People’s Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

University M’Hamed BOUGARA – Boumerdès

Institute of Electrical and Electronic Engineering

Department of Power and Control

Project Report Presented in Partial Fulfilment of the Requirements of the Degree of

MASTER

In **Electrical and Electronics Engineering**

Option**: Power Engineering**

Title:

Power Quality Disturbances Classification using Convolutional Neural Networks

Presented by:

- **BELLIL Sara**

Supervisor:

Pr. A.Ammar

Co-Supervisor :

Fayçal YKHLEF, Maître de Recherche A, CDTA, Alger.

Registration number: 2022/2023

ABSTRACT

This thesis focuses on the application of 1D Convolutional Neural Networks (CNN) for the classification of power quality disturbances. As the demand for electricity continues to rise, ensuring the reliability and efficiency of power systems has become paramount. Power quality disturbances pose a significant challenge in maintaining system stability and integrity. The aim of this research is to explore the potential of 1D CNN in accurately classifying various types of power quality disturbances, thereby contributing to the enhancement of power system reliability.

This thesis provides an overview of power quality disturbances, including an exploration of the state-of-the-art research. It delves into the field of pattern recognition, specifically focusing on the detailed architecture of 1D CNN. The proposed 1D CNN model for power quality disturbance classification is presented in detail. Three different datasets were used in this work which are noiseless dataset, dataset with 30dB noise and dataset with random noise. The accuracy results were 100%, 97.18% and 93% respectively. The 1D CNN model proposed showed effective classification ability even in the case of noise, and also a good generalization for it to be used as prediction model.

AKNOWLEGEMENTS

First and foremost, I am deeply grateful to my supervisor, Dr. YKHLEF Fayçal, for his guidance, expertise, and continuous encouragement throughout the research process. His insightful feedback and unwavering support were instrumental in shaping this thesis.

I am also thankful to Pr. AMMAR Abdelkarim for his scholarly guidance that greatly enhanced the quality of this work.

I extend my heartfelt thanks to the Advanced Technology Development Centre for providing me with the necessary resources and facilities to conduct my research effectively.

I would also like to express my gratitude for KHEMISSAT Anfel's willingness to share her knowledge and time, which have helped me understand the subject matter and enriched the quality of my work.

DEDICATION

This thesis is dedicated to my loving sister BELLIL Nesrine and my dearest four friends.

To my sister, thank you for your unwavering support, encouragement, and belief in me throughout this journey. Your constant love and support have been a source of strength and inspiration. This achievement would not have been possible without your presence in my life.

To my close friends, who I do not need to name for they know themselves, thank you for standing by my side and providing me with the much-needed companionship, laughter, and encouragement. Your friendship has been a source of joy and comfort, and I am grateful for the countless memories we have shared.

I dedicate this thesis to you all as a token of my deepest appreciation and gratitude for your unwavering support, understanding, and belief in my abilities. Your presence in my life has made this journey meaningful and fulfilling.

Thank you for being my pillars of strength and for being a constant source of motivation. This accomplishment is as much yours as it is mine.

CONTENT

Abstract (i)
Acknowledgements (ii)
Dedication (iii)
Content (iv)
List of tables (vi)
List of figures (vii)
List of abbreviations (viii)

CHAPTER 1 : INTRODUCTION

1.1 Motivation ... 9
1.2 Contribution ... **Erreur ! Signet non défini.**
1.3 Projet impact ... 9
1.4 Thesis summary ... 10

CHAPTER 2: OVERVIEW ON POWER QUALITY DISTURBANCES

2.1 Introduction ... 12
2.2 Types of power quality problems ... 12
2.3 Correction of Power-Quality Problems ... 17
2.4 State of the art on the classification of PQDs ... 19
2.5 Conclusion ... 20

CHAPTER 3: CLASSIFICATION OF POWER QUALITY DISTURBANCES USING 1D CNNS

3.1 Introduction ... 21
3.2 Pattern classification ... 21
3.3 Overview on Convolutional neural network ... 25
3.4 One dimensional CNN ... 25
3.5 Difference between a CNN 1D and a CNN 2D ... 29
3.6 The model of the 1D CNN proposed method ... 30
3.7 Performance assessment ... 33
3.8 Conclusion ... 34

CHAPTER 4: RESULTS AND INTERPRETATIONS

4.1 Introduction ... 35
4.2 Software and programming languages ... 35
4.3 Dataset generation ... 39
4.4 Results and discussion ... 40
4.5 Results assessment ... 47

4.6 Comparison with other work: 49
4.7 Conclusion 49

CHAPTER 5 : CONCLUSION AND FUTURE WORK

5.1 Conclusion 49
5.2 Future work 49

LIST OF TABLES

Table 2-1 Classification of various power quality events with their duration and voltage magnitude [5] 13
Table 4-1 Mathematical models of PQDs. [4] 38
Table 4-2 Table showing the contstraint parameters chosen for each disturbance. .. 39
Table 4-3 Accuracy of the three datasets. 44
Table 4-4 Accuracy of 16 classes of disturbances in the three datasets. 44
Table 4-5 Different key performance indicators for the sixteen classes in the three dataset with their averages. 48
Table 4-6 Comparison of the accuracy of the proposed method with the accuracy of different methods. 49

LIST OF FIGURES

Figure 2-1 Short Duration Variation disturbances. 14
Figure 2-2 Transient disturbances. 14
Figure 2-3 Waveform distortion disturbances. 15
Figure 2-4 Hybrid disturbances. 16
Figure 2-5 Power Quality Disturbances classification diagram. 17
Figure 2-7 Uninterruptible Power Supply [9]. 18
Figure 3-1 Pattern recognition system [22]. 22
Figure 3-2 Types of pattern recognition [23]. 23
Figure 3-3 Working principle of statistical pattern recognition [26] 23
Figure 3-4 Neural Network one-directional connection [31]. 24
Figure 3-5 Comparison between Machine Learning and Deep Learning [35]. 25
Figure 3-6 Typical 1D CNN architecture [38] 26
Figure 3-7 Extracting feature map through convolution with kernel size of 3 and zero padding in 1D CNN [40] 27
Figure 3-8 Neural Network before and after drop out [46]. 28
Figure 3-9 Learning capacity of traditional algorithms vs neural network [48] 29
Figure 3-10 Input and kernel sizes for 1D CNN and 2D CNN [51]. 30
Figure 3-11 Model summary showing the input and output sizes at each layer 32
Figure 3-12 (a) Confusion matrix for binary classification. (b) Confusion matrix for multi-class classification [53]. 33
Figure 4-1 User Interface of open-source dataset generator. 36
Figure 4-2 GUI example: generation of a disturbance signal with 3 sag signals. 36
Figure 4-3 Confusion matrix for noiseless dataset 41
Figure 4-4 Confusion matrix for 30dB dataset 42
Figure 4-5 Confusion matrix for random noise dataset 43
Figure 4-6 Loss curve for noieseless dataset. 45
Figure 4-7 Loss curve for 30dB noise dataset. 46
Figure 4-8 Loss curve for random noise dataset. 46

LIST OF ABBRIVIATIONS

AI: Artificial intelligence

ANN: Artificial Neural Networks

AWGN: Additive White Gaussian Noise

CNN: Convolutional Neutral Network

DL: Deep Learning

DNN: Deep Neutral Network

DT: Decision Tree

E/G: Engine-generator

ELM: Extreme Learning Machine

EMD: Empirical Mode Decomposition

FC: Fully Connected

FDST: Fast Discrete S Transform

FFT: Fast Fourier Transform

GUI: Graphical User Interface

GWO: Greg Wolf Optimization

HT: Hilbert Transform

KNN: K-Nearest Neighbor

ML: Machine Learning

NN: Neutral Network

PQDs: Power Quality Disturbances

PQDs: Power Quality Disturbances

PSO: Practical Swarm Optimization

SNR: Signal to Noise Ratio

SPR: Statistical pattern recognition

ST: S Transform

SVM: Support Vector Machine

UPSs: Uninterruptible Power Supplies UPSs

WT: Wavelet Transform

Chapter 1: Introduction

1.1 Motivation

The demand for electricity is rapidly increasing worldwide, leading to the construction of more complex power systems and grids. However, this complexity also brings a higher risk of power quality disturbances. These disturbances have a significant impact on the reliability of power systems. To enhance system reliability and efficiency, it is crucial to explore advanced techniques.

Artificial intelligence (AI) is becoming increasingly integrated into various domains, prompting me to consider its potential in the electrical field. During my research on classification techniques, I noticed a gap in applying Convolutional Neural Networks (CNNs) specifically to power quality disturbances. Most existing work focused on 2-dimensional CNNs, despite power signals being 1-dimensional. This motivated me to explore the use of 1D CNNs to classify power quality disturbances and improve grid reliability.

By leveraging 1D CNNs, we can effectively analyze the temporal and frequency patterns in power signals, enabling accurate and automated classification of disturbances. This research aims to contribute to the advancement of power quality analysis and provide valuable insights for system operators and engineers to implement proactive measures and ensure reliable power system operation.

In conclusion, my motivation for working on the classification of power quality disturbances using 1D CNNs arises from the need to address the increasing complexity and risks associated with power systems. By applying AI techniques, particularly 1D CNNs, we aim to enhance the reliability and performance of power grids. This research has the potential to make a significant impact by optimizing power systems and ensuring a stable and uninterrupted electricity supply to meet the growing demands of our modern world.

1.2 Project impact

By employing the power of 1D CNN, which has proven to be highly effective in various domains of pattern recognition and signal processing, I aim to develop a robust and accurate classification model capable of automatically identifying and categorizing different types of power quality disturbances. This approach holds great promise for enhancing the monitoring and diagnosis capabilities of power systems, leading to timely detection, effective mitigation, and ultimately improving the overall reliability and quality of electrical supply.

1.3 Aim

Through this project, I aspire to contribute to the field of power quality analysis by developing a state-of-the-art classification model based on 1D CNN. By achieving accurate and automated classification of power quality disturbances, we can empower system operators, engineers, and researchers with valuable information to effectively manage power systems, diagnose faults, and implement targeted mitigation strategies.

1.4 Thesis summary

Chapter 1: Introduction

- Provides an overview of the project and outlines the motivation behind it.
- Highlights the contributions made and discusses the potential impact of the research.

Chapter 2: Overview on Power Quality Disturbances

- Explores the field of power quality disturbances, including their types and significance.
- Examines existing tools and approaches used to optimize power quality disturbances.

Chapter 3: Classification of Power Quality Disturbances

- Discusses the field of pattern recognition and its relevance to the project.
- Introduces the concept of convolutional neural networks (CNN) for classification purposes.
- Introduce and discuss the proposed method and model.

Chapter 4: Results and Discussion

- Presents the obtained results from the implementation of the 1D CNN model for power quality disturbances classification.
- Discusses the findings and provides an analysis of the results.

Chapter 5: Conclusion and Future Work

- Summarizes the main conclusions drawn from the research.
- Reflects on the significance of the work and its implications.

- Outlines potential avenues for future research and improvement

Chapter 2: Overview on Power Quality Disturbances

2.1 Introduction

Power quality disturbances (PQDs) are of great importance in ensuring the efficiency, safety, and reliability of operations electrical systems, minimizing financial losses, and meeting regulatory requirements. Organizations and power utilities invest in power quality monitoring, analysis, and mitigation measures to address these disturbances and maintain optimal power supply conditions. In this chapter, we will go through the different types of power quality disturbances and their classification, and then we will mention some the work previously done in the field of PQDs classification.

2.2 Types of power quality problems

There are several concerns with power quality in today's modern electrical systems, which are experiencing drastic development. This change made the basis for classification of power quality disturbances vary. Bhim Singh et al. [1] classified power quality disturbances into three categories:

1. Events: transient nature disturbances (e.g. sag, swell, short-duration voltage variations and power frequency variations) and steady state nature (e.g. long-duration voltage variations, waveform distortions, DC offset, flicker and poor power factor)
2. Quantity: voltage (flicker, notches, sag, swell), current (harmonic currents, unbalanced currents, and excessive neutral current) and frequency.
3. Load or supply chain: load current (harmonics, DC offset, unbalanced current), supply chain (voltage, frequency or combination of both).

On the other hand, S.Khokhar et al. [2] classified power quality events into two main categories. First, fault events like sag or interruption in faulty phase of a three-phase system and swell in a non-faulty phase. Second, switching events like transients (e.g. impulsive and oscillatory) and harmonics (e.g. sag, swell and interruption).

IEEE standard 115-1995 also had a similar classification for power quality disturbances. It calls the events based on its nature like transient for impulsive and oscillatory events, short duration variations for interruption, sag and swell, and also steady state for harmonics, notch flicker etc. [3]- [4]

D. Saxena et al. [5] classified power quality disturbances based on the nature of the waveform distortion. Table 2-1 shows each category while mentioning its spectral content, duration and magnitude.

Table 2-1 Classification of various power quality events with their duration and voltage magnitude [5]

Category		**Duration**	**Voltage Magnitude**
Short Duration Variation	Sag	0.5-30 cycle. (instantaneous)	0.1-0.9 pu.
		30 cycles-3 sec. (momentary)	0.1-0.9 pu.
		3 sec-1min.(temporary)	0.1-0.9 pu.
	Swell	0.5-30 cycle. (instantaneous)	1.1-1.8 pu.
		30 cycles-3 sec. (momentary)	1.1-1.4 pu.
		3 sec-1 min. (temporary)	1.1-1.2 pu.
	Interruption	0.5 cycles-3 sec. (momentary)	<0.1 pu.
		3 sec-1min. (temporary)	<0.1 pu.
Transients	Impulsive	<50 nsec. (nanosecond)	
		50-1 msec. (microsecond)	
		>1 msec (millisecond)	
	Oscillatory	00.3-50 msec.(low frequency)	0-4 pu.
		20 µsec.(medium frequency)	0-8 pu.
		5 µsec. (high frequency)	0-4 pu.
Waveform Distortion	Harmonics	Steady state.	
	Notch	Steady state.	
	Noise	Steady state.	

We can observe the previously mentioned single disturbances in figure 2-1 and figure 2-2, while figure 3-3 shows the hybrid disturbances.

