Characterization of radioactivity signals by embedded machine learning

Ali Hachem

THESE DE DOCTORAT

NNT : 2024UPASP017

université PARIS-SACLAY

Characterization of radioactivity signals by embedded machine learning

Caractérisation en flux de signaux radiologiques par apprentissage automatique embarqué

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 576 : particules, hadrons, énergie et noyau : Instrumentation, image, cosmos et simulation (PHENIICS)
Spécialité de doctorat : physique nucléaire
Graduate School : physique. Référent : Faculté des sciences d'Orsay

Thèse préparée dans l'unité de recherche **Institut LIST** (Université Paris-Saclay, CEA), sous la direction de **Frédérick CARREL**, directeur de recherche, le co-encadrement de **Gwenolé CORRE**, ingénieur de recherche et **Yoann MOLINE**, ingénieur de recherche

Thèse soutenue à Paris-Saclay, le 11 mars 2024 , par

Ali Hachem

Composition du Jury

Membres du jury avec voix délibérative

Nicolas GAC Professeur des universités, Université Paris-Saclay	Président
Michel PAINDAVOINE Professeur émérite, Université Bourgogne Franche-Comté	Rapporteur & Examinateur
Antonin VACHERET Professeur des universités, Imperial College London	Rapporteur & Examinateur
Marie-Laure GALLIN-MARTEL Directrice de recherche, LPSC, Université Grenoble Alpes	Examinatrice
Jérôme GAUTHIER Ingénieur de recherche, CEA Saclay	Examinateur
Olivier LIMOUSIN Directeur de recherche, CEA Saclay	Examinateur

université PARIS-SACLAY
ÉCOLE DOCTORALE
Particules, hadrons, énergie et noyau : instrumentation, imagerie, cosmos et simulation (PHENIICS)

Titre : Caractérisation en flux de signaux radiologiques par apprentissage automatique embarqué

Mots clés : apprentissage automatique, système embarqueé, signaux radioactives

Résumé : Dans diverses applications, la classification des signaux de détecteurs de rayonnements revêt une importance cruciale. Cette thèse se concentre sur un cas d'utilisation spécifique et complexe, à savoir la discrimination des neutrons et des rayonnements gamma dans un scintillateur plastique organique, en utilisant l'apprentissage automatique embarqué. Les solutions explorées dans cette étude pourraient potentiellement être étendues à la discrimination d'autres types de radiations dans des détecteurs différents. Nous présentons tout d'abord une méthode pour créer des ensembles de données neutron-gamma étiquetés, acquis par un scintillateur organique. Ce point est crucial car toutes les sources de neutrons émettent des rayonnements gamma. Les modèles *Multilayer Perceptron* (MLP) et 1D *Convolution Neural Network* (CNN) supervisés sont entraînés et évalués avec les signaux bruts préparés en utilisant la méthode d'étiquetage. Le modèle 1D CNN surpasse le modèle MLP, qui, à son tour, surpasse l'état de l'art, en particulier pour les radiations à faible énergie ([100, 250] keVee). Une deuxième approche d'apprentissage basée sur l'extraction d'attributs a été explorée pour faire la discrimination, permettant à un signal d'être représenté par une dimension indépendante de la chaîne d'acquisition, facilitant ainsi l'utilisation de méthodes d'adaptation non supervisées. Les résultats montrent que les modèles supervisés sur les signaux bruts sont plus performants que l'approche basée sur les attributs extraits. Dans cette étude, l'attribut de "*Form Factor*" est exploré en tant que nouvelle méthode de discrimination, offrant des performances similaires à l'algorithme de l'état de l'art sans nécessiter d'ajustement de paramètres. Enfin, nous avons implémenté les modèles d'apprentissage proposés et l'algorithme de l'état de l'art sur *Field Programmable Gate Array* (FPGA), pour une discrimination en temps réel, en respectant une latence inférieure à la durée du signal. En prenant en compte la latence et la consommation de ressources comme une référence de comparaison, l'order des méthodes s'inverse.

Title: Characterization of radioactivity signals by machine learning implemented on an embedded system

Keywords: machine learning, embedded system, radioactivity signal

Abstract: In various applications, the classification of radiation detector signals is of crucial importance. This thesis focuses on a specific and complex use case, namely the discrimination of neutrons and gamma-rays in an organic plastic scintillator using integrated machine learning (ML). The solutions explored in this study could potentially be extended to the discrimination of other types of radiations in different detectors. We present a method for creating labeled neutron-gamma datasets acquired through an organic scintillator. This is critical as all neutron sources emit gamma-rays. Supervised Multilayer Perceptron (MLP) and 1D Convolution Neural (CNN) models are trained and evaluated with the prepared dataset using the labeling method. The 1D CNN model outperforms the MLP model, which, in turn, surpasses the state-of-the-art, especially for low-energy radiations ([100, 250] keVee). A second ML approach based on features extraction was explored for discrimination, allowing a signal to be represented by a dimension independent of the acquisition chain, thus facilitating the use of unsupervised adaptation methods. The results indicate that supervised models on raw signals perform better than the attribute-based approach. In this study, the "*Form Factor*" attribute is explored as a novel discrimination method, offering performance similar to the state-of-the-art algorithm without requiring parameter tuning. Finally, we implemented the proposed ML models and state-of-art algorithm on Field Programmable Gate Array (FPGA) for a discrimination on the fly, while maintaining latency below less than the signal duration. Considering latency and resource consumption as a basis for comparison, the order of the methods is reversed

There is no benefit to hope for from knowledge if it is not for the sake of supporting the oppressed and achieving justice on this earth

Today, we live in an age of advanced technologies and innovations. Nevertheless, the world continues to grapple with problems such as poverty, injustice, massacres and famines. Scientific progress is often exploited to satisfy the selfish desires and greed of individuals and institutions, instead of promoting the progress and development of humanity. My hope that this thesis, along with the competences and experiences I have acquired, will be used in the opposite direction, for a less unjust world.

Acknowledgments

In the name of ALLAH, the Beneficent, the Merciful. First, all praise is due to ALLAH, the Lord of the worlds, for all his help, grace, mercy, and the opportunities that He has given me, shaping me into the man I am today. Also, my praise to Him for His showers of blessings throughout my work, which enabled me to complete it successfully.

My appreciation and gratitude are extended by the following persons who contributed in making this study possible.

Olivier Limousin for following my work as a member of my thesis committee, for his advice and feedback.

Jérôme Gauthier for his invaluable advice and guidance, for his constant availability to discuss and exchange different ideas, for his constant inspiration to work efficiently on this thesis subject and achieve promising results.

My thesis director, Frédéric Carrel, for his efforts, his patience with me, his pedagogical and scientific qualities, as well as his frankness and sympathy.

My thesis supervisors, Gwenolé Corre and Yoann Moline, for their valuables feedback, constant encouragement and motivations, for providing me with all the necessary resources and equipment during these three years, for affording me the opportunity to supervise a trainee, which was a valuable experience for me.

My colleagues, Mathieu Trocme and Aly Elayeb, for their help to implement the experimental setups of this thesis.

Aya Kanj and Clement Lynde for the time they granted me to discuss and exchange different ideas in the field of particle physics, for generously providing some of the datasets necessary for my research, and for their assistance in deriving certain experimental results within this study.

My intern Imane Belalchheb for her participation in this work and who, I hope, benefited from her internship under my supervision and that of Gwenolé Corre and Yoann Moline.

The members of LCAE team, without whom the completion of this work would not have been possible, for their availability and for all the knowledge and distinct skills I have acquired from them.

Members of the jury, for the time and effort they provided in evaluating this work.

Finally, because the person I am today and who has achieved this work is the result of a myriad of complex and interconnected circumstances and events that I have been through, along with a few personal decisions and a little individual effort. Because every person I have met, every event, every circumstance has played a role in shaping who I am today and who I will be tomorrow, and because each of them has a share of responsibility in what I have achieved and what I will achieve. I would like to express my gratitude, appreciation and sincerest thanks to my parents who bore the burden of raising and educating me, my brothers and sisters, my whole family, the people of my village, my professors, my teachers, my colleagues and my friends. To the people of my home country, Lebanon, who have given and sacrificed to create a safe, secure, and free environment, in which I had the opportunity to grow, learn and become the person I am today. To everyone has contributed directly or indirectly to my personality and has taught me something in one way or another. I would also like to thank my wife, Racha, who resides in my heart and is the sanctuary of my soul. I want to thank her for her support and patience, which enabled me to achieve this work.

Contents

Acknowledgements **iii**

List of Figures **xi**

List of Tables **xiv**

List of Abbreviations **xv**

Résumé **1**

Introduction **5**
Context and Challenges 5

1 Presentation of Radioactivity Signals Classification **9**
1.1 Overview of Radiation Detectors 9
1.2 Scintillation Measurement Chain 12
1.2.1 Scintillator 12
1.2.2 Photodetectors 13
1.2.2.1 Photo-Multiplier Tube 14
1.2.2.2 Silicon Photomultiplier 14
1.2.2.3 Photomultiplier Tube vs Silicon Photomultiplier 15
1.2.3 Analog Signal Processing 15
1.2.4 Analog-to-Digital Converter 16
1.2.5 Digital Signal Processing 17
1.3 Neutron/Gamma-ray Discrimination in Organic Scintillator 19
1.3.1 Tail to Total Integral Ratio PSD Algorithm 20
1.3.2 Machine Learning for Neutron/Gamma-ray Discrimination in Organic Scintillator 22
1.4 Evaluation Metrics for Neutron/Gamma-ray Discrimination Algorithms 25
1.4.1 Figure of Merit 25
1.4.2 Receiver Operating Characteristic Curve 26
1.4.3 ROC vs FOM 28
1.5 Conclusion 29

2 Signal Analysis and Processing for Clean Labeled Neutron/Gamma-ray datasets **31**
2.1 Discrimination Performance According to SNR, Sampling Frequency and Vertical Resolution 31

2.1.1 Experimental Setup . . . 32
2.1.1.1 Signal Alignment . . . 32
2.1.2 Signal Frequency Analysis . . . 33
2.1.3 Signal to Noise Ratio . . . 34
2.1.4 Vertical Resolution . . . 36
2.1.5 Dataset Acquisition . . . 36
2.1.6 Experimental Results . . . 37
2.1.7 Impact of Pulse Shaping Amplifier . . . 41
2.1.7.1 Data Acquisition . . . 43
2.1.7.2 Results . . . 44
2.2 Labeling Method for Neutron/Gamma-ray Datasets in Organic Scintillator . . . 45
2.2.1 Time of Flight Setup . . . 46
2.2.2 Pile-up Detection & Evaluation . . . 47
2.2.2.1 Detection Method . . . 49
2.2.2.2 Evaluation of the Detection Method . . . 51
2.2.3 Processing & Labeling ToF Dataset . . . 51
2.2.4 Identifying & Removing Mislabeled Samples . . . 53
2.2.5 Contributions and Limitations . . . 55
2.3 Conclusion . . . 55

3 Supervised ML for Radiations Classification: A Comparison with a State of the Art Discrimination Algorithm 59
3.1 Introduction . . . 59
3.2 Multi-Layer Perceptron Model . . . 61
3.3 1D Convolution Neural Network . . . 64
3.4 Data Preparation . . . 65
3.5 Energy Calibration . . . 67
3.6 Experimental Results . . . 68
3.6.1 MLP vs. TTT_{ratio} Discrimination Algorithm . . . 69
3.6.1.1 MLP vs TTT_{ratio}: According to Incident Energy Variation . . . 69
3.6.1.2 MLP vs. TTT_{ratio}: According to Sampling Frequency . . . 70
3.6.2 MLP vs 1D CNN . . . 71
3.7 Features Extraction . . . 73
3.7.1 Definitions of Extracted Features . . . 74
3.7.2 Random Forest Model for Training on Extracted Features . . . 75
3.7.3 Performance on raw signals vs. Features Extracted . . . 77
3.7.4 Features Selection . . . 78
3.8 Form Factor: A Novel Pulse Shape Discrimination Method . . . 79
3.8.1 Dataset Acquisition . . . 79
3.8.2 Gaussian Mixture Model . . . 82
3.8.3 Experimental results . . . 82
3.9 Conclusion . . . 84

4 Pulse Shape Discrimination using embedded Neural Networks: First Steps for Developing a Specific Electronic Architecture **87**
4.1 Introduction to FPGA 88
4.2 Configuration and Implementation Process 90
4.3 Quantization of TTT_{ratio}, Form Factor, MLP and 1D CNN Discrimination Methods . . 92
4.4 Tail to Total Integral Ratio (TTT_{ratio}) Implementation 93
4.5 Form Factor Implementation 94
4.6 MLP implementation 95
4.7 1D CNN implementation 99
4.8 Conclusion 102

5 Conclusions and Future Developments **103**
Perspectives 106

Personal Bibliography **107**

Bibliography **107**

List of Figures

1.1 GN TRACKER probe developed by CEA-List institute. . . . 11
1.2 Key building blocks in a scintillation acquisition chain. . . . 12
1.3 Average of neutron and gamma-ray signals obtained by EJ276 plastic scintillator, ^{252}Cf (neutron and gamma-ray emission) and ^{60}Co (pure gamma-ray emission) sources. Neutrons emitted by ^{252}Cf source are separated from the gamma-rays using the labeling method explained in chapter 2. . . . 13
1.4 Illustration of a signal generated by the photodetector of a scintillation measurement chain. The acquisition window in this example contains five pulses corresponding to five radiations. . . . 13
1.5 Schematized view of a Photo-Multiplier Tube (PMT) displaying the focusing electrode, the dynodes, and the anode [11]. . . . 14
1.6 Schematized view on how the voltages are applied to each dynode of a PMT in order to generate electron multiplication [11]. . . . 14
1.7 Schematic of the parallel arrangement of photo-diodes with series quenching resistor in Silicon Photo-Multiplier (SiPM)s [11]. . . . 15
1.8 Pile-up example containing two pulses. . . . 17
1.9 Key blocks of an online neutron/gamma-ray discrimination system. . . . 19
1.10 Average of neutron and gamma-ray signals obtained by EJ276 plastic and BC-501A liquid scintillators, ^{252}Cf and ^{60}Co sources. Min-max normalization is applied on the average signals. . . . 20
1.11 Illustration of Q_{tail} and Q_{tot} integrals on the average of neutron and gamma-ray signals obtained by EJ276 plastic scintillator, ^{252}Cf and ^{60}Co sources. . . . 21
1.12 Bi-parametric graph of TTT_{ratio} according to Q_{tot} integral for a dataset obtained using EJ276 plastic scintillator and a ^{252}Cf source. . . . 21
1.13 Main parameters for a FOM calculation. . . . 26
1.14 The ROC space for a better and worse classifier [47]. . . . 27

2.1 Implemented acquisition chain based on the EJ276 plastic scintillator. . . . 33
2.2 Illustration of the base line, starting and ending points on a signal obtained by EJ276 plastic scintillator and a ^{252}Cf source. The sampling frequency and signal duration are 1.25 GHz and 500 ns, respectively. . . . 33
2.3 FFT power spectrum of 50,000 signals obtained by EJ276 plastic scintillator and ^{252}Cf. The acquisition was done at 1.25 GHz sampling frequency. . . . 34
2.4 Average of neutron and gamma-ray signals acquired at different SNRs, obtained by EJ276 plastic scintillator, ^{252}Cf and ^{60}Co sources. . . . 35

2.5 Illustration of the noise on a signal obtained using an EJ276 plastic scintillator and a ^{252}Cf source. . . . 36
2.6 Variation of FOM according to f_s for different SNRs (Table 2.3). A trade-off between the SNR and f_s is necessary to improve the discrimination performance. . . . 39
2.7 TTT_{ratio} distribution of neutron and gamma-ray signals acquired at different sampling frequencies and SNR levels. . . . 39
2.8 Variation of minimum sampling frequency according to SNR (equation 2.5). . . . 40
2.9 A scintillation acquisition chain with Operational Amplifier (Op-Amp) integrator. . . . 41
2.10 Ideal Op-Amp integrator circuit [53]. . . . 42
2.11 The implemented acquisition chain with Op-Amp amplifier. . . . 44
2.12 The main steps of the proposed labeling method. . . . 46
2.13 Diagram representing the configuration of ToF experiment. . . . 46
2.14 Implemented ToF experiment. . . . 47
2.15 Real detected pile-up events before and after the correlation with Gaussian kernel. . . . 48
2.16 ToF Distribution 52
2.17 Labeling of ToF output vector by K-mean algorithm. . . . 53
2.18 TTT_{ratio} according to total integral for each class of signals labeled by ToF parameter. . 53
2.19 TTT_{ratio} distribution of signals labeled as neutron by ToF. . . . 54
2.20 TTT_{ratio} distribution of gamma-rays emitted by ^{60}Co. . . . 55

3.1 An illustration of a ML model for neutron/gamma-ray discrimination, taking as input a raw signal. . . . 60
3.2 An example of a raw signal used for training MLP and 1D CNN models. The signal is obtained by EJ276 plastic scintillator and a ^{252}Cf source. The sampling frequency and signal duration are 250 MHz and 500 ns, respectively. The input length is 125 sampling points. . . . 60
3.3 An illustration of a ML model for neutron/gamma-ray discrimination, taking as input features extracted from a raw signal. . . . 60
3.4 Illustration of computation steps inside a single artificial neuron. The first step is calculation the the sum of multiplications between the neuron weights and the input. Then the bias term is added to the result. The last step is calculating the neuron output through the activation function. . . . 62
3.5 ReLU activation function used as an activation function in the hidden layers of a ML model. . . . 62
3.6 An illustration of two hidden layers MLP model. The numbers of neurons in the input, first hidden, second hidden and output layers are 3, 4, 3 and 2, respectively. . . . 63
3.7 Illustration of *1D* convolution operation. The number filter, the filter width, stride, number of input channels and input length are equal to 3, 3, 2, 2 and 7, respectively. . 65
3.8 Frequency response of Butterworth digital Finite Impulse Response filter. The window is *Kaiser* with β equal to 6 and length equal to $200 * f_s$. . . . 66
3.9 Examples of spectra obtained from ^{137}Cs and ^{22}Na sources with an EJ276 plastic scintillator and a PMT. . . . 67
3.10 TTT_{ratio} according to the total energy integral obtained from the labeled datasets at 250 MHz. The green lines correspond to the limits of the energy range [100 keVee, 1.2 MeVee]. 68

3.11 ROC curves obtained by MLP model and TTT_{ratio} discrimination algorithm on validation data, for different energy ranges, $f_s = 250$ MHz. The ROC curves for the energy ranges [500 keVee, 750 keVee] and [750 keVee, 1.2 MeVee] are superimposed. 70
3.12 ROC curves obtained by MLP model and TTT_{ratio} discrimination algorithm on testing data at different sampling rates. The figure on the right is a zoom based on the ROC curve on the left. 71
3.13 ROC curves obtained by MLP and 1D CNN models on validation dataset at 250 MHz. 72
3.14 A simple decision tree classifier with 4 features [91]. 76
3.15 RF algorithm with n decision trees [92]. 76
3.16 ROC curves obtained by RF, MLP and 1D CNN models on validation dataset at 250 MHz and energy range [100, 250] keVee. 77
3.17 The contribution score of each defined feature to the decision output. 78
3.18 Distributions of Form Factor, TTT_{ratio} and Pulse Indicator features extracted from each class of neutron/gamma-ray labeled dataset obtained by using an EJ276 plastic scintillator (sampling frequency at 250 MHz). 79
3.19 Distributions of Form Factor, TTT_{ratio} and Pulse Indicator features extracted from neutron and gamma-ray dataset obtained using the four employed acquisition chains. . . . 81
3.20 Distributions of Form Factor, TTT_{ratio} obtained with the second acquisition chain, using different values t_{long}, t_{short} and signal length. 83
3.21 Distributions of Form Factor, TTT_{ratio} obtained with the third acquisition chain, using different values t_{long}, t_{short} and signal length. 84

4.1 Key blocks of an online neutron/gamma-ray discrimination system. 88
4.2 The proposed workflow to translate a ML model into a FPGA implementation using Qkeras and Vitis HLS. 92
4.3 The three main operations in the trained MLP model. 96
4.5 Adder tree structure 96
4.4 Illustration of the first loop unrolling in dense layer. 97
4.6 The inference flow of the trained MLP model. 98
4.7 The inference flow of the trained 1D CNN model. 99
4.8 Illustration of 1D CL loops unrolling 100

List of Tables

1.1 Overview of the main benefits and drawbacks of ML methods for neutron/gamma-ray discrimination. . . . 25
1.2 Evaluation metrics for binary classification model. . . . 28

2.1 Acquisition setup configurations. Each configuration provides a dataset with specific quantization step and sampling frequency (f_s). . . . 37
2.2 Variation of SNR according to QS in each digitizer. . . . 38
2.3 FOM for different sampling frequencies, QS and SNR. . . . 38
2.4 FOM for QS = 0.1 mV/bin, different sampling frequencies and SNR. . . . 38
2.5 FOM for QS = 0.2 mV/bin, different sampling frequencies and SNR. . . . 39
2.6 Five different configurations of 2111 Timing Filter Amplifier. These configurations provide signals with different length, SNR and frequency power spectrum. . . . 44
2.7 FOM and SNR obtained on each configuration of the Timing Shaping Amplifier. . . . 44
2.8 Pile-up detection error (%) according to arrival time distance (nanoseconds) between the two contributions in a pile-up event. . . . 51

3.1 The corresponding Q_{tot} for different energy levels based on the result obtained by the simulation for the energy calibration. . . . 68
3.2 Classification report of the validation data acquired at sampling rate equal to 250 MHz. Accuracy is equal to 98%. . . . 69
3.3 The corresponding Q_{tot} for different energy levels based on the result obtained by the energy calibration. Column three represents the percentage of each energy range in the validation dataset (f_s = 250 MHz). . . . 70
3.4 Obtained accuracy, TPR and FPR with MLP on validation (testing) dataset at 250 MHz (125 MHz). . . . 71
3.5 Obtained TPR for FPR = 2% with MLP and 1D CNN models on validation dataset at 250 MHz. . . . 72
3.6 Equations of extracted features. . . . 75
3.7 TPR for FPR = 2% obtained by RF, MLP and 1D CNN models on validation dataset at 250 MHz and energy range [100, 250] keVee. . . . 78
3.8 Main components and parameters of the 4 employed acquisition chains. . . . 80
3.9 Optimal values of t_{long} and t_{short} obtained on the datasets acquired by the four employed acquisition chains. . . . 80
3.10 The results obtained by GMM fitted on Form Factor and TTT_{ratio} attributes. . . . 83

