nature.

The Savitzky-Golay filter (SGF) [312]–[314] While it shares some conceptual similarities with MA filters, it provides a different approach to noise reduction and is particularly useful for preserving higher-order signal properties. Specifically, MA filters give equal weight to all points in the window, which can distort the features of the signal. On the other hand, SGF performs a local polynomial regression on the signal data points present in the moving window to provide a more nuanced smoothing that preserves the original shape and characteristics of the signal. This approach not only smooths the data but also preserves features of the data points, such as relative maxima, minima, and width, which are often obscured by noise. The design of an SGF involves the proper selection of the window size, and the order of the polynomial [72]–[75]. The selection depends on the specific characteristics of the signal being processed and the nature of the noise. A higher polynomial order can better adapt to changes in the signal's trend, but it may also introduce noise if not chosen appropriately.

Adaptive filters leverage adaptive noise cancellation (ANC) techniques to instantly isolate and remove unwanted noise from the signal, making them suitable for real-time applications such as autonomous driving, navigation, control systems, and robotics. ANC involves dynamically adjusting the filter parameters to minimize the difference between the desired signal (primary signal of interest) and the actual signal; the primary goal is to improve the SNR without distorting the primary signal. Common ANC techniques involve the least mean square (LMS) and recursive least square (RLS) adaptive filters [315]–[318], and Kalman filtering (KF) [319]–[322]. LMS adaptive filtering, a gradient-based method, attempts to minimize the mean square error between the desired and actual output of the filter. To facilitate the generation of an estimate of the unwanted noise, this process entails using a reference signal, highly correlated with the noise contaminating the primary signal and uncorrelated with the primary signal itself. The RLS algorithm provides a recursive solution to the least-squares problem, offering faster convergence than LMS. In contrast to LMS and RLM, which require a prior reference signal, KF estimates the state of the system, represented by the signal of interest, based on noisy measurements. The estimation is based on Bayesian estimation and state-space representation of linear systems [323], making KF particularly useful in real-time applications involving time-varying signals or systems with inherent uncertainties. KF consists of two main steps: prediction and update. In the prediction phase, KF predicts the next state of the system. In the update phase, the prediction is adjusted using the new measurement data, refining the system's state estimate. In the context of noise reduction, this process inherently filters out the noise from

the measurements, isolating the primary signal. Standard KF assumes that noise is Gaussian and the system has linear dynamics. However, several extensions have been developed to handle more complex scenarios. Such as the extended Kalman filter (EKF) and the unscented Kalman filter (UKF) [324]. However, the design of KF requires a good understanding of system dynamics and noise characteristics.

Singular spectrum analysis (SSA), which is a well-known method to decompose time series into trends, seasonal components, and noise, has been explored within the signal processing field for signal denoising applications [325]–[331] due to its effectiveness in separating useful oscillatory components from noisy signals. SSA works by embedding a signal in multidimensional space, constructing a trajectory matrix by sliding a window over the signal data points. The trajectory matrix is then decomposed using Singular Value Decomposition (SVD), which separates the signal into a series of matrices associated with singular values. These matrices represent different signal components, including deterministic structures and stochastic noise. Accordingly, the decomposed components are classified into groups that represent the signal and noise, respectively. This involves evaluating singular values and the corresponding eigenvectors to differentiate between signal information and noise components. The denoised signal is then reconstructed by summing the respective groups of matrices that contain signal information. Subsequently, the reconstructed components are transformed back into the time domain. The effective implementation of SSA in the context of signal denoising involves two main aspects: Selecting an appropriate window length and accurately grouping singular values and eigenvectors corresponding to signal information and noise components. These aspects are pivotal as they directly influence the quality and reliability of the signal-denoising process. The choice of window length determines the structure of the trajectory matrix and subsequently affects the decomposition and reconstruction phases. An appropriately chosen length ensures that the resulting trajectory matrix captures the underlying structure of the signal, effectively separating signal components and noise. A window length that is extremely short may not capture the signal's dynamics adequately, while a window length that is too long may reduce denoising efficiency and increase computational complexity. However, selecting an optimal length is not straightforward; it is highly dependent on the specific characteristics of the signal, including periodicity and stationarity [326]. In practice, selecting an appropriate window length depends on prior knowledge of signals of interest and often ultimately relies on experimentation [327], [330] where empirical or heuristic methods are often employed to determine a suitable window length for a specific application. Additionally, the accurate grouping of decomposed components is fundamental for effective signal denoising as misclassification of these components can lead to either an inadequate removal of noise or the unintended discarding of useful signal information. There is no general rule for the grouping process; it is mainly determined by the specific requirements of the application, the statistical nature of the signal of interest, and associated noise [330].

Frequency-domain noise reduction is based on the signal's

frequency content, where noise is often more distinguishable due to its distinct spectral characteristics. Common frequency-domain techniques involve frequency-selective filtering and spectral subtraction (SS). Frequency-selective filters include low-pass [332]–[334], high-pass [334]–[336], band-pass [337], [338], and notch filters (band-stop) [339], [340]. These filters attenuate noise components that are outside the primary frequency range. Specifically, low-pass filters allow components with frequencies lower than their defined cutoff frequency to pass through while attenuating components with frequencies higher than the cutoff. Conversely, high-pass filters perform the opposite operation, blocking frequencies below the cutoff. Band-pass filters pass frequencies within a certain range and attenuate frequencies outside that range. Notch filters attenuate specified narrow frequency bands. Frequency-selective filtering techniques are useful in applications where noisy frequency characteristics are known, such as noise caused by power-line and equipment interference. SS [338], [341]–[347] relies on estimating the noise spectrum and subtracting it from the noisy signal's spectrum. In contrast to frequency-selective filtering, which eliminates specific frequencies in the signal spectrum, SS provides global spectrum denoising mechanisms. The main steps in SS involve estimating the power spectrum of noise and subtracting it from the power spectrum of the original signal to obtain a new power spectrum. Negative values are eliminated by setting them to zero. The new power spectrum is combined with the phase of the original signal to reconstruct the denoised signal's time waveform. The noise spectrum is commonly estimated and updated during signal-inactivity periods where only noise is present. SS generally has moderate computational complexity [348]. Hence, it offers practical solutions in cases where noise is generally characterized as a stationary or a slowly varying process independent of the desired signal. However, in practice, random noise variations can lead to an overestimation of noise's power spectrum, leading to many negative values in the subtracted spectrum. While these values can be eliminated by replacing them with zeros, this nonlinear rectification process can cause significant distortion in the recovered signal [348], necessitating the adaptation of enhancement mechanisms [338], [345] that come at the cost of increased computational complexity.

Time-frequency denoising methods are commonly applied for nonstationary signals since their varying spectral contents make traditional time-domain or frequency-domain methods less effective. The main concept of time-frequency denoising methods is centered around discriminating and eliminating noise-related coefficients within the time-frequency domain representations of the noisy signal. Time-frequency denoising involves transforming the signal into the time-frequency domain, representing it by a set of coefficients corresponding to each time-frequency instant. Since each coefficient reflects the signal's energy at that instant, signal denoising is achieved by screening these coefficients to distinguish between noise and meaningful signal components. Thresholding is a common technique employed in this stage; it involves setting a threshold value, below which coefficients are considered to represent noise and are, therefore, attenuated or set to zero. Coefficients above the threshold, which are assumed to represent the actual

signal, are either preserved or adjusted. The key idea here is that small-magnitude coefficients, representing low energy, are more likely to be associated with noise, while larger coefficients are likely to represent actual signal components. After thresholding, the time-domain signal is reconstructed by inversely transforming the modified coefficients to the time-domain. Wavelet-based thresholding [45], [349]–[356] is commonly used for signal denoising, mainly attributed to its unique characteristics of multi-resolution analysis and the availability of a large set of basic functions. Multi-resolution analysis can capture global trends and fine details, making it useful in denoising. The diverse family of wavelet base functions allows the selection of a wavelet that matches signal characteristics. In thresholding-based denoising, the noise level is determined by the threshold value. Hence, excessively large thresholds could filter out some of the useful signal representations; on the other hand, too small thresholds could cause some noise to be retained in the signal. There are several methods to guide the selection of an appropriate threshold, such as universal threshold, minimax threshold, and SURE threshold [356]. Common thresholding functions include hard thresholding and soft thresholding [351]. In hard thresholding, coefficients with absolute values less than the threshold are set to zero, while others remain unchanged. This can sometimes lead to discontinuities in the reconstructed signal. Soft thresholding, on the other hand, shrinks coefficients towards zero by the threshold value, providing a smoother result but potentially introducing bias. The choice of thresholding function and threshold values is influenced by the characteristics of the signal and the specific requirements of the application [350], [356]–[360]. In practice, it is an iterative process where thresholding and signal reconstruction are repeated, and the thresholding is adjusted at each iteration based on denoising performance.

The use of DL for signal-denoising applications is an active research field [361]–[378]; the existing approaches involve a wide range of models and methods, making them too numerous to mention. Nevertheless, the main theme is to use noise-free signals as training targets and corresponding noisy versions as training inputs. From a signal-processing perspective, such an approach can be centered around the concept of tunable denoising filters. Where a trained DL denoising model can be viewed as a tuned denoising filter whose coefficients, represented by model weights, are tuned through the training phase to remove noise from the signal. DL-based approaches are approved to be highly effective in noise removal due to their ability to learn complex, noisy patterns and related dependencies within the signal. Moreover, they can be designed to implement various operations on signal simultaneously, facilitating efficient end-to-end implementation of various signal-processing pipelines. This concept has been demonstrated in [378], where 1D convolutional autoencoders are utilized to implement a signal processing pipeline that smooths, compresses, denoises, and expands vibration signals. The pipeline is intended to facilitate power-efficient remote condition monitoring. The smoothing and compression operations were implemented simultaneously using a smoothing autoencoder with a μ -law compression-based activation function. The simultaneous denoising expansion

of the compressed noisy signal is realised by a denoising reconstruction autoencoder.

Despite these advantages, the effectiveness of DL-based denoising, in general, is limited to noise profiles that are present in the training set [379]. Furthermore, as they lack inherent capabilities to adapt to varying noise patterns, conventional DL denoising approaches are impractical for real-time applications that involve varying noise characteristics.

B. Availability of Labeled Abnormal Samples

In supervised ML models, labeled samples that are large in size and diverse—in terms of class representation—should be available to serve the training purpose. In real-world situations, the available labeled samples are usually limited in their size due to expensive and time-consuming labeling processes. Moreover, faulty or abnormal samples are not abundant since abnormal events, such as faults or anomalies, are infrequent, making it difficult to gather substantial abnormal signal data. As a result, real-world abnormal samples are usually insufficient to represent the corresponding classes during the training phase. There are several solutions in the literature to address this problem. DL-based solutions are becoming increasingly popular where transfer learning techniques are commonly adapted to avoid learning from limited labeled data [380]–[383]. These approaches overcome the problem by adapting pre-trained models to extract deep features from the limited available signal samples, eliminating the need for model training. Recent trends utilize few-shot learning (FSL) [384]–[387], enabling the pre-trained models to generalize over new categories of data using only a few labeled samples per class. While DL-based solutions show promising results, they are computationally intensive. Additionally, In contrast to computer vision, where mature pre-trained models can be developed using large-scale and well-labeled datasets [388], the development of pre-trained time-series models [389]–[391] as generic feature extractors is still an evolving area of research.

Non-ML solutions [267], [392]–[398] employ signal processing techniques for feature extraction and use similarity-based classification, providing a computationally efficient alternative to DL-based solutions. In these methods, features are extracted from available labeled normal samples, and a “normality” reference model is constructed from these features. Accordingly, signal classification is achieved by assessing the similarity between unknown samples and the reference model in the feature space. In tasks involving multiple classes, the process entails comparing unknown samples with reference samples representing different classes. The class of the unknown sample is then determined to be the class of the reference sample with the highest similarity score.

Other solutions utilize data augmentation techniques [399]–[401] to generate additional pseudo-samples. This is achieved by applying “label-invariant” transformations to available signal samples, which can considerably promote the final classification performance and generalization ability of the ML model [401]. However, compared to image augmentation [402], which relies mainly on spatial transformations, signal

augmentation involves unique challenges stemming from the complexity of the temporal and spectral characteristics of the signal. This complexity would reduce the augmentation reliability, consequently hindering the generalization of the employed model. To overcome this challenge, signals can be transformed into 2D time-frequency imagery representations, such as spectrogram and scalogram, where conventional image augmentation can be applied accordingly [403]–[405]. However, although 2D time-frequency representations and classical images possess similar data structures, they fundamentally differ regarding what information is conveyed along the x – y -color space. Thus, employed data augmentation techniques should correlate with a real-world, physical transformation of the signal [403]. While this is feasible for signals with well-defined characteristics, such as sound signals, it becomes impractical for complex signals.

Advanced approaches leverage the powerful synthesizing capabilities of generative adversarial networks (GANs) to generate new synthetic samples with reliable characteristics [406]–[411]. The advancement in computational capabilities enables the feasible implementation of these approaches despite their intensive computational requirements. Further, as an offline operation, synthesizing new samples does not impact online inferencing tasks.

C. Computational Complexity

Computational complexity is a major concern in the real-world deployment of signal-based ML systems. The complexity of these systems is twofold: the complexity associated with signal processing and the complexity inherent in the used ML model. From a signal processing perspective, the complexity is mainly attributed to three factors [44]: the size of the signal segment N_s , algorithm-based computations involved in preprocessing and feature extraction, and the size of the resultant feature vector. While the complexity stemming from algorithmic computations is a fundamental aspect of the chosen approach, several techniques can be employed to reduce the influence of segment length and feature size on the complexity.

1) *Segment Length*: The segment length, represented by the number of samples in the segment, is a significant factor in signal processing since it directly impacts the computational complexity, system delay, and reliability of extracted features. Therefore, selecting a proper segment length is generally a balancing act that necessitates a thorough consideration of the trade-offs involved. It is often guided by theoretical analysis, empirical evaluation, and domain-specific knowledge. Considering oversampled signals, which is the common case in many applications, decimation can be used to reduce the signal’s sampling rate, thereby decreasing the number of samples in the segment while maintaining the same time duration. The decimation process involves two steps: Applying low-pass filtering to the signal to prevent aliasing. The low-pass filter limits the signal’s bandwidth to satisfy the Nyquist criterion, ensuring that the highest frequency in the signal is less than half of the new reduced sampling rate. Accordingly, the second step involves down-sampling the filter signal by retaining

every M th sample and discarding the rest, where M is the decimation factor. For instance, if $M = 4$, the process would keep every fourth sample and discard the three in between. Decimation is widely utilized in digital signal processing (DSP) applications [412], [413]. However, the application of decimation requires a deep understanding of the frequencies of interest in the signal. To the best of our knowledge, the application of decimation in the context of feature extraction and, consequently, its effectiveness remain unexplored.

2) *Size of Extracted Features*: In contrast to the length of the signal segment, which influences the complexity of the signal processing pipeline, the dimensionality of the extracted features primarily affects the complexity of the employed ML model. There is a wealth of well-established methods in the ML literature that can effectively decrease feature size, thereby relaxing computational complexity and reducing feature redundancy. Common methods include dimensionality reduction [414], [414]–[421] and feature ranking and selection [8], [125], [414], [422]–[434]. In DL-based feature extraction methods, the substantial size of the resultant deep-learned features necessitates dimensionality reduction since these features constitute a multi-dimensional latent space. Moreover, since these features lack interpretability, feature ranking and selection become essential to identify the most relevant features. In the context of signal processing-based features, these techniques are commonly employed when many features—such as Fourier, wavelet, and energy-time-frequency coefficients—are used. Despite their effectiveness in reducing the computational burden and improving the efficiency of the ML model, dimensionality reduction and feature selection come at the cost of extra processing, increasing the system’s end-to-end complexity and time delay. An alternative solution is to utilize various signal processing tools to develop a small set of highly distinctive features. Although this typically demands domain knowledge and empirical evaluation, certain signal-processing tools can be particularly useful in this context. Specifically, Signal decomposition offers a reliable way to engineer a few highly-discriminative features from decomposed modes [44] that are closely associated with the inherent structure of the signal. The size of the features extracted from these modes equals $1 \times (K \times M)$ [44], where K is the number of decomposed modes and M is the number of extracted features for each mode. In [44], this approach was utilized for vibration-based condition monitoring of rolling bearings using wavelet-based decomposition. A comparative analysis shows that the approach would achieve excellent performance in fault detection with very few features compared to other approaches. Furthermore, the comparison shows that the decomposition-based method allows for shorter input segments, attributed to the high localization of the decomposed modes in both time and frequency.

VII. APPLICATIONS

This section delves into the practical application of signal-processing techniques for feature extraction in signal-based ML systems. By providing actual use cases, the section aims to bridge the theoretical foundations of signal processing with

its tangible use in real-world scenarios. To foster a collaborative research environment and support reproducibility, the Python codes related to these use cases are made publicly accessible on the Github site of the Optimized Computing and Communications (OC2) Laboratory⁴. The first part of the section is a demonstration of condition monitoring of rolling bearings, where the power spectral density (PSD) of generated vibration is analyzed to extract features indicative of the operational health of the machine. The second part delves into epilepsy detection using EEG brain signals, where wavelet packet transform (WPT) is used to decompose the signals into their elementary modes. Accordingly, the energy of each mode is calculated, facilitating efficient EEG energy-based epilepsy detection.

A. Vibration-Based Condition Monitoring of Rolling Bearings

Vibration-based condition monitoring (VBCM) is commonly used in predictive maintenance (PdM) due to its inherent advantages over alternative forms of condition monitoring that include [435], [436]:

- Vibration sensors are non-intrusive and can be contactless, facilitating non-destructive condition monitoring.
- Real-time acquisition of vibration signals can be conducted in situ, allowing for online local condition monitoring.
- Vibration sensors are cost-effective and widely available, offering various specifications to suit various requirements.
- Vibration waveform responds instantly to changes in the monitored condition and, therefore, is suitable for continuous and intermittent monitoring applications.
- Of paramount significance, signal processing techniques can be applied to vibration signals to mitigate corrupting noise and extract weak condition indications from other masking signals.