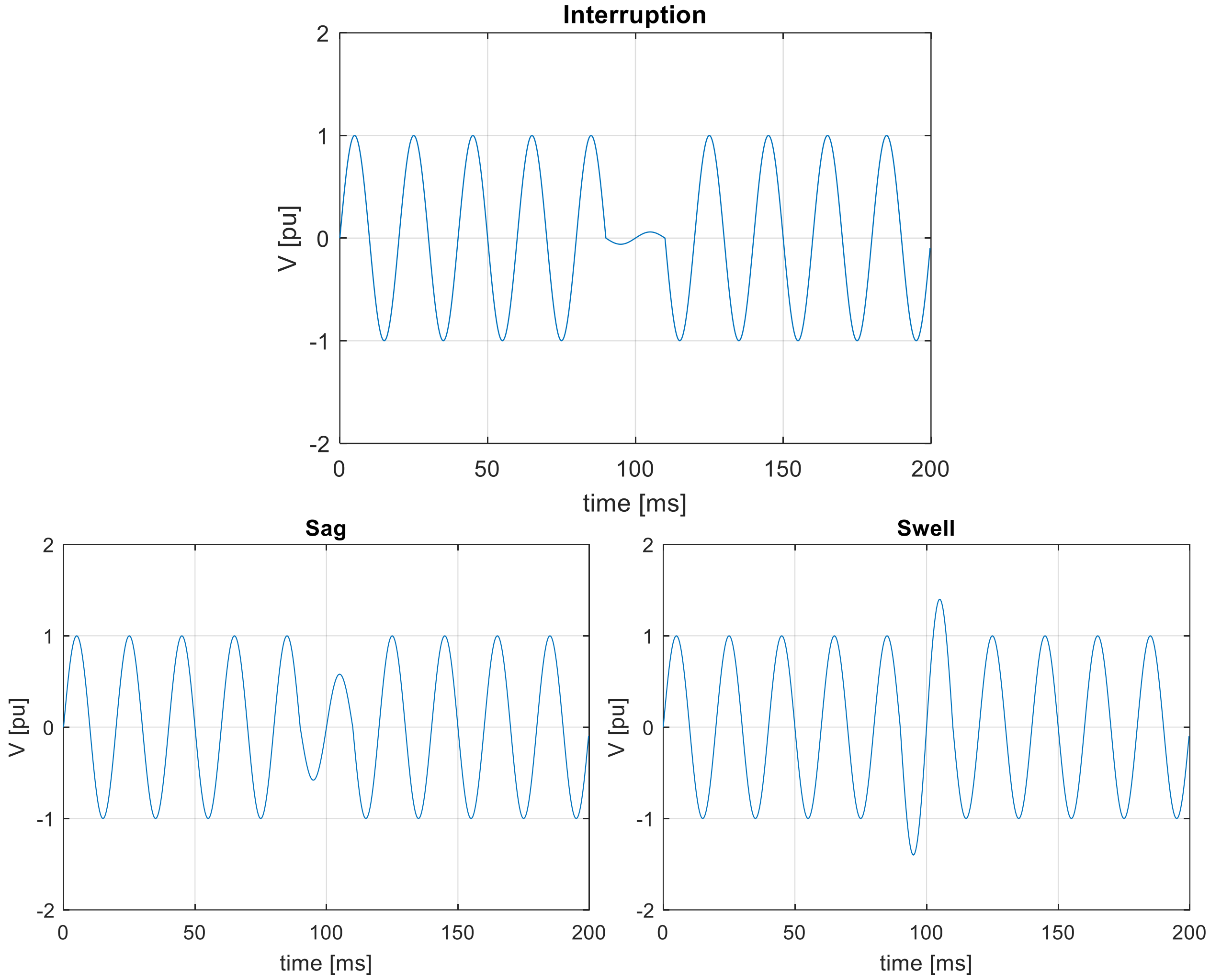


Figure 2-1 Short Duration Variation disturbances.

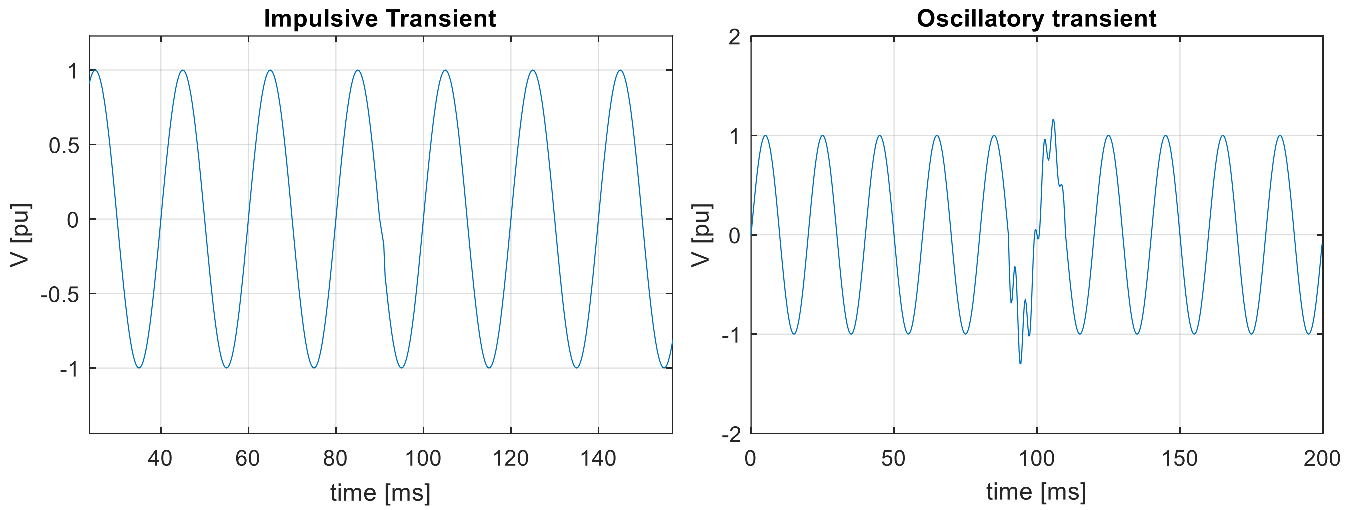


Figure 2-2 Transient disturbances.

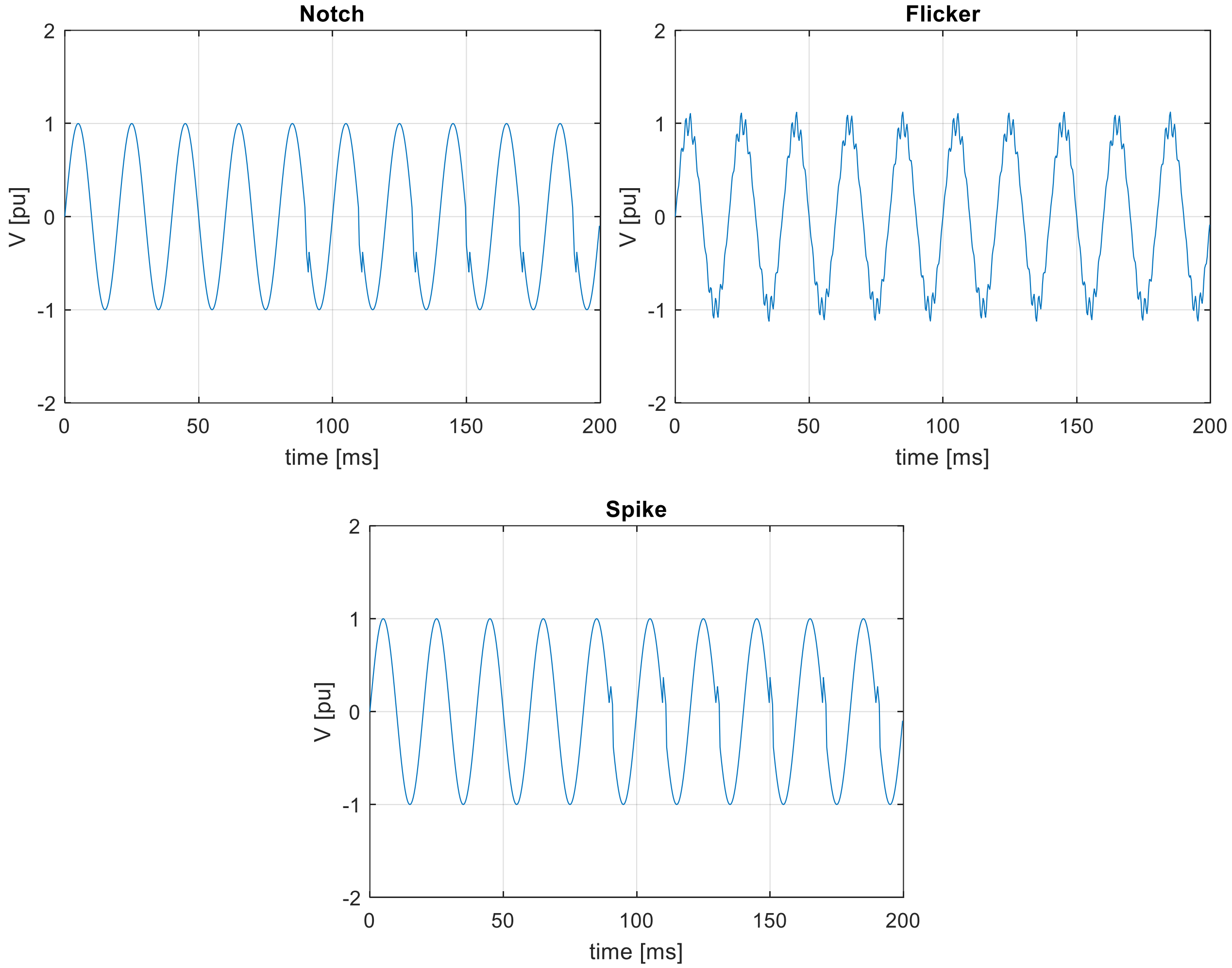


Figure 2-3 Waveform distortion disturbances.

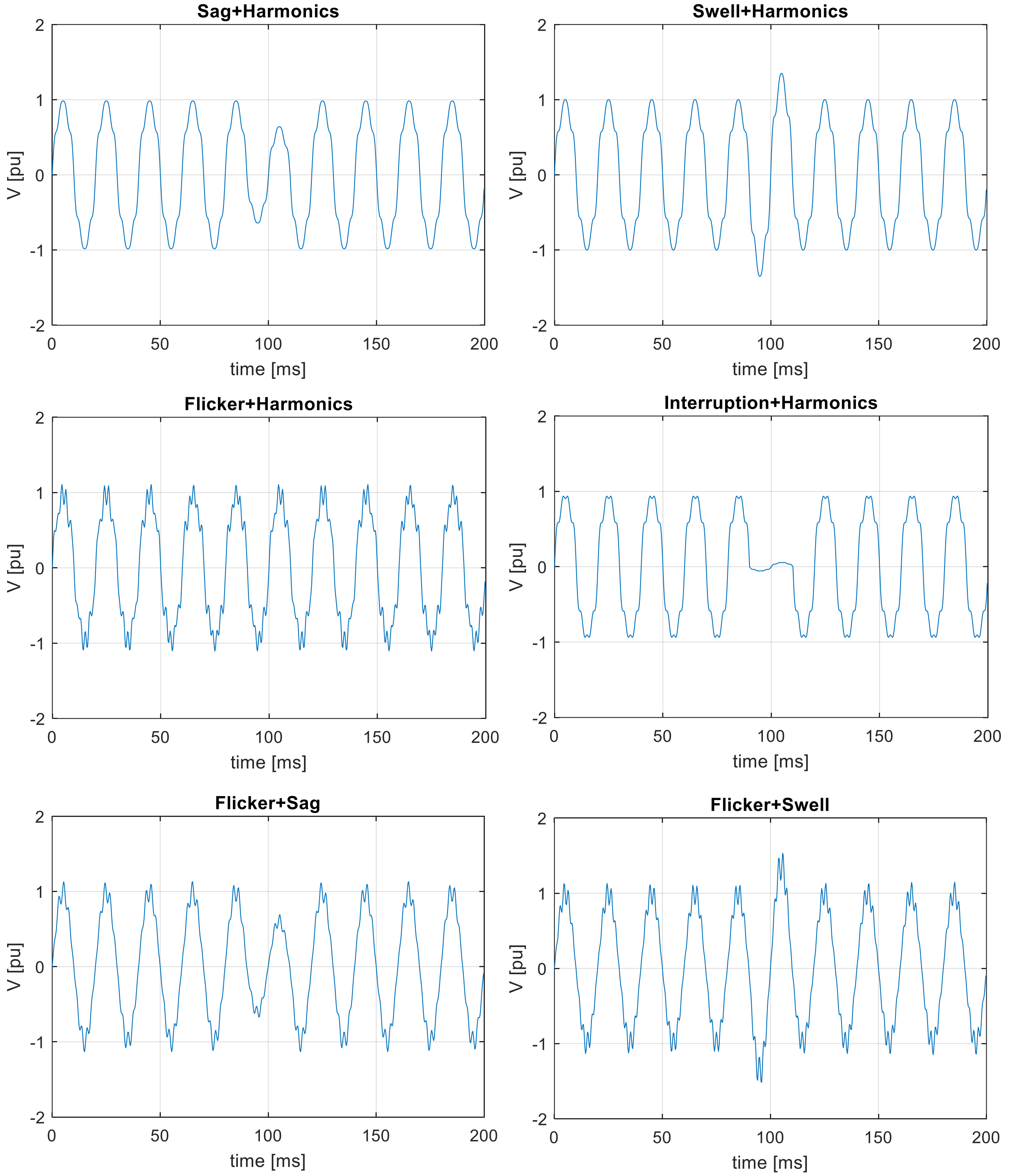


Figure 2-4 Hybrid disturbances.

Figure 2-5 shows a summary of the previous classifications based mainly on nature and quality:

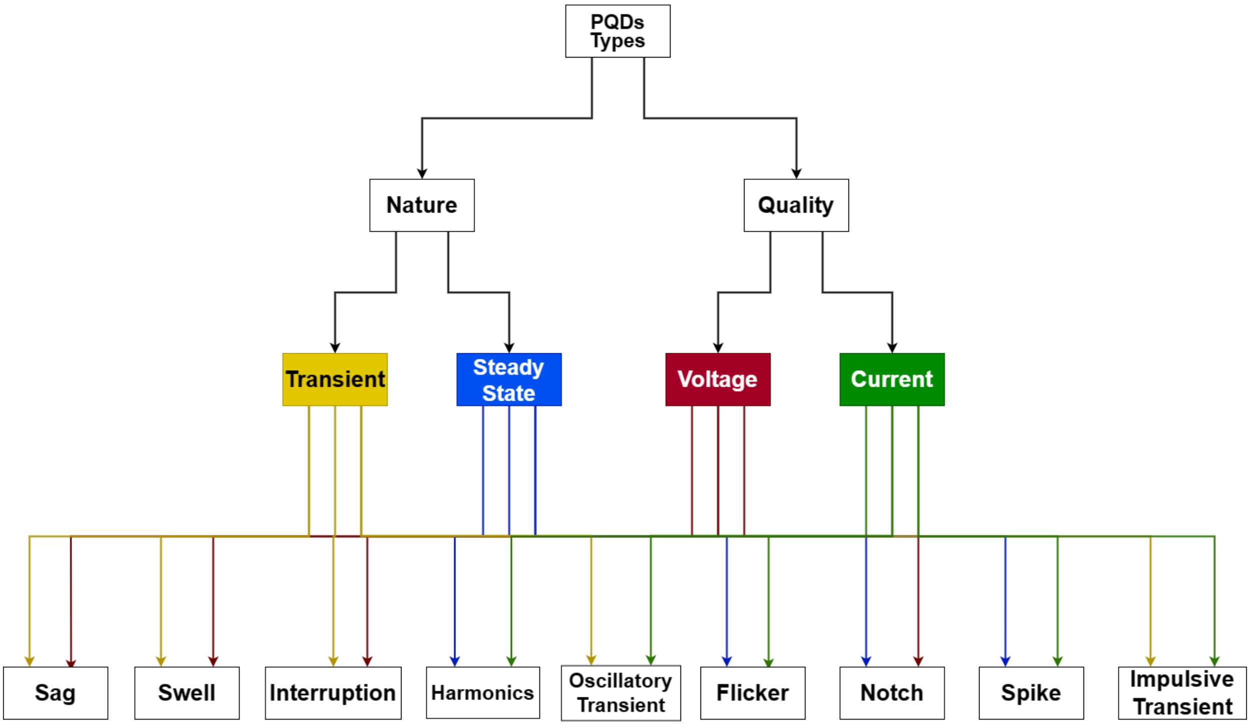


Figure 2-5 Power Quality Disturbances classification diagram.