4.1 TPR for FPR = 2% obtained by TTT_{ratio} and Form Factor, MLP and 1D CNN algorithms using floating point and fixed point types on validation dataset for the energy range [100, 250] keVee. 92
4.2 RTL synthesis reports for the two proposed solutions in FPGA implementation of the TTT_{ratio} discrimination algorithm. 94
4.3 Comparing the RTL synthesis reports for the solutions proposed in the FPGA implementation of TTT_{ratio} and Form Factor neutron/gamma-ray discrimination algorithms. . . . 95
4.4 RTL synthesis reports for the two proposed solutions in the FPGA implementation of the trained MLP model. 99
4.5 Obtained unrolling factors for each layer in the trained 1D CNN model. 101
4.6 RTL synthesis reports for the solutions proposed in the FPGA implementation of 1D CNN, MLP, TTT_{ratio} and Form Factor neutron/gamma-ray discrimination methods. . 102

List of Abbreviations

1D CL One Dimension Convolution Layer

1D CNN One Dimension Convolution Neural Network

ADC Analog-to-Digital Converter

BLR BaseLine Restoration

BRAM Block RAM

CNN Convolution Neural Network

DSP Digital Signal Processing

FF Flip Flop

FFT Fast Fourier Transform

FNR False Negative Rate

FOM Figure of Merit

FPGA Field Programmable Gate Array

FPR False Positive Rate

GMM Gaussian Mixture Model

HDL Hardware Description Language

HLS High Level Synthesis

LUT Look-up Table

MAC Multiply–Accumulate

ML Machine Learning

MLP Multilayer Perceptron Neural Network

Op-Amp Operational Amplifier

PLBs Programming Logic Blocks

PMT Photo-Multiplier Tube

PSD Pulse Shape Discrimination

QS Quantization Step

ReLu Rectified Linear Unit

RF Random Forest

RMS Root Mean Square

ROC Receiver Operating Characteristic

RTL Register Transfer Level

SiPM Silicon Photo-Multiplier

SNR Signal to Noise Ratio

SVM Support Vector Machine

TNR True Negative Rate

ToF Time of Flight

TPR True Positive Rate

TTT_{ratio} Tail to Total Integral Ratio

V_{pp} Input Voltage Range

Résumé

Le domaine de l'instrumentation nucléaire couvre un large éventail d'applications, notamment le contrôle des réacteurs nucléaires, la sécurité des frontières et la médecine. Divers types de détecteurs ont été développés pour chacune de ces applications. Certains détecteurs sont dédiés à un type de rayonnement spécifique, comme le détecteur de neutrons rempli de gaz Hélium 3. D'autres détecteurs sont sensibles à plusieurs types de rayonnement. En gros, chaque rayonnement détecté est représenté par le signal de sortie acquis par le convertisseur analogique-numérique de la chaîne de mesure. Dans le cas du capteur multi-rayons, il est nécessaire de pouvoir discriminer chaque type de rayonnement détecté, ce qui peut être réalisé en classant le signal acquis en sortie. L'amélioration des performances de discrimination peut être obtenue grâce à deux approches principales. La première consiste à améliorer la capacité de discrimination du détecteur, souvent réalisée par des modifications chimiques. La deuxième approche implique l'exploration de nouvelles techniques de traitement du signal et de méthodes de discrimination. Cette thèse se concentre principalement sur cette dernière approche, en particulier sur la mise en œuvre d'outils d'apprentissage automatique embarqués pour améliorer les performances de discrimination pour les applications contraintes par le temps.

En considérant différentes familles de détecteurs, le signal de sortie d'intérêt est impacté par le type de rayonnement incident. Parmi ces capteurs de discrimination, une grande partie de la recherche s'est concentrée sur les scintillateurs plastiques, largement utilisés pour détecter les neutrons et les rayons gamma. Ces détecteurs organiques présentent plusieurs avantages pour la détection de rayonnement. Ils peuvent être facilement modelés, produire en grande volume et ont un coût relativement bas. De plus, ils ont une durabilité accrue, des caractéristiques de non-toxicité et de non-inflammabilité. Cependant, ils ont une faible capacité intrinsèque de discrimination par rapport à d'autres types de scintillateurs organiques et les résultats obtenus par les méthodes de discrimination de l'état de l'art sont limités en termes de performances, notamment pour les radiations de faible énergie. Cette limitation est principalement due à la légère différence entre les signaux de neutrons et de rayons gamma générés par un scintillateur plastique. De plus, le manque de jeu de données de neutrons de référence fiable rend l'évaluation de toute approche de discrimination proposée difficile. La limitation principale pour l'obtention de ce jeu de données de neutrons est l'absence d'une source de neutrons pure. Toutes les sources de neutrons existantes émettent des rayons gamma. De plus, la courte durée des signaux fournis par un scintillateur plastique, qui ne dure que plusieurs centaines de nanosecondes, présente un défi supplémentaire pour la mise en œuvre d'un système de discrimination en ligne. Pour une discrimination en temps réel, il est crucial d'obtenir une implémentation intégrée avec un temps d'inférence inférieur à la durée du signal afin de minimiser le pourcentage de détection de rayonnement manqué, notamment dans un scénario de taux de comptage élevé. En gros, aborder le problème de discrimination dans ce type de détecteur pour des applications en temps réel représente l'un des problèmes de classification les

plus complexes dans le domaine de l'instrumentation nucléaire.

Ce travail de thèse vise à évaluer l'intérêt des méthodes d'apprentissage automatique (Machine Learning (ML)) pour améliorer la classification des signaux de radioactivité. Le travail abordera d'abord la discrimination des neutrons/rayons gamma dans les scintillateurs plastiques en utilisant des outils ML, en raison de la complexité relativement élevée de ce problème par rapport à d'autres problèmes de classification de particules radioactives, comme expliqué dans le paragraphe précédent. Une fois qu'une solution aura été trouvée, elle pourrait être étendue afin de classer différents types de radiations détectées par d'autres familles de détecteurs. L'objectif final est de développer une approche ML intégrée pour des applications en temps réel.

À notre connaissance, cette thèse représente le premier travail explorant l'intérêt de l'apprentissage automatique pour la discrimination des neutrons et rayons gamma avec un scintillateur plastique pour des applications en ligne. L'étude implique une évaluation des performances de discrimination des modèles ML et des méthodes de l'état de l'art utilisant un scintillateur plastique, en tenant compte de différentes gammes d'énergie de radiation et de fréquences d'échantillonnage. De plus, cette recherche examine les mises en œuvre intégrées de ces méthodes sur des systèmes embarqués, évaluant leur faisabilité pour le déploiement dans un système de discrimination en ligne.

Tout d'abord, dans le chapitre 1, nous avons introduit le sujet de la classification des signaux de radioactivité en mettant l'accent sur la discrimination des neutrons et des rayons gamma dans les scintillateurs plastiques. Ensuite, nous avons présenté les différents composants d'un système d'acquisition de scintillation. Par la suite, nous avons analysé et discuté les principales limitations de l'état de l'art concernant la discrimination des neutrons et rayons gamma dans les scintillateurs plastiques. Cette étude a montré que l'évaluation des méthodes de discrimination reste une tâche difficile, principalement en raison des effets des émissions de rayons gamma dans les sources de neutrons. De plus, le coût et la taille d'un système de discrimination intégré sont proportionnels à sa fréquence d'échantillonnage. Par conséquent, il est important de déterminer la fréquence d'échantillonnage minimale nécessaire pour la discrimination afin d'optimiser l'implémentation embarquée. Ce point est crucial dans le cadre de cette étude pour intégrer des approches ML sur des dispositifs embarqués.

Par conséquent, pour mettre en place une configuration expérimentale pratique activée à la fréquence d'échantillonnage minimale nécessaire à la discrimination, le chapitre 2 a étudié comment la performance de discrimination d'un scintillateur organique varie en fonction du rapport signal sur bruit (Signal to Noise Ratio (SNR)) et de la résolution verticale à différentes fréquences d'échantillonnage. Les résultats de l'étude ont montré qu'un compromis entre la fréquence d'échantillonnage et le SNR est nécessaire pour optimiser la performance de discrimination. Augmenter le SNR peut conduire à une réduction de la fréquence d'échantillonnage minimale requise pour une discrimination efficace. De plus, les résultats expérimentaux ont indiqué que, en utilisant la même fréquence d'échantillonnage, un SNR plus élevé peut fournir une performance de discrimination supérieure. En d'autres termes, la performance d'une méthode de discrimination dépend des paramètres et des composants de la chaîne d'acquisition. Même de légères modifications dans les conditions d'acquisition peuvent avoir un impact significatif sur la performance de discrimination. Par conséquent, pour obtenir la meilleure performance de discrimination pour une application prédéfinie, il est essentiel de concevoir et d'optimiser soigneusement la configura-

tion de mesure. En plus, la comparaison de différentes approches de discrimination à travers différentes études peut être difficile en raison des complexités liées à l'assurance de conditions et d'environnements d'acquisition cohérents. Une information détaillée sur le SNR est nécessaire pour garantir une comparaison équitable.

L'étude des conditions d'acquisition a servi de base à la proposition d'une méthode pour générer des ensembles de données de neutrons et rayons gamma propres et étiquetés avec un scintillateur organique. C'est la deuxième partie du chapitre 2. Les ensembles de données préparés peuvent être utilisés pour entraîner et évaluer avec précision divers modèles d'apprentissage automatique et comparer différentes approches de discrimination. Une limitation principale de l'approche de étiquetage de signal proposée est sa dépendance à l'algorithme de discrimination Tail to Total Integral Ratio (TTT_{ratio}). Une autre limitation est sa dépendance à la fois à la fréquence d'échantillonnage et à l'énergie des radiations incidentes. De plus, par rapport à la méthode traditionnelle, la mise en œuvre de la chaîne de mesure de l'équitage est plus compliquée, et le processus d'acquisition prend plus de temps. En revanche, la méthode proposée est capable de produire des ensembles de données étiquetés même dans des situations où la capacité de discrimination de la chaîne d'acquisition utilisée est limitée.

Dans le chapitre 3, des modèles supervisés de réseaux de neurones (Multilayer Perceptron Neural Network (MLP) et One Dimension Convolution Neural Network (1D CNN)) ont été entraînés et évalués en utilisant des ensembles de données de neutrons/rayons gamma préparés par la méthode d'étiquetage proposée. Les ensembles de données ont été obtenus en utilisant un scintillateur plastique EJ276 et acquis à la fréquence d'échantillonnage minimale obtenue au chapitre 2 (250 MHz). Nous avons comparé les performances des modèles entraînés à celles de l'algorithme de discrimination de l'état de l'art TTT_{ratio}, en tenant compte des variations de fréquence d'échantillonnage et de niveaux d'énergie des radiations. Le modèle 1D CNN a montré des performances supérieures par rapport au modèle MLP, qui, à son tour, a surpassé l'algorithme TTT_{ratio}. La différence de performance la plus notable a été observée pour les radiations de faible énergie ([100, 250] keVee), soulignant l'intérêt d'utiliser des modèles ML pour les applications nécessitant la discrimination d'événements neutron/gamma de relativement faible énergie.

Il est crucial de souligner que l'algorithme de discrimination TTT_{ratio} peut présenter une performance de discrimination améliorée pour l'intervalle d'énergie [100, 250] keVee, en utilisant le même type de scintillateur (EJ276), lorsqu'on utilise des blocs de traitement de signal analogique dédiés, tels qu'un intégrateur d'Op-Amp dans la chaîne d'acquisition, comme le montrent les résultats présentés dans le chapitre 2. Néanmoins, un modèle ML supervisé entraîné sur les signaux 1D acquis peut toujours atteindre de meilleures performances de discrimination pour les radiations de plus faible énergie par rapport à TTT_{ratio}, comme le montrent les résultats obtenus dans cette thèse. La seule exigence est de répéter la préparation de l'ensemble de données étiqueté en utilisant le nouveau dispositif expérimental proposé, puis d'entraîner le modèle. En fait, la limitation principale de cette approche ML supervisée est sa sensibilité aux composants et paramètres de la chaîne de mesure. La répétition de la préparation de signaux de neutrons étiquetés propres pour tout changement subtil dans la chaîne d'acquisition présente un travail difficile et chronophage.

L'adaptation de domaine non supervisée peut être une solution pour surmonter cette limitation.

Cependant, un défi critique pour utiliser cette approche provient de la dimension d'entrée du modèle, qui dépend des caractéristiques de la chaîne de mesure. Pour résoudre ce problème, nous avons exploré dans le chapitre 3 une approche qui est basée sur l'extraction des attributs à partir des signaux bruts pour effectuer la discrimination. Cette approche peut réduire la dimension des données à un certain niveau indépendamment de la chaîne d'acquisition et de la longueur du signal d'entrée. Dans cette étude, plusieurs attributs ont été définies en tenant compte de notre application de discrimination. Les résultats obtenus ont indiqué que les modèles supervisés entraînés sur des signaux bruts fournissent des performances supérieures par rapport à l'approche supervisée reposant sur les attributs extraits. Notre exploration de l'approche d'extraction de attributs montre que l'attribut du facteur de Forme (Form Factor) peut être utilisé comme méthode de discrimination, atteignant les performances de l'algorithme TTT_{ratio} sans nécessiter d'ajustements de paramètres. Cette indépendance vis-à-vis des paramètres de réglage peut être cruciale dans certaines applications.

Dans le chapitre 4, nous avons comparé l'implémentation embarquée des approches de discrimination proposées (1D CNN, MLP, TTT_{ratio} et Facteur de Forme) pour être utilisées sur des systèmes portables pour des applications contraintes par le temps. C'est la dernière partie du travail présenté dans ce manuscrit, où les implémentations ont été réalisées sur Field Programmable Gate Array (FPGA). La décision d'utiliser un FPGA a été motivée par sa propriété de traitement parallèle et sa capacité à traiter le signal numérisé à une fréquence d'échantillonnage de l'ordre de plusieurs centaines de mégahertz (MHz), ce qui est une exigence clé dans ce travail en raison de la courte durée des signaux de neutrons et de rayons gamma générés par un scintillateur plastique. L'objectif était d'optimiser l'implémentation de chaque méthode afin d'obtenir une latence inférieure à la durée du signal tout en minimisant la quantité de ressources consommées. Basé sur la performance de discrimination comme critère d'évaluation, le modèle 1D CNN surpasse le modèle MLP, qui, à son tour, surpasse les algorithmes TTT_{ratio} et Facteur de Forme, surtout pour les radiations de faible énergie ([100, 250] keVee). De plus, les méthodes TTT_{ratio} et Facteur de Forme présentent des performances comparables, ce dernier ayant l'avantage de ne pas dépendre de paramètres de réglage. Cependant, lors de l'évaluation est basée sur la latence et la consommation de ressources, le classement des quatre méthodes est complètement inversé. Par conséquent, le choix de la méthode dépend de l'application cible et des ressources d'implémentation disponibles. Par exemple, dans des applications impliquant la classification de radiations de faible énergie, l'utilisation plus importante des ressources des modèles MLP et 1D CNN peut être justifiée. En revanche, pour la discrimination de radiations de plus haute énergie, les algorithmes de Facteur de Forme et TTT_{ratio} sont plus avantageux car ils atteignent la même performance de discrimination que les modèles ML avec une latence et une consommation de ressources plus faibles.

Introduction

Context and Challenges

The field of nuclear instrumentation covers a wide range of applications, including nuclear reactor control, homeland security and medicine. Various types of detectors have been developed for each of these applications. Some detectors are dedicated to specific radiation type, such as the neutron detector Helium 3 gas-filled. Other detectors are sensitive to several radiation types. Roughly speaking, each detected radiation is represented by the output signal acquired by the Analog-to-Digital Converter (ADC) of the measurement chain. In the multi-rays sensor case, it is necessary to be able to discriminate each type of detected radiation, which can be achieved by classifying the output acquired signal. Improving discrimination performance can be achieved through two main approaches. The first one involves enhancing the discrimination capability of the detector, often achieved through chemical modifications. The second approach involves the exploration of novel signal processing techniques and discrimination methods. This thesis mainly focuses on the latter approach, especially the implementation of embedded machine learning tools to enhance discrimination performance for time constraint applications.

Considering different families of detectors, the output signal of interest is impacted by the type of incident radiation. Among these discriminating sensors, a large part of the research has been focused on plastic scintillators, which are largely used to detect neutrons and gamma-rays. These organic detectors present several significant benefits for radiation detection. They can be easily shaped, produced in large volumes and have a relatively low-cost. In addition, they have increased durability, non-toxicity, and non-flammability characteristics. However, they have low intrinsic discrimination ability compared to other organic scintillator types and the results obtained by state of the art discrimination methods are limited in terms of performance, especially for low energy radiations. This limitation is mainly due to the slight difference between neutron and gamma-ray signals produced by an organic plastic scintillator. Moreover, the lack of reliable reference neutron dataset makes the evaluation of any proposed discrimination approach a challenging task. The main limitation for obtaining this neutron dataset is the absence of a pure neutron source. All the existing neutron sources are emitters of gamma-rays. Moreover, the short duration of signals provided by a plastic scintillator, which only lasts for several hundreds of nanoseconds, presents an additional challenge for implementing an online discrimination system. For a real time discrimination, achieving an embedded implementation with an inference time less than the signal duration is crucial to minimize the percentage of missed radiation detection, especially in a high count rate scenario. Roughly speaking, addressing the discrimination problem in this type of detector for real time applications represents one of the most complex classification problems in the field of nuclear instrumentation.

This PhD work aims at assessing the interest of ML methods to improve the classification of radioactivity signals. The work will address first the neutron/gamma-ray discrimination in plastic scintillators using ML tools, due to the relatively high complexity of this problem compared to other radioactive particle classification problem, as explained in the previous paragraph. Once a solution has been found, it could be extended in order to classify different types of radiations detected by other detector family. The final purpose is to develop an embedded ML approach for real time applications.

To the best of our knowledge, this PhD represents the first work in exploring the interest of ML for neutron/gamma-ray discrimination with a plastic scintillator for online applications. The study involves an assessment of the discrimination performance of ML models and state of the art method using a plastic scintillator, considering various radiation energy ranges and sampling frequencies. Furthermore, this research examines the integrated implementations of these methods on edge devices, evaluating their feasibility for deployment in an online discrimination system. The work in this thesis is structured as follows.

Chapter 1 details the context and challenges of this PhD by providing an overview of the different types of radiation detectors, describing the main components of a typical scintillation acquisition system, which can significantly affect discrimination performance. The chapter analyses the state of the art neutron/gamma-ray discrimination approaches and discusses the evaluation metrics used in the literature for these discrimination approaches. By comparing the embedded machine learning solutions that will be proposed during this work with the state of the art using these metrics, their performances can be evaluated and highlighted.

Chapter 2 presents the impact of vertical resolution, sampling frequency and SNR on the discrimination ability of a radiation measurement system. This study can help to implement a convenient experimental setup activated at the minimum sampling frequency needed for discrimination. This is a critical point to optimize the embedded implementation and reduce its cost and size. The second part of this chapter proposes a method to prepare a clean labeled neutron/gamma-ray datasets acquired by an organic scintillator. These datasets can be used to train and evaluate various ML models and compare different discrimination approaches.

Chapter 3 examines whether the use of ML tools, compared to state of the art method, can result in a decrease of the minimum required sampling frequency for discrimination in plastic scintillator, as well as an improvement in classification performance. The chapter presents two different ML discrimination approaches explored in this study. The proposed ML models are trained and evaluated using the labeled datasets generated through the labeling process proposed in chapter 2. The results are compared to state of the art according to the variation of sampling frequency and radiation energy.

Chapter 4 details the main steps involved in the embedded implementation of the discrimination methods introduced in chapter 3, along with the state of the art algorithm for classifying neutron and gamma-ray signals. As a result, a comparison can be made between these different discrimination approaches, taking into account their discrimination performance, execution time and resource consumption. The main objective is to achieve the embedded implementation with no significant degradation in terms of discrimination performance, while ensuring an execution time lower than the signal duration

and using a minimum amount of resources.

Finally, we will draw the main conclusions resulting from this thesis and future developments which could be planned in the frame of this research topic.

Chapter 1

Presentation of Radioactivity Signals Classification

This chapter introduces the classification of radioactivity signals. The study focuses on a specific and challenging use case, the discrimination of neutrons and gamma-rays in organic plastic scintillator. The solutions explored in the framework of this study could potentially be extended to the discrimination of other types of radiations in different detectors. Section 1.2 describes the key building blocks of a typical scintillation acquisition system, which can significantly affect discrimination performance. Section 1.3 presents a comprehensive overview and analysis of Pulse Shape Discrimination (PSD) and ML approaches for neutron/gamma-ray discrimination in an organic scintillator. Finally, section 1.4 discusses the evaluation metrics used in literature to assess the performance of these discrimination approaches. By comparing the proposed embedded machine learning solutions with the existing literature using these metrics, their performances can be evaluated and discussed.

1.1 Overview of Radiation Detectors

Different types of detectors, such as semiconductors, scintillators, and systems based on gaz ionization, have been developed to detect the different radiations. These detectors are used in a wide range of applications associated with various fields, including medical imaging, industrial operations, environmental monitoring, and homeland security [1]. Detecting and identifying these radiations is important for ensuring the safety of workers and the public, as well as protecting the environment from their harmful effects.

Gas ionization detectors, like Geiger-Mueller counters and ionization chambers, are used for detecting and measuring ionizing radiations, such as alpha particles, beta particles, and gamma-rays [1,2]. The gas molecules of a detector are ionized after interaction with these particles, resulting in the creation of positively charged ions and negatively charged electrons. Neutrons are electrically neutral particles and cannot directly ionize gas molecules. Hence, they cannot be measured by this type of detector. They can be indirectly detected by using specific gases, such as helium-3, which can capture neutrons and generate charged particles through nuclear reactions [3, 4]. The helium-3 gas ionization detector is notably the most effective approach for detecting neutrons, especially in terms of sensitivity [1]. However, a significant challenge to use these detectors is the high cost and the limited availability of helium-3 gas. Moreover, temperature, pressure, and humidity can influence their performances. Therefore, there

has been a need to explore alternative solutions.