The Paderborn University (PU) bearing dataset [437] is used to facilitate a real-time VBCM. The dataset includes actual vibration signals from a real system during healthy and faulty operations. The fault types include inner race (IR), outer race (OR), and combined IR and OR defects. Thus, the dataset comprises four classes of operational conditions: 1- healthy, 2- IR faults, 3- OR faults, and 4- combined IR and OR faults. The vibration signals of the dataset were acquired at 64,000 samples per second. Hence, in the signal processing pipeline, the vibration signals are segmented into non-overlapping segments of $N_o = 12,800$ samples so that the resultant segment duration is 0.2 seconds. This duration is selected to facilitate precise real-time monitoring with moderate computational requirements. This segmentation results in a dataset of 7,997 vibration samples in total. Fig. 30 shows a healthy vibration pattern and faulty patterns for the PU dataset. The aim is to detect changes in the generated vibration (oscillation) patterns that may indicate a fault in the equipment. Here, power spectral analysis helps identify these oscillations since they appear as dominant frequencies in the PSD. Accordingly, PSDs of the

⁴<https://github.com/Western-OC2-Lab/Signal-Processing-for-Machine-Learning>

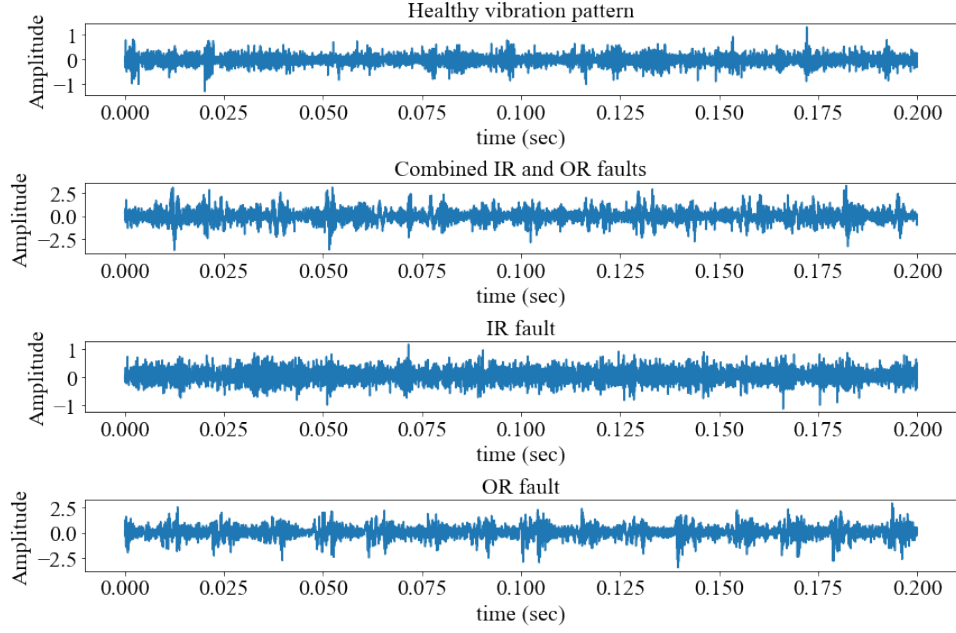


Fig. 30. Sample vibration patterns representing operational conditions in the PU dataset.

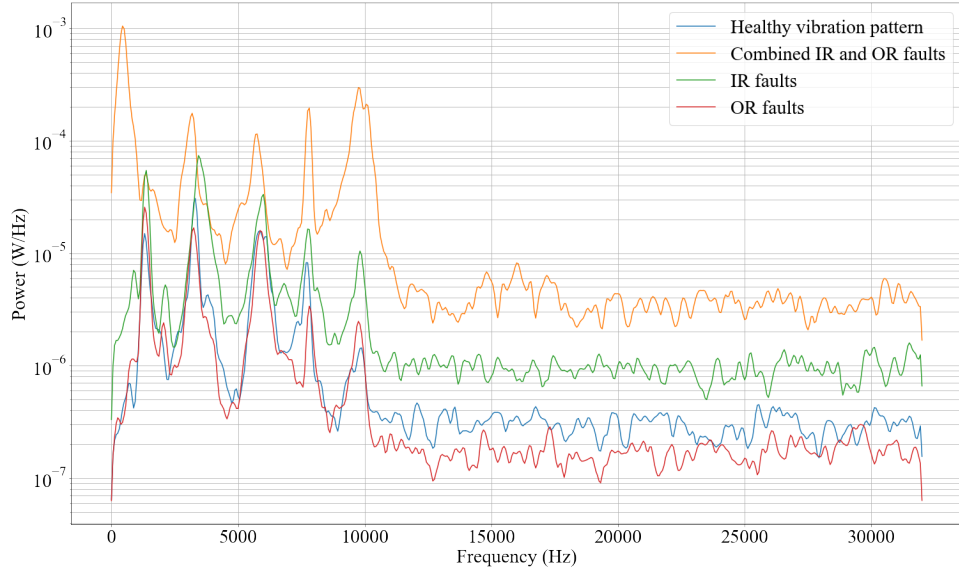


Fig. 31. PSD estimates of four vibration samples representing healthy and abnormal operational conditions in the dataset.

vibration samples are estimated by Welch's method using the Hann window and a segment length of 512 with 50% overlap. The Number of FFT points (NFFT) is set equal to 1024, which is calculated using the conventional method where NFFT is commonly set to be equal to 2^P , where P is the smallest power of 2 that is greater than or equal to the Welch segment length (512 samples). Thus, the value of p is set to 10. Fig. 31 shows PSD estimates of four vibration samples that represent healthy and abnormal operational conditions in the dataset. It is noticeable that a fault alters the PSD characteristics, and different faults have different effects on the PSD. It is worth noting that the segment length of 512 data points used in the Welch methods represents a short time duration compared

to the duration of the vibration segment. While such a short duration would reduce frequency resolution, it increases the number of estimates. This, in turn, helps to reduce noise [438] and variance of the PSD estimate, leading to a smoother and more reliable estimate of the PSD, as demonstrated in Fig. 32, which compares two different estimations of the PSD of a vibration sample. The first estimation is based on a short segment length of 512 data points, while the second estimation employs a larger segment of 6400 data points. After PSD estimation, relevant spectral features are extracted from the PSD to characterize the varying vibration patterns. These features include:

- Spectral Centroid (SC): The SC represents the center of

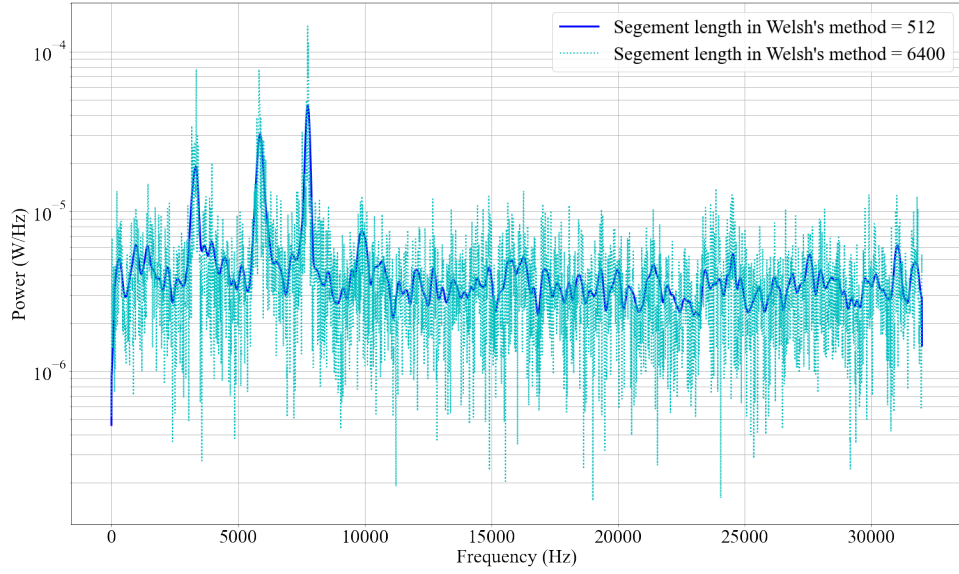


Fig. 32. Two different estimations of the PSD using short and large segments in the Welch method.

gravity of the spectral contents of a signal, indicating the average frequency at which the spectral power is concentrated. Since SC describes the spectral position of dominant oscillations in the generated vibration, it helps identify healthy and abnormal vibration patterns. SC is calculated from the PSD using the following formula [439]:

$$CS = \frac{\sum_{k=0}^{K-1} f_k P_k}{\sum_{k=0}^{K-1} P_k} \text{ Hz} \quad (58)$$

- **Spectral Spread (SS):** It describes the deviation of spectral contents with respect to the SC, providing a measure to assess how dispersed the spectral power is in relation to the SC. Mathematically, SS is obtained by calculating the spectral standard deviation with respect to SC [440]:

$$SS = \sqrt{\frac{\sum_{k=0}^{K-1} (f_k - CS)^2 P_k}{\sum_{k=0}^{K-1} P_k}} \text{ Hz} \quad (59)$$

The SC and SS are useful spectral features as they describe the position and the shape of spectral contents [440] and, hence, characterize dominant oscillatory components within the signal.

- **Spectral Entropy (SE):** SC is obtained by calculating the Shannon entropy of the power spectral values, P_k , to assess the irregularity of vibration patterns.

$$H[x[n]] = - \sum_{i=0}^{m-1} p(x_i) \log(p(x_i)), \quad (60)$$

where, $p(x_i)$ are probabilities associated with x_i representing m different values in $x[n]$. Typically, defects generate vibration with high regularity compared to healthy vibrations [441].

- **Peak Power (PP):** It represents the maximum value of spectral power in the PSD; it is expressed mathematically as $\max(P_k)$, $k = 0, 1, \dots, K$.

- **Peak Frequency (PF):** PF is the frequency pin that corresponds to the PP. The values of PP and PF characterize the dominant spike (maximum peak) in the PSD, which would be a discriminative indicator of healthy and faulty conditions as suggested by the comparison of healthy and abnormal PSDs in Fig. 31.

After the feature extraction stage, the resultant set of features is split into 5,597 (70%) samples for training and 2,400 (30%) samples for testing, and a random forest (RF) classifier is trained accordingly. The choice of the RF classifier is motivated by its higher performance compared to other classifiers, as reported in [44]. The achieved performance results and the confusion matrix (CM) are shown in Table V and Fig. 33, respectively. Further, a comparison, in terms of achieved accuracy (%), with recent deep learning (DL)-based methods on the PU dataset is presented in Table VI. The results show that the engineered features are highly effective in achieving excellent performance, similar to DL-based methods. Additionally, the signal processing approach introduced in this section is much more efficient than DL-based approaches. Specifically, unlike DL approaches, The introduced method employs a very short duration of the acquired vibration signal (only 0.2 seconds) and results in five features, sufficiently carrying out the tasks of fault detection and diagnosis. Moreover, in contrast to DL approaches, where extensive training on a large augmented dataset is required, the introduced method involves moderate training of an RF classifier.

B. Epilepsy Detection using EEG Brain Signals

The electroencephalogram (EEG) is a technique for recording electrical activity arising from the human brain. EEG signals are typically obtained by placing electrodes on the scalp to measure the differential voltage fluctuations resulting from ionic current flows within the neurons of the brain. EEG signals can detect changes over milliseconds [448], thereby providing insights into the rapid dynamics of brain activity

TABLE V
PERFORMANCE RESULTS OF THE PROPOSED AND OTHER METHODS ON THE PU DATASET.

| Accuracy (%) | F1 score | AUROC |
|--------------|----------|-------|
| 99.42% | 0.995 | 1.00 |

TABLE VI
PERFORMANCE COMPARISON BETWEEN THE PROPOSED SIGNAL PROCESSING (SP)-BASED METHOD AND OTHER DL-BASED METHODS ON THE PU DATASET.

| Method | Approach | Achieved accuracy (%) |
|----------|----------|-----------------------|
| Proposed | SP-based | 99.42% |
| [442] | DL-based | 99.44% |
| [443] | DL-based | 97.05% |
| [444] | DL-based | 96.51% |
| [445] | DL-based | 99.50% |
| [446] | DL-based | 96.67% |
| [447] | DL-based | 97.68% |

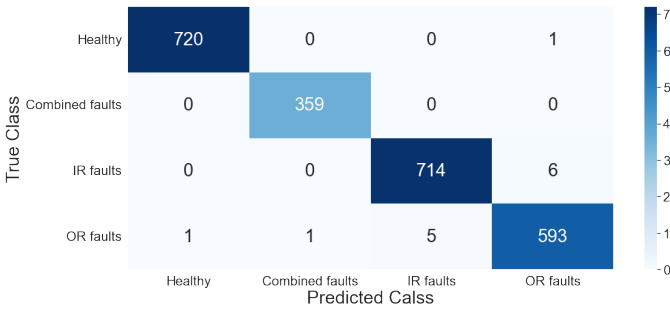


Fig. 33. Confusion matrix (CM) of VBCM of rolling bearings.

[449]. Thus, they are commonly used as a medical diagnostic tool for conditions like seizures and epilepsy due to their high temporal sensitivity. This section introduces a signal processing-based method for epilepsy detection using EEG signals. The EEG signals are obtained from the University of Bonn (UoB) EEG dataset [450], which is a well-known dataset in the field of epilepsy research. The dataset consists of five subsets, labeled A through E, each containing 100 single-channel EEG segments of 23.6 seconds. The signals were recorded with a sampling rate of 173.61 Hz, leading to 4097 data points per signal. sets C, D, and E were recorded from five patients; EEG signals in set D were taken from the epileptogenic zone, while set C was recorded from the hippocampal formation on the opposite hemisphere of the brain. Sets C and D comprise EEG signals measured during seizure-free intervals (interictal), whereas the EEG signals in set E were recorded exclusively during seizure activity (ictal) [451]. Thus, the EEG signal in the dataset can be categorized under three classes: normal (sets A and B), interictal (sets S and D), and ictal (set E). Fig. 34 shows an EEG sample from each class. The PSD-based method introduced in the previous section is adapted for EEG certification as a first attempt. The signals are segmented into non-overlapping segments, with a segment length N_o of 500 data point, which results in a segmented dataset of 4095 total EEG samples. With a sampling rate of 173.61 Hz, this gives a segment duration of 2.88 seconds of duration. For PSD estimation, the Welch's segment length is set to $\frac{N_o}{2}$ data points with segment overlap of

$\frac{N_o}{3}$ data points; NFFT is set to 1024. Samples from estimated PSDs are depicted in Fig. 35. Accordingly, SC, SS, SE, PP, and PF features are extracted from the estimated PSDs. An RF classifier is trained on the extracted features using 70% of the total samples (2866 samples) for training and the remaining 30% samples (1229) for testing. The achieved classification results are 93.409(%), 0.934, 0.989 representing, Accuracy (%), F1 score, and AUROC, respectively.

In contrast to vibrations of rolling bearings, EEG signals are characterized as low-frequency signals with high temporal sensitivity, making them particularly susceptible to various forms of artifacts, such as eye and body movements. This places the need to employ a suitable signal-denoising method to filter out these artifacts and maintain the frequency range that is of clinical significance. Typically, EEG components such as infra-slow oscillations (ISO) (less than 0.5 Hz) and high-frequency oscillations (HFOs) (greater than 30 Hz) are outside the conventional bandwidth of clinical EEG [452]. Considering this, a band-pass filter with pass and cut-off frequencies of 0.5 HZ and 30 Hz, respectively, is implemented to filter the EEG signals. Fig. 36 compares the time-domain waveforms, estimated PSDs, and frequency spectrum of unfiltered (raw) and filtered signals. After denoising the signals using the implemented filter, the PSDs of the filtered signals are estimated, and the features are extracted accordingly. Table VII shows the achieved results and compares them to those obtained using the unfiltered signals. The comparison shows that filtered EEG signals brought a noticeable improvement in epilepsy detection, demonstrating the effectiveness of the employed signal-denoising in removing irrelevant frequency components and, consequently, improving detection accuracy. The CM of the filtered EEG signals case is presented in Figure 37. It can be seen that there are high misclassifications between classes 1 and 2 (normal and interictal) in the CM. Examining PSD contents depicted in Fig. 35 reveals a high degree of similarity between the PSDs of these two classes. Furthermore, EEG signals are generally nonstationary and nonlinear [453]. These observations suggest that using time-frequency analyses for feature extraction would capture time-varying dynamics of EEG signals, thereby improving accuracy compared to pure frequency-domain approaches like PSD-

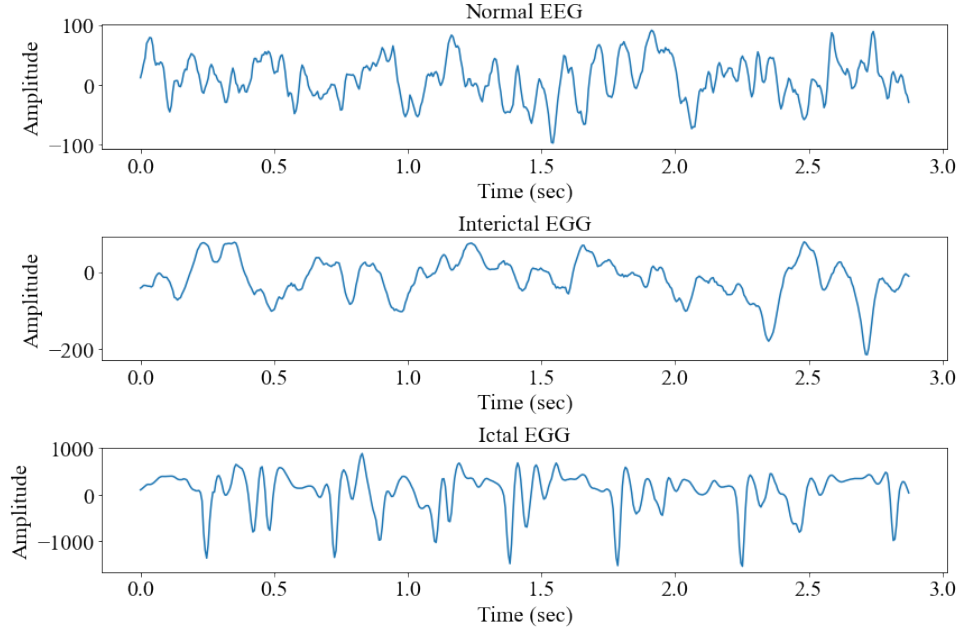


Fig. 34. Sample EEG signals representing the three classes of the UoB dataset.