2.3 Metigation of Power-Quality Problems

In the past few years, the industry witnessed a boost in its investment regarding the issues of power quality in electrical systems. While developing many techniques for recognition and classification of power quality disturbances, it is equally important to come up with new methods of correction.

The aim for these methods of correction is to make the power source qualified in meeting the requirements from the users. Alexander Kusko & Marc T.Thompson and S.Khalid & Bharti Dwivedi [6]- [7] proposed similar methods in this area.

The methods mainly cover:

- Layout of the load equipment: like switch-mode power supplies, which can be implemented in lowering harmonics in the load current as well as control the sensitivity to voltage disturbances.
- System design of the electric-power supply: can be constructed to lower source impedance, to separate loads and to prevent harmonic resonances.

- Installation of power-harmonic filters: implementing power-harmonic filters to compensate the continuous voltage distortion and blocking harmonics. A model is shown in figure 2-3.
- Use of dynamic voltage compensators: used to fix short time voltage waveforms sags.
- Installation of uninterruptible power supplies (UPSs): an electronic inverter providing an independent power source to the load in case of a power outage or interruption. We can see a real implemented supply in figure 2-4.

Figure 2-6 Uninterruptible Power Supply *[8]*.

- Dependability on standby power: Engine-generator (E/G), for example, is more effective during long-time outages. It also replaces the main power supply during a system maintenance. Diesel Generator in figure 2-5 is one of the most common used generators.

2.4 State of the art on the classification of PQDs

Power quality has been receiving a tremendous interest from customers for various reasons, mainly due to the consequences resulting from different types of disturbances. In addition, as the power grid is witnessing a great and quick evolution; new and more complicated power quality disturbances are evolving.

As a result, impressive research is being developed in the field of power quality. Many papers are published discussing new and updated methods and algorithms for detecting and classifying power quality disturbances.

Feng Zhao et al. and Amin Akbarpour et al. [9], [10] implemented different methods for PQDs detection, multi resolution S transform (ST) and extreme-point-based algorithm respectively, yet both used the same type of classifiers (K-Nearest Neighbor KNN, Support Vector Machine SVM, and Decision Tree DT) for classification. In both works, SVM scored the lowest accuracy especially in the case of hybrid disturbances with 95.9% and 76.5%. Whereas DT performed the best with 97.5% and 98.8%. Both authors concluded that DT classifier was the best options compared to other methods.

Similarly, I. W. C. Lee & P. K. [11] Dash also used ST as a method of detection of non-stationary signals in power networks. This method was combined with two types of neutral networks (NN): Feedforward NN and Probabilistic NN for classification. The combined techniques came up with a very high and constant accuracy even in the presence of noise with value of 95%, which showed that this method is significant in designing a strong recognition system for PQDs.

Umamani Subudhi & Sambit Dash [12] also compared KNN and SVM classifiers with Extreme Learning Machine (ELM) method tuned through Grey Wolf Optimization (GWO). The technique over-performed other methods of classifications by a very high accuracy of 99.41%, where ELM alone also had a high accuracy of 97.35% proving that even techniques with good performance can be enhanced when combined with optimization techniques.

Deep learning was also implemented in the field of PQDs classification. Shouxiang Wang, Haiwen Chen [13] used a deep learning method using CNN 1-D where he made a comparison of five different types of deep learning neutral networks DNNs (SEA, ResNet50, LSTM, FRU,RNN and CNN) based on loss, accuracy, training time, mode size, and number of parameters. Different methods outperformed in different parameters, but overall SEA provided the worst performance while the deep CNN proposed in the paper was the optimal choice with higher classification precision and less training time cost.

2.5 Conclusion

Power quality disturbances became a huge topic of interest among both research and consumers for many reasons. This interest resulted in the creation of many different methods of recognition as well as techniques of classifications with different and competitive results of accuracy. This constant growth is a promising sign for the development of the power system.

Chapter 3: Classification of Power Quality Disturbances using 1D CNNs

3.1 Introduction

Advancements in the field of artificial intelligence, especially deep learning, greatly influenced variety of fields and especially the field of pattern recognition on many aspects. Pattern recognition methods existed since a long time but recently has been witnessing a drastic development. Many methodologies and techniques are being invented and updated. One of the methods that has been catching scientists' attention is neural networks and specifically CNNs. While CNNs are usually correlated with image-related applications, they can also be applied to 1-dimensional signals such as voice or electrical waves. Research studies on 1D CNNs are not as wide-ranging as their 2D counterparts. Therefore, in this chapter we focus on the application of 1 D CNN for the classification of power quality disturbances. We will go through a review on pattern recognition, its description and types. Then, we will discuss in details 1D convolutional neural networks focusing on its working principle. A comparison between 1D and 2D architectures is also provided.

3.2 Pattern classification

3.2.1 Definition

Duda and Hart [14] in his book simply defined pattern recognition as "the act of taking in raw data and making an action based on the "category" of the pattern". In other words, it is the process of assigning objects to certain categories or classes based on specific features, these objects can differ depending on the nature of the study (e.g images, signal waveforms, measurements) [15]- [16]- [17].

There are two types of learning paradigms in pattern classification [18]:

1) Supervised learning: which is learning from data with labeled outputs, this type is used in prediction or classification problems.

2) Unsupervised learning: which is learning from data without given labels for the output, this type is used for grouping problems.

Pattern classification was developed starting from the 1960s, but nowadays it is being implemented in the fields of machine learning using computers. In terms of machine learning, the word “recognition” is more commonly used, which refers to the ability of a computer to recognize patterns of objects that I has learned before [19]

In figure 3-1, we can see the overall system of pattern recognition from receiving an input (object) to generating an output (class/category) [20].

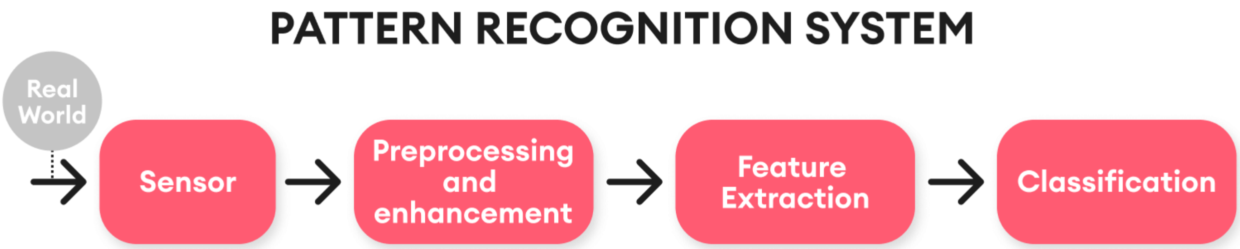


Figure 3-1 Pattern recognition system [22].

3.2.2 Types of pattern classification

Types of pattern classification can differ depending on the nature of the object and the approached method. In this thesis, we will only go through the most commonly used ones: statistical, structural (syntactic), and neutral as shown in figure 3-2 [21]

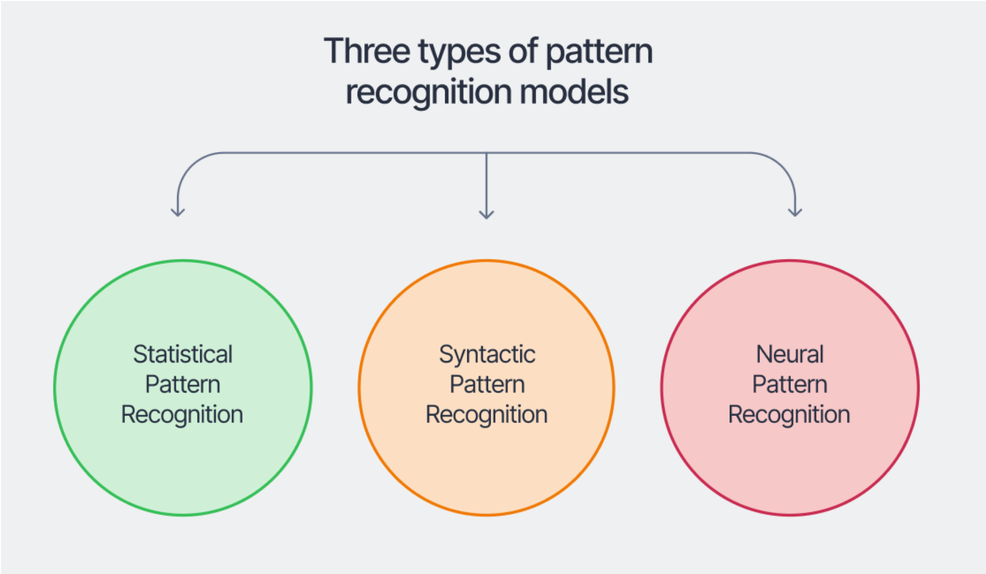


Figure 3-2 Types of pattern recognition *[21]*.

a) Statistical

Statistical pattern recognition (SPR) is using mathematical models and algorithms to statistically characterize patterns of objects in large data sets [22]. These patterns are batch of features where each batch is selected such that it does not intersect with another feature batch [23]. Few applications of SPR are handwriting, speech recognition, or objects classification in images.

From figure 3-3, it can be seen that this method is based on extracting features from new inputs using special algorithms then compare it with already learned parameters to finally assigning each input to its appropriate class or group.

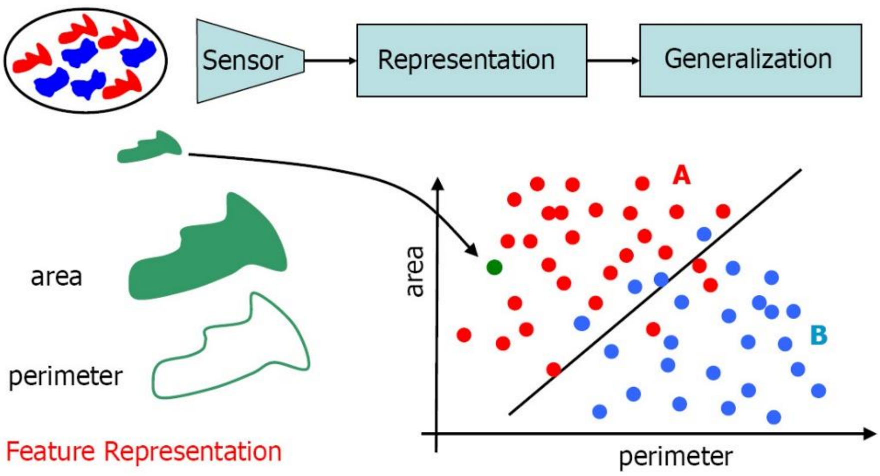


Figure 3-3 Working principle of statistical pattern recognition *[24]*

b) Syntactic

Syntactic or structural pattern recognition is used in the case of problems with more complex features and it includes two main methods: syntax analysis and structure matching [25]. This method is used when patterns are interconnected where they are composed of simple sub-patterns which themselves can be composed of simpler sub-patterns creating a hierarchical system. For better understanding of this architecture, it can be visualized as the structure of a language where the patterns are seen as sentences, sub-patterns as the alphabet, and the sentences are generated according to a grammar [26].

c) Neural

The world witnessed the birth of the first NN model in 1944, it quickly gained popularity in the field of pattern recognition to the fact that it does not required prior knowledge making them capable of earning complex nonlinear connections between the input and output successive training tools and adjust to the given data [27] [28].

This technique was inspired by the working process of a biological nervous system, i.e. the human brain. In fact, the term neural refers to the processing elements implemented in NN called neurons which are sequentially connected together creating multiple hidden layers, these connection can either be one-directional or bi-directional [16] as shown in figure 3-4.

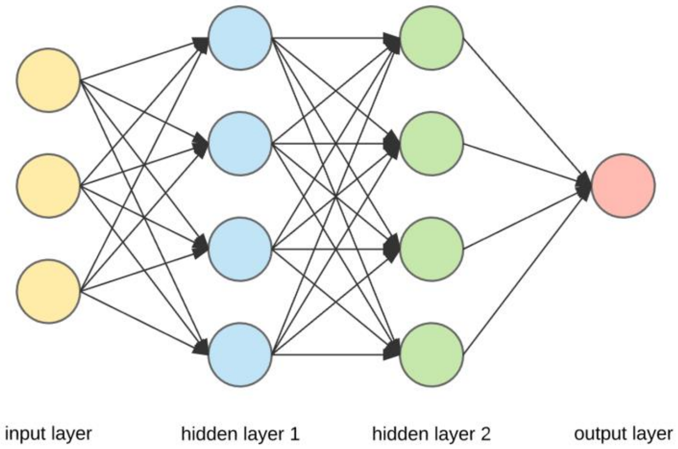


Figure 3-4 Neural Network one-directional connection *[29]*.

The main factors that made NN a widely used architecture in pattern recognition are its flexibility as it adapts to new data and information, time efficiency, tolerance against missing or noisy data, and optimal error rates [30].

3.3 Overview on Convolutional neural network

Convolutional neural network (CNN) was first proposed by Leon O. Chua and Lee Yang in 1988 [31], it quickly the most commonly used algorithm in deep analysis when it comes to pattern classification. Before introducing CNN in details, it is better to have a clearer idea about deep learning (DL).