Semiconductor detectors measure ionizing radiations using semiconducting materials, such as silicon or germanium. When ionization radiation interacts with these materials, it creates electron-hole pairs, which generate an electrical signal. Neutron detection in this type of detector is achieved indirectly by adding thin layers of specific converters, such as lithium-6 or boron-10. Both these materials generate charged particles when they are exposed to neutrons, which can be detected by semiconducting materials. Semiconductor detectors have high sensitivity and time resolution [1]. Furthermore, they can also be designed in compact and portable forms. Nevertheless, their performance can be influenced by temperature, they are costly, particularly for large volume detectors, and they are susceptible to radiation damage, especially for neutron detection. These limitations can affect their long-term stability and reliability.

Scintillator is a material capable of converting radiation into visible or ultraviolet light. Light sensors, such as Photo-Multiplier Tube (PMT) or Silicon Photo-Multiplier (SiPM), can then convert this light into an electrical signal. Scintillators may be broadly grouped into organic and inorganic materials.

- Inorganic scintillators are made of materials such as sodium iodide (NaI) or cesium iodide (CsI). They are available as single crystals, glasses and transparent ceramics.
- Organic scintillators primarily fall into two main subcategories, namely, liquid and plastic scintillators, which are made from organic compounds that contain both a fluorescent and a radiation-sensitive molecule. Liquid organic scintillators often use a solvent, such as toluene, to dissolve the molecules, while plastic scintillators typically use a solid matrix to hold them in place.

Inorganic scintillators have a higher density compared to organic scintillators. They provide also better energy resolution and higher light yields, which can be important for specific applications, such as gamma spectrometry. For instance, the NaI(Tl) inorganic scintillator has a density roughly three times greater than that of both EJ276 plastic and EJ309 liquid scintillators [5]. Moreover, the light yield of this inorganic scintillator surpasses that of its organic counterparts, reaching approximately three times the light yield of EJ276 plastic type and five times that of EJ309 liquid type [5]. In contrast, organic scintillators are easier to adapt to different designs, can be manufactured in larger sizes, and are more cost effective.

A radiation detector can be sensitive to more than one type of radiation, requiring the use of a discrimination method to identify the type of each detected radiation. Increasing the ability of detector to discriminate can help to improve the discrimination performance. This improvement is usually achieved through chemical modifications. Further, developing a novel discrimination approach can be another way to enhance the discrimination performance.

Organic scintillators are an example of multiple radiation types detectors, which have been developed to detect both neutrons and gamma-rays. As an illustration, the CEA List institute recently developed a new measurement system called GN TRACKER, illustrated in Figure 1.1. This detection system is based on a chemically modified plastic scintillator and aims to perform an online detection and classification of neutron and gamma-ray related events.

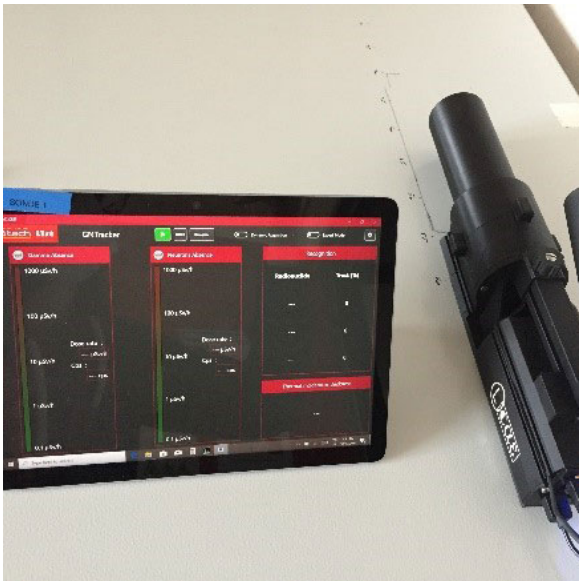


Figure 1.1: GN TRACKER probe developed by CEA-List institute.

Organic plastic scintillators have been preferred to other types of detectors in many applications due to their advantages for detecting radiations. They can be easily shaped and manufactured at a relatively low cost and high volumes. They are more durable, non-toxic and non-inflammatory [1]. However, the performance of state of the art for neutron/gamma-ray discrimination in these detectors is low compared to other organic scintillator types [6–8]. This low performance is mainly due to their limited discrimination abilities, which mainly depend on their chemical characteristics. Furthermore, signals produced by plastic scintillators have a relatively short duration, lasting only several hundred nanoseconds. This short duration can challenge the embedded implementation of a discrimination method for a classification in real time, especially in high count rate scenario. The inference time of the method should be less than the signal duration to avoid the miss detection of radiations, as it will be explained in section 1.2.5. Furthermore, the absence of a ground truth neutron dataset presents a challenge for accurately assessing any proposed discrimination approach. Neutron sources are also emitters of gamma-rays. Considering all these difficulties makes the discrimination problem in plastic scintillator one of the most challenging classification problems in the field of nuclear instrumentation, especially for real time applications.

The main objective of this thesis is to assess the potential of ML in enhancing the classification of radioactive particles, capable of being embedded on portable systems. The work will be dedicated first to explore the potential of ML tools compared to state of the art for neutron/gamma-ray discrimination problem in plastic scintillators, which is one of the most complex problem in the domain of radioactivity signals classification, as explained previously. Once a solution is identified, it can be extended to classify other types of radiation detected by different types of detectors. The final objective of this work is to develop an embedded ML approach for a classification on the fly, where the aim is to make an immediate decision after the acquisition of each pulse. Furthermore, the short duration of signals in plastic scintillators requires an acquisition at a sampling frequency on the order of several hundred megahertz (MHz) to acquire the information necessary for discrimination. The power consumption of an embedded discrimination system is proportional to its sampling frequency. Therefore, a study will

also be conducted to explore if ML could reduce the minimum sampling frequency required for discrimination compared to the state of the art. This can reduce the architecture size, power consumption, and complexity of an online discrimination system.

Next section introduces the main components in a scintillation measurement chain. The section explain the process from the interaction between the radiation and the detector, to the generation of the digital output signal.

1.2 Scintillation Measurement Chain

Figure 1.2 shows the key building blocks in a scintillation measurement chain. It is composed by a scintillator, which is connected to photo-detectors, such as PMTs or SiPMs. These sensors generally used in nuclear measurement chain convert the scintillation light into electrical signals. Depending on the application, the obtained analog signals can be digitized directly by an ADC, or go through different analog signal processing steps before digitization. Following the acquisition step, the obtained output signal can be processed by digital signal processing techniques to extract meaningful information required for the target application, such as identifying the source of each recorded signal, count rate, and energy of the incident radiation.

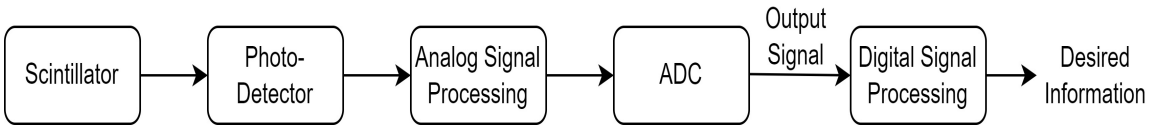


Figure 1.2: Key building blocks in a scintillation acquisition chain.

1.2.1 Scintillator

Scintillators are made from materials that can absorb radiations, and then emit light in response. When particles interact with the atoms of the scintillator material, the atoms become excited and subsequently release energy in the form of light photons as they return to their ground state. Roughly speaking, the scintillation process can be described into two steps. The process begins with a particle interacting with the scintillator material, depositing energy into it and causing the atoms or molecules to become excited. Subsequently, the excited atoms or molecules quickly return to their ground state, emitting photons in the process. The number of photons emitted is proportional to the amount of energy deposited by the incident particle.

More details about the scintillation mechanism in organic scintillators can be found in [9]. The authors in [10] explain the theory of neutron/gamma-ray discrimination in organic scintillator. The difference between two signals obtained by the interaction of the two particles arises from the density of the excited states generated by the ionizing particle. In general, proton and electron particles correspond to neutron and gamma-ray, respectively. Due to the lower ionization power of a recoiling electron compared to recoiling proton, the excitation density of the scintillator by an electron is less important. Therefore, the energy deposition of en electron produces less delayed de-excitation than that of a recoil

proton. Consequently, for equivalent energy deposition, the pulse resulting from a gamma-ray interaction is temporally less dispersed than a pulse generated by neutron interaction, as illustrated in Figure 1.3.

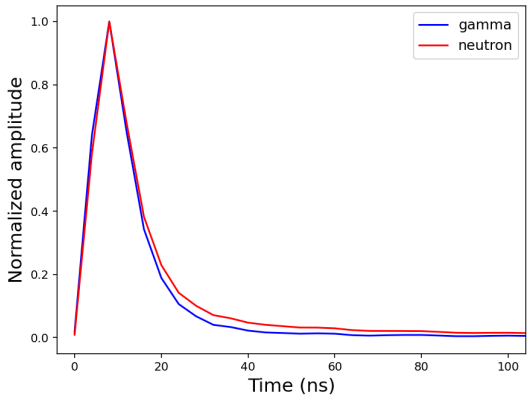


Figure 1.3: Average of neutron and gamma-ray signals obtained by EJ276 plastic scintillator, ^{252}Cf (neutron and gamma-ray emission) and ^{60}Co (pure gamma-ray emission) sources. Neutrons emitted by ^{252}Cf source are separated from the gamma-rays using the labeling method explained in chapter 2.

1.2.2 Photodetectors

The photodetector converts the light signal produced by the scintillator into an electrical signal. The output signal is in the form of a different pulses representing different incident radiations with fluctuating maximum amplitudes distributed randomly over time, as illustrated in Figure 1.4. The quality of this output signal is influenced by the type and characteristics of the photodetector. Sections 1.2.2.1 and 1.2.2.2 present the characteristics of PMTs and SiPMs photodetectors, which are the two primary photodetector types used in nuclear measurement chains. Section 1.2.2.3 compares the main advantages and disadvantages points of these two phototdetectors which can help in selecting one of them to be used in the experimental setups of this thesis.

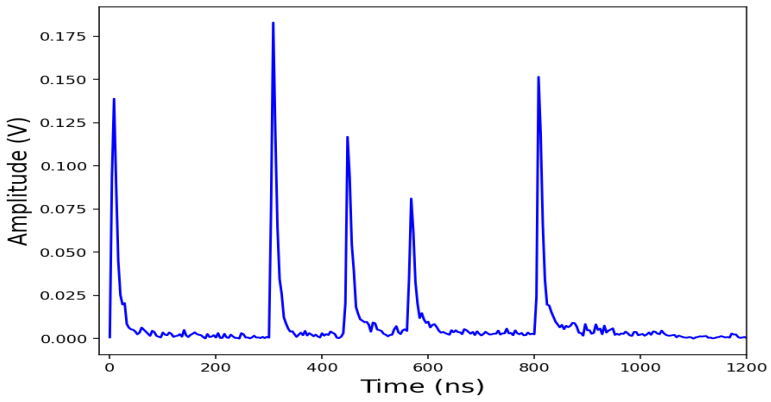


Figure 1.4: Illustration of a signal generated by the photodetector of a scintillation measurement chain. The acquisition window in this example contains five pulses corresponding to five radiations.

1.2.2.1 Photo-Multiplier Tube

PMTs have been commonly used for many years in scintillation based detectors because of their high sensitivity, fast response time, and ability to amplify very weak light signals. The basic structure of a PMT consists of a vacuum tube with a photocathode, several dynodes, and an anode, as shown in Figure 1.5. When light strikes the photocathode, electrons are emitted due to the photoelectric effect. These electrons are then accelerated by a high voltage applied between the photocathode and the first dynode, as illustrated in Figure 1.6. The first dynode is a metal electrode that is positioned close to the photocathode. As the electrons move toward the first dynode, they collide with other electrons and release more electrons through secondary emission, resulting in an amplification of the signal. The process of acceleration and secondary emission is repeated at each subsequent dynode, resulting in a cascade of electrons that produces a large output signal at the anode. The output signal is proportional to the amount of light striking the photo-cathode, and can be amplified and measured using appropriate electronic components.

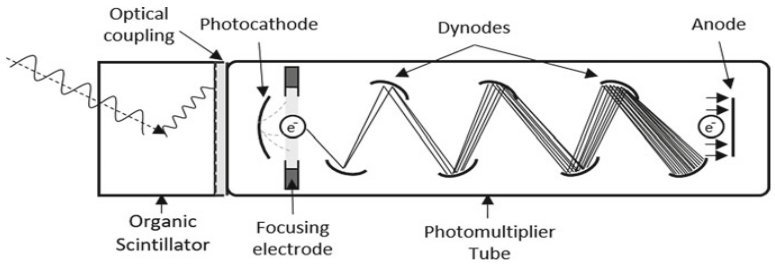


Figure 1.5: Schematized view of a PMT displaying the focusing electrode, the dynodes, and the anode [11].

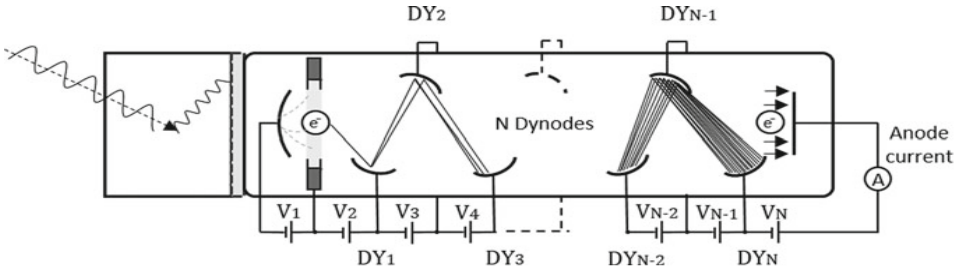


Figure 1.6: Schematized view on how the voltages are applied to each dynode of a PMT in order to generate electron multiplication [11].

1.2.2.2 Silicon Photomultiplier

In recent years, SiPMs have emerged as an alternative to PMTs for scintillation detection. The basic structure of a SiPM consists of a silicon substrate with an array of small photo-diodes connected in parallel with series quenching resistor, see Figure 1.7. Each photo-diode operates in reverse bias, which means that a voltage is applied across the diode in the opposite direction to its normal operation. When a photon strikes a photo-diode, it creates an electron-hole pair, which generates a small current. The

current generated by each photo-diode is proportional to the amount of light that strikes it. The output signal of a SiPM is the sum of the currents generated by all photo-diodes contained in the array. The signal can be amplified and measured using convenient electronic circuits.

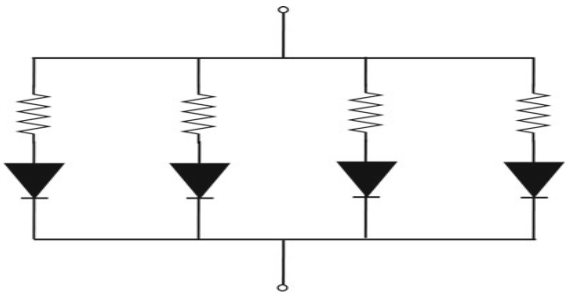


Figure 1.7: Schematic of the parallel arrangement of photo-diodes with series quenching resistor in SiPMs [11].

1.2.2.3 Photomultiplier Tube vs Silicon Photomultiplier

SiPMs have some advantages over PMTs, such as their smaller size, lower operating voltage and power consumption. They also have a high photon detection efficiency, meaning that they are able to detect a large percentage of photons entering the device. In contrast, SiPMs have many drawbacks. They have a lower dynamic range compared to PMTs, which can lead to saturation at high light levels. Moreover, discriminating the dark noise from the signal is a challenging task, the gain is significantly impacted by temperature, and the device's size is limited by the dark noise characteristics [11].

PMTs have been employed for many years in scintillation-based detectors for neutron/gamma-ray discrimination. Additionally, various studies have explored the use of ML to enhance the performance of neutron/gamma-ray discrimination in organic scintillators, incorporating PMTs into their experimental setups [12–19]. In this work, datasets were acquired using a measurement system with a PMT photo-detector. Nevertheless, the methods implemented for signal processing and classification can also be applied to data collected through an acquisition system based on a SiPM sensor. In some applications, the signals generated by the PMT or the SiPM are processed by different analog circuits before the digitization. Next section explain the main analog processing tools that can be used in a scintillation measurement chain.

1.2.3 Analog Signal Processing

The analog processing of signals generated by a SiPM or a PMT photo-detector aims to achieve basic pulse shaping that matches the digitization requirements of the target application. Roughly speaking, there are three main analog processing steps that are usually implemented:

1. Preamplifier: the main functions of this step is to recover the amplitude of the signal, to minimize capacitor effects, to match the impedance of the detector to the impedance of the coaxial cable carrying the signal and of the electronic processing chain, and perform an initial shaping of the signal. There are three types of preamplifiers: voltage, charge and current. This stage is usually unnecessary when the detector output is large enough to be processed.

2. Amplifier: in some cases, the preamplifier stage does not provide enough gain for further use of the signal. Therefore, an additional stage is used, which also serves as a pulse shaper and impedance matching for the next stage electronics.

3. Voltage limiters: high energy radiations can introduce transient instabilities in the signal acquisition chain, resulting in inconsistent measurement output. To limit their impact, voltage limiters are used. Different methods can be used to develop a voltage limiter circuit. This functionality can also be integrated directly into the ADC chip. This means that the gain of the analog stage must cover the energy range, while ensuring that the input dynamic range of the ADC is respected.

1.2.4 Analog-to-Digital Converter

The output analog signal provided by the photo-detector or the analog signal processing steps can be digitized using an ADC. In the framework of this thesis, some of the key specifications of interest for an ADC include:

1. Sampling frequency: the rate at which the ADC takes measurements of the input signal. It is measured in Hertz (Hz), which represents the number of samples per second. According to Shannon Nyquist sampling theorem, the sampling rate must be at least twice the highest frequency component of the input signal to enable a reconstruction without information loss [20].

2. Bandwidth: the frequency range within the ADC can accurately sample and digitize. In simple terms, it is the highest frequency that can be captured by the ADC without significant distortion or information loss.

3. Input Voltage Range (V_{pp}): the dynamic input voltage range that can be digitized without clipping.

4. Resolution: the number of bits used to represent each digitized signal sample. It determines the smallest voltage increment detectable by the ADC, which is calculated by dividing the V_{pp} by two to the power number of bits. Although higher resolutions (such as 12, 14, or 16 bits) provide greater accuracy, they come at the expense of decreased maximum sampling rate and bandwidth.

5. Triggering: ADCs can be activated by an external signal or programmed to trigger internally. In some cases, they can be activated permanently, in which case the sampling process is continuous. Internal triggering can be set to activate at a specific time, after a certain number of samples, or when a particular threshold is reached. It is important to set the triggering threshold appropriately to ensure accurate and reliable measurements. If the threshold is too low, the ADC may trigger on noise, leading to incorrect measurements. If the threshold is too high, the ADC may not trigger at all, resulting in missed data. The optimal triggering threshold is determined by the characteristics of the signal, such as amplitude, frequency, and noise level.

During this work, the parameters of ADC are adjusted to match the dynamic range of the input signals. The signals are acquired without using analog signal processing blocks and the different proposed ML approaches are applied directly on raw signals.

1.2.5 Digital Signal Processing

In real time applications, the main focus of this work, the sampling process is continuous, and the acquisition window could contain different pulses corresponding to different radiations, as illustrated in Figure 1.4. Therefore, triggers are needed to determine when pulse selection starts and stops. The starting point of a pulse can be identified when a particular threshold is reached. The trigger threshold must be adjusted to avoid noise triggering and minimize the number of undetected radiations. The ending point of a pulse represents the pulse duration, depending on the target application and the information required to be extracted from the pulse. For instance, determining the type of the incident radiation can require a duration on the order of several hundreds of nanoseconds, depending on the type of scintillator, the photo-detector, and the analog processing blocks.

During the processing of incident radiation to generate its corresponding signal, the scintillator may react with subsequent radiation, resulting in an occurrence known as pile-up. Figure 1.8 illustrates a typical case of pile-up, where each peak corresponds to a distinct radiation event. A shaping system, using high pass filters, can be employed to prevent pile-up before acquiring signal values of interest. If pile-up persists, the traditional approach is to reject it using an appropriate pile-up detection system [21]. Alternatively, digital methods can be employed to identify and address these events. Further details will be provided in Chapter 2, where a digital method is proposed for preparing a clean labeled dataset.

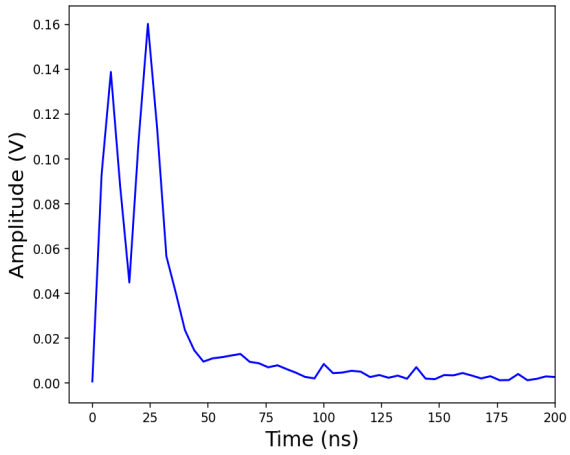


Figure 1.8: Pile-up example containing two pulses.

The rejection of pile-up events has negative impact on different applications. For instance, pile-ups can make the system to miss the identification of certain radiations. Therefore, the apparent number of incident radiation is lower than the actual number. This leads to an underestimation of the source activity. The likelihood of these events is directly proportional to the dead time of the system and the frequency of incident particles. In the presented case, dead time refers to the duration between the interaction of incident radiation and the generation of the analog signal before digitization [1]. The information provided by a photodetector coupled to a scintillator may persist for several hundreds of nanoseconds, while the energy deposition resulting from radiation interaction in the scintillator is virtually instantaneous (ranging from picoseconds to nanoseconds). Further, it is important to note while

the source activity can be managed in a laboratory setting by using a relatively low active source, in field applications where measurement duration is constrained, source activities are beyond control.