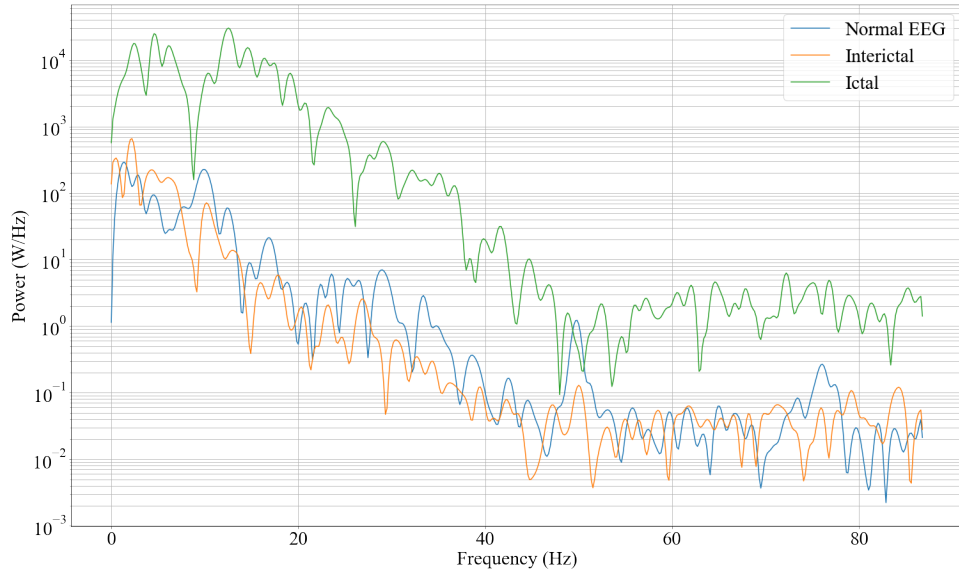


Fig. 35. Samples of estimated PSDs representing the three classes of the UoB dataset.

TABLE VII
PERFORMANCE RESULTS OF PSD-BASED EPILEPSY DETECTION.

| Input signal | Accuracy (%) | F1 score | AUROC |
|-----------------|--------------|----------|-------|
| Raw signal | 93.41% | 0.934 | 0.989 |
| Filtered signal | 94.55% | 0.945 | 0.991 |

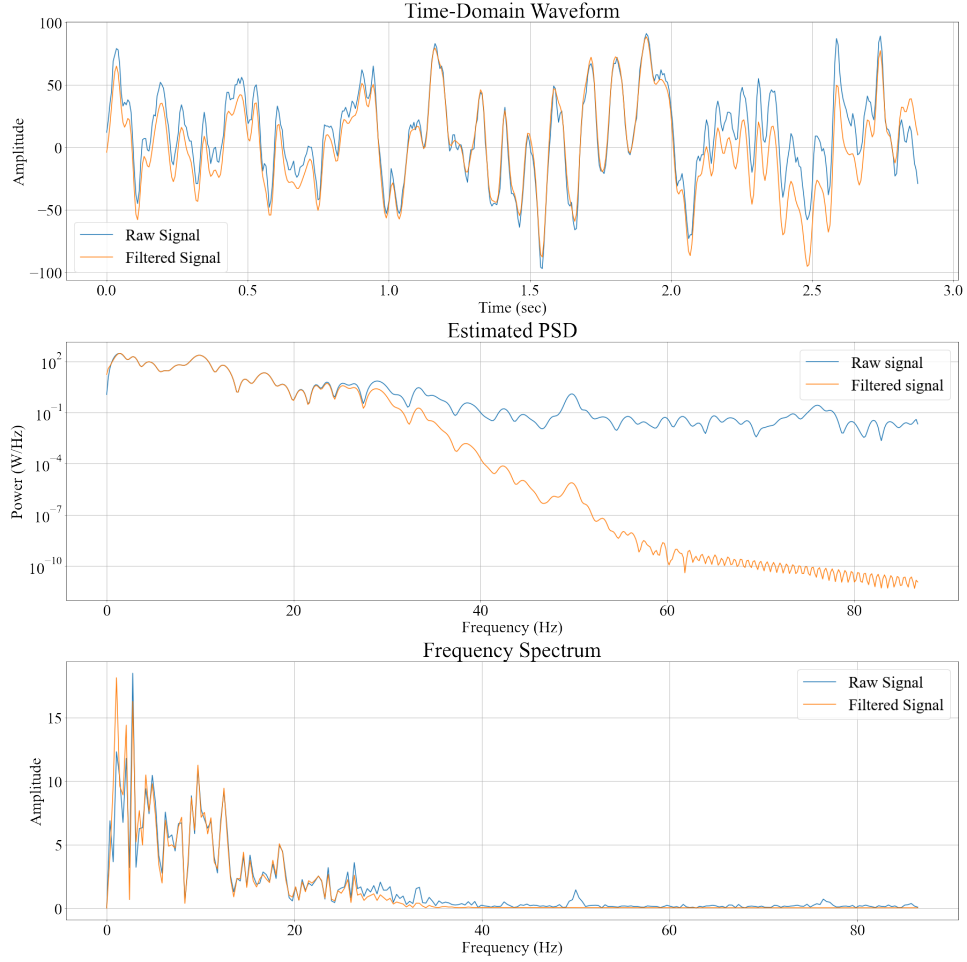


Fig. 36. A comparison between unfiltered (raw) and filtered EEG signals.

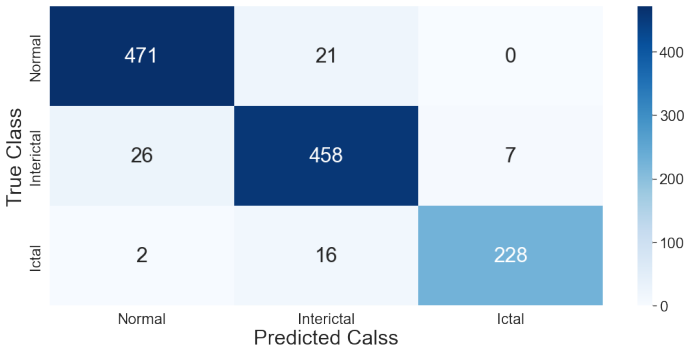


Fig. 37. Confusion matrix (CM) of the PSD-based method using filtered EEG signals.

based analysis. Thus, wavelet decomposition is employed to obtain elementary modes of EEG signals and the mode's energies are computed accordingly. This approach facilitates energy-based analysis with high time-frequency localization to capture energy-time-frequency varying dynamics within the EEG signal. The detailed steps of the method include:

- 1) Selection of a proper mother wavelet function to decompose the segmented EEG signals using WPT to the maximum decomposition level, denoted as k . This step

involves obtaining approximation and detail coefficients.

- 2) Use the obtained coefficients to construct elementary modes of the decomposed signal. This results in $M = 2^k$ constructed elementary modes with high-time frequency localization.

- 3) Compute energy E_i for each mode:

$$E_i = \sum_{l=0}^{L-1} |M_i[l]|^2, i = 1, 2, \dots, M \quad (61)$$

where L is the length of the constructed mode $M_i[l]$. Accordingly, a feature vector of size $S = x \times 2^K$ that contains energy values of each mode is created and fed into the RF classifier.

The selection of the mother wavelet is a critical step in wavelet analysis, which is often based on empirical evaluation and prior knowledge. In [454], a comparison is conducted between various mother wavelets to determine the most suitable wavelet for analyzing EEG signals. Results indicate that *db4* would be a proper selection for EEG energy analysis. Such existing results are particularly beneficial as they set guidelines for the proper selection of mother wavelets. Hence, using these findings as a guideline, Daubechies wavelets (*db4* - *db7*) are used to decompose the EEG signals, and energy features are extracted accordingly. The aim is to empirically

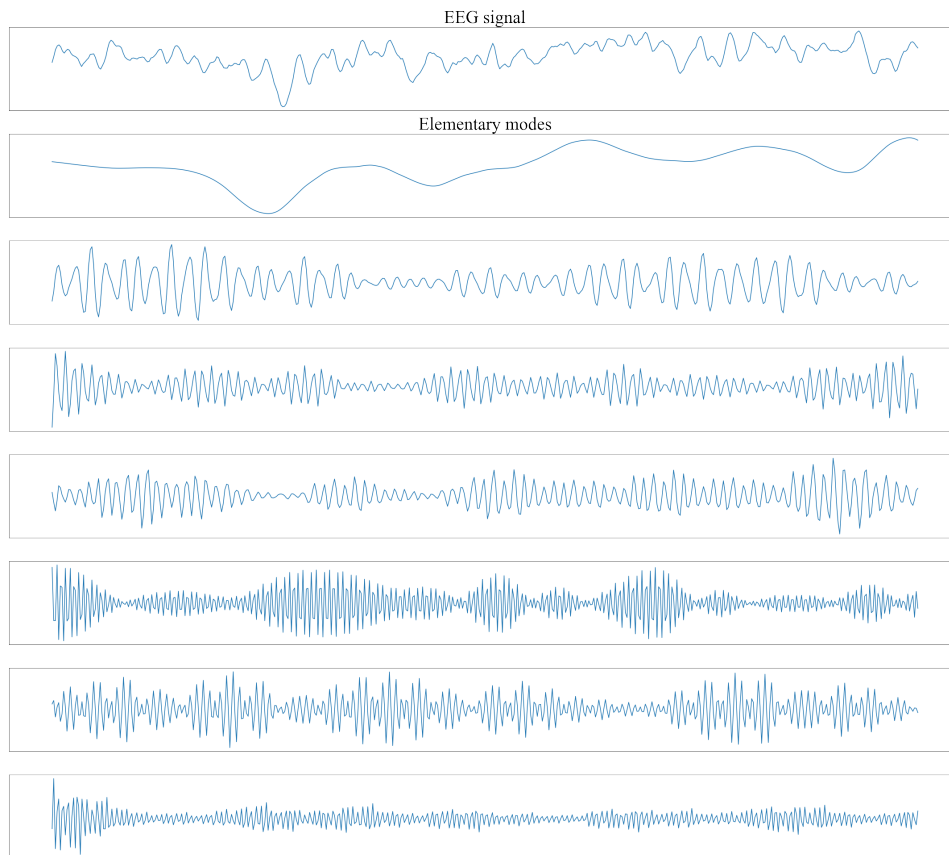


Fig. 38. EEG signal along with some of its constructed elementary modes.

TABLE VIII
PERFORMANCE RESULTS OF WAVELET-BASED EPILEPSY DETECTION.

| Mother wavelet | Accuracy (%) | F1 score | AUROC |
|----------------|--------------|----------|-------|
| <i>db4</i> | 96.83% | 0.968 | 0.998 |
| <i>db5</i> | 96.95% | 0.969 | 0.997 |
| <i>db6</i> | 96.59% | 0.996 | 0.997 |
| <i>db7</i> | 96.46% | 0.965 | 0.996 |

evaluate these wavelets to determine the most suitable one for this application. The obtained results are presented in Table VIII. A sample EEG signal and some of its constructed elementary modes are depicted in Fig38. The performance results show that all wavelets achieved comparable performance with noticeable improvements compared to PSD-based analysis. These results are based on filtered EEG signals where irrelevant frequency components (ISO and HFO) are filtered out. However, in recent years, these bands gained clinical importance with the advancements in DSP [452]. Motivated by this, the wavelet analysis is conducted on the unfiltered signals, and energy features are extracted accordingly. Table IX compares the obtained results with previous findings, including the results of PSD-based analysis. The comparison shows that in contrast to PSD-based analysis, where ISO and HFO components represent irrelevant frequency components, the presence of these components in wavelet analysis has considerably improved epilepsy detection. The results demonstrate that wavelet multi-resolution analysis reveals significant time-frequency dynamics within these bands, underscoring the

significance of including these bands in wavelet analysis.

VIII. CONCLUSION

Addressing the existing gaps related to the role of signal processing in ML, this paper has undertaken a comprehensive, integrated-article approach to present several contributions that aim to enrich the existing literature. First, the paper made a solid foundation on the topic through a comprehensive tutorial on signal processing fundamentals. Written for a diverse readership, the tutorial allows interested readers to grasp essential concepts and develop a proper background in signal processing.

Furthermore, the paper provided a comprehensive overview of a typical signal-processing pipeline, introducing a structured workflow for signal-based ML applications by categorizing tasks into preprocessing, processing, and application phases. Additionally, the paper introduced an exhaustive review of feature extraction methods through a new taxonomy that clearly distinguishes between two main concepts in feature extraction: feature learning and feature engineering, thereby

TABLE IX
PERFORMANCE RESULTS OF PSD-BASED AND WAVELET-BASED METHODS FOR EPILEPSY DETECTION.

| Approach | Input Signal | Accuracy (%) | F1 score | AUROC |
|--------------|-----------------|--------------|----------|-------|
| PSD analysis | Raw signal | 93.41% | 0.934 | 0.989 |
| PSD analysis | Filtered signal | 94.55% | 0.945 | 0.991 |
| WPT-db4 | Filtered signal | 96.83% | 0.968 | 0.998 |
| WPT-db5 | Filtered signal | 96.95% | 0.969 | 0.997 |
| WPT-db6 | Filtered signal | 96.59% | 0.996 | 0.997 |
| WPT-db7 | Filtered signal | 96.46% | 0.965 | 0.996 |
| WPT-db4 | Raw signal | 97.56% | 0.976 | 0.999 |
| WPT-db5 | Raw signal | 98.17% | 0.982 | 0.999 |
| WPT-db6 | Raw signal | 98.66% | 0.987 | 0.999 |
| WPT-db7 | Raw signal | 97.92% | 0.979 | 0.999 |

offering new insights into the topic of feature extraction. Focused on signal processing-based feature extraction, the paper reviewed various available techniques in terms of their main aspects, advantages, and limitations. The paper also addressed major challenges associated with the practical deployment of signal-based ML systems, explored their implications, and highlighted existing and potential solutions.

In an effort to connect the introduced theoretical concepts with real-world applications, the paper demonstrated the practical application of signal processing-based ML systems through two use cases: vibration-based condition monitoring of rolling bearings and epilepsy detection from EEG signals. Moreover, this work contributes to the research community by introducing a public repository of relevant Python and MATLAB codes for various signal-processing techniques.

REFERENCES

- [1] D. K. Iakovidis, M. Ooi, Y. C. Kuang, S. Demidenko, A. Shestakov, V. Sinitsin, M. Henry, A. Sciacchitano, S. Discetti, S. Donati, M. Norgia, A. Menychtas, I. Maglogiannis, S. C. Wriessnegger, L. A. B. Chacon, G. Dimas, D. Filos, A. H. Aletras, J. Töger, F. Dong, S. Ren, A. Uhl, J. Paziewski, J. Geng, F. Fioranelli, R. M. Narayanan, C. Fernandez, C. Stiller, K. Malamousi, S. Kamnis, K. Delibasis, D. Wang, J. Zhang, and R. X. Gao, "Roadmap on signal processing for next generation measurement systems," *Measurement Science and Technology*, vol. 33, p. 012002, 1 2022.
- [2] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, M. Gabbouj, and D. J. Inman, "A review of vibration-based damage detection in civil structures: From traditional methods to machine learning and deep learning applications," *Mechanical Systems and Signal Processing*, vol. 147, p. 107077, 1 2021.
- [3] J. Leukel, J. González, and M. Rieker, "Adoption of machine learning technology for failure prediction in industrial maintenance: A systematic review," *Journal of Manufacturing Systems*, vol. 61, pp. 87–96, 10 2021.
- [4] C. Zhang, A. A. Mousavi, S. F. Masri, G. Gholipour, K. Yan, and X. Li, "Vibration feature extraction using signal processing techniques for structural health monitoring: A review," *Mechanical Systems and Signal Processing*, vol. 177, p. 109175, 9 2022.
- [5] M. Romansini, P. C. C. de Aguirre, L. Compassi-Severo, and A. G. Girardi, "A review on vibration monitoring techniques for predictive maintenance of rotating machinery," *Eng*, vol. 4, pp. 1797–1817, 6 2023.
- [6] O. Matania, L. Bachar, E. Bechhoefer, and J. Bortman, "Signal processing for the condition-based maintenance of rotating machines via vibration analysis: A tutorial," *Sensors*, vol. 24, p. 454, 1 2024.
- [7] S. Sonawani, K. Patil, and P. Natarajan, "Biomedical signal processing for health monitoring applications: a review," *International Journal of Applied Systemic Studies*, vol. 10, p. 44, 2023.
- [8] Z. Khademi, F. Ebrahimi, and H. M. Kordy, "A review of critical challenges in mi-bci: From conventional to deep learning methods," *Journal of Neuroscience Methods*, vol. 383, p. 109736, 1 2023.
- [9] V. Gupta, M. Mittal, V. Mittal, and N. K. Saxena, "A critical review of feature extraction techniques for eeg signal analysis," *Journal of The Institution of Engineers (India): Series B*, vol. 102, pp. 1049–1060, 10 2021.
- [10] L. Boubchir, B. Daachi, and V. Pangracious, "A review of feature extraction for eeg epileptic seizure detection and classification," *IEEE*, 7 2017, pp. 456–460.
- [11] R. Yuvaraj, P. Thagavel, J. Thomas, J. Fogarty, and F. Ali, "Comprehensive analysis of feature extraction methods for emotion recognition from multichannel eeg recordings," *Sensors*, vol. 23, p. 915, 1 2023.
- [12] R. Jenke, A. Peer, and M. Buss, "Feature extraction and selection for emotion recognition from eeg," *IEEE Transactions on Affective Computing*, vol. 5, pp. 327–339, 7 2014.
- [13] A. Chaddad, Y. Wu, R. Kateb, and A. Bouridane, "Electroencephalography signal processing: A comprehensive review and analysis of methods and techniques," *Sensors*, vol. 23, p. 6434, 7 2023.
- [14] M. L. Duc, P. Bilik, and R. Martinek, "Harmonics signal feature extraction techniques: A review," *Mathematics*, vol. 11, p. 1877, 4 2023.
- [15] J. de Lope and M. Graña, "An ongoing review of speech emotion recognition," *Neurocomputing*, vol. 528, pp. 1–11, 4 2023.
- [16] S. Pangaonkar and A. Panat, *A Review of Various Techniques Related to Feature Extraction and Classification for Speech Signal Analysis*, 2020, pp. 534–549.
- [17] S. Nivetha, "A survey on speech feature extraction and classification techniques," *IEEE*, 2 2020, pp. 48–53.
- [18] G. Sharma, K. Umapathy, and S. Krishnan, "Trends in audio signal feature extraction methods," *Applied Acoustics*, vol. 158, p. 107020, 1 2020.
- [19] A. A. Anthony and C. M. Patil, "Speech emotion recognition systems: A comprehensive review on different methodologies," *Wireless Personal Communications*, vol. 130, pp. 515–525, 5 2023.
- [20] W. Liu, Y. Liu, S. Li, and Y. Chen, "A review of variational mode decomposition in seismic data analysis," *Surveys in Geophysics*, vol. 44, pp. 323–355, 4 2023.
- [21] S. M. Mousavi and G. C. Beroza, "Machine learning in earthquake seismology," *Annual Review of Earth and Planetary Sciences*, vol. 51, pp. 105–129, 5 2023.
- [22] P. Choudhary, N. Goel, and M. Saini, "A survey on seismic sensor based target detection, localization, identification, and activity recognition," *ACM Computing Surveys*, vol. 55, pp. 1–36, 11 2023.
- [23] A. Rojas and G. J. Dolecek, *A Review of Spectrum Sensing Techniques Based on Machine Learning*, 7 2024, pp. 1–21.
- [24] Q.-V. Pham, N. T. Nguyen, T. Huynh-The, L. B. Le, K. Lee, and W.-J. Hwang, "Intelligent radio signal processing: A survey," *IEEE Access*, vol. 9, pp. 83 818–83 850, 2021.
- [25] N. Soltanieh, Y. Norouzi, Y. Yang, and N. C. Karmakar, "A review of radio frequency fingerprinting techniques," *IEEE Journal of Radio Frequency Identification*, vol. 4, pp. 222–233, 9 2020.
- [26] Y. Li and L. Wang, "A survey of radio frequency fingerprint feature extraction and recognition based on deep learning," *Institution of Engineering and Technology*, 2022, pp. 776–782.
- [27] T. Wang, G. Yang, P. Chen, Z. Xu, M. Jiang, and Q. Ye, "A survey of applications of deep learning in radio signal modulation recognition," *Applied Sciences*, vol. 12, p. 12052, 11 2022.
- [28] A. Hazza, M. Shoaib, S. A. Alshebeili, and A. Fahad, "An overview of feature-based methods for digital modulation classification," *IEEE*, 2 2013, pp. 1–6.
- [29] S. Peng, S. Sun, and Y.-D. Yao, "A survey of modulation classification using deep learning: Signal representation and data preprocessing,"

- IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, pp. 7020–7038, 12 2022.
- [30] S. Xie and J. Ye, "Overview of automatic modulation recognition methods." *IEEE*, 4 2023, pp. 1–7.
 - [31] R. Priemer, *Introductory Signal Processing*. WORLD SCIENTIFIC, 11 1990.
 - [32] R. X. Gao and R. Yan, *Signals and Signal Processing in Manufacturing*. Springer US, 2011, pp. 1–15.
 - [33] L. F. Chaparro and A. Akan, *Frequency Analysis: The Fourier Series*. Elsevier, 2019, pp. 241–303.
 - [34] S. Braun, *SPECTRAL ANALYSIS, CLASSICAL METHODS*. Elsevier, 2001, pp. 1208–1223.
 - [35] D. L. Massart, B. G. M. Vandeginste, S. N. Deming, Y. Michotte, and L. Kaufman, Eds., *Chapter 12 - Signals and Data*. Elsevier, 2003, vol. 2, pp. 157–163. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0922348708702259>
 - [36] A. Stepchenko, J. Chizhov, L. Aleksejeva, and J. Tolujew, "Nonlinear, non-stationary and seasonal time series forecasting using different methods coupled with data preprocessing," *Procedia Computer Science*, vol. 104, pp. 578–585, 2017.
 - [37] H. Landau, "Sampling, data transmission, and the nyquist rate," *Proceedings of the IEEE*, vol. 55, pp. 1701–1706, 1967.
 - [38] G. Ellis, *The z -Domain*. Elsevier, 2012, pp. 73–96.
 - [39] A. Neupane, N. Raj, R. Deo, and M. Ali, *Development of data-driven models for wind speed forecasting in Australia*. Elsevier, 2021, pp. 143–190.
 - [40] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 7 1948.
 - [41] 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, pp. 754–767, 6 2020.
 - [42] B. Ricaud and B. Torrésani, "A survey of uncertainty principles and some signal processing applications," *Advances in Computational Mathematics*, vol. 40, pp. 629–650, 6 2014.
 - [43] P. Welch, "The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Transactions on Audio and Electroacoustics*, vol. 15, pp. 70–73, 6 1967.
 - [44] S. A. Aburakhia, R. Myers, and A. Shami, "A hybrid method for condition monitoring and fault diagnosis of rolling bearings with low system delay," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–13, 2022.
 - [45] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 9 1994.
 - [46] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 454, pp. 903–995, 3 1998.
 - [47] W. Guo and M. J. Zuo, *Adaptive Signal Decomposition Methods for Vibration Signals of Rotating Machinery*. InTech, 5 2017.
 - [48] M. Zare and N. M. Nouri, "End-effects mitigation in empirical mode decomposition using a new correlation-based expansion model," *Mechanical Systems and Signal Processing*, vol. 194, p. 110205, 7 2023.
 - [49] Y. Gao, G. Ge, Z. Sheng, and E. Sang, "Analysis and solution to the mode mixing phenomenon in emd." *IEEE*, 2008, pp. 223–227.
 - [50] Z. WU and N. E. HUANG, "Ensemble empirical mode decomposition: A noise-assisted data analysis method," *Advances in Adaptive Data Analysis*, vol. 01, pp. 1–41, 1 2009.
 - [51] K. Dragomiretskiy and D. Zosso, "Variational mode decomposition," *IEEE Transactions on Signal Processing*, vol. 62, pp. 531–544, 2 2014.
 - [52] J. S. Smith, "The local mean decomposition and its application to eeg perception data," *Journal of The Royal Society Interface*, vol. 2, pp. 443–454, 12 2005.
 - [53] J. Gilles, "Empirical wavelet transform," *IEEE Transactions on Signal Processing*, vol. 61, pp. 3999–4010, 8 2013.
 - [54] T. Eriksen and N. ur Rehman, "Data-driven nonstationary signal decomposition approaches: a comparative analysis," *Scientific Reports*, vol. 13, p. 1798, 1 2023.
 - [55] M. Civera and C. Surace, "A comparative analysis of signal decomposition techniques for structural health monitoring on an experimental benchmark," *Sensors*, vol. 21, p. 1825, 3 2021.
 - [56] Z. Feng, D. Zhang, and M. J. Zuo, "Adaptive mode decomposition methods and their applications in signal analysis for machinery fault diagnosis: A review with examples," *IEEE Access*, vol. 5, pp. 24 301–24 331, 2017.
 - [57] T. Liu, Z. Luo, J. Huang, and S. Yan, "A comparative study of four kinds of adaptive decomposition algorithms and their applications," *Sensors*, vol. 18, p. 2120, 7 2018.
 - [58] G. Lindfield and J. Penny, *Analyzing Data Using Discrete Transforms*. Elsevier, 2019, pp. 383–431.
 - [59] A. Tabrizi, L. Garibaldi, A. Fasana, and S. Marchesiello, *Influence of Stopping Criterion for Sifting Process of Empirical Mode Decomposition (EMD) on Roller Bearing Fault Diagnosis*, 2014, pp. 389–398.
 - [60] M. Ali, D. M. Khan, I. Saeed, and H. M. Alshanbari, "A new approach to empirical mode decomposition based on akima spline interpolation technique," *IEEE Access*, vol. 11, pp. 67 370–67 384, 2023.
 - [61] L. Cohen, *Time-frequency analysis*. Prentice hall New Jersey, 1995, vol. 778.
 - [62] J. E. Torres-Fernández, A. Prieto-Guerrero, and G. Espinosa-Paredes, "Decay ratio estimation using the instantaneous autocorrelation function," *Nuclear Engineering and Design*, vol. 240, pp. 3559–3570, 10 2010.
 - [63] F. Hlawatsch and G. Boudreaux-Bartels, "Linear and quadratic time-frequency signal representations," *IEEE Signal Processing Magazine*, vol. 9, pp. 21–67, 4 1992.
 - [64] Y. Wu and X. Li, "Elimination of cross-terms in the wigner–ville distribution of multi-component lfm signals," *IET Signal Processing*, vol. 11, pp. 657–662, 8 2017.
 - [65] P. Flandrin, "Some features of time-frequency representations of multi-component signals," in *ICASSP '84. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 9, 1984, pp. 266–269.
 - [66] —, *Time-frequency time-scale analysis*. Academic press, 1998.
 - [67] 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, pp. 862–871, 6 1989.
 - [68] 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, pp. 1084–1091, 7 1990.
 - [69] N. Abdennour, T. Ouni, and N. B. Amor, "The importance of signal pre-processing for machine learning: The influence of data scaling in a driver identity classification." *IEEE*, 11 2021, pp. 1–6.
 - [70] N. Mourad, "Robust smoothing of one-dimensional data with missing and/or outlier values," *IET Signal Processing*, vol. 15, pp. 323–336, 7 2021.
 - [71] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Analytical Chemistry*, vol. 36, pp. 1627–1639, 7 1964.
 - [72] S. R. Krishnan and C. S. Seelamantula, "On the selection of optimum savitzky-golay filters," *IEEE Transactions on Signal Processing*, vol. 61, pp. 380–391, 1 2013.
 - [73] Tanu and D. Kakkar, "Accounting for order-frame length tradeoff of savitzky-golay smoothing filters." *IEEE*, 2 2018, pp. 805–810.
 - [74] K. C. B. Rao and B. T. Krishna, "Comparative analysis of integer and non-integer order savitzky-golay digital filters." *IEEE*, 1 2017, pp. 26–31.
 - [75] X. Lv, F. Zhou, B. Li, and B. Yan, "Incipient fault feature extraction of rolling bearing based on signal reconstruction," *Electronics*, vol. 12, p. 3749, 9 2023.
 - [76] I. M. A. Wirawan, R. Wardoyo, D. Lelono, and S. Kusrohmaniah, "Comparison of smoothing methods to remove artifacts in emotion recognition based on electroencephalogram signals." *IEEE*, 12 2022, pp. 1–8.
 - [77] P. D. Rosero-Montalvo, V. F. López-Batista, and D. H. Peluffo-Ordóñez, "A new data-preprocessing-related taxonomy of sensors for iot applications," *Information*, vol. 13, p. 241, 5 2022.
 - [78] A. Kawala-Sterniuk, M. Podpora, M. Pelc, M. Błaszczyński, E. J. Gorzelanczyk, R. Martinek, and S. Ozana, "Comparison of smoothing filters in analysis of eeg data for the medical diagnostics purposes," *Sensors*, vol. 20, p. 807, 2 2020.
 - [79] P. Kowalski and R. Smyk, "Review and comparison of smoothing algorithms for one-dimensional data noise reduction." *IEEE*, 5 2018, pp. 277–281.
 - [80] H. W. Ott, *Noise reduction techniques in electronic systems*, 2nd ed. Wiley-Interscience, 1988, vol. 442.
 - [81] V. J. Barclay, R. F. Bonner, and I. P. Hamilton, "Application of wavelet transforms to experimental spectra: smoothing, denoising, and data set compression," *Analytical Chemistry*, vol. 69, pp. 78–90, 1 1997.
 - [82] H. Ashraf, A. Waris, M. Jamil, S. O. Gilani, I. K. Niazi, E. N. Kamavuako, and S. H. N. Gilani, "Determination of optimum segmentation schemes for pattern recognition-based myoelectric control:

- A multi-dataset investigation," *IEEE Access*, vol. 8, pp. 90 862–90 877, 2020.
- [83] A. Dehghani, O. Sarbishei, T. Glatard, and E. Shihab, "A quantitative comparison of overlapping and non-overlapping sliding windows for human activity recognition using inertial sensors," *Sensors*, vol. 19, p. 5026, 11 2019.
- [84] B. Fida, I. Bernabucci, D. Bibbo, S. Conforto, and M. Schmid, "Varying behavior of different window sizes on the classification of static and dynamic physical activities from a single accelerometer," *Medical Engineering Physics*, vol. 37, pp. 705–711, 7 2015.
- [85] V. J. D. Vieira, S. C. Costa, and S. E. N. Correia, "Non-stationarity-based adaptive segmentation applied to voice disorder discrimination," *IEEE Access*, vol. 11, pp. 54 750–54 759, 2023.
- [86] M. M. Milani, P. E. Abas, and L. C. D. Silva, "A critical review of heart sound signal segmentation algorithms," *Smart Health*, vol. 24, p. 100283, 6 2022.
- [87] K. Chang, Y. Yoo, and J.-G. Baek, "Anomaly detection using signal segmentation and one-class classification in diffusion process of semiconductor manufacturing," *Sensors*, vol. 21, p. 3880, 6 2021.
- [88] J. Jirka, M. Prauzek, O. Krejcar, and K. Kuca, "Automatic epilepsy detection using fractal dimensions segmentation and gprndash;svm classification," *Neuropsychiatric Disease and Treatment*, vol. Volume 14, pp. 2439–2449, 9 2018.
- [89] M. H. M. Noor, Z. S. Salic, and K. I.-K. Wang, "Adaptive sliding window segmentation for physical activity recognition using a single tri-axial accelerometer," *Pervasive and Mobile Computing*, vol. 38, pp. 41–59, 7 2017.
- [90] S. Bombiński, K. Błażej, M. Nejman, and K. Jemielniak, "Sensor signal segmentation for tool condition monitoring," *Procedia CIRP*, vol. 46, pp. 155–160, 2016.
- [91] A. Prochazka, M. Mudrova, O. Vysata, R. Hava, and C. P. S. Araujo, "Multi-channel eeg signal segmentation and feature extraction," *IEEE*, 5 2010, pp. 317–320.
- [92] H. O. A. Ahmed and A. K. Nandi, "Vibration image representations for fault diagnosis of rotating machines: A review," *Machines*, vol. 10, p. 1113, 11 2022.
- [93] C.-L. Yang, Z.-X. Chen, and C.-Y. Yang, "Sensor classification using convolutional neural network by encoding multivariate time series as two-dimensional colored images," *Sensors*, vol. 20, p. 168, 12 2019.
- [94] N. Sobahi, B. Ari, H. Cakar, O. F. Alcin, and A. Sengur, "A new signal to image mapping procedure and convolutional neural networks for efficient schizophrenia detection in eeg recordings," *IEEE Sensors Journal*, vol. 22, pp. 7913–7919, 4 2022.
- [95] L. Guo, Z. Lu, X. Wen, S. Zhou, and Z. Han, "From signal to image: Capturing fine-grained human poses with commodity wi-fi," *IEEE Communications Letters*, vol. 24, pp. 802–806, 4 2020.
- [96] P. Singh, P. Pandey, K. Miyapuram, and S. Raman, "Eeg2image: Image reconstruction from eeg brain signals," *IEEE*, 6 2023, pp. 1–5.
- [97] K. Gupta and Z. Wei, "Rm-net: Rasterizing markov signals to images for deep learning," *IEEE*, 10 2023, pp. 654–659.
- [98] C. M. Yilmaz, B. H. Yilmaz, and C. Kose, "A novel signal-to-image conversion approach with ensembles of pretrained cnns for motor imagery eeg signals," *IEEE*, 9 2023, pp. 49–53.
- [99] J. Venton, P. J. Aston, N. A. S. Smith, and P. M. Harris, "Signal to image to classification: Transfer learning for ecg," *IEEE*, 7 2020, pp. 1–2.
- [100] B. H. Yilmaz, C. M. Yilmaz, and C. Kose, "Classification of the eeg signals for the cursor movement with the signal-to-image transformation," *IEEE*, 10 2019, pp. 1–4.
- [101] T. Tayeh, S. Aburakhia, R. Myers, and A. Shami, "An attention-based convlstm autoencoder with dynamic thresholding for unsupervised anomaly detection in multivariate time series," *Machine Learning and Knowledge Extraction*, vol. 4, no. 2, pp. 350–370, 2022. [Online]. Available: <https://www.mdpi.com/2504-4990/4/2/15>
- [102] L. Yang and A. Shami, "A transfer learning and optimized cnn based intrusion detection system for internet of vehicles," in *ICC 2022 - IEEE International Conference on Communications*, 2022, pp. 2774–2779.
- [103] A. Moubayed, M. Sharif, M. Luccini, S. Primak, and A. Shami, "Water leak detection survey: Challenges research opportunities using data fusion federated learning," *IEEE Access*, vol. 9, pp. 40 595–40 611, 2021.
- [104] M. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, "Machine learning towards intelligent systems: applications, challenges, and opportunities," *Artificial Intelligence Review*, vol. 54, pp. 3299–3348, 6 2021.
- [105] L. Yang and A. Shami, "Iot data analytics in dynamic environments: From an automated machine learning perspective," *Engineering Applications of Artificial Intelligence*, vol. 116, p. 105366, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0952197622003803>
- [106] —, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925231220311693>
- [107] —, "A multi-stage automated online network data stream analytics framework for iiot systems," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 2107–2116, 2023.
- [108] V. Narayan, S. Awasthi, N. Fatima, M. Faiz, and S. Srivastava, "Deep learning approaches for human gait recognition: A review," *IEEE*, 2 2023, pp. 763–768.
- [109] S. Ahmad, K. Styp-Rekowski, S. Nedelkoski, and O. Kao, "Autoencoder-based condition monitoring and anomaly detection method for rotating machines," *IEEE*, 2 2020, pp. 4093–4102.
- [110] M. Liao, C. Liu, C. Wang, and J. Yang, "Research on a rolling bearing fault detection method with wavelet convolution deep transfer learning," *IEEE Access*, vol. 9, pp. 45 175–45 188, 2021.
- [111] D. Hassan, H. I. Hussein, and M. M. Hassan, "Heart disease prediction based on pre-trained deep neural networks combined with principal component analysis," *Biomedical Signal Processing and Control*, vol. 79, p. 104019, 2 2023.
- [112] Q. Nawaz, M. Younas, I. Hamid, and S. M. M. Gilani, "Recurrent neural network based human emotion recognition using eeg brain signals," *IEEE*, 2 2023, pp. 117–122.
- [113] F. S. Butt, M. F. Wagner, J. Schafer, and D. G. Ullate, "Toward automated feature extraction for deep learning classification of electrocardiogram signals," *IEEE Access*, vol. 10, pp. 118 601–118 616, 2022.
- [114] S. S. Xu, M.-W. Mak, and C.-C. Cheung, "Towards end-to-end eeg classification with raw signal extraction and deep neural networks," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, pp. 1574–1584, 2 2019.
- [115] W. Chen, H. Zhou, L. Cheng, and M. Xia, "Condition monitoring and anomaly detection of wind turbines using temporal convolutional informer and robust dynamic mahalanobis distance," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–14, 2023.
- [116] A. Mohammad-Alikhani, B. Nahid-Mobarakeh, and M.-F. Hsieh, "One-dimensional lstm-regulated deep neural network for data-driven fault detection in electric machines," *IEEE Transactions on Industrial Electronics*, vol. 71, pp. 3083–3092, 2 2024.
- [117] D. Zhang, Y. Lu, Y. Li, W. Ding, B. Zhang, and J. Xiao, "Frequency learning attention networks based on deep learning for automatic modulation classification in wireless communication," *Pattern Recognition*, vol. 137, p. 109345, 2 2023.
- [118] B. Zali-Vargahan, A. Charmin, H. Kalbkhani, and S. Barghandan, "Deep time-frequency features and semi-supervised dimension reduction for subject-independent emotion recognition from multi-channel eeg signals," *Biomedical Signal Processing and Control*, vol. 85, p. 104806, 2 2023.
- [119] W. Jia, M. Sun, J. Lian, and S. Hou, "Feature dimensionality reduction: a review," *Complex Intelligent Systems*, vol. 8, pp. 2663–2693, 2 2022.
- [120] M. Jalayer, C. Orsenigo, and C. Vercellis, "Fault detection and diagnosis for rotating machinery: A model based on convolutional lstm, fast fourier and continuous wavelet transforms," *Computers in Industry*, vol. 125, p. 103378, 2 2021.
- [121] Y. Zhang, K. Xing, R. Bai, D. Sun, and Z. Meng, "An enhanced convolutional neural network for bearing fault diagnosis based on time-frequency image," *Measurement*, vol. 157, p. 107667, 2 2020.
- [122] H. V. Dang, H. Tran-Ngoc, T. V. Nguyen, T. Bui-Tien, G. D. Roeck, and H. X. Nguyen, "Data-driven structural health monitoring using feature fusion and hybrid deep learning," *IEEE Transactions on Automation Science and Engineering*, vol. 18, pp. 2087–2103, 2 2021.
- [123] K. Zaman, M. Sah, C. Direkoglu, and M. Unoki, "A survey of audio classification using deep learning," *IEEE Access*, vol. 11, pp. 106 620–106 649, 2023.
- [124] Y. Miao, Y. V. Zakharov, 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, pp. 952–962, 2 2021.
- [125] N. P. G. Seshadri, S. Agrawal, B. K. Singh, B. Geethanjali, V. Mahesh, and R. B. Pachori, "Eeg based classification of children with learning disabilities using shallow and deep neural network," *Biomedical Signal Processing and Control*, vol. 82, p. 104553, 2 2023.