DL is a subset of machine learning (ML) which was inspired by how the human brain processes patterns in objects or information and creates connection between them. It functions by analyzing a generous amount of data and automatically extracting patterns to label the given inputs. Even though DL is a part of ML, it stands out due to its utilization of artificial neural networks (ANN), which gives it a more complex structure compared to traditional ML methods. This complexity makes DL more practical when dealing with large and complicated data. Not to mention how self-sufficient DL is comparing to ML in terms of human or other types of interventions as shown in figure 3-5 [32].

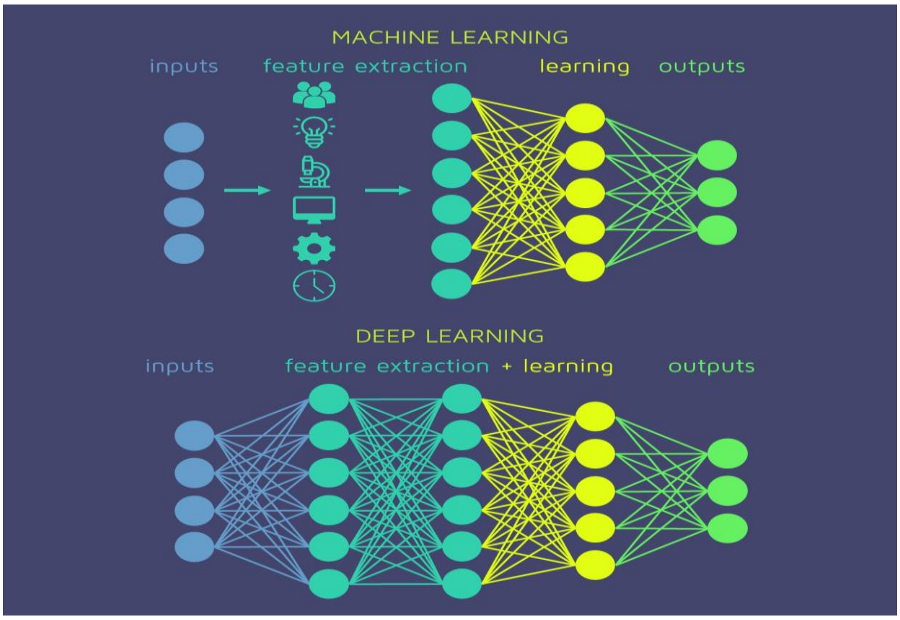


Figure 3-5 Comparison between Machine Learning and Deep Learning *[33]***.**

3.4 One dimensional CNN

1D CNN is one of DL algorithms; it is a type of feed-forward (multi-layer) neutral network consisting of three main layers: input layer, output layer, and multiple hidden layers in between where these layers represent convolutional structures. Since CNN functions based on the human brain structure, every biological parameter in the brain is met with an artificial parameter in the architecture of CNN where biological neurons are artificial neuron; kernels are receptors; activation functions are neural electric signals. These structures perform multiple operations (convolution, pooling, flattening) with the help of other techniques like early-stopping, batch-normalization, and dropout to create multiple hidden layers before the output layer like shown in figure 3-6 [34]- [35].

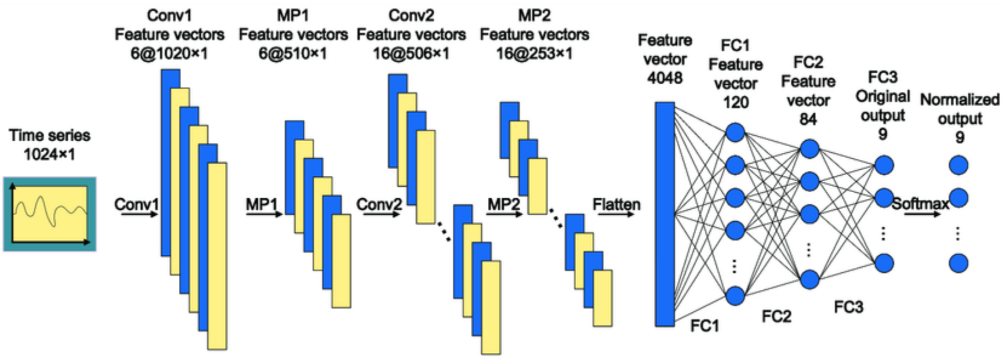


Figure 3-6 Typical 1D CNN architecture *[36]*

1D CNN has three main layers: convolutional layer supported by activation functions, pooling layer, and fully connected layers.

- Convolutional layer:

Convolutional layer is the fundamental component in 1D CNN when it comes to feature extraction. It consists of multiple convolutional filters called kernels, these filters convolve with the input which is considered a matrix and is called a tensor, to then generate an output called feature map. The size of these filters can be adjusted depending on other parameters [37].

Convolution is done such that the size of the kernel is applied across the tensor under a dot product operation between the two over a single dimension. The result of this operation is summed and sent to the next layer to perform the same operation. Multiple operation are done consecutively until a feature map is generated [37].

While the kernel is sliding across the tensor performing the convolution operation, the outermost elements of the tensor can be left out causing a reduction in the size of the feature map compared to the tensor. To avoid this, padding is introduced [37].

Padding, or commonly used zero padding, is the adding rows and columns of zeros in each side of the tensor. This way, the dimension of the tensor is kept in the feature map [37]. These operations are clearly illustrated in figure 3-8.

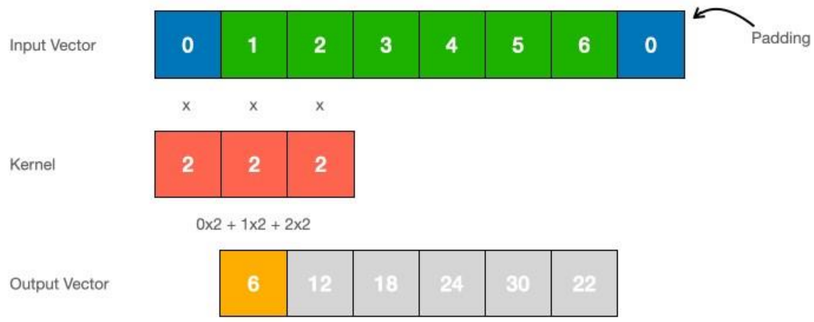


Figure 3-7 Extracting feature map through convolution with kernel size of 3 and zero padding in 1D CNN *[38]*

After generating a feature map, next is to pass this layer through an activation function. There exist multiple activation functions with different characteristics. The most commonly used one is the ReLU function because of its ability to overcome the vanishing gradient problem [39] and lowering the computational load [32].

- Pooling layer:

Pooling layer is responsible for reducing the dimension of the 1D feature map, i.e. decreasing the computations required to process the 1D input data. This layer helps extracting the dominant features making the model training more effective [32], [40].

This operation comes in different forms, the most popular one is max pooling where it returns the maximum value from the portion selected in the feature map by the kernel while ignoring all the other values [40].

- Fully connected (FC) layer:

In this layer, all the neurons are connected to the neurons of the previous layer, hence fully connected. After the feature extraction is done, the last pooling layer is flattened into a one-dimensional array of numbers resulting of an input for the FC layer. This connection results in what we call the dense layers [40], [32].

In addition to these layers, some functions are introduced in order to make the training process more effective like early-stopping, batch-normalization, and dropout functions.

- Early stopping: a regularization technique used to stop the training when it is no longer producing improved results i.e. convergence in order to avoid over fitting [41].
- Batch Normalization: in NN, an output of a layer is an input of another, this shift causes disturbance in the distribution of the input in the training, especially in the case of large number of layers. This disturbance is known as internal covariate shift. Batch normalization is a method introduced to reduce this problem by normalizing each activation input to have the same distribution over every batch. In CNN, usually batch normalization is inserted directly after the convolutional layer [42].
- Dropout: The term "dropout" refers to the dispose of units in the neural network. The units are temporarily and randomly removed with all of its connection, like shown in figure 3-9. The range of values for dropout is from 0.1 to 1, where 0.1 means we drop 10% of the neuron units in previous layer are randomly dropped and 90% are kept active for next iterations. This technique was also introduced to avoid overfitting [43].

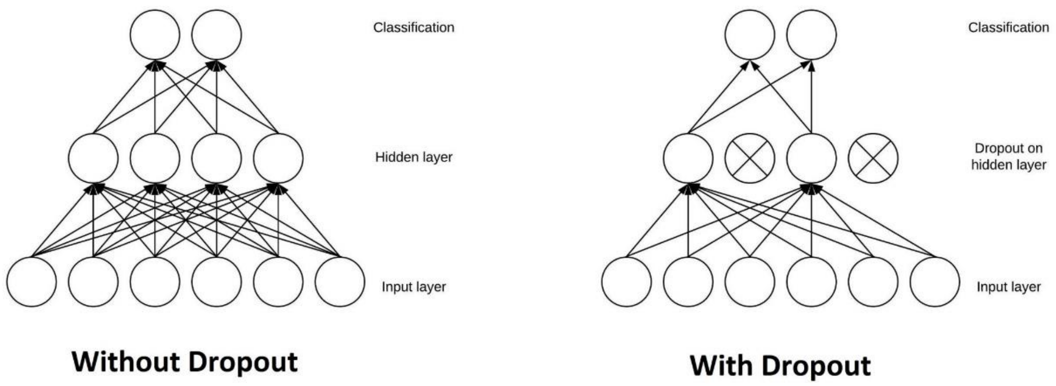


Figure 3-8 Neural Network before and after drop out *[44]*.

This complex architecture of CNN and NN in general, made it a famous classification tool for modern applications. Considering the amount of data collected nowadays, traditional methods are no longer practical as its learning process is limited, i.e. it does not learn from all the data given to it, unlike NN, which its performance is proportional to the amount of available data to be trained [45]. Figure 3-10 clearly shows the relation between the amount of data with traditional learning methods and different levels of complexity of NN networks.

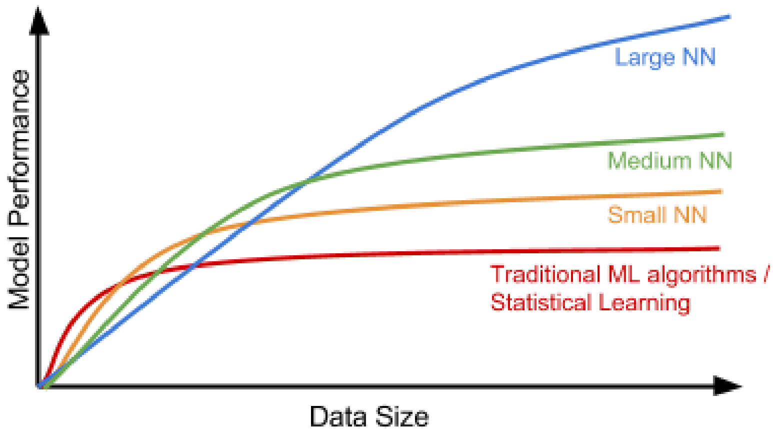


Figure 3-9 Learning capacity of traditional algorithms vs neural network *[46]*

3.5 Difference between a CNN 1D and a CNN 2D

It is worth to mention that when CNN started taking off, it was mostly used on 2D type of objects like images and videos. Later on, a modified version of CNN appeared which is 1D CNN. This version was implemented on applications handling 1D data like speech recognition, ECG monitoring and other types of signals. In this type of data, CNN showed a superiority in performance in terms of computational complexity, facility of training and implementation, and speed [47].

Even though it is mentioned that 1D CNN handles 1D data and 2D CNN handles 2D data, we must clarify that the actual shape of the input data is 3D and 4D respectively. Input in 1D CNN is a matrix with three components: [batch size, width, input channels], while in 2D CNN the input is a matrix with four elements: [batch size, height, width, input channels] [48]

The two types of CNN do not only differ in the dimensionality of the input, but also the dimensionality of the kernel used to extract features. Kernels follow the same dimensionality as the input, i.e. kernels used for 1D CNN are vectors with three components: [width, input channels, output channels], while kernels for 2D CNN are matrices with four elements: [height, width, input channels, output channels [48] This difference is illustrated in figure 3-11.

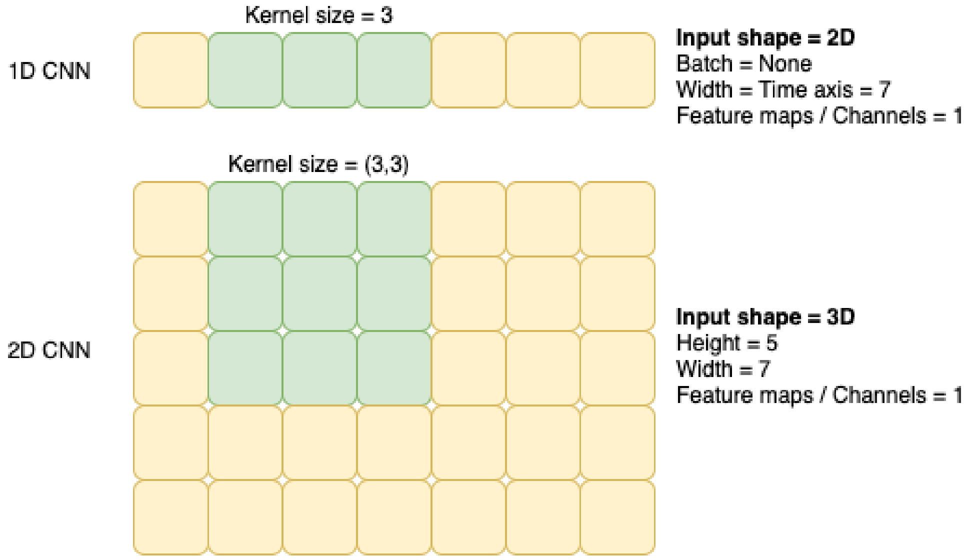


Figure 3-10 Input and kernel sizes for 1D CNN and 2D CNN *[49]*

Number of channels is related to the type of input data. In 1D CNN, the number of channels is typically 1. As for 2D CNN, number of channels is related to the color representation of the images. For example, in case of grayscale images the number of channels Is 1, while for RGB images the number of channels is 3 [49].

In this thesis, for the classification of power quality disturbances in electrical signals using 1D CNN and since these signals are considered one dimensional, we used only 1 input channel i.e. one feature is extracted time step. The resulting input shape of our input is [batch size, time_steps, input channels].

3.6 The model of the 1D CNN proposed method

The model has an input layer, six convolutional layers, three max-pooling layer, two fully connecting layers, and four batch normalization layers.

The shape of our input is [n_timesteps, n_features, n_outputs] where n_timesteps is the length of the input data, n_features is the number of channels and n_outputs is the number of classes in the output labels. From out data, the input shape is [640, 1, 16].

As for the convolutional layers, the size of each layer is 32,32,64,64,128 and 128, respectively. The kernel slides over the input signal by using (3) to produce the feature map. The max-pooling layers are added after every two convolutional layers with size of 3 and a stride of 1. The batch normalization layers are added directly after every max-pooling layer.