In nuclear instrumentation, extracting the pulse in real time during the acquisition involves several steps. The initial step is to identify the starting and ending points of the pulse and apply the BaseLine Restoration (BLR) method to eliminate the DC component. Various methods can be employed to determine the starting point, such as a certain percentage of the maximum or when the sample points of the signal reach a predefined threshold. The stop point in the presented case corresponds to the duration required for discrimination. In this thesis, the BLR method involves calculating the average of the previous n samples from the starting point and subtracting it from the input data. In certain situations, it is necessary to include pile-up detection method before pulse extraction, especially in a high count rate scenario where the probability of these events is increased as explained previously. Consequently, the incorporation of a pile-up detection block takes on added significance as it plays a central role in detecting and eliminating these events, or alternatively, isolating and distinguishing their individual pulses. The work in [22] proposes a a macro-pipelined architecture based on FPGA to execute all these steps for real time applications.

Roughly speaking, FPGA integrated circuit has been used in different studies to perform the acquisition with organic scintillator due to its capacity to process the digitized signal at a sampling frequency on the order of several hundred Megahertz [23–26]. This relatively high sampling rate is necessary due to the short duration of signals produced by a plastic scintillator (in order of several hundreds of nanoseconds), which can help to acquire the signal without loss of information. Furthermore, FPGA circuit has different advantages. It is reconfigurable device with parallel processing capability, low latency and high energy efficiency. Chapter 4 delves into further details about this integrated circuit, including its architecture and the process of its reconfiguration.

In the context of real time applications, the deployment of a discrimination system with FPGA can be achieved through two distinct approaches. The initial approach involves processing each acquired sample on the flow for discrimination without the need for storage [22]. In contrast, the second approach entails buffering all acquired samples representing one incident radiation event [23]. This approach is employed when the discrimination method, as is the case for some ML models, cannot be executed until all the samples representing the duration required for discrimination are acquired. Therefore, the FPGA implementation of proposed ML approaches for neutron/gamma-ray discrimination in plastic scintillator during this work will be preformed and compared to state of the art under the assumption that the discrimination method takes as input an already buffered pulse, as illustrated in Figure 1.9. The Figure shows the key blocks of an online discrimination system. The first block involves pulse extraction, as explained previously. The second block is a pile-up rejection system which can help to perform the discrimination on a single pulse. In this context, a pile-up is defined as any event containing at least two radiations separated by a time less than the signal duration required for discrimination. The third block is a buffer designed to store the samples acquired by the ADC. The output of this block serves as the input for the discrimination method block. To ensure continuous processing without any loss of information, a ping pong buffer system can be employed to store the sampling points of subsequent pulses while the discrimination block is reading the current one. The final block is responsible for the classification decision, where a predefined threshold is used to determine whether the input signal

represents a neutron or a gamma-ray. The value of this threshold depends on the discrimination method and the performance constraints of the target application.

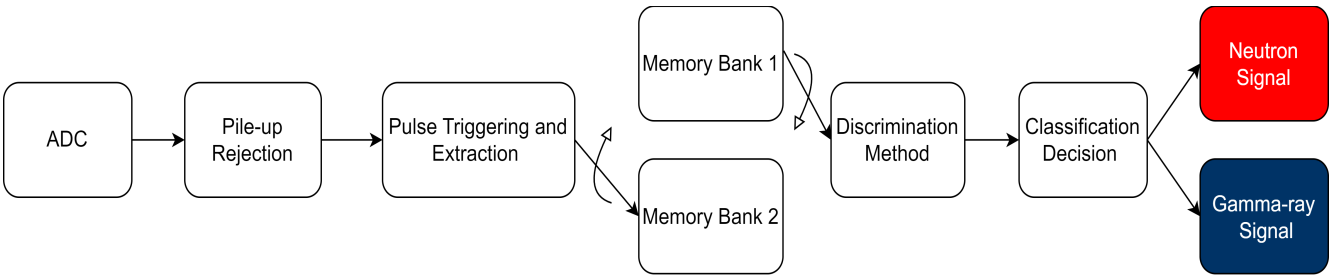


Figure 1.9: Key blocks of an online neutron/gamma-ray discrimination system.

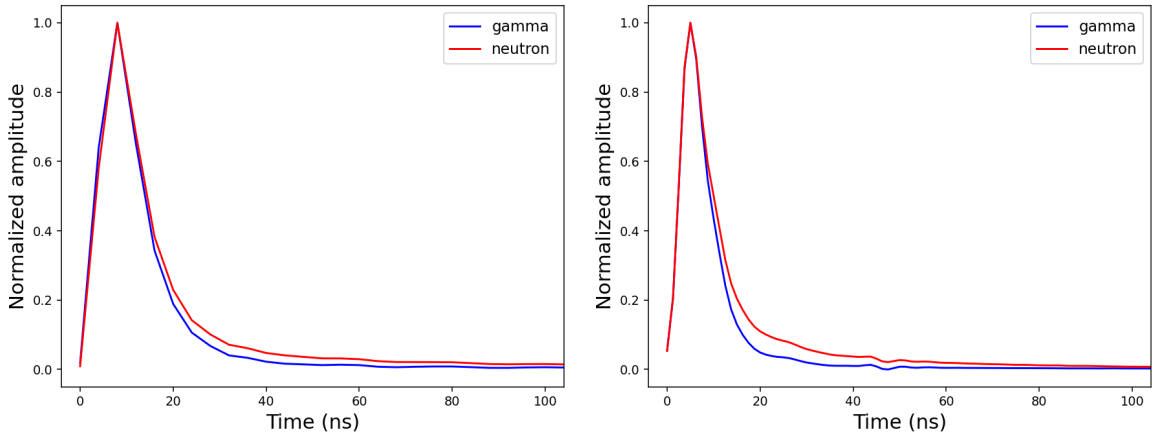
It is important to note that the pulse extraction and pile-up rejection blocks in Figure 1.9 should operate at the same sampling rate as the ADC. The discrimination method and classification blocks can function at different clock frequencies under an execution time constraint for each block, ensuring it is less than the pulse duration. This execution time is called electronic dead time of the system. Violating this dead time constraint leads to a subsequent period during which the system is blocked, making it either totally or partially unavailable for the complete processing of a subsequent pulse. The higher the frequency of the incident particles, the greater the loss of information. These losses affect the observed count rate, especially if this rate is relatively high. In some applications, an accurate count rate is a requirement. Furthermore, the longer the block time of the system, the fewer pulses are processed in relation to the pulse frequency actually detected. This means that information is lost, and a higher execution time is needed to acquire a greater number of pulses before making a decision (alarm threshold, for example), which is not a feasible choice in real time applications where the duration of measures is constrained. Therefore, a proposed approach to perform neutron/gamma-ray discrimination with a plastic scintillator in real time should have an inference time less than the signal duration. The short duration of signals in a plastic scintillator, which lasts for several hundreds of nanoseconds, challenges the embedded implementation under this relatively low time constraint.

During this thesis, the signals are first acquired and registered by a digitizer such as, an oscilloscope, directly connected to the PMT. The acquired raw signals are processed offline to be labeled and cleaned from unwanted events. Thereafter, different ML models are trained and compared to the state of the art neutron/gamma-ray discrimination algorithm. Finally, for real time applications, the FPGA implementations of the proposed discrimination methods are compared based on their discrimination performance, execution time, and resource consumption. Next section presents different approaches that have been proposed in the literature for neutron/gamma-ray discrimination in organic scintillator.

1.3 Neutron/Gamma-ray Discrimination in Organic Scintillator

PSD algorithms, such as zero crossing [27], Tail to Total Integral Ratio (TTT_{ratio}) [28], and curve-fitting [29] have been proposed to discriminate neutron and gamma-ray in organic scintillators. These algorithms rely on the difference in shape of the signals to classify them. Neutron interaction produces

a longer signal than the one generated by gamma-ray [10]. In liquid and stilbene scintillators, the difference between the two created signals is more significant than the plastic counterpart, as shown in Figure 1.10. Therefore, all discrimination approaches perform better with these types of detectors [6–8].



(a) EJ276 plastic scintillator

(b) BC-501A liquid scintillator

Figure 1.10: Average of neutron and gamma-ray signals obtained by EJ276 plastic and BC-501A liquid scintillators, ^{252}Cf and ^{60}Co sources. Min-max normalization is applied on the average signals.

Compared to zero crossing and curve-fitting PSD methods, TTT_{ratio} is widely used as the reference algorithm to be compared with different ML approaches proposed to enhance neutron/gamma-ray discrimination in organic scintillators [13–15, 17–19]. Furthermore, this algorithm is very well documented in the literature for different applications. It is implemented to characterize the discrimination ability of EJ309 liquid [30] and EJ276 plastic [31] scintillators. It is also used to calibrate the neutron response in EJ276 plastic scintillator [32]. Another studies consider TTT_{ratio} to compare the discrimination ability between plastic and stilbene scintillators [7], and between liquid and plastic scintillators [33, 34]. Therefore, this algorithm is defined in the frame of this work as the reference to be compared with proposed ML models.

1.3.1 Tail to Total Integral Ratio PSD Algorithm

TTT_{ratio} algorithm computes the ratio between the tail and total integral of the signal to identify the nature of the incident radiation according to equation 1.1.

$$TTT_{ratio} = \frac{Q_{tail}}{Q_{total}} \tag{1.1}$$

where $Q_{tail} = \int_{t_{short}}^{t_{long}} f(t)$ and $Q_{total} = \int_{0}^{t_{long}} f(t)$. Figure 1.11 illustrates Q_{tail} and Q_{tot} integrals on the average of neutron and gamma-ray signals obtained by EJ276 plastic scintillator.

The amplitude value of a signal is proportional to the detected radiation energy [35]. Therefore, this energy can be represented by the Q_{total} integral. The values of t_{long} and t_{short} enabling to optimize the discrimination performance are dependent on the implemented experimental setup. In this thesis,

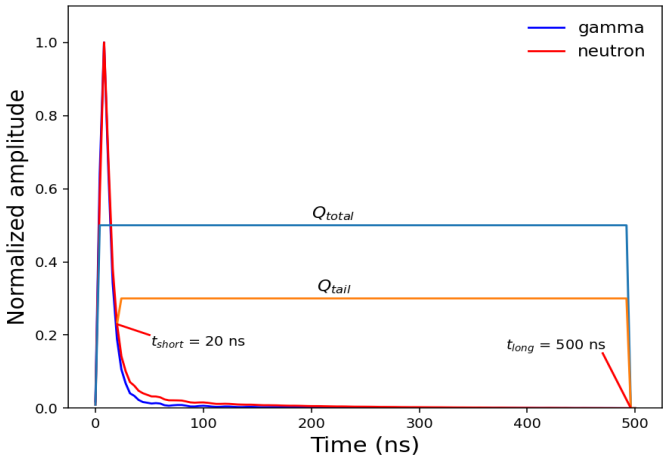


Figure 1.11: Illustration of Q_{tail} and Q_{tot} integrals on the average of neutron and gamma-ray signals obtained by EJ276 plastic scintillator, ^{252}Cf and ^{60}Co sources.

the algorithm implemented in [36] is used to obtain their optimal values whenever TTT_{ratio} algorithm is employed for discrimination. This optimization algorithm tunes t_{long} and t_{short} in order to maximize Figure of Merit (FOM) evaluation metric, which is explained in section 1.4.1.

The discrimination performance of this algorithm is proportional to the energy range of incident radiations. Figure 1.12 illustrates the bi-parametric graphical representation of the TTT_{ratio} as a function of the total energy (Q_{total}) for a dataset obtained by EJ276 plastic scintillator and a mixed neutron/gamma-ray source (^{252}Cf). We can distinguish two main classes. The overlapping between both of them illustrates the difficulty of the TTT_{ratio} algorithm to discriminate all signals, especially for low Q_{total} values ([0, 0.2] a.u.), which correspond to low energy radiations ([0, 250] keVee). Nonetheless, it is crucial to highlight that a key benefit of the TTT_{ratio} algorithm lies in its computational simplicity and relatively low inference time [23, 24]. This makes it a preferred choice in applications where the classification should be achieved on the fly.

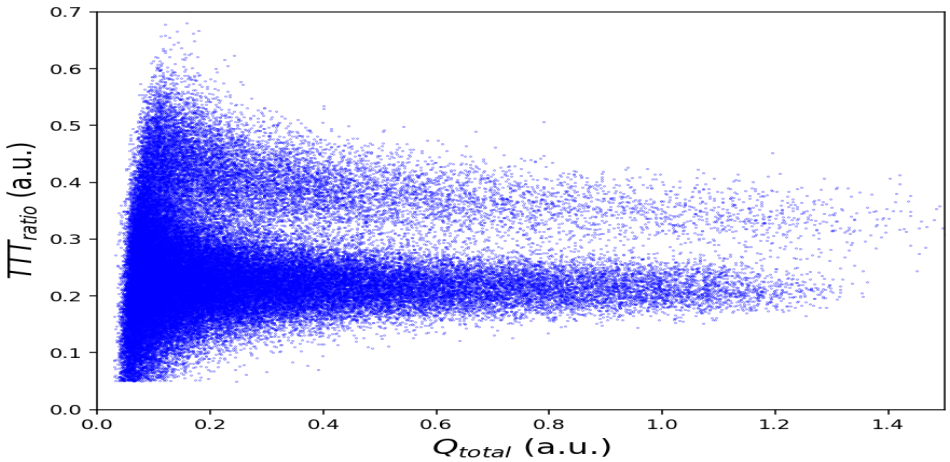


Figure 1.12: Bi-parametric graph of TTT_{ratio} according to Q_{tot} integral for a dataset obtained using EJ276 plastic scintillator and a ^{252}Cf source.

1.3.2 Machine Learning for Neutron/Gamma-ray Discrimination in Organic Scintillator

ML is a field of artificial intelligence that involves training algorithms to learn patterns and make predictions from data. This process allows machines to identify complex relationships and extract valuable insights from data. It can be separated into three main groups: supervised, unsupervised and semi-supervised.

1. Supervised learning involves training an algorithm on a dataset where the desired output is known, called labeled data. The algorithm learns to map inputs to outputs by minimizing the difference between its predicted output and the true output. One example of supervised machine learning is image recognition, where a model is trained to recognize different objects within an image. Another example is email classification, where a model is trained to classify emails as either spam or not spam. Some popular models used in supervised machine learning include Support Vector Machine (SVM) [37], Random Forest (RF) [38], MLP [39] and Convolution Neural Network (CNN) [40]. These models can be applied to a wide range of applications, including but not limited to, healthcare, finance, and social media analysis [41, 42].

2. Unsupervised learning involves training an algorithm on a dataset where the desired output is unknown, called unlabeled data [43]. The algorithm learns to find patterns and structure in the data, without any explicit guidance. One example of unsupervised learning is clustering, where the algorithm groups similar data points together based on their features. Another example is dimensionality reduction, where the algorithm identifies the most important features in the data and reduces the dimensionality of the dataset. Some popular models used in unsupervised learning include k-means clustering, Gaussian Mixture Model (GMM), and encoded-decoder architecture. These models can be applied to a wide range of applications, including but not limited to, customer segmentation, anomaly detection, and image compression.

3. Semi-supervised learning involves a combination of labeled and unlabeled dataset to train a ML model [44]. The labeled data is used to guide the learning process, while the unlabeled data helps to improve the accuracy and generalization of the model. One example of semi-supervised learning is text classification, where only a small portion of the data is labeled. The model can use the labeled data to learn the patterns and relationships between the features and the labels, and then apply that knowledge to the unlabeled data.

Recently, ML techniques have been explored to improve discrimination performance with liquid and stilbene scintillators. The authors of [12] propose a non negative matrix and tensor factorization (TF) methods to discriminate neutrons and gamma-rays in a stilbene scintillator. The obtained results in this study show that the origin of acquired signals is due to two main independent components. These components represent the neutron and gamma-ray respectively. The computation of the cross correlation function between these obtained components for each signal and both pure neutron and gamma signals allows the authors to discriminate neutrons and gamma-rays accurately. However, the authors do not mention how this accurate performance is measured. Furthermore, they do not provide the preparation details of pure neutron and gamma-ray signals used for the evaluation.

Another study related to the discrimination of neutron and gamma-ray signals is proposed in [13] using a stilbene detectors and a MLP model. The authors train their model on neutron and gamma-ray

signals obtained by a mixed neutron/gamma-ray californium source (^{252}Cf). Then, they evaluate the model on a dataset acquired in a high rate scenario using ^{252}Cf source and ^{137}Cs pure gamma-ray source. The neutron/gamma-ray signals obtained by ^{252}Cf are labeled by TTT_{ratio} algorithm. The trained model misclassifies only 1.4% of neutrons in this proposed high rate scenario (gamma-ray neutron ratio of approximately 400 to 1).

The authors in [14] employ a GMM model for neutron/gamma-ray discrimination in EJ309 liquid scintillator. The authors compare their GMM model to TTT_{ratio} algorithm across varying incident radiation energies ([10, 100] keVee). The results reveal that the trained GMM outperforms TTT_{ratio}, mainly for low energy radiations. As the energy radiation increases, TTT_{ratio} algorithm begins to exhibit a similar performance compared to the trained GMM. The trained GMM model is evaluated and compared to TTT_{ratio} algorithm using neutron/gamma-ray datasets labeled by Time of Flight (ToF) discrimination setup and an interval selection method to minimize the labeling error, as explained in [45]. ToF setup relies on the speed difference between particles to identify their nature. This involves using two scintillators separated by a certain distance to detect radiation and associated speed. Section 2.2 provides further information about this setup, the method proposed in [45] to reduce the labeling error, and the error reduction approach proposed in this thesis when this setup is implemented to prepare labeled datasets.

SVM model is proposed in [15] to perform the neutron/gamma-ray discrimination with EJ299-33 plastic scintillator. The model is trained and evaluated using dataset labeled by TTT_{ratio}. The study compares the performance of the pulse gradient analysis (PGA) discrimination algorithm and the proposed SVM model. The results show that the SVM method can achieve high discrimination accuracy of 99.1%, which is much better than that of PGA (92.1%).

The article [16] proposes a MLP approach with the EJ301 liquid scintillator. The main advantage of this study is that model training and evaluation are achieved on a dataset labeled by ToF method. Nevertheless, the proposed approach for reducing labeling error in ToF has some limitations that will be discussed in section 2.2. In addition, the energy range of the prepared dataset is relatively high ([740 keVee, 1.225 MeVee]).

1D CNN models are proposed to perform the neutron/gamma-ray discrimination in CLYC inorganic scintillator [17] and EJ200 plastic scintillator coupled to 6LiF:ZnS(Ag) neutron sensitive screen [18]. The authors of both studies compared their CNN models to different discrimination approaches such as MLP and TTT_{ratio}. The obtained results in both works showed the outperformance of CNN model. The first proposed approach depends on datasets labeled by TTT_{ratio} algorithm and have an energy range between 2000 keVee and 4500 keVee. The signals of second approach have an energy range between 1 MeVee and 10 MeVee and are labeled by a PSD method based on the time-over-threshold (TOT) and maximum peak amplitude (MPA) of signals. This method relies on the fact that neutron signals have longer decay time compared to gamma-ray signals. In general, for a predefined threshold, neutron signals have higher TOT values. The analogy of this method can be compared to TTT_{ratio} algorithm by replacing TTT_{ratio} and Q_{tot} integral in Figure 1.12 by TOT and MPA, respectively. Therefore, as for TTT_{ratio}, the main drawback of this method is its dependence on the energy range of incident radiations. The discrimination is challenging for relatively low energy radiations. Nevertheless, the authors take

the advantage of using sensitive neutron screen to overpass this limitation and choosing TOT and MPA thresholds minimizing the overlap between the distributions of TOT as a function of MPA of neutron and gamma-ray classes.

Another approach based on CNN model is proposed to perform the neutron/gamma-ray discrimination in EJ276 plastic scintillator [19]. The datasets used to train and evaluate the model is labeled by TTT_{ratio} and have [250, 1000] keVee energy range. The approach relies on two CNN models. The initial model classifies the input signal as either a gamma-ray or neutron. The second model then discriminates the neutrons predicted by the first model, labeling them as either true or fake neutrons. In other words, the second model validates the neutron predictions of the first. The authors found this dual-model approach to be more efficient than using a single model for signal classification, contributing to a reduction in false alarm rate.

Despite the improvement obtained by all previous ML work on the neutron/gamma-ray discrimination task, there are several drawback points that cannot be ignored:

- The majority of ML approaches for neutron/gamma-ray discrimination have been developed using liquid and stilbene scintillators, which have a higher discrimination capability than plastic scintillators.
- Some of these models are trained on datasets only labeled by the TTT_{ratio} discrimination algorithm. The accuracy of the labeling by this algorithm decreased when there is an overlap between the TTT_{ratio} distributions of neutrons and gamma-rays, as shown in Figure 1.12. Choosing one TTT_{ratio} threshold to label the signals results to a significant number of mislabeled samples. Furthermore, if an energy radiation threshold is selected, above which neutrons and gamma-rays can be distinguished from each other, then using ML would become unnecessary. This is because simpler methods can be used to separate the two types of radiation.
- Some of these studies do not take into consideration the comparison between the trained ML model and state of the art algorithms based on their embedded implementations for a real time discrimination. This point is critical in some applications, as explained in section 1.2.5.
- The discrimination performance of some of these ML approaches are not compared to that of state of the art methods concerning variations in the energy of incident radiations. This comparison is important to highlight the outperformance of the proposed ML model since the discrimination is more challenging for low energy radiation, as shown in Figure 1.12.
- The discrimination performance is not evaluated according to the variation in sampling frequency. This helps to find the minimum sampling rate that can be used to perform the discrimination, which is an important point to reduce the complexity and power consumption of an online discrimination system.

Table 1.1 provides an overview of the main benefits and drawbacks of each one of the previous mentioned ML method for neutron/gamma-ray discrimination. To the best of our knowledge, this PhD is the first work to explore the interest of using ML for neutron/gamma-ray discrimination with a plastic scintillator for online applications. This work compares the discrimination performance of ML models and TTT_{ratio} algorithm in a plastic scintillator, for different radiation energy ranges, at different

Table 1.1: Overview of the main benefits and drawbacks of ML methods for neutron/gamma-ray discrimination.

	Model	Scintillator type	Labeling method	Energy range (keVee)	Minimum f_s	Embedded implementation
[12]	TF	stilbene	?	?	?	no
[13]	MLP	stilbene	TTT_{ratio}	?	?	yes
[14]	GMM	EJ309 (liquid)	ToF	[10, 100]	?	yes
[15]	SVM	EJ299 (plastic)	TTT_{ratio}	?	?	no
[16]	MLP	EJ301 (liquid)	ToF	[740, 1225]	?	no
[17]	1D CNN	CLYC (inorganic)	TTT_{ratio}	[2000, 4500]	250 MHz	no
[18]	1D CNN	EJ200 + 6LiF:ZnS(Ag)	TOT	?	no	no
[19]	1D CNN	EJ276 (plastic)	TTT_{ratio}	[250, 1000]	no	no

sampling frequencies. In addition, it compares their FPGA implementations for use in an online discrimination system.