- [126] H. Altaheri, G. Muhammad, M. Alsulaiman, S. U. Amin, G. A. Altuwaijri, W. Abdul, M. A. Bencherif, and M. Faisal, "Deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: a review," *Neural Computing and Applications*, vol. 35, pp. 14 681–14 722, 2 2023.
- [127] L. Ilias, D. Askounis, and J. Psarras, "Multimodal detection of epilepsy with deep neural networks," *Expert Systems with Applications*, vol. 213, p. 119010, 2 2023.
- [128] S.-H. Kim, C.-B. Moon, J.-W. Kim, and D.-S. Kim, "A hybrid deep learning model for automatic modulation classification," *IEEE Wireless Communications Letters*, vol. 11, pp. 313–317, 2 2022.
- [129] S. Zheng, X. Zhou, L. Zhang, P. Qi, K. Qiu, J. Zhu, and X. Yang, "Toward next-generation signal intelligence: A hybrid knowledge and data-driven deep learning framework for radio signal classification," *IEEE Transactions on Cognitive Communications and Networking*, vol. 9, pp. 564–579, 2 2023.
- [130] S. Peng, S. Sun, and Y.-D. Yao, "A survey of modulation classification using deep learning: Signal representation and data preprocessing," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, pp. 7020–7038, 2 2022.
- [131] T. Chen, S. Gao, S. Zheng, S. Yu, Q. Xuan, C. Lou, and X. Yang, "Emd and vmd empowered deep learning for radio modulation recognition," *IEEE Transactions on Cognitive Communications and Networking*, vol. 9, pp. 43–57, 2 2023.
- [132] W. Li, K. Wang, L. You, and Z. Huang, "A new deep learning framework for hf signal detection in wideband spectrogram," *IEEE Signal Processing Letters*, vol. 29, pp. 1342–1346, 2022.
- [133] K. Qiu, S. Zheng, L. Zhang, C. Lou, and X. Yang, "Deepsig: A hybrid heterogeneous deep learning framework for radio signal classification," *IEEE Transactions on Wireless Communications*, p. 1, 2023.
- [134] Y. Lei, Z. He, and Y. Zi, "A new approach to intelligent fault diagnosis of rotating machinery," *Expert Systems with Applications*, vol. 35, pp. 1593–1600, 2 2008.
- [135] B. R. Nayana and P. Geethanjali, "Analysis of statistical time-domain features effectiveness in identification of bearing faults from vibration signal," *IEEE Sensors Journal*, vol. 17, pp. 5618–5625, 2 2017.
- [136] P. Geethanjali, Y. K. Mohan, and J. Sen, "Time domain feature extraction and classification of eeg data for brain computer interface," *IEEE*, 2 2012, pp. 1136–1139.
- [137] X. Wang, Y. Zheng, Z. Zhao, and J. Wang, "Bearing fault diagnosis based on statistical locally linear embedding," *Sensors*, vol. 15, pp. 16 225–16 247, 2 2015.
- [138] S. M. S. Mocar, I. Rida, and S. Boudaoud, "Time-domain features for semg signal classification: A brief survey," *JETSAN 2023*, 2023.
- [139] A. Mert, N. Kılıç, and A. Akan, "Evaluation of bagging ensemble method with time-domain feature extraction for diagnosing of arrhythmia beats," *Neural Computing and Applications*, vol. 24, pp. 317–326, 2 2014.
- [140] S. U. Jan, Y.-D. Lee, J. Shin, and I. Koo, "Sensor fault classification based on support vector machine and statistical time-domain features," *IEEE Access*, vol. 5, pp. 8682–8690, 2017.
- [141] D. Tkach, H. Huang, and T. A. Kuiken, "Study of stability of time-domain features for electromyographic pattern recognition," *Journal of NeuroEngineering and Rehabilitation*, vol. 7, p. 21, 2 2010.
- [142] B. Sreejith, A. K. Verma, and A. Srividya, "Fault diagnosis of rolling element bearing using time-domain features and neural networks," *IEEE*, 2 2008, pp. 1–6.
- [143] Y. Paul, V. Goyal, and R. A. Jaswal, "Comparative analysis between svm amp; knn classifier for emg signal classification on elementary time domain features," *IEEE*, 2 2017, pp. 169–175.
- [144] R. Barioul, S. Fakhfakh, H. Derbel, and O. Kanoun, "Evaluation of emg signal time domain features for hand gesture distinction," *IEEE*, 2 2019, pp. 489–493.
- [145] F. O.K. and R. R., "Time-domain exponential energy for epileptic eeg signal classification," *Neuroscience Letters*, vol. 694, pp. 1–8, 2 2019.
- [146] K. Nazarpour, A. R. Sharafat, and S. M. Firoozabadi, "A novel feature extraction scheme for myoelectric signals classification using higher order statistics," *IEEE*, pp. 293–296.
- [147] K. Nazarpour, A. R. Sharafat, and S. M. P. Firoozabadi, "Application of higher order statistics to surface electromyogram signal classification," *IEEE Transactions on Biomedical Engineering*, vol. 54, pp. 1762–1769, 2 2007.
- [148] J. Chen, B. Xu, and X. Zhang, "A vibration feature extraction method based on time-domain dimensional parameters and mahalanobis distance," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–12, 7 2021.
- [149] A. Saci, A. Al-Dweik, and A. Shami, "Autocorrelation integrated gaussian based anomaly detection using sensory data in industrial manufacturing," *IEEE Sensors Journal*, vol. 21, no. 7, pp. 9231–9241, 2021.
- [150] E. Ali, R. K. Udhayakumar, M. Angelova, and C. Karmakar, "Performance analysis of entropy methods in detecting epileptic seizure from surface electroencephalograms," *IEEE*, 2 2021, pp. 1082–1085.
- [151] W. Jiao, G. Li, Y. Jiang, R. Baim, C. Tang, T. Yan, X. Ding, and Y. Yan, "Multi-scale sample entropy-based energy moment features applied to fault classification," *IEEE Access*, vol. 9, pp. 8444–8454, 2021.
- [152] C. Liang and C. Chen, "Generalized composite multiscale diversity entropy and its application for fault diagnosis of rolling bearing in automotive production line," *IEEE Access*, vol. 9, pp. 84 545–84 558, 2021.
- [153] M. Li and W. Chen, "Fft-based deep feature learning method for eeg classification," *Biomedical Signal Processing and Control*, vol. 66, p. 102492, 2 2021.
- [154] N. F. Güler and S. Koçer, "Classification of emg signals using pca and fft," *Journal of Medical Systems*, vol. 29, pp. 241–250, 2 2005.
- [155] M. K. Delimayanti, B. Purnama, N. G. Nguyen, M. R. Faisal, K. R. Mahmudah, F. Indriani, M. Kubo, and K. Satou, "Classification of brainwaves for sleep stages by high-dimensional fft features from eeg signals," *Applied Sciences*, vol. 10, p. 1797, 2 2020.
- [156] E. C. Djamal, W. I. Furi, and F. Nugraha, "Detection of eeg signal post-stroke using fft and convolutional neural network," *IEEE*, 2 2019, pp. 18–23.
- [157] H. Arabaci and A. Mohamed, "A knowledge-based diagnosis algorithm for broken rotor bar fault classification using fft, principal component analysis and support vector machines," *International Journal of Intelligent Engineering Informatics*, vol. 8, p. 19, 2020.
- [158] C. Altun and O. Er, "Comparison of different time and frequency domain feature extraction methods on elbow gesture's emg," *European Journal of Interdisciplinary Studies*, vol. 2, p. 35, 2 2016.
- [159] S. K. Narudin, N. M. Nasir, A. Joret, N. Fuad, B. S. K. K. Ibrahim, and M. S. Huq, "Classification of hand movements of stroke patients using combination of statistical features and artificial neural network," 2023, p. 60011.
- [160] L. S. Dhamande and M. B. Chaudhari, "Compound gear-bearing fault feature extraction using statistical features based on time-frequency method," *Measurement*, vol. 125, pp. 63–77, 2 2018.
- [161] M. Murugappan and S. Murugappan, "Human emotion recognition through short time electroencephalogram (eeg) signals using fast fourier transform (fft)," *IEEE*, 2 2013, pp. 289–294.
- [162] S. A. Shufni and M. Y. Mashor, "Ecg signals classification based on discrete wavelet transform, time domain and frequency domain features," *IEEE*, 2 2015, pp. 1–6.
- [163] M. Altaf, T. Akram, M. A. Khan, M. Iqbal, M. M. I. Ch, and C.-H. Hsu, "A new statistical features based approach for bearing fault diagnosis using vibration signals," *Sensors*, vol. 22, p. 2012, 2 2022.
- [164] L. Jiang, J. He, H. Pan, D. Wu, T. Jiang, and J. Liu, "Seizure detection algorithm based on improved functional brain network structure feature extraction," *Biomedical Signal Processing and Control*, vol. 79, p. 104053, 2 2023.
- [165] P. Singh, S. D. Joshi, R. K. Patney, and K. Saha, "Fourier-based feature extraction for classification of eeg signals using eeg rhythms," *Circuits, Systems, and Signal Processing*, vol. 35, pp. 3700–3715, 2 2016.
- [166] M. N. Alam, M. I. Ibrahimy, and S. M. A. Motakabber, "Feature extraction of eeg signal by power spectral density for motor imagery based bci," *IEEE*, 2 2021, pp. 234–237.
- [167] Q. Gao, A. H. Omran, Y. Baghersad, O. Mohammadi, M. A. Alkhafaji, A. K. J. Al-Azzawi, S. H. Al-Khafaji, N. Emami, D. Toghraie, and M. J. Golkar, "Electroencephalogram signal classification based on fourier transform and pattern recognition network for epilepsy diagnosis," *Engineering Applications of Artificial Intelligence*, vol. 123, p. 106479, 2 2023.
- [168] A. Zhang, B. Yang, and L. Huang, "Feature extraction of eeg signals using power spectral entropy," *IEEE*, 2 2008, pp. 435–439.
- [169] S. Babeetha and S. S. Sridhar, *EEG Signal Feature Extraction Using Principal Component Analysis and Power Spectral Entropy for Multi-class Emotion Prediction*, 2023, pp. 435–448.
- [170] D. Konar, S. De, P. Mukherjee, and A. H. Roy, "A novel human stress level detection technique using eeg," *IEEE*, 2 2023, pp. 1–6.
- [171] I. Shaer and A. Shami, "Sound event classification in an industrial environment: Pipe leakage detection use case," in *2022 International Wireless Communications and Mobile Computing (IWCMC)*, 2022, pp. 1212–1217.

- [172] G. R. D, "Analysis of mfcc features for eeg signal classification," *International Journal of Advances in Signal and Image Sciences*, vol. 2, p. 14, 12 2016.
- [173] M. Deng, T. Meng, J. Cao, S. Wang, J. Zhang, and H. Fan, "Heart sound classification based on improved mfcc features and convolutional recurrent neural networks," *Neural Networks*, vol. 130, pp. 22–32, 10 2020.
- [174] Y. Arpitha, G. L. Madhumathi, and N. Balaji, "Spectrogram analysis of ecg signal and classification efficiency using mfcc feature extraction technique," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, pp. 757–767, 2 2022.
- [175] M. Boussaa, I. Atouf, M. Atibi, and A. Bennis, "Ecg signals classification using mfcc coefficients and ann classifier," *IEEE*, 5 2016, pp. 480–484.
- [176] S. A. A. Yusuf and R. Hidayat, "Mfcc feature extraction and knn classification in ecg signals," *IEEE*, 9 2019, pp. 1–5.
- [177] M. Mohan, P. Dhanalakshmi, and R. S. Kumar, "Speech emotion classification using ensemble models with mfcc," *Procedia Computer Science*, vol. 218, pp. 1857–1868, 2023.
- [178] A. Abbaskhah, H. Sedighi, and H. Marvi, "Infant cry classification by mfcc feature extraction with mlp and cnn structures," *Biomedical Signal Processing and Control*, vol. 86, p. 105261, 9 2023.
- [179] X. Mu and C.-H. Min, "Mfcc as features for speaker classification using machine learning," *IEEE*, 6 2023, pp. 0566–0570.
- [180] F. Alharbi, S. Luo, H. Zhang, K. Shaukat, G. Yang, C. A. Wheeler, and Z. Chen, "A brief review of acoustic and vibration signal-based fault detection for belt conveyor idlers using machine learning models," *Sensors*, vol. 23, p. 1902, 2 2023.
- [181] F. Rustam, A. Ishaq, M. S. A. Hashmi, H. U. R. Siddiqui, L. A. D. López, J. C. Galán, and I. Ashraf, "Railway track fault detection using selective mfcc features from acoustic data," *Sensors*, vol. 23, p. 7018, 8 2023.
- [182] Z. K. Abdul and A. K. Al-Talabani, "Mel frequency cepstral coefficient and its applications: A review," *IEEE Access*, vol. 10, pp. 122 136–122 158, 2022.
- [183] A. P. Petropulu, "Higher-order spectra in biomedical signal processing," *IFAC Proceedings Volumes*, vol. 27, pp. 47–52, 3 1994.
- [184] C. Zhu, T. Cao, L. Chen, X. Dai, Q. Ge, and X. Zhao, "High-order domain feature extraction technology for ocean acoustic observation signals: A review," *IEEE Access*, vol. 11, pp. 17 665–17 683, 2023.
- [185] K. C. Chua, V. Chandran, U. R. Acharya, and C. M. Lim, "Application of higher order statistics/spectra in biomedical signals—a review," *Medical Engineering Physics*, vol. 32, pp. 679–689, 9 2010.
- [186] S. A. Hosseini, "A hybrid approach based on higher order spectra for clinical recognition of seizure and epilepsy using brain activity," *Basic and Clinical Neuroscience Journal*, vol. 8, pp. 479–492, 11 2017.
- [187] E. B. Assi, L. Gagliano, S. Rihana, D. K. Nguyen, and M. Sawan, "Bispectrum features and multilayer perceptron classifier to enhance seizure prediction," *Scientific Reports*, vol. 8, p. 15491, 10 2018.
- [188] K. Tan, W. Yan, L. Zhang, Q. Ling, and C. Xu, "Semi-supervised specific emitter identification based on bispectrum feature extraction cgan in multiple communication scenarios," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, pp. 292–310, 2 2023.
- [189] Y. Zhou, R. Zhou, C. Wang, X. Wang, Y. Yu, and X. Zhang, "A modulation recognition method based on bispectrum and ensemble learning," *IEEE*, 9 2022, pp. 124–128.
- [190] X. Li, M. Yu, Y. Liu, and X. Xu, "Feature extraction of underwater signals based on bispectrum estimation," *IEEE*, 9 2011, pp. 1–4.
- [191] Z. Dong, F. Lv, T. Wan, K. Jiang, X. Fang, and L. Zhang, "Radar signal modulation recognition based on bispectrum features and deep learning," *IEEE*, 6 2021, pp. 63–67.
- [192] Y. Liu, Y. Zhou, W. Xin, Y. He, and P. Fan, "Bispectrum analysis for feature extraction of pitting fault in wind turbine gearbox," *IEEE*, 8 2010, pp. 1298–1301.
- [193] D. G. Bhalke, C. B. R. Rao, and D. S. Bormane, "Musical instrument classification using higher order spectra," *IEEE*, 2 2014, pp. 40–45.
- [194] A. C. McCormick and A. K. Nandi, "Bispectral and trispectral features for machine condition diagnosis," *IEE Proceedings - Vision, Image, and Signal Processing*, vol. 146, p. 229, 1999.
- [195] K. Muthuvel, L. P. Suresh, T. J. Alexander, and S. H. K. Veni, *Classification of ECG Signal Using Hybrid Feature Extraction and Neural Network Classifier*, 2015, pp. 1537–1544.
- [196] M. K. J. A. T. M. P. T. Thangam, and P. S. L., "Hybrid feature extraction technique for electrocardiograph arrhythmia signal classification," *IEEE*, 8 2023, pp. 1518–1524.
- [197] C. K. Kovach and M. A. Howard, "Decomposition of higher-order spectra for blind multiple-input deconvolution, pattern identification and separation," *Signal Processing*, vol. 165, pp. 357–379, 12 2019.
- [198] M. K. Gourisaria, R. Agrawal, M. Sahni, and P. K. Singh, "Comparative analysis of audio classification with mfcc and stft features using machine learning techniques," *Discover Internet of Things*, vol. 4, p. 1, 1 2024.
- [199] A.-C. Tsai, J.-J. Luh, and T.-T. Lin, "A novel stft-ranking feature of multi-channel emg for motion pattern recognition," *Expert Systems with Applications*, vol. 42, pp. 3327–3341, 5 2015.
- [200] T. Huynh-The, Q.-V. Pham, T.-H. Vu, D. B. da Costa, and V.-P. Hoang, "Intelligent spectrum sensing with convnet for 5g and lte signals identification," *IEEE*, 7 2023, pp. 140–144.
- [201] J. Huang, B. Chen, B. Yao, and W. He, "Ecg arrhythmia classification using stft-based spectrogram and convolutional neural network," *IEEE Access*, vol. 7, pp. 92 871–92 880, 2019.
- [202] Y. Hu, L. Wang, and W. Fu, "Eeg feature extraction of motor imagery based on wt and stft," *IEEE*, 8 2018, pp. 83–88.
- [203] O. K. Cura, G. C. Yilmaz, H. S. Ture, and A. Akan, "Deep time-frequency feature extraction for alzheimer's dementia eeg classification," *IEEE*, 10 2022, pp. 1–4.
- [204] A. Buyya, T. Ogeti, G. Suhas, P. Kashapogula, and A. K. Panigrahy, "Arrhythmias classification by using stft-based spectrograms, transfer learning and concatenation of features," *IEEE*, 5 2023, pp. 1–5.
- [205] H. Liu, L. Li, and J. Ma, "Rolling bearing fault diagnosis based on stft-deep learning and sound signals," *Shock and Vibration*, vol. 2016, pp. 1–12, 2016.
- [206] P. Ewert, B. Wicher, and T. Pajchrowski, "Application of the stft for detection of the rotor unbalance of a servo-drive system with an elastic interconnection," *Electronics*, vol. 13, p. 441, 1 2024.
- [207] Y. Hao, C. Zhang, Y. Lu, L. Zhang, Z. Lei, and Z. Li, "A novel autoencoder modeling method for intelligent assessment of bearing health based on short-time fourier transform and ensemble strategy," *Precision Engineering*, vol. 85, pp. 89–101, 1 2024.
- [208] H. Shayeste and B. M. Asl, "Automatic seizure detection based on gray level co-occurrence matrix of stft imaged-eeg," *Biomedical Signal Processing and Control*, vol. 79, p. 104109, 1 2023.
- [209] Z. Suryady, A. W. A. Rahman, and D. Oktarina, "Human emotion classification based on phonocardiography signals (pcg)," 2024, p. 020002.
- [210] N. Sharma, G. G. and R. S. Anand, "Epileptic seizure detection using stft based peak mean feature and support vector machine," *IEEE*, 8 2021, pp. 1131–1136.
- [211] H. Zehir, T. Hafs, S. Daas, and A. Nait-Ali, "An ecg biometric system based on empirical mode decomposition and hilbert-huang transform for improved feature extraction," *IEEE*, 6 2023, pp. 1–4.
- [212] L. Li, S. Wang, Y. Zhao, and F. Wang, "Fault diagnosis for single-phase grounding in distribution network based on hilbert-huang transform and siamese convolution neural network," *IEEE*, 5 2023, pp. 3571–3576.
- [213] P. Warule, S. P. Mishra, S. Deb, and D. Joshi, "Hilbert-huang transform-based time-frequency analysis of speech signals for the identification of common cold," *IEEE*, 10 2023, pp. 876–880.
- [214] Y. P. Bórnea, A. L. O. Vitor, M. F. Castoldi, A. Goedtel, and W. A. Souza, "Classification of bearing faults in induction motors with the hilbert-huang transform and feature selection," *IEEE*, 8 2023, pp. 567–573.
- [215] D. Walsh, S. Dev, and A. Nag, "Hilbert-huang-transform based features for accent classification of non-native english speakers," *IEEE*, 6 2023, pp. 1–6.
- [216] B. S. Munir, M. Reza, A. Trisetayrso, and B. S. Abbas, "Feature extraction using hilbert-huang transform for power system oscillation measurements," *IEEE*, 10 2017, pp. 93–96.
- [217] D. Du, J. Zhang, Y. Fang, and J. Tian, "Motor bearing fault diagnosis based on hilbert-huang transform and convolutional neural networks," *IEEE*, 10 2022, pp. 1–5.
- [218] D. H. Kisa, M. A. Ozdemir, O. Guren, and A. Akan, "Classification of hand gestures using semg signals and hilbert-huang transform," *IEEE*, 8 2022, pp. 1293–1297.
- [219] N. M. Bhiri, S. Ameur, I. Jegham, M. A. Mahjoub, and A. B. Khalifa, "Fisher-hht: A feature extraction approach for hand gesture recognition with a leap motion controller," *IEEE*, 5 2022, pp. 1–6.
- [220] J. Li, Z. Liang, Y. Qi, L. Li, G. Li, N. Zheng, and Z. Wang, "Feature extraction and state identification of transformer dc bias vibration based on hht-svm," *IEEE*, 4 2022, pp. 746–750.
- [221] J. Chen, H. Li, L. Ma, H. Bo, and X. Gao, "Application of eemd-hht method on eeg analysis for speech evoked emotion recognition," *IEEE*, 8 2020, pp. 376–381.