After all these layers comes the dropout layer with rate of 0.7 and a flatten layer.

Two dense layers are applied after the flatten layer with 256 and 128 units respectfully with a batch normalization layer in between.

In every convolutional layer, an activation operation is performed using ReLU, and regularization of rate 0.3.

The figure 3-12 summarizes our model by showing the input and output sizes at each layer.

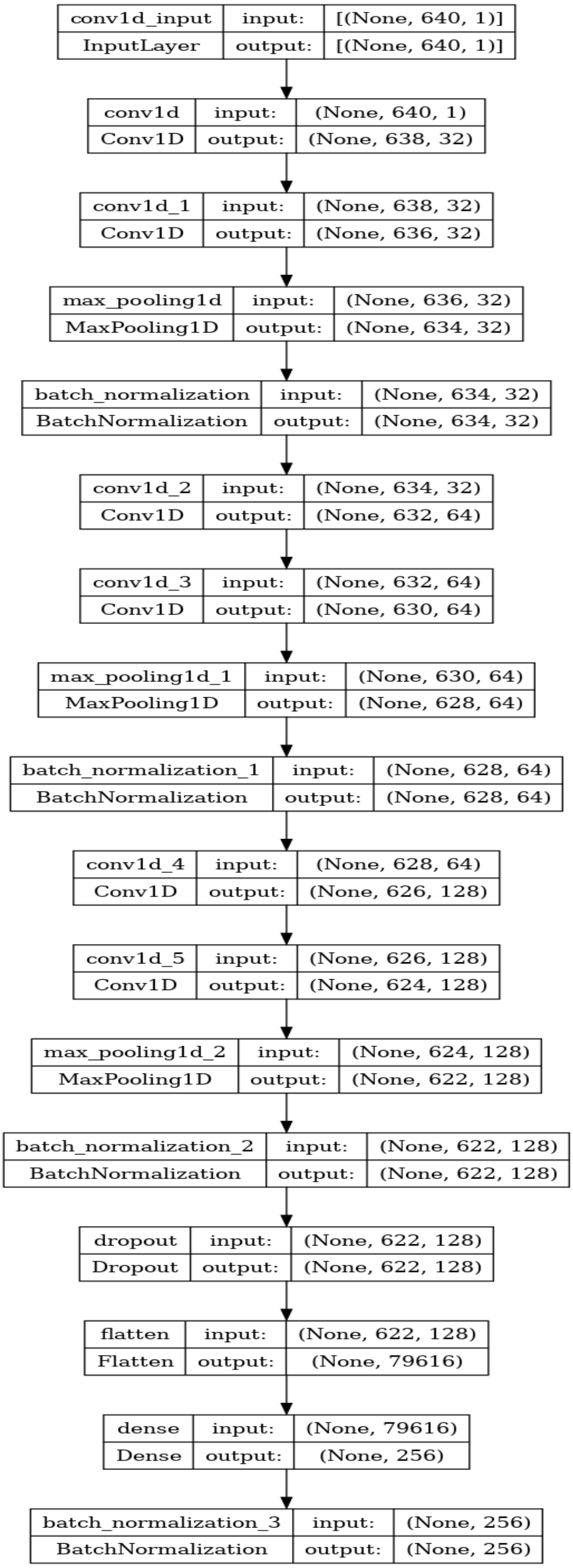


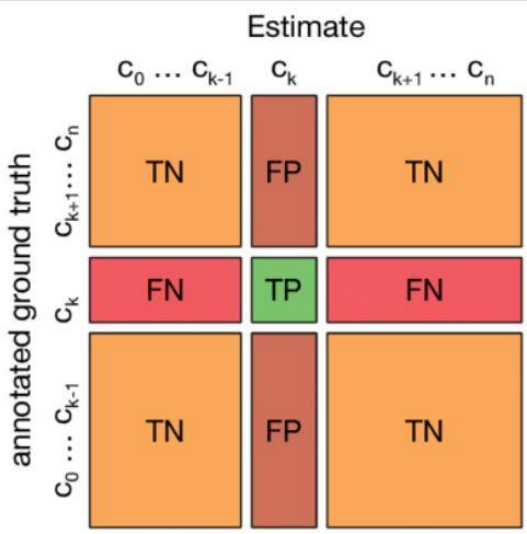
Figure 3-11 Model summary showing the input and output sizes at each layer

3.7 Performance assessment

There are several measures to estimate the performance of a classification algorithm. Some techniques are scalar values calculated though the confusion matrix shown in figure 3-13, like accuracy, sensitivity, precision, and specificity. Alternatively, there exist graphical methods like Receiver operating characteristics (ROC) and Precision-Recall curves (PR) [50]. The latters are used in case of unbalanced dataset.

		Predicted Class	
		Positive	Negative
Actual Class	Positive	TP	FN
	Negative	FP	TN

(a)



(b)

Figure 3-12 (a) Confusion matrix for binary classification. *[51]*
(b) Confusion matrix for multi-class classificaition. *[52]*

In our case, since our data is balanced, we will be using the confusion matrix to evaluate our model which will be shown with the results in chapter 4.

- Confusion Matrix: It is an N×N matrix where N is the number of classes.

Where: TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative.

Key Performance indicators: The previously mentioned indicators are calculated from the confusion matrix using the formulas as follows [51]:

- Accuracy: (TP+TN)/(TP+TN+FP+FN).
- Precision: TP/(TP+FP).
- Recall (Sensitivity): TP/(TP+FN).
- Specificity: TN/(TN+FP).

3.8 Conclusion

Pattern recognition is a problem-solving technique that touched upon a wide-range of fields from medical applications to data mining to financial analysis. Merging this technique in the field of artificial intelligence, and precisely deep learning, resulted in a speed growth in the pattern recognition methodologies and especially in neural networks. Convolutional neural network specifically had profound impact on pattern recognition whether it is for 1D applications like speech recognition, ECG signal analysis, and electrical signals, or for 2D applications like image classification and facial recognition. 1D CNN has being receiving more attention in the field of pattern recognition after proving its robust and effectiveness in the field due to hierarchical representations and exhibit translation invariance. In this project, we have proposed a 1D CNN architecture for the classification of power quality disturbances. This architecture, designed empirically based on PGD data, aims to accurately identify and categorize various disturbances in electrical power systems. The proposed architecture is not very complex with only six convolutional layer, which contributed with the computational efficiency. We will undergo a comprehensive analysis and discussion of the model's performance results in the next chapter.

Chapter 4: Results and interpretations

4.1 Introduction

The reliability of the power system is an important for ensuring the stable and uninterrupted supply of electricity to consumers. In recent years, 1D CNN has proved its robustness and effectiveness in various fields, including power quality analysis. In this chapter, we will go through the results of classifying power quality disturbances using 1D CNN architecture. The method will be analyzed using special performance assessment parameters. We will then briefly compare the performance of our method with different previously suggested techniques.

4.2 Software and programming languages

4.2.1 MATLAB

This work was accomplished using MATLAB software. MATLAB is a powerful programming language for computational tasks. It is a user-friendly tool that combines computation and visualization for solving problems using numerous and familiar mathematical representations [53]. The 2021 version (MATLAB R2021a) was used in this project.

4.2.2 Python

Python was created by Guido van Rossum as a successor of the language called ABC, its first version was released in 1991 [54]. Python is a scripted programming language that focuses on organizing and structuring codes around objects with just-in-time compilation technique, making it easy and fast to edit and debug [55]. Python has many different libraries with different function for multiple applications.

For signal and image processing, we used the Keras library. Keras is a potent, user-friendly, charge-free and an open source for building and assessing deep learning models [56]. This library was used through the Kaggle platform.

Kaggle is a famous platform for its wide use in Data Science. It hosts competitions, provides datasets and models, and is a very useful environment for data analysis and machine learning projects thanks to the variety of libraries it offers like Keras. Kaggle primarily supports the Python programming language. Aside from all these advantages, the main reason we resorted to kaggle is that it offers different GPU accelerators. We executed our model using the GPU P100[1] which tremendously helped with the time optimization.

4.2.3 The open-source software package

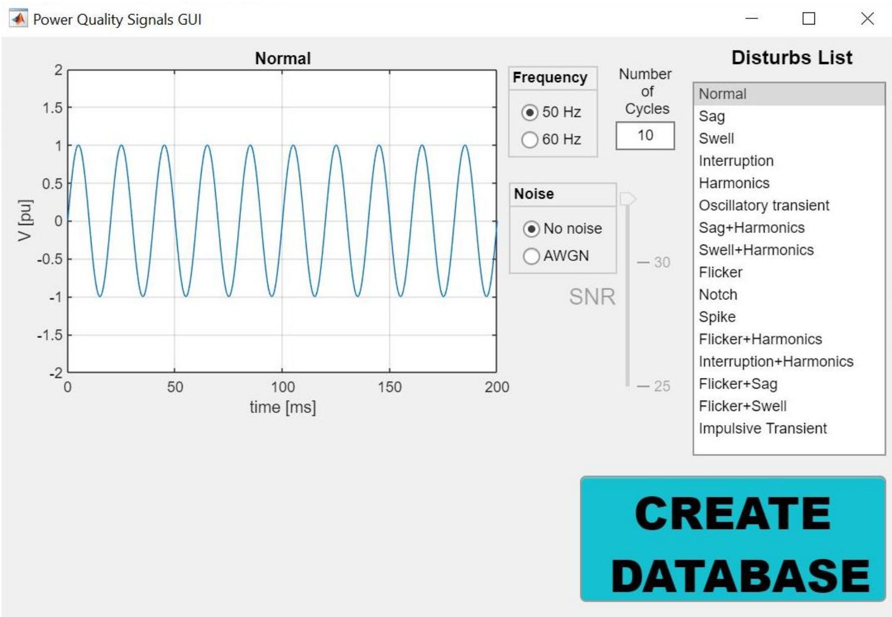


Figure 4-1 User Interface of open-source dataset generator.

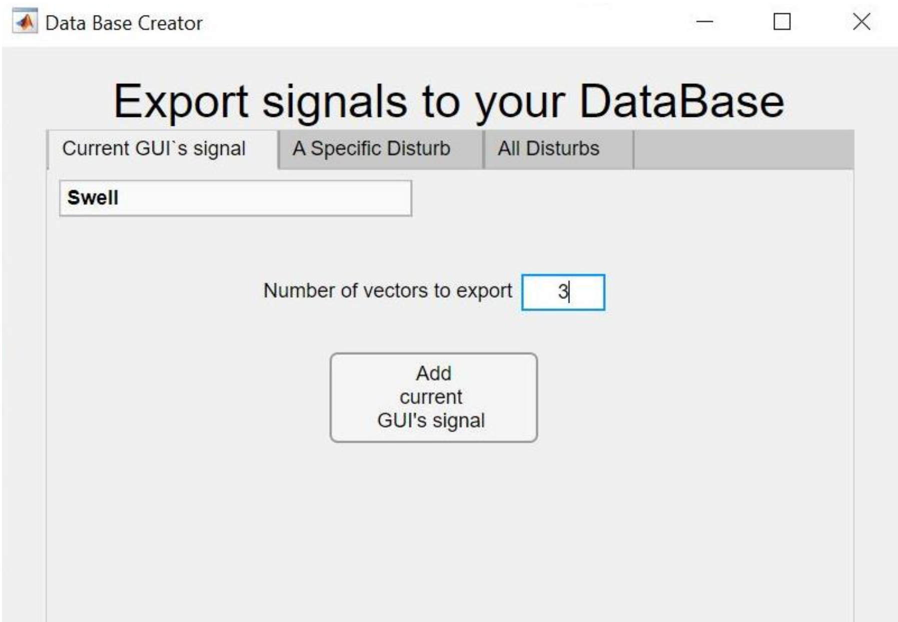


Figure 4-2 GUI example: generation of a disturbance signal with 3 sag signals.

[1] GPU: Graphics Processing Unit, is a specialized electronic circuit which is used for parallel processing tasks related to graphics and visualization. GPU P100 refers to the NVIDIA Tesla P100.

A synthetic dataset generator was designed by R.Machlew et *al*. to generate sixteen different types of PQDs. The sixteen type consist of single disturbances (sag, swell, interruption, harmonics, oscillatory transient, flicker, notch, spike, impulsive transient) and hybrid distrubances (sag+harmonics, swell+harmonics, flicker+harmonics, interruption+harmonics, flicker+sag, flicker+swell).

The mathemathical formulas of these PQ disturbances follow IEEE-1159 standard as shown in table 1-4. The code to this software as well as the Graphical User interface (GUI) were both built using MATLAB 2016B. The interface is shown in figure 4-1.

The software contains two main GUI, the first generates and analyzes the disturbance signals based on the mathematical formulas shown in table 1-4. The second one is a dataset assembler which creates different PQD vectors into a single ".mat" file. These two GUIs are shown in figure 4-1 and 4-2.