In this section we presented and discussed the main techniques and their limitations for discriminating between neutrons and gamma-rays in organic scintillators. The following section presents the evaluation metrics proposed to assess these techniques and compare their performances.

1.4 Evaluation Metrics for Neutron/Gamma-ray Discrimination Algorithms

In order to assess the ability of organic scintillators in differentiating between neutrons and gamma-rays, two evaluation methods have been considered: the FOM and the Receiver Operating Characteristic (ROC) curve. Sections 1.4.1 and 1.4.2 present these two metrics, respectively.

1.4.1 Figure of Merit

Researchers in physics and chemistry commonly use FOM evaluation metric to assess the ability of a measurement chain to distinguish between two radiation types within a specific energy range [46]. For instance, chemists use FOM metric to compare several formulations of scintillators and to select the most effective one. This evaluation metric assesses the ability to discriminate using the TTT_{ratio} algorithm by calculating equation 1.2, making the assumption that TTT_{ratio} distributions of neutrons and gamma-rays follow a Gaussian [46].

$$FOM = \frac{S}{FWHM_\gamma + FWHM_n} \tag{1.2}$$

where $S = \mu_n - \mu_\gamma$. μ_n and μ_γ are the mean values of the Gaussian fitting equations of the neutron and gamma-ray distributions respectively, and $FWHM_n$ and $FWHM_\gamma$ the corresponding Full Width at

Half Maximum. Figure 1.13 illustrates the computation of this metric on a neutron/gamma-ray dataset acquired by a plastic scintillator. A higher FOM value indicates a higher discrimination ability.

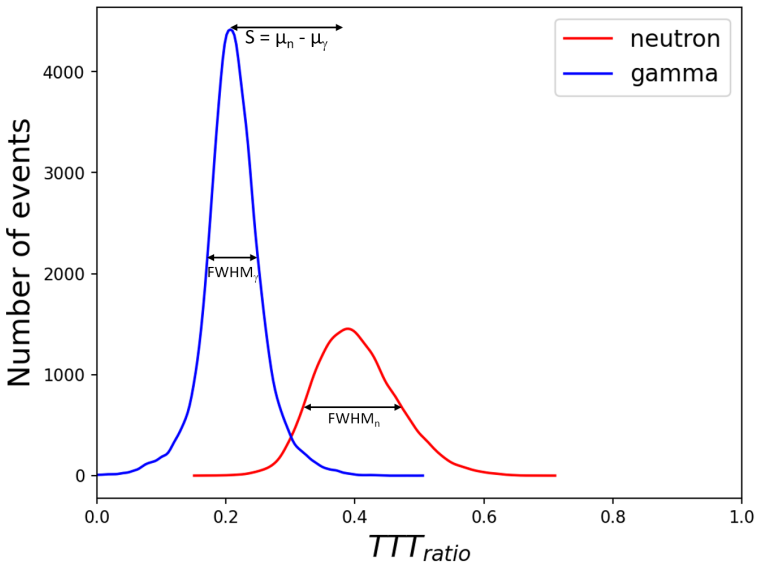


Figure 1.13: Main parameters for a FOM calculation.

The discrimination ability of the acquisition chain can be significantly influenced by various components and parameters, such as the scintillator type, sampling frequency, ADC noise, and vertical resolution, as presented in chapter 2, section 2.1. Therefore, the calculated FOM of two different studies are only comparable if their pulse acquisition chains are exactly the same. Furthermore, the definition of the FOM is based on a Gaussian distribution assumption. In this thesis, when FOM metric is used to evaluate the discrimination performance, the Gaussian fit of the experimental data and the computation of the FOM are both carried out using the algorithm implemented in [36]. This algorithm tunes t_{long} and t_{short} parameters of TTT_{ratio} algorithm to optimize the discrimination performance by maximizing the value of FOM.

1.4.2 Receiver Operating Characteristic Curve

The performance of a binary classification system depends on the chosen classification threshold used to classify the input. For instance, this can be observed in the use of the TTT_{ratio} discrimination algorithm to differentiate neutrons from gamma-rays, where the optimal threshold is dependent on the various components and parameters used in the implemented acquisition chain.

The ROC curve is a graphical display illustrating the diagnostic ability of a binary classification system as the discrimination threshold is varied. This curve is generated by plotting the True Positive Rate (TPR) versus the False Positive Rate (FPR) obtained at different classification thresholds, as illustrated in Figure 1.14. These metrics, along with others are used to evaluate binary classification problems. They are defined as:

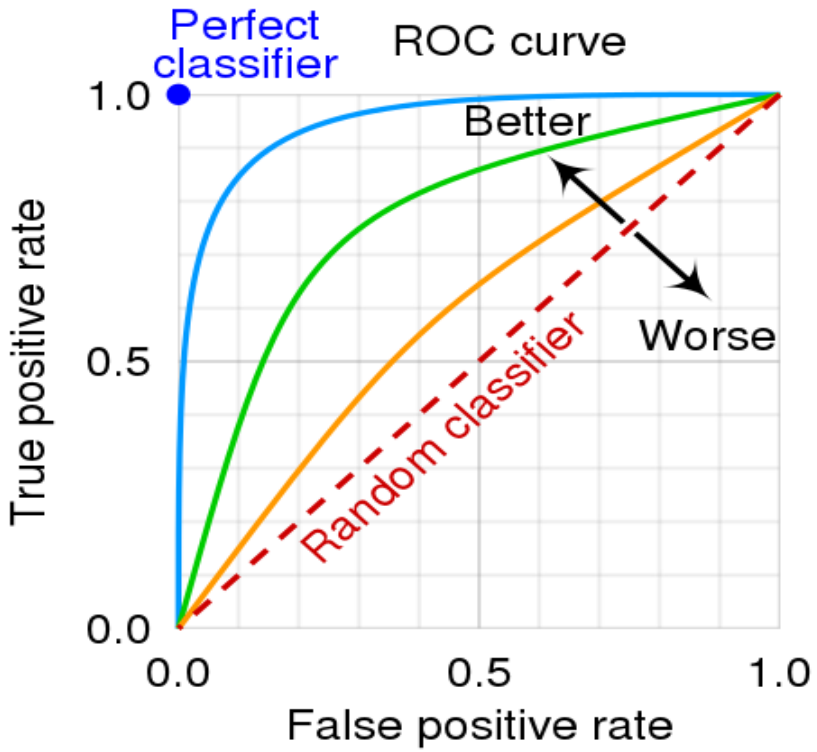


Figure 1.14: The ROC space for a better and worse classifier [47].

1. True Positive Rate (also called Recall or Sensitivity): number of items correctly identified as positive out of the total actual positives. TPR represents the probability that an actual positive will be predicted positive.

2. True Negative Rate (TNR) (also called Specificity): number of items correctly identified as negative out of the total actual negatives. TNR represents the probability that an actual negative will be predicted negative.

3. False Positive Rate: number of items incorrectly identified as positive out of the total actual negatives (FP/(TN+FP)). FPR is the probability that a false alarm will be raised. It can be computed from the TNR, where FPR = 1 - TNR.

4. False Negative Rate (FNR): number of items incorrectly identified as negative out of the total actual positive (FN/(TP+FN)). FNR is the probability that a true positive will be missed by the prediction. It can be computed from the TPR, where FNR = 1 - TPR.

5. Accuracy: number of items correctly identified as either truly positive or truly negative out of the total number of items.

6. Precision: number of items correctly identified as positive (negative) out of the total items identified as positive (negative).

7. F1-Score: the harmonic average of the precision and recall, it measures the effectiveness of identification when just as much importance is given to recall as to precision.

Table 1.2 summarizes all these metrics used to evaluate a binary classification problem. TPR and FPR metrics are considered as the most crucial criteria when evaluating a system designed to discriminate neutrons and gamma-rays. Thanks to the labeling method which will be presented in chapter 2, section 2.2, the ROC curve can be used to accurately evaluate and compare various discrimination methods, including the proposed ML models, where neutrons and gamma-rays are designated as the positive and negative classes, respectively.

Table 1.2: Evaluation metrics for binary classification model.

		Predicted		
		Positive	Negative	$TPR = recall_p = sensitivity = \frac{TP}{TP+FN}$
True label	Positive	TP	FN	$TNR = recall_N = specifity = \frac{TN}{TN+FP}$
	Negative	FP	TN	$FPR = \frac{FP}{TN+FP} = 1 - TNR$
		$accuracy = \frac{TP+TN}{TP+TN+FP+FN}$		$FNR = \frac{FN}{TP+FN} = 1 - TPR$
		$F1 = \frac{2*precision*recall}{precision+recall}$		$precision = \frac{TP}{TP+FP} or \frac{TN}{TN+FN}$

1.4.3 ROC vs FOM

FOM evaluation metric evaluates the discrimination ability of a scintillation acquisition system. In contrast, the ROC curve shows the variation of false rate alarm of neutron detection according to the true neutron detection rate, Figure 1.14. Therefore ROC can be considered to be more precise for the evaluation. Nevertheless, creating the ROC curve requires a ground truth neutron dataset, which can be difficult to obtain as explained in section 1.1. On the other side, determining the discrimination ability of a measurement system through the FOM metric only requires a mixed dataset of neutrons and gamma-rays, which can be generated using various radiation sources, such as californium (^{252}Cf) and americium–beryllium (^{241}AmBe).

In the context of this thesis, FOM is used to assess and compare the discrimination abilities of different implemented acquisition systems. In contrast, the ROC curve is presented for an accurate evaluation and comparison between different proposed discrimination approaches. This graphical representation will be achieved by using labeled datasets obtained through the labeling method proposed in chapter 2, section 2.2.

For real time applications, the embedded implementation of a neutron/gamma-ray discrimination method is a requirement. The implementation should be achieved while respecting an inference time less than the signal duration to avoid the miss detection of some radiations, as explained in section 1.2.5. Therefore, for these applications, the comparison between different discrimination approaches should go beyond their performances. Their embedded implementations under this time constraint should also be

compared. This can be done by comparing their execution time, resource consumption and discrimination performance.

1.5 Conclusion

This chapter defined the problem context and discussed the state of the art regarding this thesis. The topic of classifying radioactivity signals was introduced, with focus on a challenging use case, which is the discrimination of neutrons and gamma-rays in organic plastic scintillators. We presented the various components of a typical scintillation acquisition system (section 1.2), analyzed and discussed the main limitations of the existing discrimination approaches (section 1.3) and analyzed the evaluation metrics used in the literature (section 1.4).

The cost and size of an embedded discrimination system increase in accordance with its sampling frequency. Therefore, it is crucial to identify the minimal sampling frequency required for discrimination to enhance the embedded implementation. This aspect holds significant importance within the context of this study, particularly concerning the integration of ML approaches on edge devices. Furthermore, the evaluation of discrimination methods remains a challenging task even with the use of the presented evaluation metrics, mainly due to the effects of gamma-ray emissions in neutron sources. The next chapter addresses the first concern by analyzing the impact of various factors, such as vertical resolution, sampling frequency, SNR, and the use of an integrator amplifier, on the ability to differentiate between neutrons and gamma-rays in EJ276 plastic scintillator. The results of this study can help in implementing an experimental setup that operates at the minimal sampling frequency necessary for discrimination. Moreover, chapter 2 proposes a method to generate a clean labeled neutron dataset in an organic scintillator for an accurate assessment of different discrimination approaches.

Chapter 2

Signal Analysis and Processing for Clean Labeled Neutron/Gamma-ray datasets

Determining the minimum sampling frequency required for discrimination is an important step to optimize the embedded implementation of a discrimination method. The power consumption of a digital processing system is proportional to its sampling rate. Furthermore, an accurate evaluation of any discrimination method is a difficult task due to the lack of a reference neutron dataset. The main limitation for obtaining this dataset is the impact of gamma-ray emissions in neutron sources.

This chapter is segmented in two main sections. The first one presents the effect of vertical resolution, sampling frequency, and SNR on the discrimination ability of a radiation measurement system. In the presented case, the acquisition chain is implemented using EJ276 plastic scintillator. Thereafter, this first section shows the impact of using an integrator amplifier in the acquisition chain on discrimination performance and the minimum sampling rate required for accurate discrimination. The results of this section can help in implementing an experimental setup activated at the minimum sampling frequency needed for discrimination using EJ276 plastic scintillator. This is a critical point to optimize the embedded implementation and reduce its cost and size. The second section of the chapter introduces a method for preparing clean and labeled neutron/gamma-ray datasets in an organic scintillator, which will be used to train and evaluate various ML models and compare different discrimination approaches. The method is mainly based on a ToF setup and TTT_{ratio} algorithm. The main conclusions of this study are summarized at the end of this chapter.

2.1 Discrimination Performance According to SNR, Sampling Frequency and Vertical Resolution

In the literature, several studies have examined the characteristics of acquisition chains that have an impact on the discrimination ability with an organic scintillator. The authors in [48] concluded that achieving optimal discrimination with liquid scintillator required a balance between vertical resolution and sampling frequency. However, the impact of SNR was not taken into account in this study. Another study found that a minimum sampling frequency of 200 MHz was necessary to achieve discrimination with BC501A liquid scintillator [49]. Nevertheless, the impact of SNR and vertical resolution on this minimum frequency was not explored. A higher SNR may reduce this obtained sampling rate.

The discrimination results with an organic scintillator are impacted by the sampling frequency, vertical resolution, and SNR. This impact has never been examined in the literature. This chapter proposes to study this effect, specifically showing how the sampling frequency can be reduced to simplify ML implementation on portable devices and reduce their dynamic power consumption. First, the implemented setup used to acquire the datasets of this study is described in section 2.1.1. Then, section 2.1.2 explains the analysis of the frequency components of the acquired signals. Sections 2.1.3 and 2.1.4 explain the SNR and vertical resolution and how they are used in this study, respectively. Thereafter, the acquisition of different neutron/gamma-ray datasets at various sampling frequencies and vertical resolutions is described in section 2.1.5. Section 2.1.6 presents the analysis achieved to determine the relation between SNR, vertical resolution, and the minimum sampling frequency required for discrimination. FOM evaluation metric explained in section 1.4.1 is used to measure the discrimination performance. Furthermore, in section 2.1.7, we explore the impact of using an integrator amplifier in a measurement system on the minimum sampling rate required for discrimination and its performance.

2.1.1 Experimental Setup

In the past, various commercial plastic scintillators have been developed. One of these detectors is EJ276, manufactured by Eljen Technology, which has been used in different studies in the literature [6–8, 11, 19]. Its pulse shape discrimination properties are sufficient to achieve neutron/gamma-ray discrimination [50]. Nevertheless, they are still degraded compared to those offered by a liquid scintillator [6, 7]. Furthermore, the discrimination ability is dependent on the energy of the incident radiation, with lower energy levels being more difficult to discriminate. This thesis uses this scintillator type in the implemented acquisition chains. By comparing the acquired results with the discrimination performance documented in the literature, we can assess and validate the various discrimination approaches proposed during this work.

Figure 2.1 illustrates the implementation of a scintillation acquisition chain that was used during this study. An EJ276 plastic scintillator is coupled to a PMT model 9821B from ET Enterprises [51]. The PMT operated at 1700 V and is coupled to the scintillator via SCIONIX EJ-500 optical grease, which ensures efficient transmission of scintillation photons to the PMT's photo-cathode. The PMT is connected to HDO6104A-MS oscilloscope, which has a bandwidth of 1 GHz. The mixed neutron/gamma-ray radiation source ^{252}Cf is placed at 5 cm from the scintillator.

It is important to note that the signal processing methods employed during this study are not dedicated solely to the analysis of the datasets obtained from this chain. They can be applied on any dataset acquired by a different acquisition chain, which be created by modifying at least one component of the schematic presented in Figure 2.1.

2.1.1.1 Signal Alignment

In general, the starting point of each signal is the first sampling point reaching the triggering threshold. However, the rise time of each signal depends on the energy of the incident radiation, thus resulting in different peak positions for different signals. The alignment method of this work defines the starting point of a signal as the time when the pulse reaches 10% of its maximum value, see Figure 2.2. The

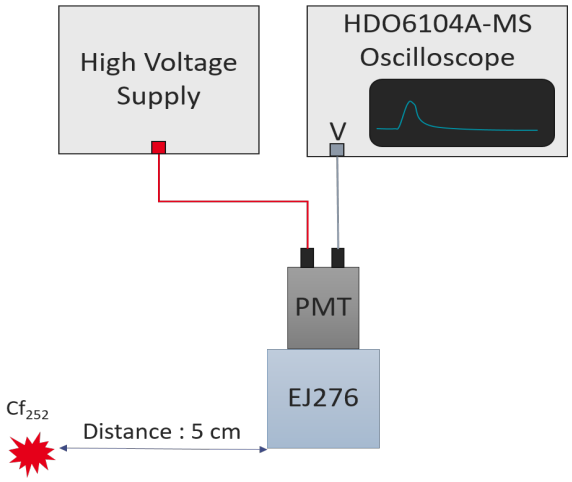


Figure 2.1: Implemented acquisition chain based on the EJ276 plastic scintillator.

length of the signal from the detected starting point can be chosen differently for each acquisition chain. To remove the DC component and its drifts, we apply a BLR method on each signal. This method is based on calculating the average of the previous n samples of the starting point and subtracting it from the input data, as shown in Figure 2.2. In this work, n corresponds to a duration of 100 ns.

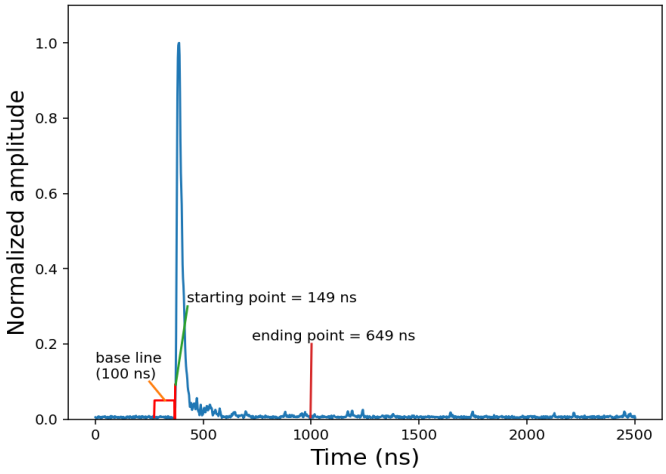


Figure 2.2: Illustration of the base line, starting and ending points on a signal obtained by EJ276 plastic scintillator and a ^{252}Cf source. The sampling frequency and signal duration are 1.25 GHz and 500 ns, respectively.

2.1.2 Signal Frequency Analysis

The Shannon Nyquist sampling theorem states that a digital signal can be reconstructed without losing information if it is sampled at a rate of at least twice its highest frequency component [20]. Therefore, a dataset is acquired by the measurement chain shown in Figure 2.1. The acquisition parameters are set to a triggering threshold of 60 mV, a sampling frequency of 1.25 GHz, a time window

of 1 μs, and a voltage peak-to-peak of 1.6 V. The sampling rate of 1.25 GHz is specifically chosen to ensure the detection of relatively high frequency components in the signals. Following the alignment and BLR steps, the processed signals pass through frequency analysis using the Fast Fourier Transform (FFT) implemented in the *sklearn* Python library. The alignment step and BLR are performed with a signal and baseline duration of 500 ns and 100 ns, respectively. A 500 ns signal duration is enough for ensuring that all pulses return to the noise level, as illustrated in Figure 2.2.

The FFT average of the recorded signals in Figure 2.3 indicates frequency components up to 100 MHz. As a result, the minimum required sampling frequency for these signals is approximately 200 MHz, as supported by the obtained results of this study.

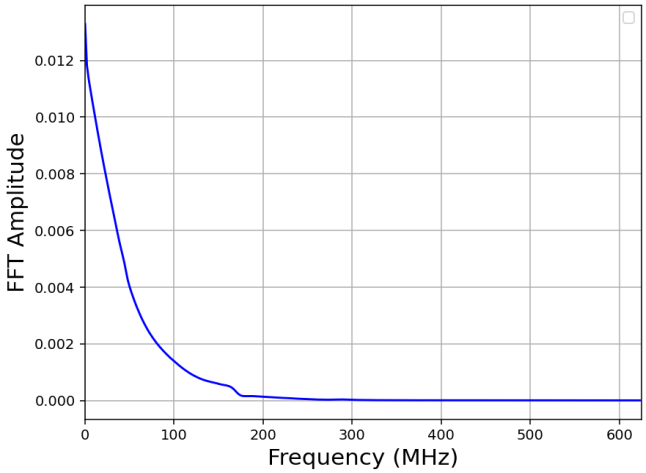


Figure 2.3: FFT power spectrum of 50,000 signals obtained by EJ276 plastic scintillator and ^{252}Cf. The acquisition was done at 1.25 GHz sampling frequency.

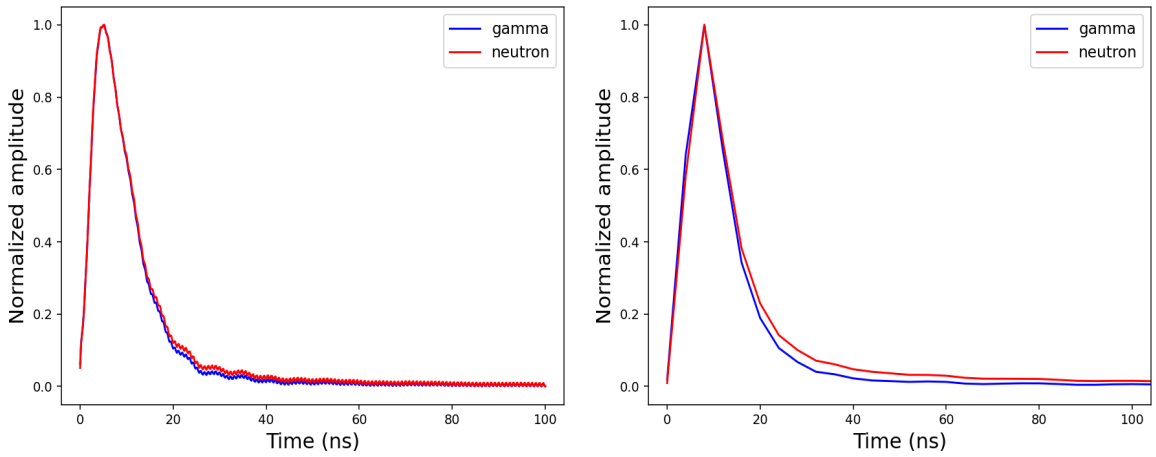
2.1.3 Signal to Noise Ratio

SNR is a quantitative measure of the ratio between the signal strength and the level of noise in a signal. A higher SNR indicates better separation of the signal from noise. The specific formula used to calculate SNR may vary depending on the application. For instance, in audio processing, engineers may compare the power levels of the signal and the noise to determine the SNR.