- [222] K. Cai, W. Cao, L. Aarniovuori, H. Pang, Y. Lin, and G. Li, "Classification of power quality disturbances using wigner-ville distribution and deep convolutional neural networks," *IEEE Access*, vol. 7, pp. 119 099–119 109, 2019.
- [223] M. Zulfiqar, M. F. U. Butt, A. Rama, and I. Shafi, "Abnormality detection in cardiac signals using pseudo wigner-ville distribution with pre-trained convolutional neural network," *IEEE*, 12 2019, pp. 1–5.
- [224] G. Wei, T. Jiang, Y. Liu, and W. Liu, "Dynamic gesture recognition using software defined radio based on smooth pseudo wigner-ville distribution," *IEEE*, 12 2015, pp. 1–5.
- [225] F. Akdeniz, I. Kayikcioglu, I. Kaya, and T. Kayikcioglu, "Using wigner-ville distribution in ecg arrhythmia detection for telemedicine applications," *IEEE*, 6 2016, pp. 409–412.
- [226] R. Abayneh, M. Motala, J. Ren, and X. Yang, "Efficient wi-fi device rf fingerprint detection and classification using wigner-ville distribution," *IEEE*, 11 2023, pp. 504–508.
- [227] S. Taran, S. K. Khare, P. V. K. Krishna, and V. S. Kardam, *Focal and non-focal EEG signal classification using the Wigner-Ville distribution and deep feature extraction*. IOP Publishing, 2 2024, pp. 2–12–14.
- [228] K. Kumar, P. Kumar, R. K. Patel, M. Sharma, V. Bajaj, and U. R. Acharya, "Time frequency distribution and deep neural network for automated identification of insomnia using single channel eeg-signals," *IEEE Latin America Transactions*, vol. 22, pp. 186–194, 3 2024.
- [229] X. Zou, H. Mu, R. Wang, K. Fan, Z. Cheng, Y. He, and G. Zhang, "An efficient accuracy improvement method for cable defect location based on instantaneous filtering in time-frequency domain," *Measurement*, vol. 226, p. 114178, 2 2024.
- [230] S. K. Dash and G. S. Rao, "Arrhythmia detection using wigner-ville distribution based neural network," *Procedia Computer Science*, vol. 85, pp. 806–811, 2016.
- [231] P. Shegokar and P. Sircar, "Continuous wavelet transform based speech emotion recognition," *IEEE*, 12 2016, pp. 1–8.
- [232] C. Peng, W. Cheng, Z. Song, and R. Dong, "A noise-robust modulation signal classification method based on continuous wavelet transform," *IEEE*, 6 2020, pp. 745–750.
- [233] R. N. Toma, F. H. Toma, and J.-M. Kim, "Comparative analysis of continuous wavelet transforms on vibration signal in bearing fault diagnosis of induction motor," *IEEE*, 9 2021, pp. 1–4.
- [234] A. Belkhou, A. Achmamad, and A. Jbari, "Classification and diagnosis of myopathy emg signals using the continuous wavelet transform," *IEEE*, 4 2019, pp. 1–4.
- [235] R. Singh, S. Gangwar, S. Bartwal, and V. Tomar, "Wavelet-based feature extraction and rcnn for ecg signal classification," *IEEE*, 6 2023, pp. 1–6.
- [236] J. Fu, X. Xu, and Y. Chen, "Feature selection and optimization method based on cwt," *IEEE*, 10 2017, pp. 1–4.
- [237] N. A. Aditiya, Z. Darojah, D. R. Sanggar, and M. R. Dharmawan, "Fault diagnosis system of rotating machines using continuous wavelet transform and artificial neural network," *IEEE*, 9 2017, pp. 173–176.
- [238] S. Tang, B. C. Khoo, Y. Zhu, K. M. Lim, and S. Yuan, "A light deep adaptive framework toward fault diagnosis of a hydraulic piston pump," *Applied Acoustics*, vol. 217, p. 109807, 2 2024.
- [239] J. Dong, H. Jiang, D. Su, Y. Gao, T. Chen, and K. Sheng, "Transfer learning rolling bearing fault diagnosis model based on deep feature decomposition and class-level alignment," *Measurement Science and Technology*, vol. 35, p. 046006, 4 2024.
- [240] C. Bhagavatula, S. Venugopalan, R. Blue, R. Friedman, M. O. Griofa, M. Savvides, and B. V. K. V. Kumar, "Biometric identification of cardiosynchronous waveforms utilizing person specific continuous and discrete wavelet transform features," *IEEE*, 8 2012, pp. 4311–4314.
- [241] O. Almanza-Conejo, D. L. Almanza-Ojeda, J. L. Contreras-Hernandez, and M. A. Ibarra-Manzano, "Emotion recognition in eeg signals using the continuous wavelet transform and cnns," *Neural Computing and Applications*, vol. 35, pp. 1409–1422, 1 2023.
- [242] R. Dwyer, "Detection of non-gaussian signals by frequency domain kurtosis estimation," *Institute of Electrical and Electronics Engineers*, pp. 607–610.
- [243] J. Antoni, "The spectral kurtosis: a useful tool for characterising non-stationary signals," *Mechanical Systems and Signal Processing*, vol. 20, pp. 282–307, 2 2006.
- [244] Y. Wang, J. Xiang, R. Markert, and M. Liang, "Spectral kurtosis for fault detection, diagnosis and prognostics of rotating machines: A review with applications," *Mechanical Systems and Signal Processing*, vol. 66–67, pp. 679–698, 1 2016.
- [245] S. Liu, S. Hou, K. He, and W. Yang, "L-kurtosis and its application for fault detection of rolling element bearings," *Measurement*, vol. 116, pp. 523–532, 2 2018.
- [246] J. Wodecki, "Time-varying spectral kurtosis: Generalization of spectral kurtosis for local damage detection in rotating machines under time-varying operating conditions," *Sensors*, vol. 21, p. 3590, 5 2021.
- [247] I. Ahmad, C. Yao, L. Li, Y. Chen, Z. Liu, I. Ullah, M. Shabaz, X. Wang, K. Huang, G. Li, G. Zhao, O. W. Samuel, and S. Chen, "An efficient feature selection and explainable classification method for eeg-based epileptic seizure detection," *Journal of Information Security and Applications*, vol. 80, p. 103654, 2 2024.
- [248] Y. Liu, F. Yu, and J. Gao, "Sparse representation-based classification for rolling bearing fault diagnosis," *IEEE*, 11 2019, pp. 3058–3061.
- [249] J. C. Chan, H. Ma, and T. K. Saha, "Partial discharge pattern recognition using multiscale feature extraction and support vector machine," *IEEE*, 2013, pp. 1–5.
- [250] S. Sekkate, M. Khalil, and A. Adib, "Fusing wavelet and short-term features for speaker identification in noisy environment," *IEEE*, 4 2018, pp. 1–8.
- [251] M. Tibdewal, H. Dey, M. Mahadevappa, A. Ray, and M. Malokar, "Wavelet transform based multiple features extraction for detection of epileptic/ non-epileptic multichannel eeg," *IEEE*, 2016, pp. 1648–1653.
- [252] Y. Xie and T. Zhang, "Feature extraction based on dwt and cnn for rotating machinery fault diagnosis," *IEEE*, 5 2017, pp. 3861–3866.
- [253] F. J. Floresca, C. K. Tobias, and C. Ostia, "Naïve bayes classification technique for brushless dc motor fault diagnosis with discrete wavelet transform feature extraction," *IEEE*, 3 2022, pp. 140–144.
- [254] S. Katiyar and I. Sharma, "Emg signal decomposition methods for physical actions: A comparative analysis," *IEEE*, 9 2023, pp. 1–5.
- [255] C. M. Ugwu, C. P. Mukamakuza, and E. Tuyishimire, "Ecg-signals-based heartbeat classification: A comparative study of artificial neural network and support vector machine classifiers," *IEEE*, 1 2024, pp. 000 217–000 222.
- [256] S. Guo, S. Lin, and Z. Huang, "Feature extraction of p300s in eeg signal with discrete wavelet transform and fisher criterion," *IEEE*, 10 2015, pp. 200–204.
- [257] H. U. Amin, A. S. Malik, R. F. Ahmad, N. Badruddin, N. Kamel, M. Hussain, and W.-T. Chooi, "Feature extraction and classification for eeg signals using wavelet transform and machine learning techniques," *Australasian Physical Engineering Sciences in Medicine*, vol. 38, pp. 139–149, 3 2015.
- [258] A. Prochazka, J. Kukal, and O. Vysata, "Wavelet transform use for feature extraction and eeg signal segments classification," *IEEE*, 3 2008, pp. 719–722.
- [259] Q. Zhao and L. Zhang, "Ecg feature extraction and classification using wavelet transform and support vector machines," *IEEE*, pp. 1089–1092.
- [260] S. Y. Shah, H. Larijani, R. M. Gibson, and D. Liarokapis, "Epileptic seizure classification based on random neural networks using discrete wavelet transform for electroencephalogram signal decomposition," *Applied Sciences*, vol. 14, p. 599, 1 2024.
- [261] Q. Gong, K. Peng, Q. Gao, L. Feng, and C. Xiao, "Series arc fault identification method based on wavelet transform and feature values decomposition fusion dnn," *Electric Power Systems Research*, vol. 221, p. 109391, 8 2023.
- [262] J. Xi and L. Lin, "Multidimensional feature extraction based on vibration signals of rolling bearings," *IEEE*, 5 2013, pp. 1093–1096.
- [263] L.-Y. Zhao, L. Wang, and R.-Q. Yan, "Rolling bearing fault diagnosis based on wavelet packet decomposition and multi-scale permutation entropy," *Entropy*, vol. 17, pp. 6447–6461, 9 2015.
- [264] M. J. Gómez, P. Marklund, D. Strombergsson, C. Castejón, and J. C. García-Prada, "Analysis of vibration signals of drivetrain failures in wind turbines for condition monitoring," *Experimental Techniques*, vol. 45, pp. 1–12, 2 2021.
- [265] X. Ding, L. Lin, D. He, L. Wang, W. Huang, and Y. Shao, "A weight multinet architecture for bearing fault classification under complex speed conditions," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021.
- [266] I. Attoui, N. Boutasseta, N. Fergani, B. Oudjani, and A. Deliou, "Vibration-based bearing fault diagnosis by an integrated dwt-fft approach and an adaptive neuro-fuzzy inference system," *IEEE*, 5 2015, pp. 1–6.
- [267] S. Aburakhia and A. Shami, "Sb-pdm: A tool for predictive maintenance of rolling bearings based on limited labeled data," *Software Impacts*, vol. 16, p. 100503, 5 2023.
- [268] Z. Mezni, C. Delpha, D. Diallo, and A. Braham, "A comparative study for ball bearing fault classification using kernel-svm with kullback leibler divergence selected features," *IEEE*, 10 2019, pp. 6969–6974.