Table 4-1 Mathematical models of PQDs. *[4]*

#	Disturbance	Characteristic equation	Parameters
1	Normal	[1 ± α(u(t − t_1) − u(t − t_2))]sin(ωt)	α < 0.04, T ≤ (t_2 − t_1) ≤ 9T
2	Sag	[1 − α(u(t − t_1) − u(t − t_2))]sin(ωt)	0.1 ≤ α < 0.9, T ≤ (t_2 − t_1) ≤ 9T
3	Swell	[1 + α(u(t − t_1) − u(t − t_2))]sin(ωt)	0.1 ≤ α ≤ 0.8, T ≤ (t_2 − t_1) ≤ 9T
4	Interruption	[1 − α(u(t − t_1) − u(t − t_2))]sin(ωt)]	0.9 ≤ α ≤ 1, T ≤ (t_2− t_1) ≤ 9T
5	Harmonics	$\alpha_1$sin(ω_t) + $\alpha_3$sin(3ω_t) + $\alpha_5$sin(5ω_t) + $\alpha_7$sin(7ω_t)	0.05 ≤ α_3, α_5, α_7 ≤ 0.15, ∑(α_i^2 i) = 1
6	Flicker	[1 + $\alpha_f$sin(βω_t)]sin(ω_t)	0.1 ≤α_f ≤ 0.2, 5 ≤ β ≤ 20Hz
7	Oscillatory Transient	sin(ω_t) + α− (t− t_1)/τ sin(ω_n (t − t_1))(u(t_2) − u(t_1))	0.1 < α ≤ 0.8, 0.5T ≤ (t_2 − t_1) ≤ 3T, 8 ≤ τ ≤ 40, 300 ≤ 2πω_n ≤ 900
8	Impulsive Transient	[1 − α(u(t − t_1) − u(t − t_2))]sin(ω_t)	0.1 ≤ α ≤ 0.414, T/20 ≤ (t_2 − t_1) ≤ T/10
9	Notch	sin(ω_t) − sign(sin(ω_t))×$\sum_{n=0}^{9}$ k[u(t − (t_1 − 0.02n)) − u(t − (t_2 − 0.02n))]	0 ≤ t_1, t_2 ≤ 0.5T, 0.1 ≤ K ≤ 0.4, 0.01T ≤ t_2 − t_1 ≤ 0.05T
10	Spike	sin(ωt) + sign(sin(ωt))× $\sum_{n=0}^{9}$ k[u(t − (t_1 − 0.02n)) − u(t − (t_2 − 0.02n))]	0 ≤ t_1, t_2 ≤ 0.5T, 0.1 ≤ K ≤ 0.4, 0.01T ≤ (t_2 − t_1) ≤ 0.05T
11	Sag+harmonics	[1 − α(u(t − t_1) − u(t − t_2))] × [$\alpha_1$sin(ωt) + $\alpha_3$sin(3ω_t) + $\alpha_5$sin(5ω_t) + $\alpha_7$sin(7ω_t)]	0.1 ≤ α < 0.9, T ≤ (t_2 − t_1) ≤ 9T, 0.05 ≤ α_3, α_5, α_7 ≤ 0.15, ∑(α_i^2) = 1
12	Swell+harmonics	[1 + α(u(t − t_1) − u(t − t_2))] × [$\alpha_1$sin(ωt) + $\alpha_3$sin(3ω_t) + $\alpha_5$sin(5ω_t) + $\alpha_7$sin(7ω_t)	0.9 ≤ α ≤ 1, T ≤ (t_2 − t_1) ≤ 9T
13	Interruption+harmonics	[1 − α(u(t − t_1) − u(t − t_2))] × [$\alpha_1$sin(ω_t) + $\alpha_3$sin(3ω_t) + $\alpha_5$sin(5ω_t) + $\alpha_7$sin(7ω_t)]	0.05 ≤ α_3, α_5, α_7 ≤ 0.15, ∑(α_2 i) = 1 0.1 ≤ α_f ≤ 0.2, 5 ≤ β ≤ 20
14	Flicker+harmonics	[1 + $\alpha_f$sin(βω_t)] × [$\alpha_1$sin(ω_t) + $\alpha_3$sin(3ω_t) + α5sin(5ω_t) + $\alpha_7$sin(7ω_t)]	0.05 ≤ α_3, α_5, α_7 ≤ 0.15, ∑(α_2 i) = 1 0.1 ≤ α_f ≤ 0.2, 5 ≤ β ≤ 20
15	Flicker+sag	[1 + $\alpha_f$sin(βω_t)][1 − α(u(t − t_1) − u(t − t_2))]sin(ωt)	0.1 ≤ α ≤ 0.9, T ≤ (t_2 − t_1) ≤ 9T 0.1 ≤ αf ≤ 0.2, 5 ≤ β ≤ 20
16	Flicker+swell	[1 + $\alpha_f$sin(βω_t)][1 + α(u(t − t_1) − u(t − t_2))]sin(ω_t)	0.1 ≤ α ≤ 0.8, T ≤ (t_2 − t_1) ≤ 9T

4.3 Dataset generation

Using the previous dataset generator [4] simulated on MATLAB software R2021a, we generated the sixteen classes with 4800 signal for each class. That left us with 76800 signals total. We generated three type of datasets: pure (noiseless), with 30dB noise and with random noise. The random noise is an additive white Gaussian noise with signal-to-noise ratio (SNR) between 20-50 dB. The sampling rate for all signals is 3.2 kHz, and the nominal frequency was set to 50Hz with length of 10 cycles, i.e. 0.2 s, 640 sampling points.

By manipulating the constraints parameters, an infinite number of data can be generated. Table 4-2 shows the specific parameters we chose for our dataset.

Table 4-2 Table showing the contstraint parameters chosen for each disturbance.

Disturbance	Parameter
Sag	α=0.42
Swell	α=0.4
Interruption	α=0.94
Harmonics	α_3=0.09, α_3=0.11, α_7=0.13
Flicker	α_f=0.14, β=17.5
Oscillatory Transient	α=0.4, T=20, F_n=400
Impulsive Transient	α=0.414
Notch	K=0.4
Spike	K=0.4
Sag+harmonics	α=0.4, α_3=0.09, α_5=0.11, α_7=0.13
Swell+harmonics	α=0.4, α_3=0.09, α_5=0.11, α_7=0.13
Interruption+harmonics	α=0.94, α_3=0.09, α_5=0.11, α_7=0.13
Flicker+harmonics	α_f=0.14, β=1.75, α_3=0.09, α_5=0.11, α_7=0.13
Flicker+sag	α=0.92, α_f=0.16, β=10
Flicker+swell	α=0.42, α_f=0.18, β=12.5

It is also worth mentioning that the disturbances accure for duration of 20 ms only and for some signals for only 1 second.

The dataset came in .mat format, so in order to efficiently use it we had to convert the .mat files into .csv files. We used MATLAB code for the conversion.

4.4 Results and discussion

We split our data, 76800 sample, such that 20% is used for testing, 33% for validation and the rest 47% is for training.

The previously mentioned model was used for the training with parameters of 150 epoch and 32 batch size. We obtained 20,511,584 trainable parameters. The hardware parameters are the ones offered by kaggle: CPU with 13GB and GPU P100 with 15.9GB memory.

In order to calculate the accuracy for each type of disturbance, we apply the previously mentioned equation in Chapter three.

Table 4-3 summarizes the accuracy in each dataset while table 4-4 shows the accuracy of the 16 disturbances in each dataset (noiseless, 30 dB noise, random noise).

These values were calculated through applying the above equation on the confusion matrix of each dataset. The confusion matrices are shown in figure 4-3, 4-4, and 4-5 for noiseless, 30dB, and random noise datasets respectively.

The types of disturbances are labelled from 0 to 15. The labelling is shown in the table 4-4.

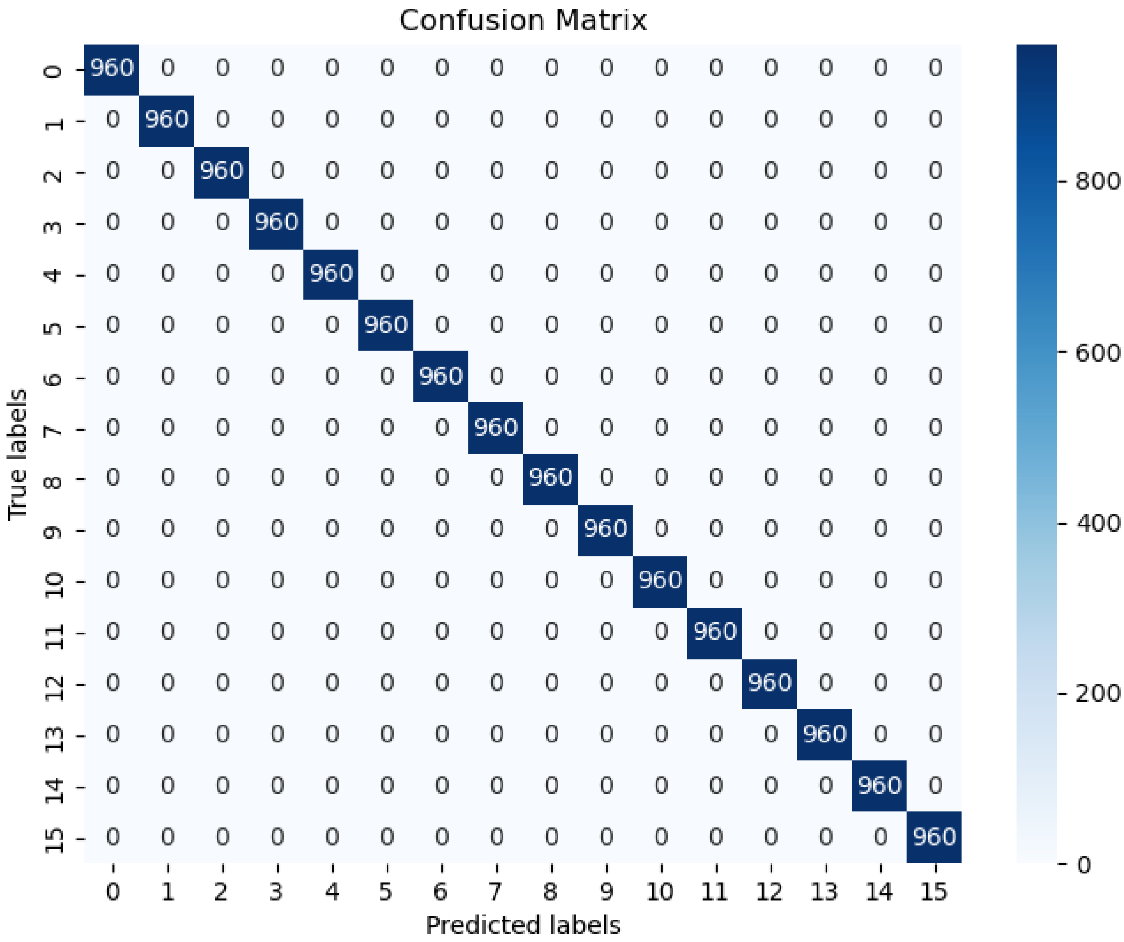


Figure 4-3 Confusion matrix for noiseless dataset

The noiseless dataset was the easiest one to train since the signals were pure, i.e. ideal case, so the model with that many layers easily classified the disturbances with an accuracy of 100% for each class and 100% average accuracy.

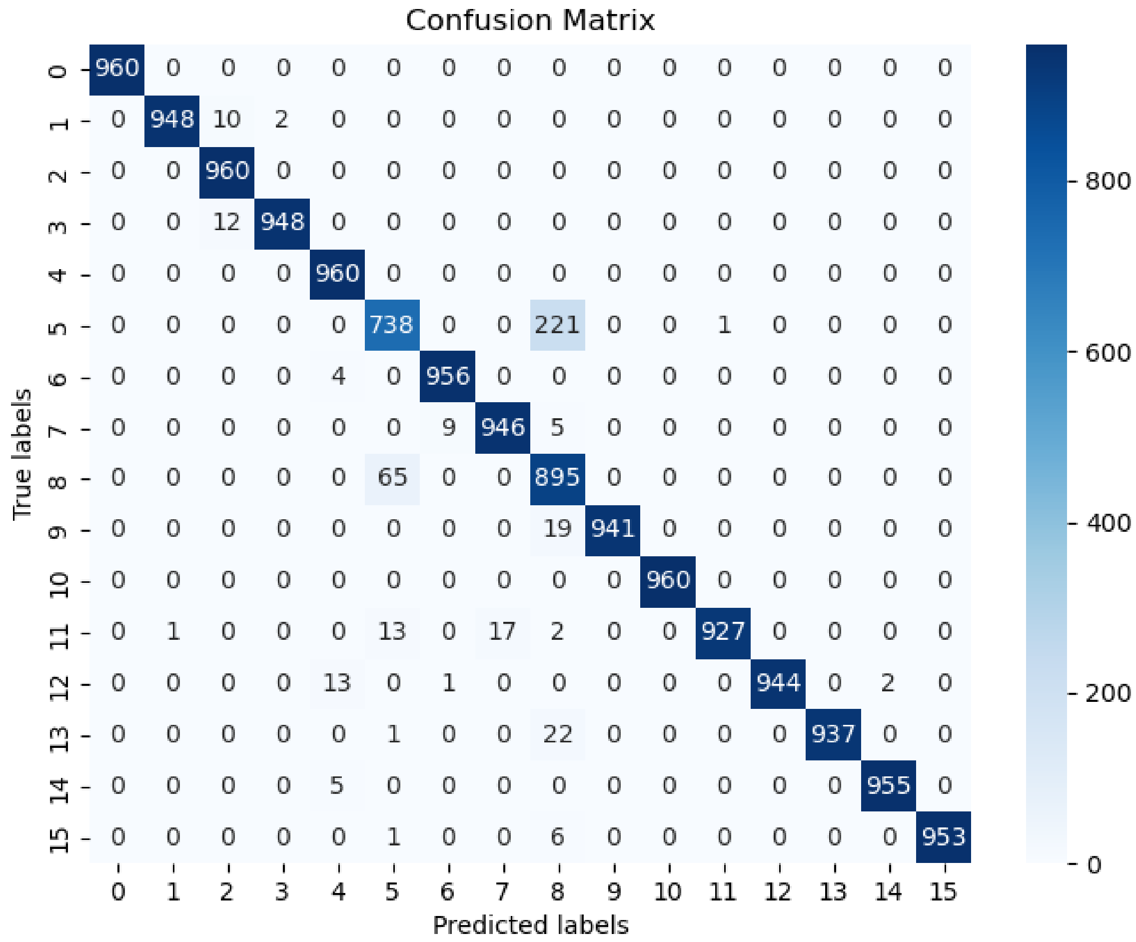


Figure 4-4 Confusion matrix for 30dB dataset

From the confusion matrix and results of the accuracy in the table, we can sense the effect of the added noise comparing to the noiseless signals. We can see some minor misclassification but the one must be mentiond is the misclassification of 221 signals of the impulsive transient class with the normal class. Since impulsive transients are considered short-duration disturbances, we assume that adding noise to the signals made it tricky for the model to distinguish between the two signals which led to the misclassification.

Despite the misclassification, this data set had a good average accurancy of 97.18%.