Two types of noise are traditionally encountered in the measurement of radioactivity. The first is thermal noise which is considered as white Gaussian noise. The second is shot noise which is often negligible compared to thermal noise [52]. White Gaussian noise is a type of random signal with a constant power spectral density across all frequencies. In other words, the noise energy is distributed uniformly across the entire frequency spectrum. It follows a Gaussian distribution, meaning that the probability of any given sample is normally distributed around zero with equal likelihood for positive and negative values.

The distinguishing feature between neutrons and gamma-rays detected by an organic scintillator is primarily concentrated in the decay phase of the signal, as illustrated in Figure 2.4. Therefore, we

are interested in the values of the sample points rather than the power of the signal. Moreover, the high variation of the noise around the signal can lead to the loss of this distinction. In other words, as the standard deviation of the noise increases, discrimination becomes more difficult. This is clear in Figure 2.4(a) and 2.4(b), where the SNR of the latter is 14 times higher than that of the former. Therefore, the appropriate SNR for a signal is determined by the ratio between the average of the sample points of the signal and the standard deviation of the noise, as shown in equation 2.1.



(a) $f_s = 5$ GHz, $SNR = 0.65$ (b) $f_s = 250$ MHz, $SNR = 9$

Figure 2.4: Average of neutron and gamma-ray signals acquired at different SNRs, obtained by EJ276 plastic scintillator, ^{252}Cf and ^{60}Co sources.

$$SNR = \frac{\mu_{signal}}{\sigma_{noise}} \tag{2.1}$$

where the signal average (μ_{signal}) and noise standard deviation (σ_{noise}) are computed by equation 2.2 and 2.3, respectively.

$$\mu_{signal} = \frac{\sum_{i=SP}^{N} x_i}{N} \tag{2.2}$$

with SP and N are the starting point and the length of signal, respectively. In the presented case, SP is the point when the signal reaches 10% of its maximum value.

The noise within each signal can be found in the sample points that occur both before the starting point and after the signal length (N). Therefore, the standard deviation of the noise in a signal can be calculated using equation 2.3.

$$noise = [signal[0:SP],\ signal[SP+N:M]]$$

$$\sigma_{noise} = \sqrt{\frac{\sum_{i=0}^{NoiseLength}(noise_i - \mu_{noise})^2}{NoiseLength}} \tag{2.3}$$

M is the length of the acquisition window. Figure 2.5 illustrates the noise of a signal acquired at 500 MHz sampling frequency. M and N are respectively 1000 ns and 500 ns. The starting point of the signal in this figure (10% of the maximum) is 148 ns from the start of the acquisition window.

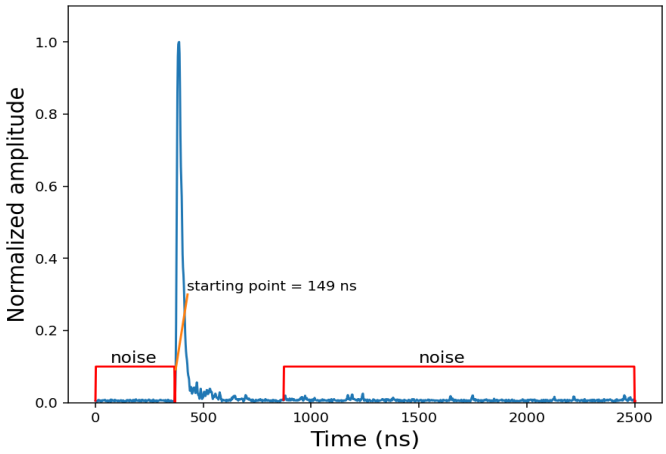


Figure 2.5: Illustration of the noise on a signal obtained using an EJ276 plastic scintillator and a ^{252}Cf source.

2.1.4 Vertical Resolution

Vertical resolution is a key specification for the ADC implemented in an oscilloscope. It is the number of bits used by the ADC to digitize input samples. Increasing vertical resolution of an ADC comes at the expense of decreasing maximum sampling rate and bandwidth.

An ADC of n bits vertical resolution divides the V_{pp} into 2^n levels. In other words, the minimum voltage or the Quantization Step (QS) that an ADC can theoretically distinguish is the input voltage range divided by the number of quantization step as expressed in equation 2.4. In our study, the QS is representative of the vertical resolution and its unit is in mV/bin. It is important to acknowledge that the presence of noise typically prevents the ADC to trigger on its theoretical minimum voltage level. Moreover, an ADC may work across a range of selectable V_{pp} that determine the maximum voltage that can be applied without causing the input waveform to clip.

$$QS\ (mV/bin) = \frac{V_{pp}}{2^n} \tag{2.4}$$

The QS directly affects the SNR of the output signal. The relationship between these two parameters is inversely proportional. In other words, the SNR is directly proportional to the number of bits used by the ADC for digitization and inversely proportional to the input voltage range employed during the acquisition.

2.1.5 Dataset Acquisition

Datasets with varying SNR levels can be generated by adjusting the QS during the acquisition process. This adjustment can be achieved by employing ADCs with different vertical resolutions or by modifying

the input voltage range (V_{pp}). Therefore, to obtain these datasets, the same acquisition system presented in Figure 2.1 is implemented with the two following oscilloscopes:

- LeCroy WaveRunner-640Zi: 8 bits vertical resolution.
- LeCroy HDO6104A-MS: 12 bits vertical resolution.

With each oscilloscope, the acquisition is done on three different V_{pp} levels (1600 mV, 800 mV and 400 mV). For each V_{pp} level, six datasets are acquired at six different frequencies. The radioactive source is ^{252}Cf. Table 2.1 summarizes the parameters of this acquisition setup.

Table 2.1: Acquisition setup configurations. Each configuration provides a dataset with specific quantization step and sampling frequency (f_s).

	WaveRunner-640Zi	HDO6104A-MS
Vertical resolution	8 bits	12 bits
Bandwidth (-3 db, 50 Ohms)	4 GHz	1 GHz
f_s (GHz)	5, 2.5, 1, 0.5, 0.25, 0.1	5, 2.5, 1.25, 0.5, 0.25, 0.1
QS (mV/bin)	6, 3, 1.5	0.4, 0.2, 0.1

In order to examine how the SNR parameter affects discrimination performance on an individual level, datasets with the same QS level but varying SNRs are required. This can be done by adding white Gaussian noise with mean equal to zero to a recorded dataset, effectively reducing its SNR.

The QS is directly proportional to V_{pp}, as shown in equation 2.4. According to the experimental setup of this study, a V_{pp} value of 400 mV results in a minimum QS of 0.1 mV/bin. Furthermore, the acquisition threshold is also proportional to V_{pp}, with the exact proportionality depending on the digitizer used and its noise level. In the presented case, the results of the experiment indicated that the threshold should be set to a minimum of 60 mV for the highest V_{pp} value of 1600 mV (which corresponds to a QS of 6 mV/bin). As a result, the acquisition threshold is set at 60 mV and any signal with a maximum amplitude exceeding 400 mV is eliminated. This implies that all of the acquired datasets are exposed to radiation with a comparable energy range. This energy range is between 200 and 1300 keVee. It is obtained by calibrating the dataset acquired at 250 MHz with a V_{pp} (QS) equal to 800 mV (0.2 mV/bin), using HDO6104A-MS oscilloscope. The calibration process is explained in section 3.5 of chapter 3. The acquisition time window and size of each dataset are respectively 1000 ns and 50000

2.1.6 Experimental Results

In order to ensure both datasets can be compared, the length of the signal in equation 2.2 is set to 500 ns for all signals. For each dataset, the SNR is determined by calculating the median value of the SNR values for all the individual signals within that dataset. The results showed that, for a given QS, the SNR remained relatively consistent according to the variation of the sampling frequency.

As expected, the SNR decreases as the QS increases (Table 2.2). Furthermore, for QS values of 2 mV/bin and 1 mV/bin, datasets with different SNR levels are obtained by adjusting the standard deviation of the added Gaussian noise.

Table 2.2: Variation of SNR according to QS in each digitizer.

	HDO6104A-MS			WaveRunner-640Zi		
SNR	9	7	2	2	1	0.65
QS (mV/bin)	0.1	0.2	0.4	1.5	3	6

This study uses TTT_{ratio} algorithm explained in section 1.3.1 to perform the discrimination between neutrons and gamma-rays. The obtained optimal values of the algorithm's parameters, t_{long} and t_{short}, are within the range of [100, 500] ns and [10, 30] ns, respectively. The discrimination performance is measured using the FOM evaluation metric. The experimental results showed that when the FOM was less than 0.6, it became difficult to distinguish the difference between neutrons and gamma-rays TTT_{ratio} distributions. Therefore, any dataset with a FOM below this threshold is considered to have no discrimination potential in this work.

Table 2.3 shows that the FOM remains approximately constant across all sampling rates when the QS is multiplied by 3 at an SNR of 2. Tables 2.4 and 2.5 indicate that decreasing SNR on the same QS increases the necessary sampling rate to achieve the discrimination. These results suggest that the performance of neutron/gamma-ray discrimination is primarily dependent on SNR and sampling frequency. The discrimination at a certain SNR level can be performed if the sampling frequency exceeds a specific threshold, as shown in Figure 2.6. However, datasets obtained at a lower sampling rate and higher SNR are more suitable for discrimination than those obtained at a higher sampling rate with lower SNR, as illustrated in Figure 2.7.

Table 2.3: FOM for different sampling frequencies, QS and SNR.

SNR	9	7	2	2	1	0.65
QS (mV/bin)	0.1	0.2	0.4	1.5	3	6
fs = 5 GHz	0.98	0.88	0.71	0.73	0.68	0.63
fs = 2.5 GHz	0.97	0.86	0.69	0.72	0.62	0
fs = 1.25 GHz	0.96	0.86	0.68	-	-	-
fs = 1 GHz	-	-	-	0.62	0	0
fs = 500 MHz	0.94	0.83	0	0	0	0
fs = 250 MHz	0.82	0.72	0	0	0	0
fs = 100 MHz	0	0	0	0	0	0

Table 2.4: FOM for QS = 0.1 mV/bin, different sampling frequencies and SNR.

SNR	9	7	4	2	1	0.65
fs = 5 GHz	0.98	0.92	0.84	0.73	0.67	0.65
fs = 2.5 GHz	0.97	0.88	0.82	0.71	0.62	0
fs = 1.25 GHz	0.96	0.87	0.78	0.65	0	0
fs = 500 MHz	0.94	0.85	0.72	0	0	0
fs = 250 MHz	0.82	0.73	0	0	0	0
fs = 100 MHz	0	0	0	0	0	0

Table 2.5: FOM for QS = 0.2 mV/bin, different sampling frequencies and SNR.

SNR	7	4	2	1	0.65
fs = 5 GHz	0.88	0.85	0.71	0.65	0.62
fs = 2.5 GHz	0.86	0.82	0.68	0.61	0
fs = 1.25 GHz	0.86	0.8	0.66	0	0
fs = 500 MHz	0.83	0.79	0	0	0
fs = 250 MHz	0.72	0	0	0	0
fs = 100 MHz	0	0	0	0	0

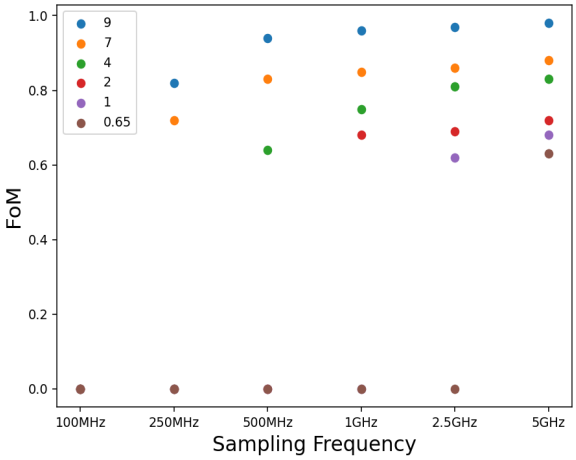
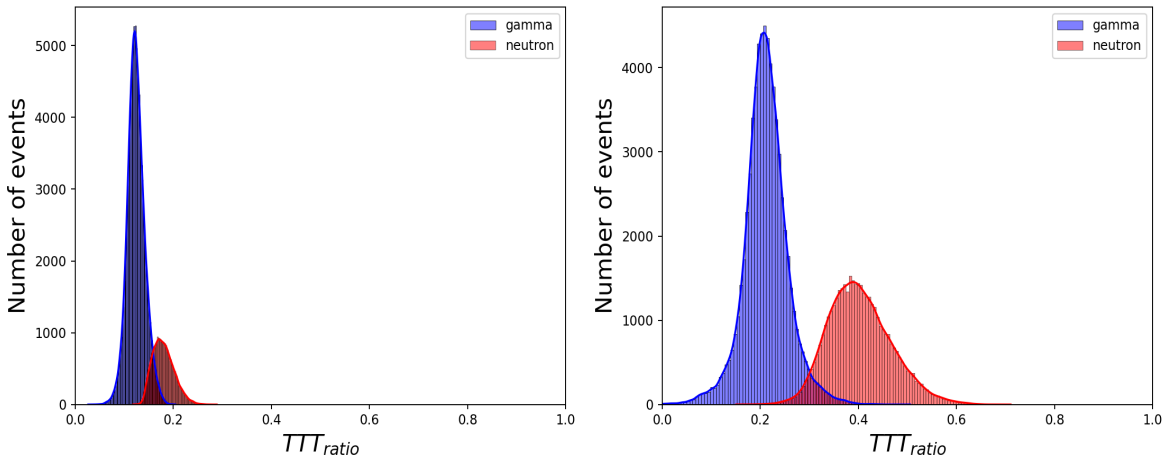


Figure 2.6: Variation of FOM according to f_s for different SNRs (Table 2.3). A trade-off between the SNR and f_s is necessary to improve the discrimination performance.



(a) f_s = 5 GHz, *SNR = 0.65, FOM = 0.63* (b) f_s = 250 MHz, *SNR = 9, FOM = 0.82*

Figure 2.7: TTT_{ratio} distribution of neutron and gamma-ray signals acquired at different sampling frequencies and SNR levels.

The QS indirectly affects discrimination by having an inverse proportional relationship with SNR. Therefore, the QS can be decreased as long as the SNR can be improved to an acceptable level that

matches the required sampling rate for the application. However, it is important to consider the amplitude range of the signals because decreasing the QS may also require reducing the ADC's input voltage range (equation 2.4). Furthermore, Figure 2.8 shows that the minimum sampling frequency required for discrimination decreases as the SNR increases. The relation between the two variables is inversely exponential, which is represented by equation 2.5. The fitting's mean absolute error is 0.15 GHz. The sampling frequency can be reduced to 250 MHz if the SNR is higher than 4. Nevertheless, the discrimination cannot be achieved if the sampling rate is less than 200 MHz, even if the SNR is high, as showed by the frequency analysis in section 2.1.2 and the results obtained in Table 2.3.

$$minimum_{fs} = 17 * exp^{(-2*SNR)} + 0.4 \tag{2.5}$$

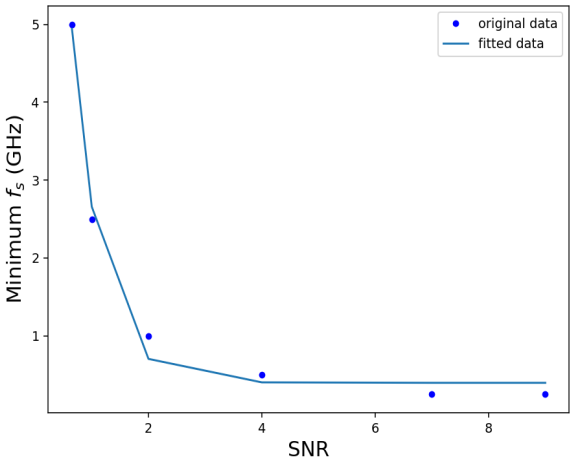


Figure 2.8: Variation of minimum sampling frequency according to SNR (equation 2.5).

In summary, there are four main conclusions drawn from the results of this study:

1. The discrimination performance of a measurement system is not solely dependent on its scintillator and the discrimination algorithm. SNR which depends on each component and parameter of the measurement chain has also an important impact. For this reason, conclusions of a study can be biased by the use of a setup with a low SNR.

2. Neutron/gamma-ray discrimination can be achieved at lower sampling frequency if the SNR is increased. The relation between these two factors is inversely exponential. In the proposed experimental setup, a minimum sampling frequency of 250 MHz is possible if the SNR is higher than 4.

3. At the same sampling rate, the discrimination performance can be improved by increasing the SNR.

4. Information about SNR is essential for ensuring a fair comparison among discrimination approaches across various studies. Each component in the acquisition chain influences discrimination

performance, making the comparison sensitive to subtle differences in acquisition conditions and environments. However, the SNR is often not detailed in studies focusing on neutron/gamma-ray discrimination.

The implemented acquisition chain of this study requires a minimum sampling frequency of 250 MHz to effectively perform neutron/gamma-ray discrimination, for a SNR higher than four. Furthermore, the discrimination performance obtained at this sampling rate can be improved by increasing the SNR. One possible solution to further increase the SNR is by introducing an Op-Amp integrator between the PMT and the ADC, as shown in Figure 2.9. This Op-Amp can also help to reduce the sampling rate required for discrimination. Reducing the sampling frequency can help in minimizing the size and cost of the embedded implementation, which is critical in some applications. The next section explains the concept of this amplifier and its impact on the SNR and the sampling rate required for discrimination.

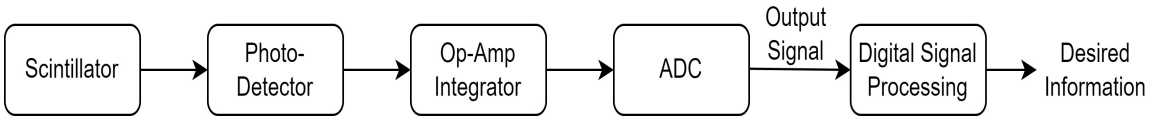


Figure 2.9: A scintillation acquisition chain with Op-Amp integrator.

2.1.7 Impact of Pulse Shaping Amplifier

An Op-Amp integrator is an electronic component widely used in various circuits due to its ability to amplify and manipulate signals. The circuit of this component performs mathematical integration on an input signal. The output voltage of this device is proportional to the integral of the input voltage over time. Figure 2.10 shows the diagram circuit of an ideal Op-Amp integrator. The voltage at the inverting (-) input is equal to the voltage at the non-inverting (+) input as a virtual ground. The input voltage passes a current V_{in}/R_1 through the resistor producing a compensating current flow through the capacitor C_f to maintain the virtual ground. This charges or discharges the capacitor over time. The output voltage (V_{out}) is taken from the output terminal of the Op-Amp, which is connected to the capacitor C_f in the feedback loop. The circuit can be analyzed by applying Kirchhoff current law at the inverting input node:

$$i_1 \;=\; I_b \;+\; i_f \tag{2.6}$$

For an ideal Op-Amp $I_b = 0$ amps:

$$i_1 \;=\; i_f \tag{2.7}$$

Furthermore, the i_f can be expressed in function of capacitor C_f and output voltage V_o by the following equation:

$$i_f = \; C_f \frac{d(V_2 - V_o)}{dt} \tag{2.8}$$

Substituting i_f in equation 2.7 by its equation 2.8:

$$\frac{V_{in} - V_2}{R_1} = C_f \frac{d(V_2 - V_o)}{dt} \tag{2.9}$$

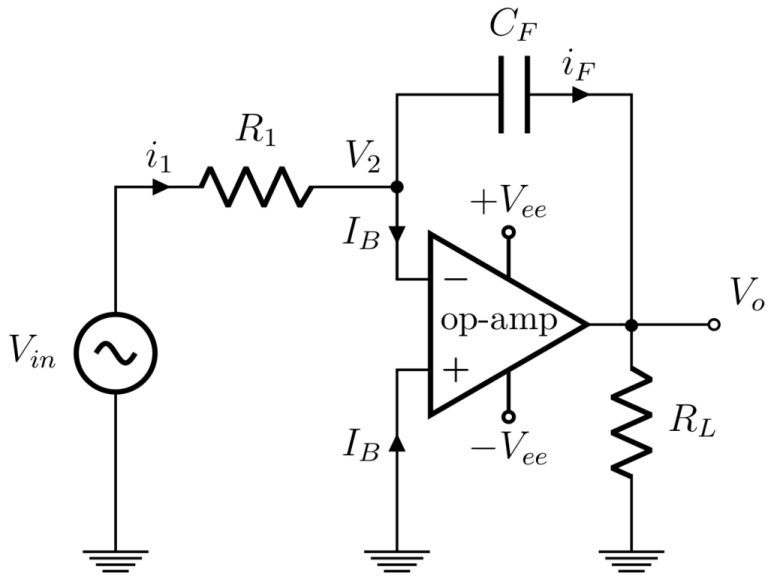


Figure 2.10: Ideal Op-Amp integrator circuit [53].

For an ideal Op-Amp, $V_2 = 0$ V:

$$\frac{V_{in}}{R_1} = -C_f \frac{d(V_o)}{dt} \tag{2.10}$$

Integrating both sides with respect to time:

$$\int_0^t \frac{V_{in}}{R_1} dt = C_f \int_0^t \frac{d(V_o)}{dt} \tag{2.11}$$

If the initial value of V_{in} is V_0, the output voltage will be:

$$V_o = -\frac{1}{R_1 C_f} \int_0^t V_{in} dt + \ V_0 \tag{2.12}$$

Equation 2.12 shows that the output voltage is proportional to the negative integral of the input voltage plus the initial input voltage V_0. R_1C_f is the integration time constant (τ), which determines how fast the capacitor charges or discharges in response to the input voltage. A larger τ means a slower integration, which can extend the length of a signal in the sense that it transforms a short-duration input signal into a longer-duration output signal. This property of signal extension can be useful in noise reduction. Rapid fluctuations or noise in a signal can be smoothed out through integration, resulting in a more stable and less noisy output.