- [269] F. Wang, G. Deng, C. Liu, W. Su, Q. Han, and H. Li, "A deep feature extraction method for bearing fault diagnosis based on empirical mode decomposition and kernel function," *Advances in Mechanical Engineering*, vol. 10, p. 168781401879825, 9 2018.
- [270] J. Shen, X. Zhang, G. Wang, Z. Ding, and B. Hu, "An improved empirical mode decomposition of electroencephalogram signals for depression detection," *IEEE Transactions on Affective Computing*, vol. 13, pp. 262–271, 1 2022.
- [271] L. Zhan, F. Ma, J. Zhang, C. Li, Z. Li, and T. Wang, "Fault feature extraction and diagnosis of rolling bearings based on enhanced complementary empirical mode decomposition with adaptive noise and statistical time-domain features," *Sensors*, vol. 19, p. 4047, 9 2019.
- [272] A. Ahmed, Y. Serrestou, K. Raoof, and J.-F. Diouris, "Empirical mode decomposition-based feature extraction for environmental sound classification," *Sensors*, vol. 22, p. 7717, 10 2022.
- [273] H. Shang, K. Lo, and F. Li, "Partial discharge feature extraction based on ensemble empirical mode decomposition and sample entropy," *Entropy*, vol. 19, p. 439, 8 2017.
- [274] S. Nalband, A. Prince, and A. Agrawal, "Entropy-based feature extraction and classification of vibroarthrographic signal using complete ensemble empirical mode decomposition with adaptive noise," *IET Science, Measurement Technology*, vol. 12, pp. 350–359, 5 2018.
- [275] M. Van and H. Kang, "Bearing-fault diagnosis using non-local means algorithm and empirical mode decomposition-based feature extraction and two-stage feature selection," *IET Science, Measurement Technology*, vol. 9, pp. 671–680, 9 2015.
- [276] Y. Li, J. hao Luo, Q. yun Dai, J. K. Eshraghian, B. W.-K. Ling, C. yan Zheng, and X. li Wang, "A deep learning approach to cardiovascular disease classification using empirical mode decomposition for ecg feature extraction," *Biomedical Signal Processing and Control*, vol. 79, p. 104188, 1 2023.
- [277] N. M. El-Kafrawy, D. Hegazy, and M. F. Tolba, *Features Extraction and Classification of EEG Signals Using Empirical Mode Decomposition and Support Vector Machine*, 2014, pp. 189–198.
- [278] R. Jia, F. Ma, H. Wu, X. Luo, and X. Ma, "Coupling fault feature extraction method based on bivariate empirical mode decomposition and full spectrum for rotating machinery," *Mathematical Problems in Engineering*, vol. 2018, pp. 1–10, 2018.
- [279] H. Yang, T. Ning, B. Zhang, X. Yin, and Z. Gao, "An adaptive denoising fault feature extraction method based on ensemble empirical mode decomposition and the correlation coefficient," *Advances in Mechanical Engineering*, vol. 9, p. 168781401769644, 4 2017.
- [280] A. Mert, "Ecg feature extraction based on the bandwidth properties of variational mode decomposition," *Physiological Measurement*, vol. 37, pp. 530–543, 4 2016.
- [281] P. K. Sahu and R. N. Rai, "Fault diagnosis of rolling bearing based on an improved denoising technique using complete ensemble empirical mode decomposition and adaptive thresholding method," *Journal of Vibration Engineering Technologies*, vol. 11, pp. 513–535, 2 2023.
- [282] Y. Yang, H. Liu, L. Han, and P. Gao, "A feature extraction method using vmd and improved envelope spectrum entropy for rolling bearing fault diagnosis," *IEEE Sensors Journal*, vol. 23, pp. 3848–3858, 2 2023.
- [283] P. Poludasu, S. J. Bandaru, A. K. Paleru, S. Venkatadas, and P. A. K., "Biometric authentication using ecg signals," *IEEE*, 5 2023, pp. 1–6.
- [284] M. Feldman, "Hilbert transform in vibration analysis," *Mechanical Systems and Signal Processing*, vol. 25, pp. 735–802, 4 2011.
- [285] G. Soto and S. Torres, "Feature extraction based on circular summary statistics in ecg signal classification," *IEEE*, 11 2009, pp. 142–144.
- [286] M. An-bo, Z. Wei, H. Hao-hao, X. Xin-yu, and C. Fu-lei, "Incipient fault feature extraction of main bearing by iterative squared envelope analysis," *IEEE*, 7 2017, pp. 1–6.
- [287] C. Liu, J. Wang, L. Liang, B. Niu, and D. Liu, "Periodic impact feature extraction for bearings fault based on envelope correlation kurtosis of singular value components," *IEEE*, 12 2022, pp. 359–363.
- [288] L. Han, J. Yan, S. Fan, M. Xu, Z. Liu, Y. Geng, and C. Guan, "Feature extraction of uhf pd signals based on diode envelope detection and linear discriminant analysis," *IEEE*, 10 2019, pp. 598–602.
- [289] D. Chowdhury, M. Hasan, and M. Z. R. Khan, "Statistical features extraction from current envelopes for non-intrusive appliance load monitoring," *IEEE*, 3 2020, pp. 1–5.
- [290] P. Zuo-wei, A. Hong-wen, L. Shuang-yin, and L. Yi-bing, "Amplitude envelope analysis for feature extraction of power plant blower bearing failure," *IEEE*, 5 2014, pp. 2821–2824.
- [291] S. Marquez-Figueroa, Y. S. Shmaliy, and O. Ibarra-Manzano, "Removing artifacts and optimal features extraction from emg envelope," *IEEE*, 7 2020, pp. 90–95.
- [292] L. Zhou, F. Yang, Y. Zhang, and S. Hou, "Feature extraction and classification of partial discharge signal in gis based on hilbert transform," *IEEE*, 10 2021, pp. 208–213.
- [293] M. R. Mehrjou, N. Mariun, M. Karami, N. Mison, and M. A. M. Radzi, "Broken rotor bar detection in ls-pmsms based on statistical features analysis of start-up current envelope," *IEEE*, 11 2015, pp. 1–6.
- [294] T. Deghboudj, A. Tlemçani, and A. Derghal, "Identifying broken rotor bars in squirrel cage induction motors: Examination of statistical characteristics derived from start-up current envelope," *IEEE*, 11 2023, pp. 1–7.
- [295] D.-H. Lee, C. Hong, W.-B. Jeong, and S. Ahn, "Time-frequency envelope analysis for fault detection of rotating machinery signals with impulsive noise," *Applied Sciences*, vol. 11, p. 5373, 6 2021.
- [296] S. Kim, D. An, and J.-H. Choi, "Diagnostics 101: A tutorial for fault diagnostics of rolling element bearing using envelope analysis in matlab," *Applied Sciences*, vol. 10, p. 7302, 10 2020.
- [297] Q. Hu, A. Qin, Q. Zhang, J. He, and G. Sun, "Fault diagnosis based on weighted extreme learning machine with wavelet packet decomposition and kpca," *IEEE Sensors Journal*, vol. 18, pp. 8472–8483, 10 2018.
- [298] L. E. Crooks, *Noise reduction techniques in electronic systems (2nd ed.)*, Henry W. Ott. Wiley-Interscience, New York, 1988, 6 1989, vol. 10.
- [299] K. E. Barner and G. R. Arce, *21 Order-statistic filtering and smoothing of time-series: Part II*, 1998, pp. 555–602.
- [300] Z. Hao, Z. Cui, S. Yue, and H. Wang, "Impulsive noise removal in multi-frequency phase sensitive demodulation for electrical impedance tomography," *IEEE*, 5 2018, pp. 1–6.
- [301] D. Malathi, P. Naveen, V. Sahasra, and S. Rithika, "Vlsi implementation of median filter for noise removal in emg signal," *IEEE*, 3 2023, pp. 422–426.
- [302] W. Wierzchowski and M. Pawelczyk, "Median filtering approach for active control of impulsive noise," *IEEE*, 9 2014, pp. 322–326.
- [303] T. Pander, "A suppression of an impulsive noise in ecg signal processing," *IEEE*, pp. 596–599.
- [304] J. Wang, J. Li, S. Yan, W. Shi, X. Yang, Y. Guo, and T. A. Gulliver, "A novel underwater acoustic signal denoising algorithm for gaussian/non-gaussian impulsive noise," *IEEE Transactions on Vehicular Technology*, vol. 70, pp. 429–445, 1 2021.
- [305] P. B. Swamy, R. S. Hallikar, and M. U. Kumari, "Enhancement of speech signals corrupted by impulsive noise using wavelets and adaptive median filtering," *IEEE*, 9 2013, pp. 1–6.
- [306] G. Tan, S. Yan, and B. Yang, "Impulsive noise suppression and concatenated code for ofdm underwater acoustic communications," *IEEE*, 10 2022, pp. 1–6.
- [307] R. Chaturvedi and Y. Yadav, "Application of moving average filter in ecg denoising," *World Academics Journal of Engineering Sciences*, p. 2004, 2014.
- [308] V. Pandey and V. K. Giri, "High frequency noise removal from ecg using moving average filters," *IEEE*, 3 2016, pp. 191–195.
- [309] A. K. Tanji, M. A. G. de Brito, M. G. Alves, R. C. Garcia, G.-L. Chen, and N. R. N. Ama, "Improved noise cancelling algorithm for electrocardiogram based on moving average adaptive filter," *Electronics*, vol. 10, p. 2366, 9 2021.
- [310] S. K. Salih, S. A. Aljunid, S. M. Aljunid, and O. Maskon, "Adaptive filtering approach for denoising electrocardiogram signal using moving average filter," *Journal of Medical Imaging and Health Informatics*, vol. 5, pp. 1065–1069, 9 2015.
- [311] Z. Shan, J. Yang, M. A. F. Sanjuán, C. Wu, and H. Liu, "A novel adaptive moving average method for signal denoising in strong noise background," *The European Physical Journal Plus*, vol. 137, p. 50, 1 2022.
- [312] Y. Liu, B. Dang, Y. Li, H. Lin, and H. Ma, "Applications of savitzky-golay filter for seismic random noise reduction," *Acta Geophysica*, vol. 64, pp. 101–124, 2 2016.
- [313] N. S. Haider, R. Periyasamy, D. Joshi, and B. K. Singh, "Savitzky-golay filter for denoising lung sound," *Brazilian Archives of Biology and Technology*, vol. 61, 2018.
- [314] J. Seo, H. Ma, and T. K. Saha, "On savitzky-golay filtering for online condition monitoring of transformer on-load tap changer," *IEEE Transactions on Power Delivery*, vol. 33, pp. 1689–1698, 8 2018.
- [315] A. S. Vaishnavi, T. Greeshma, R. P. Teja, T. Padma, and C. U. Kumari, "Noise removal from ecg signal using lms adaptive filter implementation in xilinx system generator," *IEEE*, 5 2023, pp. 1488–1491.
- [316] R. Shaddeli, N. Yazdanjue, S. Ebadollahi, M. M. Saberi, and B. Gill, "Noise removal from ecg signals by adaptive filter based on variable step size lms using evolutionary algorithms," *IEEE*, 9 2021, pp. 1–7.

- [317] A. Mirza, S. Kabir, S. Ayub, and S. A. Sheikh, "Impulsive noise cancellation of ecg signal based on ssrls," *Procedia Computer Science*, vol. 62, pp. 196–202, 2015.
- [318] S. Dixit and D. Nagaria, "Lms adaptive filters for noise cancellation: A review," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, p. 2520, 10 2017.
- [319] S. Yadav, S. K. Saha, and R. Kar, "An application of the kalman filter for eeg/erp signal enhancement with the autoregressive realisation," *Biomedical Signal Processing and Control*, vol. 86, p. 105213, 9 2023.
- [320] R. I. Alfian, A. Ma'arif, and S. Sunardi, "Noise reduction in the accelerometer and gyroscope sensor with the kalman filter algorithm," *Journal of Robotics and Control (JRC)*, vol. 2, 2021.
- [321] K. H. Eom, S. J. Lee, Y. S. Kyung, C. W. Lee, M. C. Kim, and K. K. Jung, "Improved kalman filter method for measurement noise reduction in multi sensor rfid systems," *Sensors*, vol. 11, pp. 10266–10282, 10 2011.
- [322] S. So, A. E. W. George, R. Ghosh, and K. K. Paliwal, "Kalman filter with sensitivity tuning for improved noise reduction in speech," *Circuits, Systems, and Signal Processing*, vol. 36, pp. 1476–1492, 4 2017.
- [323] Y. Kim and H. Bang, *Introduction to Kalman Filter and Its Applications*. IntechOpen, 5 2019.
- [324] M. Khodarahmi and V. Maihami, "A review on kalman filter models," *Archives of Computational Methods in Engineering*, vol. 30, pp. 727–747, 1 2023.
- [325] N. Golyandina, A. Korobeynikov, and A. Zhigljavsky, *Singular Spectrum Analysis with R*. Springer Berlin Heidelberg, 2018.
- [326] S. K. Mukhopadhyay and S. Krishnan, "A singular spectrum analysis-based model-free electrocardiogram denoising technique," *Computer Methods and Programs in Biomedicine*, vol. 188, p. 105304, 5 2020.
- [327] O. Traore, L. Pantera, N. Favretto-Cristini, P. Cristini, S. Viguier-Pla, and P. Vieu, "Structure analysis and denoising using singular spectrum analysis: Application to acoustic emission signals from nuclear safety experiments," *Measurement*, vol. 104, pp. 78–88, 7 2017.
- [328] W. Kuang, S. Wang, Y. Lai, and W.-K. Ling, "Efficient and adaptive signal denoising based on multistage singular spectrum analysis," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–20, 2021.
- [329] M. Saeed, C. C. Took, and S. R. Alty, "Efficient algorithm to implement sliding singular spectrum analysis with application to biomedical signal denoising," *IEEE*, 5 2020, pp. 1026–1029.
- [330] M. Saeed and S. R. Alty, "Removal of electrocardiogram interference from diaphragmatic electromyogram signals using sliding singular spectrum analysis," *IEEE*, 7 2018, pp. 1238–1241.
- [331] J. Barrios-Muriel, F. Romero, F. J. Alonso, and K. Gianikellis, "A simple ssa-based de-noising technique to remove ecg interference in emg signals," *Biomedical Signal Processing and Control*, vol. 30, pp. 117–126, 9 2016.
- [332] I. W. Selesnick, H. L. Graber, D. S. Pfeil, and R. L. Barbour, "Simultaneous low-pass filtering and total variation denoising," *IEEE Transactions on Signal Processing*, vol. 62, pp. 1109–1124, 3 2014.
- [333] P. Bhaskar and M. Uplane, "High frequency electromyogram noise removal from electrocardiogram using fir low pass filter based on fpga," *Procedia Technology*, vol. 25, pp. 497–504, 2016.
- [334] M. T. Almalchy, V. Ciobanu, and N. Popescu, "Noise removal from ecg signal based on filtering techniques," *IEEE*, 5 2019, pp. 176–181.
- [335] J. van Driel, C. N. Olivers, and J. J. Fahrenfort, "High-pass filtering artifacts in multivariate classification of neural time series data," *Journal of Neuroscience Methods*, vol. 352, p. 109080, 3 2021.
- [336] I. Winkler, S. Debener, K.-R. Muller, and M. Tangermann, "On the influence of high-pass filtering on ica-based artifact reduction in eeg-erp," *IEEE*, 8 2015, pp. 4101–4105.
- [337] N. Economou, "Time-varying band-pass filtering gpr data by self-inverse filtering," *Near Surface Geophysics*, vol. 14, pp. 207–217, 4 2016.
- [338] L. Hao, S. Cao, P. Zhou, L. Chen, Y. Zhang, K. Li, D. Xie, and Y. Geng, "Denoising method based on spectral subtraction in time-frequency domain," *Advances in Civil Engineering*, vol. 2021, pp. 1–12, 7 2021.
- [339] A. Slimane and A. Zaid, "Real-time fast fourier transform-based notch filter for single-frequency noise cancellation: Application to electrocardiogram signal denoising," *Journal of Medical Signals Sensors*, vol. 11, p. 52, 2021.
- [340] U. Biswas and M. Maniruzzaman, "Removing power line interference from ecg signal using adaptive filter and notch filter," *IEEE*, 4 2014, pp. 1–4.
- [341] W. Wang, J. Li, and S. Lu, "Application of signal denoising technology based on improved spectral subtraction in arc fault detection," *Electronics*, vol. 12, p. 3147, 7 2023.
- [342] R. Hussein, K. B. Shaban, and A. H. El-Hag, "Denoising different types of acoustic partial discharge signals using power spectral subtraction," *High Voltage*, vol. 3, pp. 44–50, 3 2018.
- [343] M. Karam, H. F. Khazaal, H. Aglan, and C. Cole, "Noise removal in speech processing using spectral subtraction," *Journal of Signal and Information Processing*, vol. 05, pp. 32–41, 2014.
- [344] L. C. Yong, K. R. W. Jee, and K. T. Mei, "Evaluation of partial discharge denoising using power spectral subtraction," *IEEE*, 9 2020, pp. 86–89.
- [345] Q. Liu, Y. Yu, B. S. Han, and W. Zhou, "An improved spectral subtraction method for eliminating additive noise in condition monitoring system using fiber bragg grating sensors," *Sensors*, vol. 24, p. 443, 1 2024.
- [346] R. Hussein, K. B. Shaban, and A. H. El-Hag, "Acoustic partial discharge signal denoising using power spectral subtraction," *IEEE*, 10 2015, pp. 330–333.
- [347] T.-K. La and S.-D. Kwon, "Reducing ambient sensor noise in wind tunnel tests using spectral subtraction method," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 244, p. 105631, 1 2024.
- [348] S. V. Vaseghi, *Spectral Subtraction*. Vieweg+Teubner Verlag, 1996, pp. 242–260.
- [349] C. P. Dautov and M. S. Ozerdem, "Wavelet transform and signal denoising using wavelet method," *IEEE*, 5 2018, pp. 1–4.
- [350] K. Yu, L. Feng, Y. Chen, M. Wu, Y. Zhang, P. Zhu, W. Chen, Q. Wu, and J. Hao, "Accurate wavelet thresholding method for ecg signals," *Computers in Biology and Medicine*, vol. 169, p. 107835, 2 2024.
- [351] A. Halidou, Y. Mohamadou, A. A. A. Ari, and E. J. G. Zacko, "Review of wavelet denoising algorithms," *Multimedia Tools and Applications*, vol. 82, pp. 41539–41569, 11 2023.
- [352] A. Golmohammadi, N. Hasheminejad, D. Hernando, S. Vanlanduit, and W. V. den bergh, "Performance assessment of discrete wavelet transform for de-noising of fbg sensors signals embedded in asphalt pavement," *Optical Fiber Technology*, vol. 82, p. 103596, 1 2024.
- [353] M. Grobelaar, S. Phadikar, E. Ghaderpour, A. F. Struck, N. Sinha, R. Ghosh, and M. Z. I. Ahmed, "A survey on denoising techniques of electroencephalogram signals using wavelet transform," *Signals*, vol. 3, pp. 577–586, 8 2022.
- [354] G. Baldazzi, G. Solinas, J. D. Valle, M. Barbaro, S. Micera, L. Raffo, and D. Pani, "Systematic analysis of wavelet denoising methods for neural signal processing," *Journal of Neural Engineering*, vol. 17, p. 066016, 12 2020.
- [355] G. Chen, W. Xie, and Y. Zhao, "Wavelet-based denoising: A brief review," *IEEE*, 6 2013, pp. 570–574.
- [356] D. Baleanu, *Advances in Wavelet Theory and Their Applications in Engineering, Physics and Technology*, D. Baleanu, Ed. InTech, 4 2012.
- [357] C. Ouyang, L. Cai, B. Liu, and T. Zhang, "An improved wavelet threshold denoising approach for surface electromyography signal," *EURASIP Journal on Advances in Signal Processing*, vol. 2023, p. 108, 10 2023.
- [358] M. Srivastava, C. L. Anderson, and J. H. Freed, "A new wavelet denoising method for selecting decomposition levels and noise thresholds," *IEEE Access*, vol. 4, pp. 3862–3877, 2016.
- [359] C. He, J. Xing, J. Li, Q. Yang, and R. Wang, "A new wavelet threshold determination method considering interscale correlation in signal denoising," *Mathematical Problems in Engineering*, vol. 2015, pp. 1–9, 2015.
- [360] R. Hussein, K. B. Shaban, and A. H. El-Hag, "Energy conservation-based thresholding for effective wavelet denoising of partial discharge signals," *IET Science, Measurement Technology*, vol. 10, pp. 813–822, 10 2016.
- [361] A. Rasti-Meymandi and A. Ghaffari, "A deep learning-based framework for ecg signal denoising based on stacked cardiac cycle tensor," *Biomedical Signal Processing and Control*, vol. 71, p. 103275, 1 2022.
- [362] S. Pongponsoi and X.-H. Yu, "An adaptive filtering approach for electrocardiogram (ecg) signal noise reduction using neural networks," *Neurocomputing*, vol. 117, pp. 206–213, 10 2013.
- [363] Z. Gao, S. Zhang, J. Cai, L. Hong, and J. Zheng, "Research on deep convolutional neural network time-frequency domain seismic signal denoising combined with residual dense blocks," *Frontiers in Earth Science*, vol. 9, 7 2021.
- [364] S. Yu, J. Ma, and W. Wang, "Deep learning for denoising," *GEO-PHYSICS*, vol. 84, pp. V333–V350, 11 2019.