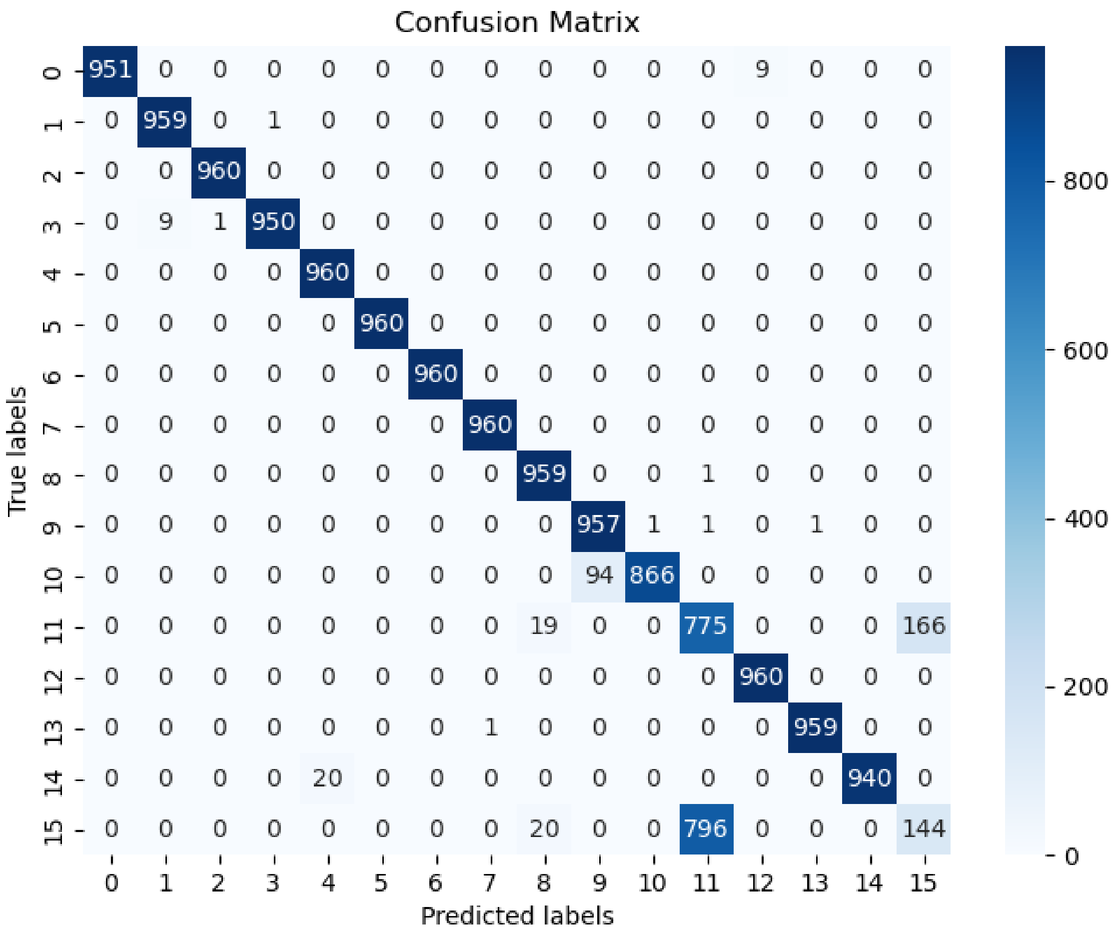


Figure 4-5 Confusion matrix for random noise dataset

The random noise dataset had an overall accuracy of 93%. The lowest accuracies were between class 11 (sag) and class 15 (swell). The model misclassified 166 sag signals for swell, and misclassified 796 swell signals for sag signal. The random noise ranged from 20-50 dB so the signals range from obscured signals to clearer signals. Even though these two types of signals have different characteristics in terms of their amplitude, where swell is a sudden increase in voltage or current while sag is sudden increase, adding random AWGN noise corrupted the signals which caused an overshadow to these characteristics making it challenging for the model to accurately detect the correct disturbance. Not to mention the short duration of the disturbances (20ms) making the changes in the signals somewhat rapid.

Table 4-3 Accuracy of the three datasets.

Dataset type	Accuracy
Noiseless	100%
30dB noise	97.18%
Random noise	93%

Table 4-4 Accuracy of 16 classes of disturbances in the three datasets.

Label	Type of disturbance	Accuracy		
		Noiseless	30dB	Random
0	Flicker harmonics	100%	100%	99.06%
1	Flicker sag	100%	98.75%	99.89%
2	Flicker	100%	100%	100%
3	Flicker swell	100%	98.75%	98.95%
4	Harmonics	100%	100%	100%
5	Impulsive transient	100%	76.87%	100%
6	Interruption harmonics	100%	99.58%	100%
7	Interruption	100%	98.54%	100%
8	Normal	100%	93.23%	99.89%
9	Notch	100%	98.02%	99.68%
10	Oscillatory transient	100%	100%	90.20%
11	Sag	100%	96.56%	78.84%
12	Sag harmonics	100%	98.33%	100%
13	Spike	100%	97.6%	99.89%
14	Swell harmonics	100%	99.48%	97.91%
15	Swell	100%	99.27%	15%

Another way to capture our model performance is through the learning curve. One of the most famous learning curves is loss over time. We decided to focus on improving the loss curve since it gives an insight into model behavior during the training process, as well as it gives a major indication on the generalization of the model. It is also an important tool to detect issues like data insufficiency, overfitting or underfitting. Focusing on improving this curve through tuning our model helped improve its performance. The figures : 4-4, 4-7, and 4-8 are the loss curves for the noiseless, 30dB noise, and random noise datasets respectively.

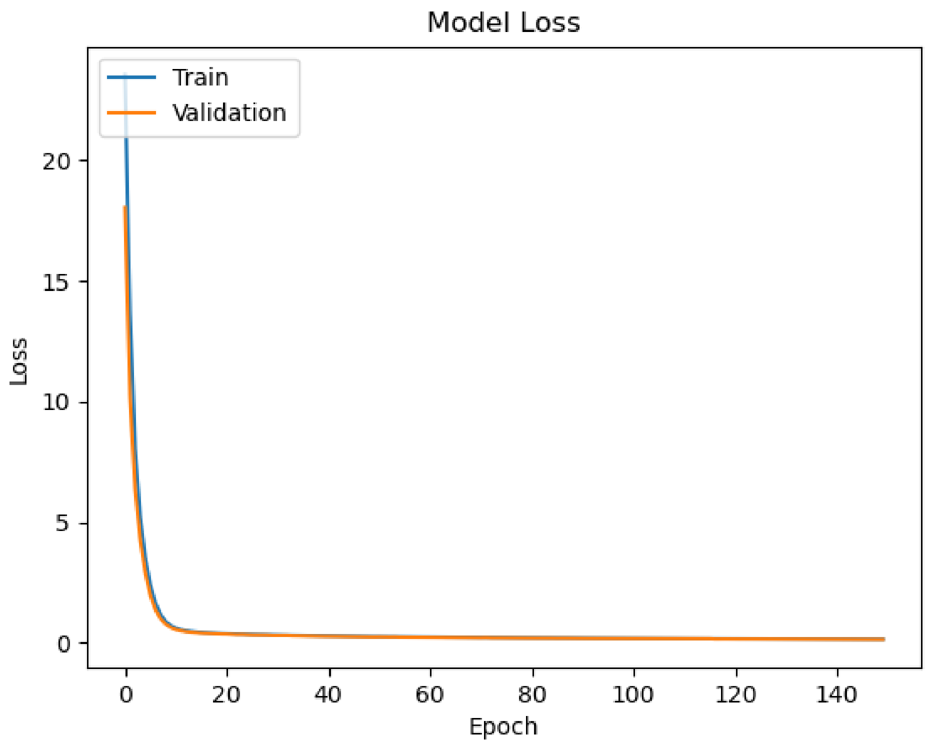


Figure 4-6 Loss curve for noieseless dataset.

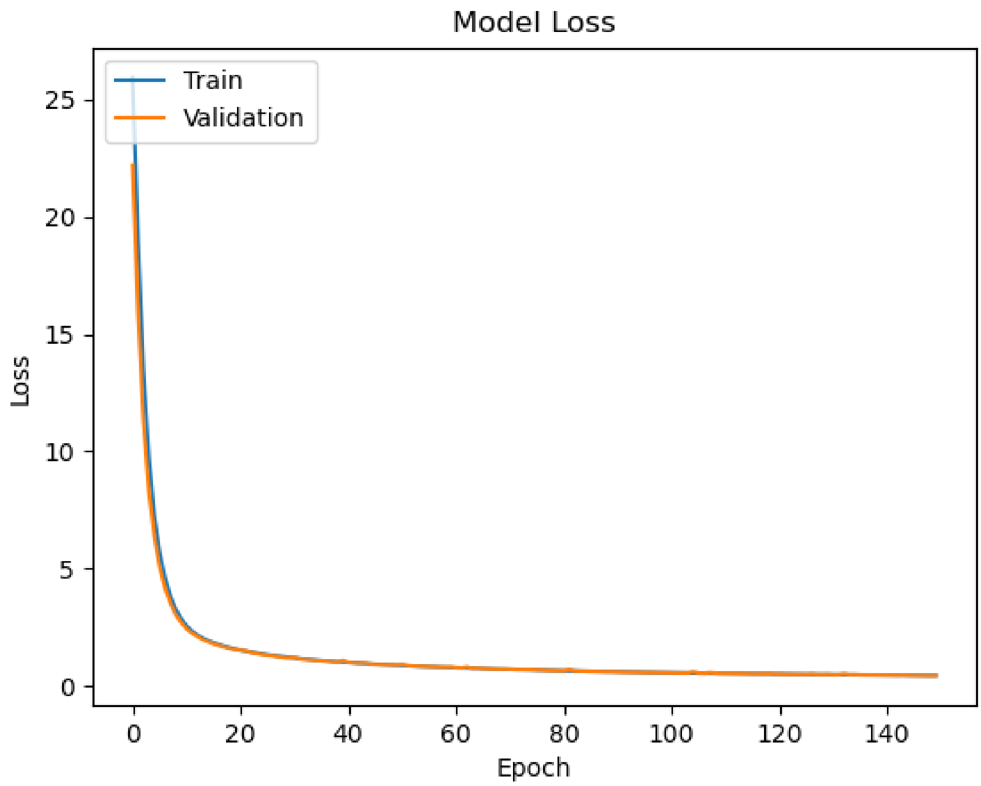


Figure 4-7 Loss curve for 30dB noise dataset.

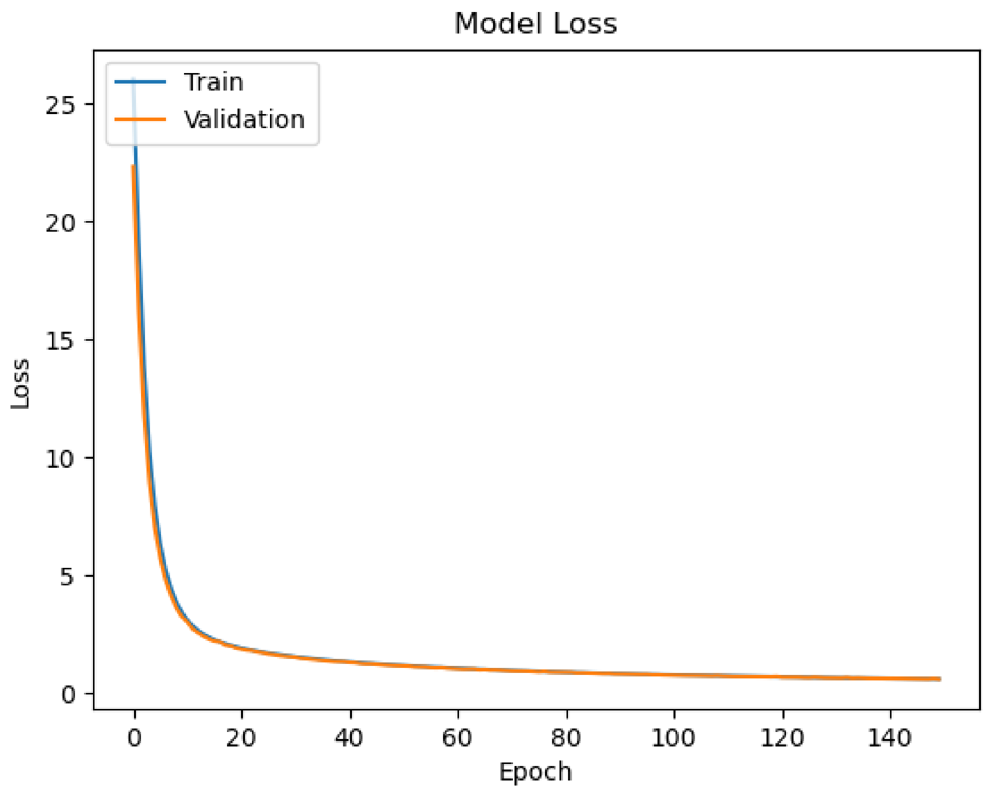


Figure 4-8 Loss curve for random noise dataset.

4.5 Results assessment

In order to evaluate our model performance on a deeper level, we will calculate the Key Performance indicators we mentioned before in chapter 3 section 7.

We already calculated the accuracy for each dataset using the confusion matrices. Now we will calculate the recall, precision and specificity using the same matrices according to the equation formulas mentioned before.

To calculate the above parameters, we need the value of FN, FP and TN which are also calculated through the confusion matrix such that:

FN= the sum of values in the class's corresponding row excluding TP.

FP= the sum of values in the class's corresponding column excluding TP.

TN= sum of all columns and rows excluding the class's corresponding column and row.

After calculating these values, we calculate the recall, precision, specificity and F1 for each class, then later calculate the average values of these parameters for every dataset.

The results are displayed in table 4-5, where: R: Recall, P: Precision, and S: Specificity.

From the table, we can clearly see that the model overall performance is high.

The first dataset showed ideal results in all four key performance indicators, due to the data being pure.

In the second dataset, the recall value of 0.97 indicates that the model is effective in minimizing the false negatives and has a low tendency to miss positive instances. While the precision value shows that the model has low rate of false positives and is able to make accurate positive predictions. The specificity value of 0.99 demonstrates a high ability to correctly identify negative cases, minimizing the number of false positives. F1 score of 0.99 also shows a high performance and good balance between minimizing false positives and false negatives.

As for the third dataset, overall results were also good but some of the classes, specifically swell class, showed a really low value for both recall and precision which ultimately affected the value of F1. This indicates that the model still have room for improvement.

Overall, the model showed promising results in its performance even in case of added noise, which highlights the potential of the model to be utilized in real-world applications.

Table 4-5 Different key performance indicators for the sixteen classes in the three dataset with their averages.

		Dataset type											
		Noiseless				**30Db**				**Random**			
Label	**PQD type**	**R**	**P**	**S**	**F1**	**R**	**P**	**S**	**F1**	**R**	**P**	**S**	**F1**
0	Flicker harmonics	1	1	1	1	1	1	1	1	0.99	1	1	0.99
1	Flicker sag	1	1	1	1	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99
2	Flicker	1	1	1	1	1	0.97	0.99	0.98	0.99	0.99	0.99	0.99
3	Flicker swell	1	1	1	1	0.98	0.99	0.99	0.99	0.97	0.99	0.99	0.98
4	Harmonics	1	1	1	1	1	0.97	0.99	0.98	1	0.97	0.99	0.98
5	Impulsive transient	1	1	1	1	0.76	0.9	0.99	0.83	1	1	1	1
6	Interruption harmonics	1	1	1	1	0.99	0.98	0.99	0.99	1	1	1	1
7	Interruption	1	1	1	1	0.98	0.98	0.99	0.98	1	0.99	0.99	0.99
8	Normal	1	1	1	1	0.93	0.77	0.98	0.84	0.99	0.71	0.99	0.83
9	Notch	1	1	1	1	0.97	1	1	0.98	0.99	0.91	0.99	0.95
10	Oscillatory transient	1	1	1	1	1	1	1	1	0.90	0.99	0.99	0.94
11	Sag	1	1	1	1	0.96	0.99	0.99	0.98	0.73	0.49	0.94	0.58
12	Sag harmonics	1	1	1	1	0.98	1	1	0.99	1	0.99	0.99	0.99
13	Spike	1	1	1	1	0.97	1	1	0.98	0.99	0.99	0.99	0.99
14	Swell harmonics	1	1	1	1	0.99	0.99	0.99	0.99	0.97	1	1	0.98
15	Swell	1	1	1	1	0.99	1	1	0.99	0.15	0.31	0.97	0.20
Average		1	1	1	1	0.97	0.97	0.99	0.97	0.91	0.89	0.99	0.90

4.6 Comparison with other work:

Table 4-6 Comparison of the accuracy of the proposed method with the accuracy of different methods.