The frequency response of an ideal Op-Amp depends on the reactance of the capacitor in the feedback loop ($X_c = 1/(2\pi f C_f)$). At very low frequencies (DC), the gain is theoretically infinite, resulting in perfect integration of the input signal over time. However, as frequency rises, the gain starts to decrease, causing the integrator to attenuate higher frequency components of the input signal. This means that the frequency response of an ideal Op-Amp integrator involves a transition from high gain at low frequencies to decreasing gain as frequency increases. In other words, an Op-Amp integrator can

be an equivalent to an active low pass filter.

An ideal Op-Amp integrator has infinite gain and bandwidth, enabling the integration of input signals of varying frequencies and amplitudes without any loss or distortion. However, practical Op-Amp integrators are subject to certain factors restricting their performance and precision, such as:

1. Op-Amp characteristics: a practical Op-Amp has finite gain and bandwidth. Furthermore, it has input and output impedance, offset voltage, bias current and certain noise level. These parameters affect the output voltage by introducing errors and deviations from the ideal behavior.

2. Capacitor leakage: the capacitor within the feedback loop of a real-world Op-Amp is not a perfect capacitor. It includes some leakage resistance, which allows a small current to pass through it, resulting in gradual discharge. This phenomenon diminishes the integration impact and contributes to drift in the output voltage.

3. Input bias current: a practical Op-Amp has an input bias current, which flows either into or out of its terminals based on its design and type. This current induces a voltage drop across R_{in} and influences the input voltage perceived by the Op-Amp, leading to an associated deviation in the output voltage.

4. Frequency response: the frequency response of a practical Op-Amp introduces both phase shifts and distortion in the output signal. Additionally, at extremely low frequencies (DC), the voltage gain may surpass the Op-Amp output range, leading to saturation or clipping effects.

Practical Op-Amp integrators are usually modified and include additional stages and components to surpass the above mentioned limitations. In this work, the Op-Amp integrator is used as a tool to improve the SNR of the neutron/gamma-ray signal and reduce the required sampling frequency for the discrimination. Employing a novel approach of integration dedicated for radiation measurement is not the objective of the study. Therefore, we used the commercially amplifier "2111 Timing Filter Amplifier" available in our laboratory [54]. By adjusting the integration time constant setting of this device, the user can extend the pulse decay time. However, this increase in decay time results in a reduction in amplitude. The user can therefore compensate this loss, or even boost the signal amplitude, by adjusting the coarse and fine gain settings.

2.1.7.1 Data Acquisition

The data acquisition for this study is achieved by adding the 2111 Timing Filter Amplifier between the PMT and the oscilloscope, as shown in Figure 2.11. The acquisition is done with a time window, triggering threshold, quantization step, and baseline equal to 10 microseconds, 15 mV, 0.1 mV/bin, and 1 microsecond, respectively.

The variation of integration time parameter provides signals with different lengths and power frequency densities. To examine this, five datasets are acquired at a sampling rate of 250 MHz using 5 different integration time configurations. For each configuration, the coarse and fine gain are tuned to obtain a dataset that has approximately the same maximum amplitude distribution of the raw signals. Frequency analysis is then conducted on all the datasets. As it is expected, longer delay times led to a decrease in the maximum frequency components, as indicated in Table 2.6.

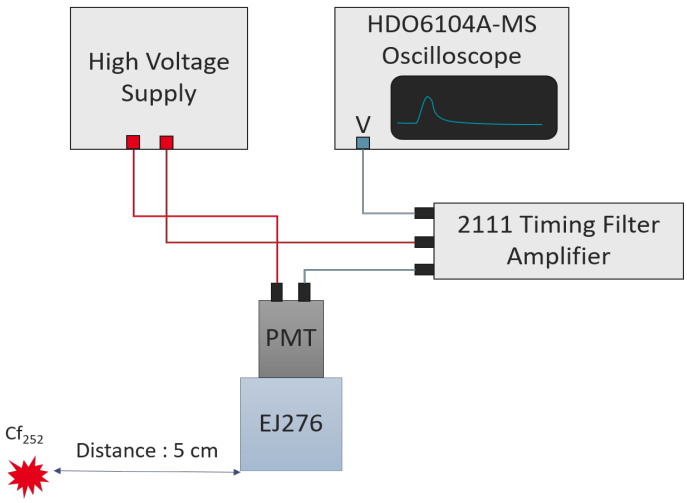


Figure 2.11: The implemented acquisition chain with Op-Amp amplifier.

Table 2.6: Five different configurations of 2111 Timing Filter Amplifier. These configurations provide signals with different length, SNR and frequency power spectrum.

Fine Gain	Coarse Gain	Integrate (ns)	Maximum frequency component
0	3	20	40 MHz
0.75	3	50	20 MHz
1.25	3	100	12 MHz
1.75	3	200	10 MHz
1.25	10	500	5 MHz

2.1.7.2 Results

For each setting of the Timing Amplifier, a dataset is collected at the corresponding minimum sampling frequency (twice the highest frequency component). The resulting optimal values for the long gate parameter are used to calculate the SNR. The results presented in Table 2.7 show that discrimination is possible at the minimum sampling rate for each integration time configuration, primarily due to the relatively high SNR. Furthermore, the comparison between the results obtained on the raw signals and the signals acquired with an integrator value of 20 ns, shows that the discrimination can be achieved on the latter with better performance at lower sampling rate, using the same signal duration. The FOM values of the former and the latter are 0.72 and 1.02, respectively.

Table 2.7: FOM and SNR obtained on each configuration of the Timing Shaping Amplifier.

Integrate (ns)	$f_s(MHz)$	Long gate (ns)	Short gate (ns)	FOM	SNR
20	100	500	20	1.02	24
50	50	1000	100	0.98	23
100	25	2000	100	0.95	28
200	25	4000	500	0.92	25
500	10	7000	500	0.9	26

To summarize, connecting a PMT to an integrator amplifier can improve the discrimination performance and reduce the sampling rate required for discrimination. In other words, it can help to minimize the cost and size of embedded implementation for online classification. However, using an Op-Amp integrator can increase the dead time and negatively affect the maximum acceptable counting rate, as explained in section 1.2. Thus, the use of an integrator amplifier could be a viable option to reduce the sampling frequency, depending on the considered application. In this work, we are interested to improve the online discrimination performance directly on raw signals using embedded ML approach. This can help to minimize the loss of information during the acquisition for a classification on the fly, as detailed in section 1.2.

2.2 Labeling Method for Neutron/Gamma-ray Datasets in Organic Scintillator

In order to be able to correctly assess a discrimination method based on organic scintillator data, it is necessary to have a reliable dataset of both neutron and gamma-ray emissions, which can be also used to train a supervised ML model and compare different discrimination approaches. While obtaining a dataset of gamma-rays is possible using pure gamma-ray sources like ^{60}Co and ^{137}Cs, creating a dataset for neutron emissions is more difficult due to the fact that neutron sources are also gamma-ray emitters.

In section 1.3 of chapter 1, we showed that TTT_{ratio} PSD algorithm is widely used and very well documented in the literature to achieve the neutron/gamma-ray discrimination in organic scintillator. However, the discrimination performance of this algorithm is proportional to the energy of the incident radiation as explained in section 1.3.1. ToF is another technique used to label and discriminate neutrons from gamma-rays. This method relies on the speed difference between gamma-rays and neutrons to discriminate them, considering the gamma-ray is faster than the neutron. Nevertheless, acquired labeled datasets often contain mislabeled samples, which have various origins, such as the overlap between the gamma and neutron arrival time distributions and natural background radiations. These mislabeled samples have a penalizing impact on the evaluation of any discrimination approach.

In this section, a process for generating labeled and clean datasets of neutrons and gamma-rays in an organic scintillator, as illustrated in Figure 2.12. The first step in the process consists in acquiring the signals via an implemented ToF setup. The next step involves identifying and removing pile-up events, as experimental results showed that these events are relatively prevalent in the collected datasets. A pile-up event is a signal containing several overlapped pulses. The remaining signals from the pile-up detection step are then processed to decrease the number of background events and labeled based on ToF information. The final step of the process aims at reducing the percentage of mislabeled samples present in the obtained datasets using TTT_{ratio} discrimination algorithm. Recently, a similar approach has been proposed to remove the mislabeled samples in the dataset acquired by ToF setup using EJ309 liquid organic scintillator [55]. However, the energy distributions of classified neutrons and gamma-rays do not correspond to the chosen thresholds to remove the mislabeled samples. At the end of this section, a summary of the main contributions and limitations of this labeling process is proposed.

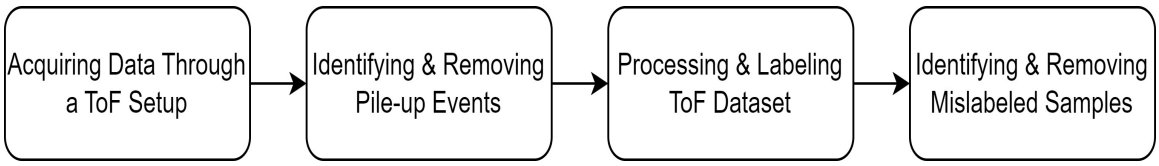


Figure 2.12: The main steps of the proposed labeling method.

2.2.1 Time of Flight Setup

The experimental ToF setup involves two detectors separated by a distance L, as illustrated in Figure 2.13. The radiation source is placed in front of the start detector. When the oscilloscope identifies the detection of radiations at both detectors and the time between the two identifications is less than a preset time threshold, it records two signals corresponding to the radiation detected by each detector. Neutron and gamma-ray signals of the stop detector can be separated by the ToF between the two detectors, which is calculated from the time duration between the peaks of both recorded signals. However, obtained labeled signals often contain mislabeled samples that should be removed to obtain a clean reference neutron/gamma-ray datasets.

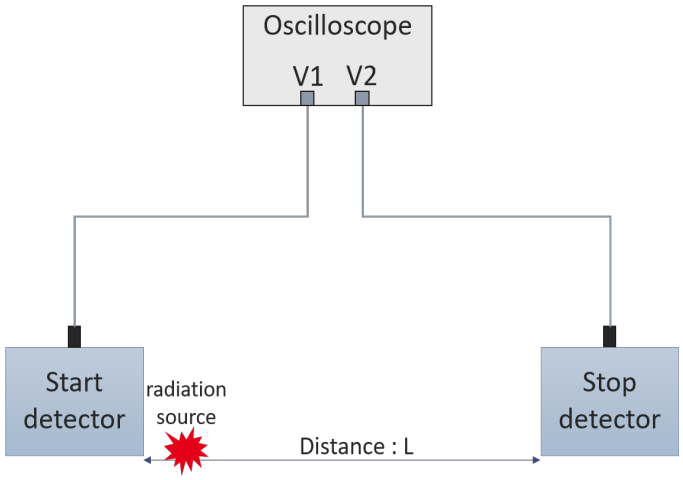


Figure 2.13: Diagram representing the configuration of ToF experiment.

The article [56] introduces an approach for minimizing the labeling error of ToF setups that is based on liquid scintillator. However, the implemented method cannot remove background events that coincidentally contribute to peaks in the ToF distribution. An alternative method for improving the precision of ToF measurements with liquid scintillator involves the use of an optimization algorithm to identify the time interval within the ToF distribution that has the lowest number of mislabeled samples [45]. Both approaches generated training and validation datasets of ML models presented in [57] and [16], respectively. In our study, the proposed method to reduce the error combines the results of ToF and TTT_{ratio} discrimination algorithm. By using this approach, mislabeled samples from the entire ToF distribution of the acquired dataset can be detected and removed.

As illustrated in Figure 2.14, the components used to implement the ToF setup of this study are:

- Start scintillator: EJ200 non-discriminating plastic scintillator made by Eljen Technology. This scintillator is not able to discriminate between neutrons and gamma-rays.
- Stop scintillator: EJ276 discriminating plastic scintillator.
- Photomultiplier: 9821B model produced by ET Enterprises, which was supplied by 1700V.
- Oscilloscope: WaveRunner-640Zi with 8 bits vertical resolution.
- Radiation source: ^{252}Cf.

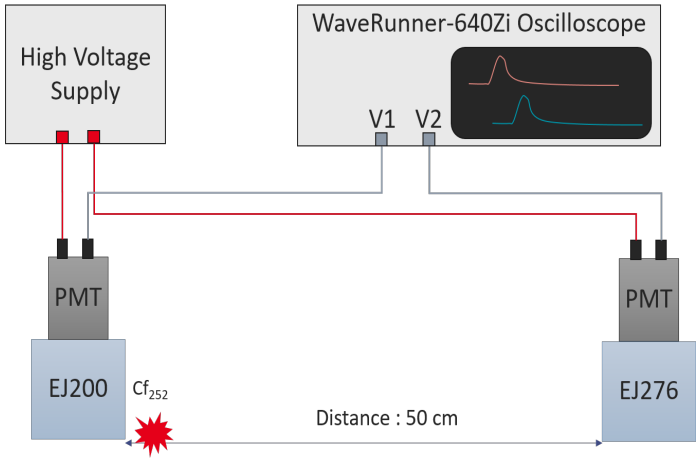


Figure 2.14: Implemented ToF experiment.

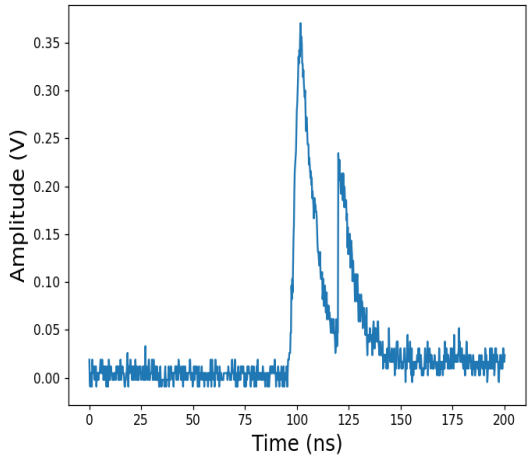
The distance between the two detectors in Figure 2.14 is 50 cm. The acquisition parameters are set to a triggering threshold of 60 mV, a sampling frequency of 5 GHz, a time window of 200 ns, a V_{pp} of 3.2 V, and a maximum duration between the triggering of the start and stop detectors of 60 ns. After the acquisition, a duration of 100 ns is assigned to the signal from its identified starting point, while the baseline is given a length of 50 ns. It should be highlighted that the experimental results obtained from this ToF setup showed a significant percentage of pile-up events within the collected dataset. Thus, an offline detection method described in next section was implemented to remove these events.

2.2.2 Pile-up Detection & Evaluation

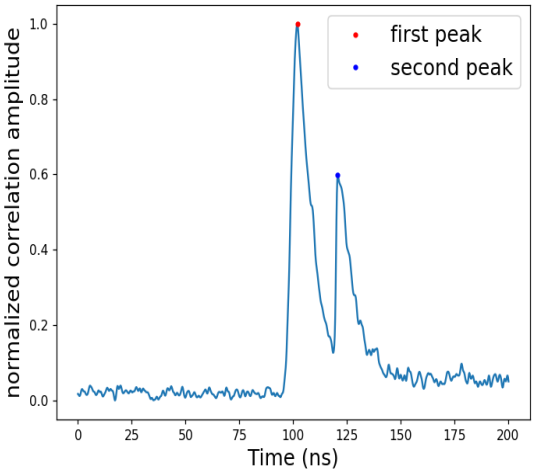
Pile-up is a phenomenon that can occur when measuring signals with non-zero durations, such as neutrons and gamma-rays, whose arrival times follow a Poisson distribution. This event occurs when more than one pulse is detected within the recorded duration, resulting in the detection of multiple local peaks in the overall acquired signal, see Figure 2.15(a). These events are usually detected and rejected using a dedicated electronic system [58]. Nevertheless, some of them may be missed in the implemented ToF setup due to its coincidence acquisition characteristic and natural background radiations. As a result, an offline method is necessary to process these events and obtain a reliable datasets for neutron/gamma classification. Furthermore, evaluating the performance of the detection method can be challenging, as pile-up events in the acquired dataset cannot be easily identified. Therefore, a method for synthesizing

pile-up events must be included in the evaluation process to assess the detection performance.

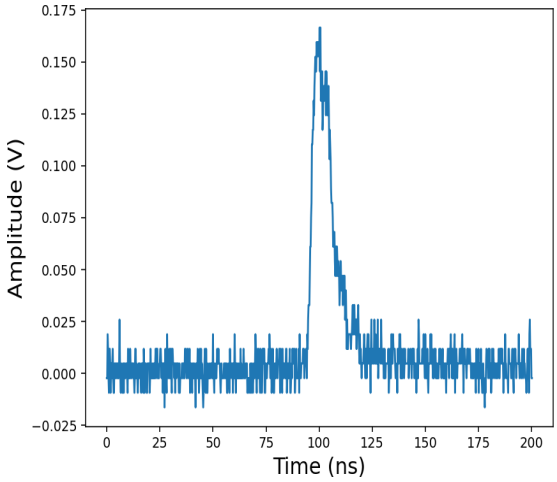
The performance of a pile-up detection algorithm is measured by determining its detection error, which is the percentage of undetected signals in a dataset consisting exclusively of pile-up events. Detecting pile-up events becomes more challenging as the time difference between the arrival of the two contributions within an event becomes smaller, see Figure 2.15(c). Consequently, the evaluation must take this factor into account to identify the minimum threshold at which the detection error remains within an acceptable range. In our research, we consider a detection error of less than 1% to be acceptable.



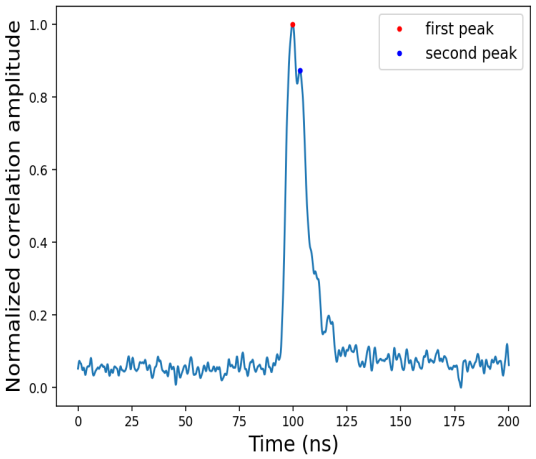
(a) A synthetic pile-up event with large arrival time difference between the two events



(b) Correlation output



(c) A real pile-up event with short arrival time difference between the two events



(d) Correlation output

Figure 2.15: Real detected pile-up events before and after the correlation with Gaussian kernel.

2.2.2.1 Detection Method

There are various algorithms available for handling digital detection of pile-up events. In [59], the authors propose a pile-up identification method based on first order derivative. Their method relies on the fact that zero-crossing point corresponds to the peak point of the original signal. Since the presence of noise in the original signal can cause many false zero-crossings, the first-order derivative is smoothed first using a moving-average filter. Then, the search for zero-crossing points is achieved. Each zero-crossing point that have a slope higher than a predefined threshold and its corresponding point in the original signal exceeds a certain amplitude threshold is counted as a peak. Finally, a signal is identified as pile-up event if two or more peaks are detected.

Another method for detecting pile-up events in germanium detectors has also been proposed, which benefits from the relatively long signal length (in microseconds) to identify them [60]. The method calculates the width of a signal at 20% of its maximum value and compares it with an average value calculated using a dataset of single events. If the pulse width is larger than the average value, it is considered as pile-up.

Fitting method is proposed in [61] for pile-up detection in liquid scintillators. First, the pulse signal from is modeled using a double exponential with two decay time constants. The parameters of the model are calculated using database acquired by an implemented experimental setup. Then, peak detection algorithm based on a least squares fit is used to identify the number of peaks in a acquired signal. However, the authors do not provide more details about this algorithm. Furthermore, in order to avoid false peak detection due to noise, the minimum distance between consecutive peaks has been set to 15 ns. In other words, pile-ups with overlapping pulses whose peaks are separated by less than 15 ns cannot be identified. These events are therefore classified as single pulses.

Recently, a MLP model has been developed that can simultaneously classify neutron, gamma-ray, and pile-up events of a stilbene detector [13]. The model is trained on neutron/gamma datasets labeled by TTT_{ratio} algorithm and a synthetic pile-up dataset. Despite the high accuracy achieved, the model is not evaluated for the minimum difference between the arrival times of the two contributions for which the model can still detect a pileup with high certainty.

In nuclear instrumentation, pile-up detection and rejection constitute a broad field of study. In this work, the objective of the pile-up detection method is to clean up the dataset acquired by the ToF setup. Introducing a novel pile-up detection method with an organic scintillator is beyond the scope of this research. Furthermore, evaluating various pile-up detection techniques and selecting the most effective one can be a challenging and time consuming work, which may divert our attention from the main objective, improving the neutron/gamma-ray discrimination in plastic scintillator using ML for real time application. Therefore, the pile-up detection method from [62] is adapted to clean the ToF dataset. The proposed adapted method can detect pile-up events with a satisfying level of performance, as we will show in the evaluation step in Section 2.2.2.2. While acknowledging that a more powerful method may exist in the literature, the performance achieved by this method satisfies the need for clean labeled data.

The proposed method is based on the cross-correlation between the output signal of the detector and a predefined Gaussian kernel. Cross-correlation measures the similarity between two vectors over

time and the shape of an output signal produced by an organic scintillator is similar to the shape of a Gaussian window. This means that each peak in the cross-correlation output could indicate an event in the acquired signal, as shown in Figures 2.15(b) and 2.15(d). However, a Gaussian window that is too wide will cause the peaks of two closely occurring events to merge together. Conversely, a narrow window will result in detecting most signals as pile-ups, since any noise peak can be similar to the created narrow kernel and be identified as a pulse peak.