- [365] E. Almazrouei, G. Gianini, N. Almoosa, and E. Damiani, "A deep learning approach to radio signal denoising," *IEEE*, 4 2019, pp. 1–8.
- [366] C. T. Arsene, R. Hankins, and H. Yin, "Deep learning models for denoising ecg signals," *IEEE*, 9 2019, pp. 1–5.
- [367] L. Wang, W. Zheng, X. Ma, and S. Lin, "Denoising speech based on deep learning and wavelet decomposition," *Scientific Programming*, vol. 2021, pp. 1–10, 7 2021.
- [368] F. L. Mvuh, C. O. V. E. Ko'a, and B. Bodo, "Multichannel high noise level ecg denoising based on adversarial deep learning," *Scientific Reports*, vol. 14, p. 801, 1 2024.
- [369] Y. Jin, C. Qin, J. Liu, Y. Liu, Z. Li, and C. Liu, "A novel deep wavelet convolutional neural network for actual ecg signal denoising," *Biomedical Signal Processing and Control*, vol. 87, p. 105480, 1 2024.
- [370] X. Li, Q. Qi, Y. Yang, P. Duan, and Z. Cao, "Removing abnormal environmental noise in nodal land seismic data using deep learning," *GEOPHYSICS*, vol. 89, pp. WA143–WA156, 1 2024.
- [371] P. Xie, L. Zhang, M. Li, S. F. S. Lau, and J. Huang, "Elevator vibration signal denoising by deep residual u-net," *Measurement*, vol. 225, p. 113976, 2 2024.
- [372] S. N. Ali, S. B. Shuvo, M. I. S. Al-Manzo, A. Hasan, and T. Hasan, "An end-to-end deep learning framework for real-time denoising of heart sounds for cardiac disease detection in unseen noise," *IEEE Access*, vol. 11, pp. 87887–87901, 2023.
- [373] A. Zhou, W. Zhang, X. Li, G. Xu, B. Zhang, Y. Ma, and J. Song, "A novel noise-aware deep learning model for underwater acoustic denoising," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–13, 2023.
- [374] T. Lin, M. Wei, and Y. Zhang, "Deep learning for denoising: An attempt to recover the effective magnetic resonance sounding signal in the presence of high level noise," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2022.
- [375] J. Wang, J. J. Sohn, Y. Lei, W. Nie, J. Zhou, S. Avery, T. Liu, and X. Yang, "Deep learning-based protoacoustic signal denoising for proton range verification," *Biomedical Physics Engineering Express*, vol. 9, p. 045006, 7 2023.
- [376] B. Porr, S. Daryanavard, L. M. Bohollo, H. Cowan, and R. Dahiya, "Real-time noise cancellation with deep learning," *PLOS ONE*, vol. 17, p. e0277974, 11 2022.
- [377] P. Manimegalai, D. Pamela, P. A. Michael, K. G. J. Nigel, P. K. Stanley, and J. R. Jenny, "Comparison on denoising of electro cardiogram signal using deep learning techniques," 2023, p. 060042.
- [378] S. Aburakhia and A. Shami, "On the peak-to-average power ratio of vibration signals: Analysis and signal companding for an efficient remote vibration-based condition monitoring," 10 2023.
- [379] R. Ma, S. Li, B. Zhang, and Z. Li, "Generative adaptive convolutions for real-world noisy image denoising," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 1935–1943, 6 2022.
- [380] P. Yan, A. Abdulkadir, P.-P. Luley, M. Rosenthal, G. A. Schatte, B. F. Grewe, and T. Stadelmann, "A comprehensive survey of deep transfer learning for anomaly detection in industrial time series: Methods, applications, and directions," *IEEE Access*, vol. 12, pp. 3768–3789, 2024.
- [381] Y. Ma, S. Chen, S. Ermon, and D. B. Lobell, "Transfer learning in environmental remote sensing," *Remote Sensing of Environment*, vol. 301, p. 113924, 2 2024.
- [382] G. V. D. Vyver, Z. Liu, K. Dolui, D. Hughes, and S. Michiels, "Adapted spectrogram transformer for unsupervised cross-domain acoustic anomaly detection," *IEEE*, 11 2022, pp. 890–896.
- [383] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang, "A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals," *Sensors*, vol. 17, p. 425, 2 2017.
- [384] M.-A. Tnani, P. Subarnaduti, and K. Diepold, "Efficient feature learning approach for raw industrial vibration data using two-stage learning framework," *Sensors*, vol. 22, p. 4813, 6 2022.
- [385] D. Wang, M. Zhang, Y. Xu, W. Lu, J. Yang, and T. Zhang, "Metric-based meta-learning model for few-shot fault diagnosis under multiple limited data conditions," *Mechanical Systems and Signal Processing*, vol. 155, p. 107510, 6 2021.
- [386] A. Zhang, S. Li, Y. Cui, W. Yang, R. Dong, and J. Hu, "Limited data rolling bearing fault diagnosis with few-shot learning," *IEEE Access*, vol. 7, pp. 110895–110904, 2019.
- [387] L. Sun, M. Zhang, B. Wang, and P. Tiwari, "Few-shot class-incremental learning for medical time series classification," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–11, 2024.
- [388] S. Aburakhia, T. Tayeh, R. Myers, and A. Shami, "A transfer learning framework for anomaly detection using model of normality," in *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2020, pp. 0055–0061.
- [389] Q. Ma, Z. Liu, Z. Zheng, Z. Huang, S. Zhu, Z. Yu, and J. T. Kwok, "A survey on time-series pre-trained models," 5 2023.
- [390] K. Kashiparekh, J. Narwariya, P. Malhotra, L. Vig, and G. Shroff, "ConvtimeNet: A pre-trained deep convolutional neural network for time series classification," *IEEE*, 7 2019, pp. 1–8.
- [391] P. Malhotra, V. TV, L. Vig, P. Agarwal, and G. Shroff, "Timenet: Pre-trained deep recurrent neural network for time series classification," 6 2017.
- [392] M. A. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Processing*, vol. 99, pp. 215–249, 6 2014.
- [393] B. Unnikrishnan, P. R. Singh, X. Yang, and M. C. H. Chua, "Semi-supervised and unsupervised methods for heart sounds classification in restricted data environments," 6 2020.
- [394] S. Das, S. Pal, and M. Mitra, "Acoustic feature based unsupervised approach of heart sound event detection," *Computers in Biology and Medicine*, vol. 126, p. 103990, 11 2020.
- [395] I. Attoui, "Novel fast and automatic condition monitoring strategy based on small amount of labeled data," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, pp. 1266–1275, 2 2022.
- [396] I. Attoui, N. Boutasseta, and N. Fergani, "Novel machinery monitoring strategy based on time-frequency domain similarity measurement with limited labeled data," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–8, 2021.
- [397] S. Aburakhia, T. Tayeh, R. Myers, and A. Shami, "Similarity-based predictive maintenance framework for rotating machinery," *IEEE*, 12 2022, pp. 1–6.
- [398] P. Arun, S. A. Lincon, and N. Prabhakaran, "An automated method for the analysis of bearing vibration based on spectrogram pattern matching," *Journal of Applied Research and Technology*, vol. 17, 10 2019.
- [399] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, pp. 279–283, 3 2017.
- [400] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, pp. 1469–1477, 9 2015.
- [401] F. Xie, H. Wen, J. Wu, W. Hou, H. Song, T. Zhang, R. Liao, and Y. Jiang, "Data augmentation for radio frequency fingerprinting via pseudo-random integration," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, pp. 276–286, 6 2020.
- [402] T. Tayeh, S. Aburakhia, R. Myers, and A. Shami, "Distance-based anomaly detection for industrial surfaces using triplet networks," in *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2020, pp. 0372–0377.
- [403] G. Zhou, Y. Chen, and C. Chien, "On the analysis of data augmentation methods for spectral imaged based heart sound classification using convolutional neural networks," *BMC Medical Informatics and Decision Making*, vol. 22, p. 226, 8 2022.
- [404] H.-C. Chu, Y.-L. Zhang, and H.-C. Chiang, "A cnn sound classification mechanism using data augmentation," *Sensors*, vol. 23, p. 6972, 8 2023.
- [405] Z. Mushtaq, S.-F. Su, and Q.-V. Tran, "Spectral images based environmental sound classification using cnn with meaningful data augmentation," *Applied Acoustics*, vol. 172, p. 107581, 1 2021.
- [406] D. Cabrera, F. Sancho, J. Long, R.-V. Sanchez, S. Zhang, M. Cerrada, and C. Li, "Generative adversarial networks selection approach for extremely imbalanced fault diagnosis of reciprocating machinery," *IEEE Access*, vol. 7, pp. 70643–70653, 2019.
- [407] J. Liu, F. Qu, X. Hong, and H. Zhang, "A small-sample wind turbine fault detection method with synthetic fault data using generative adversarial nets," *IEEE Transactions on Industrial Informatics*, vol. 15, pp. 3877–3888, 7 2019.
- [408] W. Mao, Y. Liu, L. Ding, and Y. Li, "Imbalanced fault diagnosis of rolling bearing based on generative adversarial network: A comparative study," *IEEE Access*, vol. 7, pp. 9515–9530, 2019.
- [409] M. Zareapoor, P. Shamsolmoali, and J. Yang, "Oversampling adversarial network for class-imbalanced fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 149, p. 107175, 2 2021.
- [410] H. Shao, H. Jiang, Y. Lin, and X. Li, "A novel method for intelligent fault diagnosis of rolling bearings using ensemble deep auto-encoders," *Mechanical Systems and Signal Processing*, vol. 102, pp. 278–297, 3 2018.
- [411] V. Bui, T. L. Pham, H. Nguyen, and Y. M. Jang, "Data augmentation using generative adversarial network for automatic machine fault detection based on vibration signals," *Applied Sciences*, vol. 11, p. 2166, 3 2021.

- [412] S. Narieda, "Downsampling technique for computational complexity reduction in narrowband wireless communication systems," *Electronics Letters*, vol. 48, p. 241, 2012.
- [413] S.-H. Hsieh, C.-S. Lu, and S.-C. Pei, "Sparse fast fourier transform by downsampling," *IEEE*, 5 2013, pp. 5637–5641.
- [414] D. Goyal, S. S. Dhami, and B. S. Pabla, "Non-contact fault diagnosis of bearings in machine learning environment," *IEEE Sensors Journal*, vol. 20, pp. 4816–4823, 5 2020.
- [415] M. Zubair, M. V. Belykh, M. U. K. Naik, M. F. M. Gouher, S. Vishwakarma, S. R. Ahmed, and R. Kongara, "Detection of epileptic seizures from eeg signals by combining dimensionality reduction algorithms with machine learning models," *IEEE Sensors Journal*, vol. 21, pp. 16861–16869, 8 2021.
- [416] Y. Liu, B. Jiang, J. Feng, J. Hu, and H. Zhang, "Classification of eeg signals for epileptic seizures using feature dimension reduction algorithm based on lpp," *Multimedia Tools and Applications*, vol. 80, pp. 30261–30282, 8 2021.
- [417] M. H. Soleimani-Babakamali, R. Soleimani-Babakamali, R. Sarlo, M. F. Farghally, and I. Lourentzou, "On the effectiveness of dimensionality reduction for unsupervised structural health monitoring anomaly detection," *Mechanical Systems and Signal Processing*, vol. 187, p. 109910, 3 2023.
- [418] M. Rezamand, M. Kordestani, R. Cariveau, D. S.-K. Ting, and M. Saif, "A new hybrid fault detection method for wind turbine blades using recursive pca and wavelet-based pdf," *IEEE Sensors Journal*, vol. 20, pp. 2023–2033, 2 2020.
- [419] F. Gharibnezhad, L. E. Mujica, and J. Rodellar, "Applying robust variant of principal component analysis as a damage detector in the presence of outliers," *Mechanical Systems and Signal Processing*, vol. 50–51, pp. 467–479, 1 2015.
- [420] J. Alvarez-Montoya, A. Carvajal-Castrillón, and J. Sierra-Pérez, "In-flight and wireless damage detection in a uav composite wing using fiber optic sensors and strain field pattern recognition," *Mechanical Systems and Signal Processing*, vol. 136, p. 106526, 2 2020.
- [421] V. R. Gharehbaghi, H. Kalbkhani, E. N. Farsangi, T. Y. Yang, and S. Mirjalili, "A data-driven approach for linear and nonlinear damage detection using variational mode decomposition and garch model," *Engineering with Computers*, vol. 39, pp. 2017–2034, 6 2023.
- [422] A. F. Gill, S. A. Fatima, A. Nawaz, A. Nasir, M. U. Akram, S. G. Khawaja, and S. Ejaz, "Time domain analysis of eeg signals for detection of epileptic seizure," *IEEE*, 9 2014, pp. 32–35.
- [423] R. Lu, M. Xu, C. Zhou, Z. Zhang, S. He, Q. Yang, M. Mao, and J. Yang, "A novel fault diagnosis method based on neeemd-ruslp feature selection and btstsvm," *IEEE Access*, vol. 11, pp. 113965–113994, 2023.
- [424] L. Du, Z. Xu, H. Chen, and D. Chen, "Feature selection-based low-voltage ac arc fault diagnosis method," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–12, 2023.
- [425] J. Yu, C. Xia, J. Xie, and H. Zhang, "Research on feature importance of gait mechanomyography signal based on random forest," *IEEE*, 7 2020, pp. 191–196.
- [426] A. Zhang and Q. Li, "Influence of emg data from different days on feature selection methods," *IEEE*, 10 2020, pp. 1437–1442.
- [427] S. Bekiryazici, A. Demir, and G. Yilmaz, "Feature selection and analysis eeg signals with sequential forward selection algorithm and different classifiers," *IEEE*, 10 2020, pp. 1–4.
- [428] A. U. Haq, J. Li, M. H. Memon, M. H. Memon, J. Khan, and S. M. Marium, "Heart disease prediction system using model of machine learning and sequential backward selection algorithm for features selection," *IEEE*, 3 2019, pp. 1–4.
- [429] M. R. Islam, M. M. M. Islam, and J.-M. Kim, "Feature selection techniques for increasing reliability of fault diagnosis of bearings," *IEEE*, 12 2016, pp. 396–399.
- [430] A. Nandy, M. A. Alahe, S. M. N. Uddin, S. Alam, A.-A. Nahid, and M. A. Awal, "Feature extraction and classification of eeg signals for seizure detection," *IEEE*, 1 2019, pp. 480–485.
- [431] D. L. de Vargas, J. T. Oliva, M. Teixeira, D. Casanova, and J. L. G. Rosa, "Feature extraction and selection from electroencephalogram signals for epileptic seizure diagnosis," *Neural Computing and Applications*, vol. 35, pp. 12195–12219, 6 2023.
- [432] M. H. Kabir, S. Mahmood, A. A. Shiam, A. S. M. Miah, J. Shin, and M. K. I. Molla, "Investigating feature selection techniques to enhance the performance of eeg-based motor imagery tasks classification," *Mathematics*, vol. 11, p. 1921, 4 2023.
- [433] V. Bhandari and M. D.H., "Improved ensemble learning model with optimal feature selection for automated epileptic seizure detection," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging Visualization*, vol. 11, pp. 135–165, 3 2023.
- [434] T. Buckley, B. Ghosh, and V. Pakrashi, "A feature extraction amp; selection benchmark for structural health monitoring," *Structural Health Monitoring*, vol. 22, pp. 2082–2127, 5 2023.
- [435] A. K. N. H. Ahmed, *Condition Monitoring with Vibration Signals: Compressive Sampling and Learning Algorithms for Rotating Machines*. Wiley-IEEE Press, 2020.
- [436] R. B. Randall, *Vibration-based Condition Monitoring*. Wiley, 1 2011.
- [437] C. Lessmeier, J. K. Kimotho, D. Zimmer, and W. Sextro, "Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification," 2016. [Online]. Available: <https://api.semanticscholar.org/CorpusID:4692476>
- [438] I. M. Rosado-Mendez, K. Nam, T. J. Hall, and J. A. Zagzebski, "Task-oriented comparison of power spectral density estimation methods for quantifying acoustic attenuation in diagnostic ultrasound using a reference phantom method," *Ultrasonic Imaging*, vol. 35, pp. 214–234, 7 2013.
- [439] M. H. Sheikh, E. R. Das, A. Islam, A. Ghosh, N. S. B. Shakhawat, A. A. Mahmood, and J. A. Rahman, "Development of a bengali speech-based emotion analysis system," *IEEE*, 4 2023, pp. 156–164.
- [440] T. Giannakopoulos and A. Pikrakis, *Audio Features*. Elsevier, 2014, pp. 59–103.
- [441] Y. Li, X. Wang, Z. Liu, X. Liang, and S. Si, "The entropy algorithm and its variants in the fault diagnosis of rotating machinery: A review," *IEEE Access*, vol. 6, pp. 66723–66741, 2018.
- [442] Y. Kim and Y.-K. Kim, "Physics-informed time-frequency fusion network with attention for noise-robust bearing fault diagnosis," *IEEE Access*, vol. 12, pp. 12517–12532, 2024.
- [443] L. Cui, X. Tian, Q. Wei, and Y. Liu, "A self-attention based contrastive learning method for bearing fault diagnosis," *Expert Systems with Applications*, vol. 238, p. 121645, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0957417423021474>
- [444] Z. Zhang and L. Wu, "Graph neural network-based bearing fault diagnosis using granger causality test," *Expert Systems with Applications*, vol. 242, p. 122827, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0957417423033298>
- [445] Y. Xue, C. Wen, Z. Wang, W. Liu, and G. Chen, "A novel framework for motor bearing fault diagnosis based on multi-transformation domain and multi-source data," *Knowledge-Based Systems*, vol. 283, p. 111205, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0950705123009553>
- [446] H. Geng, Y. Peng, L. Ye, and Y. Guo, "Fault identification of rolling bearing with variable speed based on generalized broadband mode decomposition and det," *SSRN Electronic Journal*, 2022.
- [447] K. Zhang, C. Fan, X. Zhang, H. Shi, and S. Li, "A hybrid deep-learning model for fault diagnosis of rolling bearings in strong noise environments," *Measurement Science and Technology*, vol. 33, p. 065103, 6 2022.
- [448] S. Zenian, T. Ahmad, and A. Idris, *Advanced Fuzzy Set: An Application to Flat Electroencephalography Image*, 2020, pp. 649–657.
- [449] E. K. S. Louis, L. C. Frey, J. W. Britton, J. L. Hopp, P. J. Korb, M. Z. Koubeissi, W. E. Lievens, and E. Pestana-Knight, "Electroencephalography (eeg): An introductory text and atlas of normal and abnormal findings in adults, children, and infants," 2016. [Online]. Available: <https://api.semanticscholar.org/CorpusID:151886401>
- [450] R. G. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David, and C. E. Elger, "Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state," *Physical Review E*, vol. 64, p. 061907, 11 2001.
- [451] I. Ullah, M. Hussain, E. ul Haq Qazi, and H. Aboalsamh, "An automated system for epilepsy detection using eeg brain signals based on deep learning approach," *Expert Systems with Applications*, vol. 107, pp. 61–71, 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0957417418302513>
- [452] C. Nayak and A. Anilkumar, "Eeg normal waveforms." *C StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing, Accessed: Mar 2024*, 1 2024. [Online]. Available: <https://www.ncbi.nlm.nih.gov/books/NBK539805/>
- [453] R. Djemili and I. Djemili, "Nonlinear and chaos features over emd/vmd decomposition methods for ictal eeg signals detection," *Computer Methods in Biomechanics and Biomedical Engineering*, pp. 1–20, 10 2023.

- [454] E. L. Lema-Condo, F. L. Bueno-Palomeque, S. E. Castro-Villalobos, E. F. Ordonez-Morales, and L. J. Serpa-Andrade, "Comparison of wavelet transform symlets (2-10) and daubechies (2-10) for an electroencephalographic signal analysis." *IEEE*, 8 2017, pp. 1–4.