Ref	Method	Dataset type		
		Noiseless	30dB	Random
[57]	2D CNN	97.67%	96.67%	-
	SqueezNet	72.33	66.67%	
	GoogleNet	80%	70.67%	
	ResNet-50	100%	98.3%	
[58]	2D CNN	-	0.95%	
[4]	BiLSTM	96%	-	98.13%
[59]	ST	94.7%	-	-
	EMD-HT	97.9%		
[10]	Extreme points	98.8%		-
[60]	WT and PSO	98%	93.6%	-
[61]	FFT and ANNs		95.65	-
[62]	FDST and DT	99.28%	97.49%	-
	Proposed method (1D CNN)	100%	97.18%	93%

To prominently show our model's performance, we did a small comparison between our proposed method and other well-known methods like BiLSTM, 2D CNN and ST…etc. The studies mentioned mostly worked with noiseless datasets or datasets with different degrees of AWGN noise than 30dB, which is way it was hard to find papers which worked with random added noise in its dataset.

From this comparison, it becomes evident that our proposed 1D CNN model displays impressive performance, attaining high accuracies. Furthermore, it maintains a balanced trade-off between computational efficiency and predictive power, distinguishing itself from more complex models.

4.7 Conclusion

From the results obtained from the training and evaluating of the proposed 1D CNN model, and after comparing it with other famous methodologies, we can easily say that this method proved its effectiveness in classifying in accurately identifying and classifying multiple power quality disturbances even when subjected to varying levels of noise. This model with further improvements can easily be applied in real life power system to make a step forward in the field of the smart grid and improve its reliability.

Chapter 5: Conclusion and future work

5.1 Conclusion

This work focused on the design of a classification model using a specific type of convolutional neural network, namely 1D CNN. In this method, three synthetic datasets were generated from an open-source data set generation tool, allowing the generation of sixteen different disturbances with controllable AWGN noise.

This research project successfully designed and implemented a 1D CNN-based classification model for power quality disturbances with high accuracies even in the presence of different level of noise. Through extensive evaluation and comparison with existing works, the effectiveness and performance of the proposed model were demonstrated in terms of model complexity, time efficiency, and memory usage. The findings contribute to the field of power quality disturbances classification and provide valuable insights for further research and practical applications.

5.2 Future work

Future work consists of:

- Improve the training time and memory usage even more.
- Measure the performance of the model on data with lower level of noise.
- Use the model for prediction with real-life data.

References

[1] S. Bhim, C. Ambrish and K. Al-Hadda, “Power quality problems and mitigation techniques”, John Wiley & Sons Ltd, United Kingdom, 2015, p. 579.

[2] S.Khokhar, A. Zin, A.S.Mokhtar, N. Ismail and N.Zareen, "Automatic Classification of Power Quality Disturbances: A Review," in *IEEE Student Conference on Research and Development (SCOReD)*, Malaysia, 2013.

[3] K. S. Manish and K. Rajiv, "Classification of power quality events - A review," *Electrical Power and Energy Systems,* vol. 43, pp. 11-19, 2012.

[4] R.Machlev, A.Chachkes, J.Beikov, Y.Beck and Y.Levron, "Open source dataset generator for power quality disturbances with deep-learning reference classifiers," *Electric Power Systems Research,* vol. 195, pp. 1-7, 2021.

[5] D.Saxena, K.S.Verma and S. Singh, "Power quality event classification: an overview and key issues," *International Journal of Engineering, Science and Technology ,* vol. 2, no. 3, pp. 186-199, 2010.

[6] K. Alexander and T. Marc, "Methods for Correction of Power-Quality Problems," in *Power Quality in Electrical Systems*, The McGraw-Hill Companies, Inc, 2007, pp. 110-111.

[7] S.Khalid and D. Bharti, "Power Quality Issues, Problems, Standars & their Effects in Iindustry with Corrective Means," *International Journal of Advances in Engineering & Technology,* vol. 1, no. 2, pp. 1-11, May 2011.

[8] "MIRUS International Inc." [Online]. Available: https://shorturl.at/vAEK. (Accessed: April 27, 2023)

[9] R. S. Blog, *Uninterruptible Power Supply: Why Your Data Center Needs It,* 2019. [Online]. Available: https://rb.gy/4rhq6. (Accessed: May 1,2023)

[10] Y. HOLDINGS, *Diesel Generators(Over 500kw).* [Online]. Available: https://rb.gy/9ihvp. (Accessed : May 1, 2023)

[11] Z. Feng, L. Di, C. Xiaoqiang and W. Ying, "Recognition of Hybrid PQ Disturbances Based on Multi-Resolution S-Transform and Decision Tree," *Tech Science Press,* vol. 120, no. 5, pp. 1133-1148, 2022.

[12] A. Amin, N. Mehdi and S. Mohsen, "Multiple power quality disturbances detection and classification," *Electrical Engineering,* vol. 104, pp. 2333-2343, 2022.

[13] I. Lee and P. Dash, "S-Transform-Based Intelligent System for Classification of Power Quality Disturbance Signals," *IEEE Transactions On Industrial Electronics,* vol. 50, no. 4, pp. 800-805, August 2003.

[14] S. Umamani and D. Sambit, "Detection and classification of power quality disturbances using GWO ELM," *Journal of Industrial Information Integration ,* vol. 22, 2021.

[15] W. Shouxiang and C. Haiwen, "A novel deep learning method for the classification of power quality disturbances using deep convolutional neural network," *Applied Energy,* vol. 235, pp. 1126-1140, 2019.

[16] O. Rchiard, E. Peter and G. David, in *Pattern Classification Second Edition*, New York, John Wiley & Sons, Inc, 2001.

[17] T. Sergios and K. Konstantinos, in *Pattern Recognition Fourth Edition*, Academic Press, Elsevier Inc, 2009.

[18] Z. Jun-Hai, Z. Su-Fang and W. Xi-Zhao, "An Overview of Pattern Classification Methodologies," in *the Fifth International Conference on Machine Learning and Cybernetics*, Dalian, 13-16 August 2006.

[19] W. Satosi, in *Pattern Recognition: Human and Mechanical*, New York, John Wiley & Sons.

[20] N. Andrew, "Machine Learning Specialization," 2022. [Online]. Available: [YouTube playlist]. Retrieved from https://youtube.com/playlist?list=PLkDaE6sCZn6FNC6YRfRQc_FbeQrF8BwGI.

[21] Y.Anzai, "Pattern Recognition and Machine Learning. 1st edition," 1992. [Online].

[22] "SuperData. "Pattern recognition: Overview and applications"," 2023. [Online]. Available: https://shorturl.at/et023. (Accessed: May 18, 2023)

[23] "V7. Pattern Recognition in Machine Learning [Basics & Examples]," 2022. [Online]. Available: https://www.v7labs.com/blog/pattern-recognition-guide. (May 21, 2023)

[24] A. K. Jain, R. P. W. Duin and Jianchang Mao, "Statistical pattern recognition: a review," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4-37, Jan. 2000.

[25] S. Priyanka and M. Kaur, "Classification in Pattern Recognition: A Review," *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 3, no. 4, pp. 298-306, April, 2013.

[26] “Pattern Recognition Tools," 2012. [Online]. Available: https://37steps.com/189/statisticalpr/#. (Accessed: May 24, 2023)

[27] L. Jie, S. Jigui and W. Shengsheng, "Pattern Recognition: An Overview," *International Journal of Computer Science and Network Security,* vol. 6, pp. 57-61, June2006.

[28] K. Anil, P. Robert and M. Jianchang, "Statistical Pattern Recognition: A Review," *IEEE Transactions On Pattern Analysis and Machine Intelligence,* vol. 22, no. 1, pp. 4-37, January 2000.

[29] K. Jayanta, B. Debnath and K. Tai-hoon, "Use of Artificial Neural Network in Pattern Recognition" *International Journal of Software Engineering and Its Applications,* vol. 4, no. 2, pp. 23-34, April 2010.

[30] M. Chris, "Neural networks and their applications," American Institute of Physics, 1994.

[31] "Medium. "How does a Neural Network work intuitively in code?," 2019. [Online]. Available: https://shorturl.at/nwSY9. (Accessed: May 30, 2023)

[32] C. B. James, "On the Relationship Between Neural Networks, Pattern Recognition and Intelligence," *International Journal of Approximate Reasoning ,* vol. 6, pp. 85-107, 1992.

[33] C. L.O and Y. L, "Cellular neural networks; Theory and Applications," *IEEE Transactions on Circuits and Systems,* vol. 35, no. 10, pp. 1267-1290, October 1988.

[34] A. Laith, W. Jinglan, J. Amjad, et *al*. "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data,* vol. 8, no. 53, pp. 1-74, 2021.

[35] "QuantDare. "What is the difference between deep learning and machine learning?"," 2019. [Online]. Available: https://shorturl.at/gAI12. (Jun 2, 2023)

[36] Z. Akos, R. Csaba, S. Peter and O. Leaon, "Overview of CNN Research: 25 Years History and the Current Trends," *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*.

[37] L. Zewen, L. Fan, Y. S. P. Wenjie and Z. Jun, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," *IEEE Transactions On Neural Networks and Learning Systems,* 2021.

[38] Y. Xijun, C. Yongjie, L. Airong, W. Xinwei, Z. Yinghao and H. Nan, "Parallel convolutional neural networktoward high efficiency and robuststructural damage identification.," *Structural Health Monitoring,* pp. 1-22, 2023.

[39] Y. Rikiya, N. Mizuho, K. G. D. Richard and T. Kaori, "Convolutional Neural Networks: an overview and application in radiology," *Insights into imaging,* vol. 9, pp. 611-629, 2018.

[40] M. ,. Amir, "Sefidian Academy," 2019. [Online]. Available: http://www.sefidian.com/2019/02/24/understanding-1d-2d-and-3d-convolutional-layers-in-deep-neural-networks/. (Accessed: Jun 5, 2023)

[41] K. Serhat, A. Kemal and C. Mete, "An overview of the activation functions used in deep learning algorithms," *Journal of New Results in Science,* vol. 10, no. 3, pp. 75-88, 2021.

[42] N. HamidReza and M. Vahid, "Novel CNN with investigation on accuracy by modifying stride, padding, kernel size and filter numbers," *Multimedia Tools and Applications,* vol. 82, pp. 23673-23691, 2023.

[43] P. Lutz, "Early Stopping- But When?," in *Neural Networks: Tricks of the Trade*, Germany, G.B. Orr, K.−R. Müller (Eds.), 1998, pp. 55-69.

[44] L. Sergey and S. Christian, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift." In *International conference on machine learning*, pp. 448-456. (2015)

[45] S. Nitish, H. Geoffrey, K. Alex, S. Ilya and S. Ruslan, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research,* vol. 15, pp. 1929-1958. (2014).

[46] "Dive into deep learning. "Dropout". [Online]. Available: https://d2l.ai/chapter_multilayer-perceptrons/dropout.html. (Accessed: Jun 10 2023)

[47] A. Ng, Machine Learning Yearning: Technical Strategy for AI Engineers, In the Era of Deep Learning (draft version), 2018.

[48] "Medium. "Machine Learning: A Gentle Introduction"," 2018. [Online]. Available: https://towardsdatascience.com/machine-learning-a-gentle-introduction-17e96d8143fc.

[49] K. Serkan, A. Onur, A. Osama, I. Turker, G. Moncef and J. Daniel, "1D convolutional neural networks and applications: A survey," *Mechanical Systems and Signals Processing,* vol. 151, 2021.

[50] T. Ayush, "Intuitive understanding of 1D, 2D, and 3D convolutions in convolutional neural networks. "Weights & Biases platform". [Online]. Available: https://shorturl.at/beAR9.

[51] "Stackoverflow.," 2021, [Online]. Available: https://stackoverflow.com/questions/66220774/difference-between-the-input-shape-for-a-1d-cnn-2d-cnn-and-3d-cnn.

[52] T. Alaa, Classification Assessment Methods, Germany: Elsevier, 2018.

[53] M. Ioannis, R. Ioannis, G. Ioannis, K. George, D. Anastasios and D. Nikolaos, Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem, Basel, Switzerland, 2021.

[54] "MATLAB," [Online]. Available: https://fr.mathworks.com/. [Accessed March 2020].

[55] S. Dipanjan, B. Raghav and S. Tushar, Practical Machine Learning with Python, 2018.

[56] python.org, "What is Python? Executive Summary," [Online]. Available: https://www.python.org/doc/essays/blurb/.

[57] B. Jason, "Machine Learning Mastery- Making Developers Awesome at Machine Learning," 2022. [Online]. Available: https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/.

[58] S. Rafael, Salles and F. Paulo, "The use of deep learning and 2-D wavelet scalograms for power quality disturbances classification," *Electric Power Qyqtems Research ,* vol. 214, no. 108834, 2014.

[59] L. Jinsong, L. Hao, W. Dengke and B. Tianshu, "Classification of Power Quality Disturbance Based on S-Transform and Convolution Neural Netwok," *Frontiers in Energy Research,* vol. 9, no. 708131, 2021.

[60] B. Biswal , M.Biswal , S.Mishra , R.Jalaja (2014) Automatic classification of power quality events using balanced neural tree. IEEE Trans Ind Electron 61(1):521–530.

[61] R. Hooshmand and A. Enshaee, "Detection and classification of single and combined power quality disturbances using fuzzy systems oriented by particle swarm optimization algorithm," *Electric Power Systems Research,* vol. 80, no. 12, pp. 1552-1561, 2010.

[62] F.A. S. Borges, R. A. S. Fernandes, I. N. Silva and C. B. S. Silva, "Feature Extraction and Power Quality Disturbances Classification Using Smart Meters Signals," in *IEEE Transactions on Industrial Informatics*, vol. 12, no. 2, pp. 824-833, April 2016, doi: 10.1109/TII.2015.2486379.

[63] M. Biswal and P.k. Dash, "Detection and characterization of multiple power quality disturbances with fast S-Transform and decision tree based classifier," *Digital Signal Processing,* vol. 23, no. 4, pp. 1071-1083, 2013.