The optimal width of the kernel window depends on the standard deviation used to create the kernel and the number of points in the kernel array. In the presented case, the kernel size is the same as the signal length and the standard deviation should be adjusted to find its optimal value. The tuning can be done by gradually decreasing the standard deviation until the algorithm starts to misidentify the majority of the signals in a collected dataset as pile-ups. In other words, the optimal value is the point at which decreasing the standard deviation further would lead to a substantial misidentification of actual signals, while increasing it would lead to a considerable number of misidentified pile-up events. In the presented case, the obtained optimal value of the standard deviation is equal to 4. The first four lines of algorithm 1 summarize these steps. However, it should be noted that using the optimal standard deviation value may misidentify some signals as pile-up events. The objective of this detection method is to prepare clean neutron/gamma-ray datasets and the performance in terms of counting rate is not a concern.

Algorithm 1 Pile-up detection

0: **function** $pile-up-detection(signals)$
1: $sigma = obtainedoptimalvalue$
2: $window = Gaussian(length(signal), sigma)$
3: $corr = correlate(signal, window)$
4: $corr = (corr\text{-}min(corr))/(max(corr)\text{-}min(corr))$
5: $index_{peaks}$= *find-peaks (corr, high, width, distance)*
6: **if** $length(index_{peaks} > 1)$ **then**
7: $pile-up-event = 1$
8: **else**
9: $pile-up-event = 0$
10: **end if**
11: **return** $pile-up-event$

After the correlation, in line five of algorithm 1, the peaks in the obtained output are detected using the implemented *find_peaks* function in *scipy* library in Python. The function finds all local maximums by a simple comparison of neighboring values. Some of the detected peaks refer to noise signals. Therefore, by specifying conditions for the peaks properties, a subset of the detected peaks can be selected, Figures 2.15(b) and 2.15(d). These properties are width, amplitude, and the distance between two peaks. Finally, if the number of peaks that are found is higher than one, the acquired signal is considered as a pile-up.

2.2.2.2 Evaluation of the Detection Method

To evaluate this detection method, a synthetic pile-up dataset is generated using real signals acquired by the acquisition chain depicted in Figure 2.1 with a ^{60}Co and ^{252}Cf sources. The oscilloscope is exchanged by WaveRunner-640Zi and the acquisition is carried out with identical ToF measurement chain settings. The size of each obtained dataset was 50,000 signals.

A synthetic pile-up event is created by selecting randomly two acquired signals, shifting the second one, then adding them together. Figure 2.15(a) shows an example of synthesised pile-up event. The shifting is determined by the *arrival time difference* parameter, which can be assigned to a constant or random value within a given predefined time interval. The shape of a created pile-up signal is quite realistic and similar to the shape of a real pile-up event. This is because the PMT current is proportional to the incident radiation fluxes magnitude, which are linearly added when multiple radiations are detected in the same time window of the acquisition [35]. Moreover, this method allows us to precisely control the temporal arrival difference between the two components of a synthetic pile-up event. Therefore, the detection error can be evaluated according to the variation of this parameter. Table 2.8 summarizes the results of this evaluation, where the value of *arrival time distance* is adjusted between 5 and 15 nanoseconds. For each value, the execution is repeated 10 times, each time 100000 pile-up signals are generated.

Table 2.8: Pile-up detection error (%) according to arrival time distance (nanoseconds) between the two contributions in a pile-up event.

Arrival time difference (ns)	5	6	7	8	9	10	11	12	13	14	15
Detection error	77.6%	66.5%	44.2%	29.8%	22.8%	17.1%	10.2%	3.9%	1.1%	0.6%	0.5%

The minimum *arrival time difference* required to keep the error less than 1% is 14 nanoseconds, meaning the proposed detection method can accurately identify a signal as a pile-up event if the second contribution arrives 14 ns after the first one. This does not imply that the method cannot detect pile-up events with a shorter arrival time difference between two incoming radiations. Nonetheless, as indicated in Table 2.8, the likelihood of detection decreases when the distance between the two arrival times is less than this threshold. These results show that the simple implemented detection method with a single tuning parameter (standard deviation of the Gaussian kernel) is adapted to clean up pile-up events from the signals acquired by the ToF setup with a satisfying level of performance.

Finally, the obtained optimal value of the standard deviation of the Gaussian kernel is used to apply the algorithm on the datasets obtained by the ToF acquisition chain. The percentage of the detected pile-up events is 8.8% of the signals. Some of the detected pile-up events may be misidentified. In the presented case, this is not a critical concern, as the primary objective is to generate clean neutron/gamma-ray datasets.

2.2.3 Processing & Labeling ToF Dataset

The remaining signals from the pile-up removal step contains two main classes: neutron and gamma-ray. This can be clearly seen in Figure 2.16, which visualizes the ToF distribution. The left and right distributions correspond to the gamma-rays and neutrons, respectively. As gamma-rays are faster than

neutrons, their ToF values are lower. However, the two ToF regions contain mislabeled gamma and neutron samples as we mentioned in section 2.2.1. In section 2.2.4, we will work on reducing the percentage of these events using TTT_{ratio} discrimination algorithm.

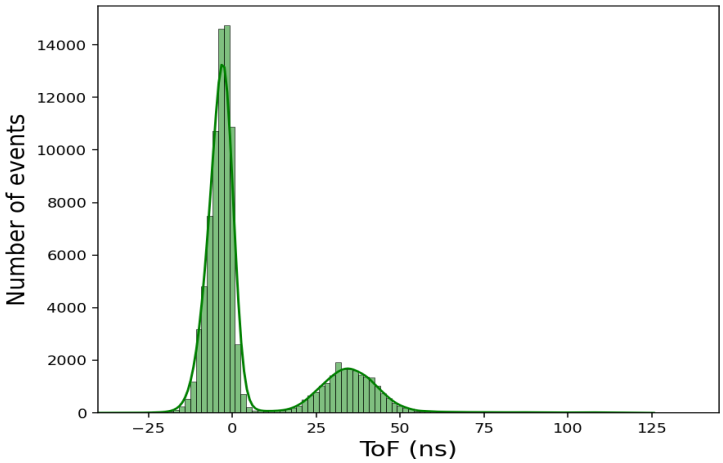


Figure 2.16: ToF Distribution

It is worth mentioning that the obtained average ToF value of the neutron class (37 ns) is compatible with the expected value of the theory. This value is equal to 35.7 ns, considering the experimental set-up previously described. It can be computed by considering that the distance between the two detectors is equal to 50 cm and the mode energy of the neutrons emitted by the ^{252}Cf is 1 MeV which corresponds to a speed equal to 1.4 cm/ns [63]. Concerning the negative average ToF value of the gamma signals (-3 ns) is due to the difference response time of two detectors and the time resolution of the oscilloscope, since the speed of gamma is approximately equal to the speed of light in the air.

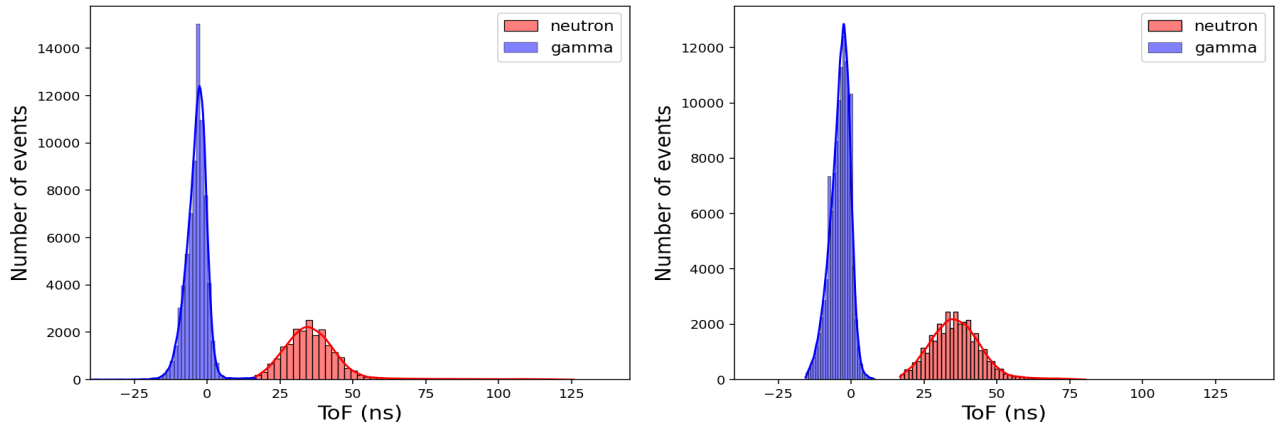
The samples located at the extremes of each class's distribution are primarily either mislabeled or irrelevant background events. Moreover, the region lying in between the two distributions is associated with significant uncertainty in classification. Therefore, in order to ensure accurate classification of acquired signals based on ToF parameter and to obtain well-labeled datasets, it is necessary to eliminate these samples. This can be done through the following steps:

1. Separate the detected radiations into two classes by applying the K-mean algorithm [64], taking the ToF values of the signals as input. The effectiveness of this method is illustrated in Figure 2.17(a), which provides an example of how the data can be well-separated into two classes.

2. Calculate the ToF mean (μ) and standard deviation (σ) of each cluster.

3. Remove signal in each cluster i, if its ToF value is greater than $\mu_i + \beta * \sigma_i$, or less than $\mu_i - \beta * \sigma_i$, where μ_i and σ_i are the mean and the standard deviation of cluster i.

The percentage of removed signals is inversely proportional to the value of β. The ToF distributions for both classes are normally distributed. In this work, the value of β is set to 3, resulting in removing only 0.2% of the signals from each distribution. Figure 2.17(b)) shows the ToF distribution of neutron and gamma-ray classes after removing the samples from the uncertain regions. This approach ensures that the neutron and gamma-ray clusters are well-separated with low percentage of signal rejection.

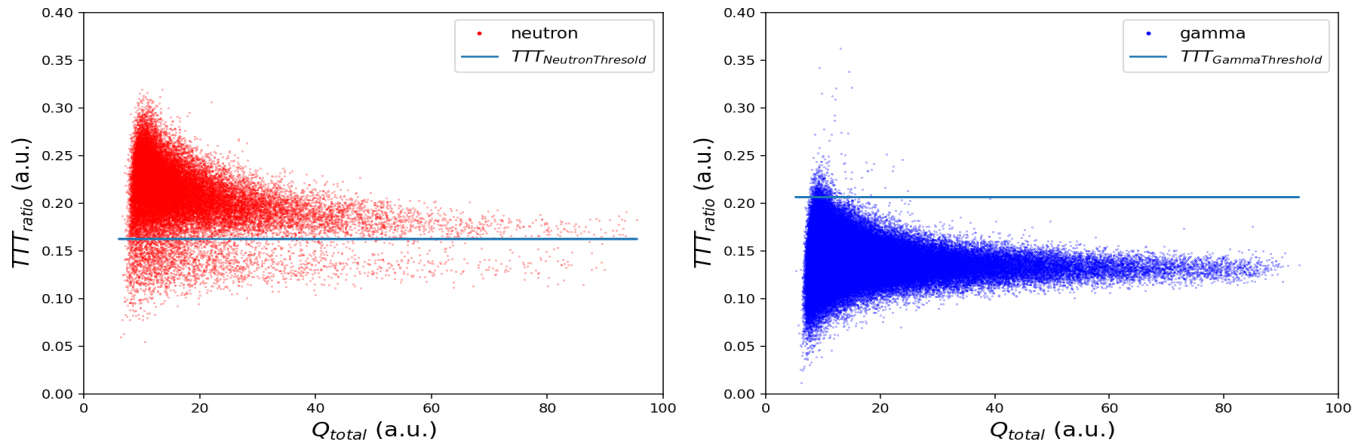


(a) ToF distribution of signals labeled by K-mean. (b) ToF distribution for β equal to 3.

Figure 2.17: Labeling of ToF output vector by K-mean algorithm.

2.2.4 Identifying & Removing Mislabeled Samples

Figure 2.18(a) shows the 2D graph of TTT_{ratio} according to total energy for signals labeled as neutron by ToF parameter. The figure contains two main clusters representing true and mislabeled neutron radiations. To separate them and obtain a clean neutron dataset, a convenient $TTT_{NeutronThreshold}$ must be chosen. The TTT_{ratio} distribution of both clusters in Figure 2.19 reveals two peaks corresponding to each cluster. $TTT_{NeutronThreshold}$ is chosen to be equal to the TTT_{ratio} value of the first peak plus half-width at half maximum, as shown in Figure 2.19. The ToF distribution of the removed samples is approximately similar to the original ToF distribution of the neutron cluster. Thus, with this approach, even background events that contribute to the peaks in the ToF spectrum can be removed.



(a) neutron class (b) gamma class

Figure 2.18: TTT_{ratio} according to total integral for each class of signals labeled by ToF parameter.

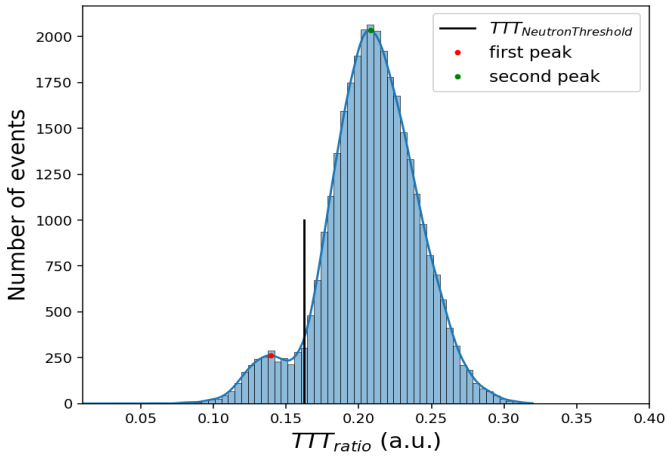


Figure 2.19: TTT_{ratio} distribution of signals labeled as neutron by ToF.

It is important to note that all background events cannot be identified by this method, and some neutron events may be misclassified. Furthermore, the removed signals may contain true neutron samples. Nevertheless, the performance of neutron counting rate is not a parameter of interest for this work.

Concerning the signals labeled as gamma-rays, the TTT_{ratio} distribution according to total energy has one main cluster, see Figure 2.18(b). In other words, the contamination by mislabeled samples is relatively low. In order to increase the labeling accuracy, a $TTT_{GammaThreshold}$ is defined, which is calculated based on the normally distributed TTT_{ratio} obtained from a pure gamma dataset emitted by ^{60}Co, by adding three times the standard deviation ($3 * \sigma_{\gamma_{TTT}}$) to the mean ($\mu_{\gamma_{TTT}}$), see Figure 2.20. Consequently, the rejection rate is approximately equal to 0.5%. Finally, at the end of this labeling pipeline, four datasets can be obtained.

1. neutron: signals are labeled as neutron using the proposed strategy.
2. gamma-ray: signals are labeled as gamma-ray using the proposed strategy.
3. mislabeled-neutron: signals classified as gamma-ray by TTT_{ratio} method and neutron by ToF technique.
4. mislabeled-gamma-ray: signals classified as neutron by TTT_{ratio} method and gamma-ray by ToF technique.

It should be emphasized that the primary objective of the proposed labeling process is to create a pure neutron dataset. Gamma-ray signals can be directly obtained from gamma-ray sources, such as ^{60}Co or ^{137}Cs. Furthermore, in this labeling method, the cutoff for the minimum energy radiation is exclusively implemented during the acquisition, using the trigger parameter of the ADC. Subsequently, all acquired signals undergo processing for labeling or rejection.

To sum up, the ToF setup initially acquires 165,000 signals. After that, pile-up events are identified and removed by the proposed detection algorithm, which account for 8.8% of the dataset. The remaining signals are further processed and separated into neutron and gamma-ray clusters. The final stage consists in removing the mislabeled samples by TTT_{ratio} algorithm. The resulting neutron and gamma-ray datasets contain 29600 and 117600 signals, respectively.

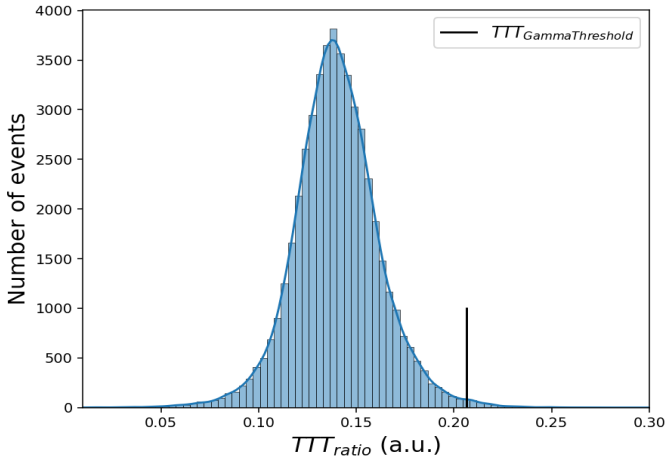


Figure 2.20: TTT_{ratio} distribution of gamma-rays emitted by ^{60}Co.

2.2.5 Contributions and Limitations

The presented labeling process can identify and remove mislabeled samples in the entire ToF spectrum, including those that contribute to peak values. This process cleans ToF dataset from pile-up events, which can significantly impact experimental results and the conclusions extracted from them. This point is usually ignored in the literature when ToF discrimination approach is employed to label the signals. Furthermore, the process can provide labeled datasets, even when the discrimination ability of the implemented acquisition chain is relatively low. In the presented experimental setup, the obtained FOM is equal to 0.6, see Table 2.3. More accurate labeled dataset can be obtained in a similar acquisition chain with higher discrimination ability, which is primarily dependent on SNR, as demonstrated by the results of this chapter.

The dependency on TTT_{ratio} discrimination algorithm is one of the main limitations of the proposed signal labeling method. Another limitation is the dependency on the sampling frequency and on the energy range of the incident radiations. The energy of the particle and its type determine both the distance it can travel and its speed during travel. Therefore, when dealing with lower energy radiations, the distance between detectors needs to be reduced, resulting in lower ToF values for higher energy radiations. To detect these lower ToF values, the sampling rate must be increased. Moreover, compared to the traditional method, the ToF measurement chain is more complicated to be implemented, and the acquisition process takes longer time due to coincidence detection. This becomes particularly challenging as a large dataset is required to effectively identify the peak of mislabeled neutron dataset in the TTT_{ratio} distribution of the neutron signals labeled by ToF.

2.3 Conclusion

Reducing the sampling frequency is a crucial step in order to minimize the cost and size of a discrimination approach. This point is key in the framework of our study for embedding ML approaches on edge devices. Therefore, in the first section of this chapter we analyzed the discrimination performance between neutrons and gamma-rays in an organic scintillator according to the variation of SNR and ver-

tical resolution at different sampling frequencies. TTT_{ratio} algorithm and FOM were used to assess the discrimination ability of implemented measurement system. The study found that this discrimination ability mainly depends on the SNR and sampling frequency, while vertical resolution has an indirect impact since SNR is proportional to it. According to the experimental setup implemented in this study using EJ276 plastic scintillator, the minimum required sampling rate for discrimination decreases as SNR increases, and it can potentially reach a minimum of 250 MHz. Furthermore, considering the same sampling rate, the obtained results showed that a higher SNR can offer a better discrimination performance. Moreover, using an integrator amplifier in the acquisition system can significantly reduce the necessary sampling frequency for discrimination and improve the discrimination performance compared to raw signals, as demonstrated by the experimental results of this study. However, this decrease in sampling rate is associated to an increase in signal duration, which could be a severe limitation, as soon as the counting rate will be a parameter to be taken into account.

In summary, the performance of a discrimination approach is strongly dependent on the parameters and components of the acquisition chain. Even subtle changes in the acquisition conditions and environment can significantly affect the discrimination performance. Therefore, it is essential to carefully design and optimize the acquisition chain for a specific application to obtain the best discrimination performance. Comparing discrimination approaches across various studies can be difficult due lack of information regarding acquisition conditions. Providing the SNR can be a solution to ensure a fair comparison. Finally, it is worth to mention that the discrimination performance is proportional to the energy range of the incident radiations. This factor was not considered in this study. The acquired datasets by different acquisition chains were ensured to be in the same energy domain.

The second section of this chapter presents a method to prepare clean and labeled neutron/gamma-ray datasets using an organic scintillator. The initial step is to gather the dataset through an implemented ToF setup. The subsequent stage comprises detecting and removing pile-up events using a proposed and evaluated detection method. The remaining signals are then processed to decrease the number of background events and classified into neutron and gamma-ray classes. Finally, TTT_{ratio} discrimination algorithm is employed to reduce the percentage of mislabeled samples present in the obtained datasets.

The described labeling procedure has the capability to identify and remove mislabeled samples throughout the entire ToF spectrum, even those contributing to the peak values. This method cleans the ToF dataset from pile-up events, which can have a significant impact on experimental results and the conclusions extracted from them. This aspect is often ignored in the literature when the ToF discrimination approach is used to label signals. Additionally, this process can generate labeled datasets even in cases where the discrimination ability of the acquisition system is relatively low (FOM equal to 0.6). A key limitation of this proposed labeling method is its reliance on the TTT_{ratio} discrimination algorithm. Another limitation is its dependence on the sampling frequency and the energy range of incident radiation. Furthermore, when compared to the traditional method, the ToF measurement setup is more complex to implement, and the acquisition process is more time-consuming due to the need for coincidence detection.

Next chapter explores the discrimination of neutrons and gamma-rays in an organic plastic scintilla-

tor by supervised and unsupervised ML models. Different ML models and the state of the art algorithm (TTT_{ratio}) will be evaluated based on the variation of sampling frequency and the energy range of the incident radiations. Training and validation datasets of the implemented models will be obtained by the proposed labeling process.

Chapter 3

Supervised ML for Radiations Classification: A Comparison with a State of the Art Discrimination Algorithm

Chapter 2 introduced a labeling process to generate clean labeled neutron/gamma-ray datasets, which can be used to train and evaluate effectively ML models for the discrimination task. The objective of this chapter is to explore whether the use of ML tools, compared to the state of the art algorithm (TTT_{ratio}), can lead to a reduction of the minimum required sampling frequency in EJ276 plastic scintillator, as well as an enhancement in the classification performance, especially for low energy radiations.

3.1 Introduction

MLP and 1D CNN ML models have been used in various studies for neutron/gamma-ray discrimination [13, 16, 19, 40], as detailed in section 1.3.2. The models in these studies are trained directly on 1D time signals, as illustrated in Figure 3.1. They exhibit superior performance compared to PSD algorithms. However, these approaches have distinct limitations related to dataset preparation and embedded implementation, as discussed in section 1.3.2. In this chapter, these two ML model types will be trained and compared to TTT_{ratio} algorithm, considering variations in sampling frequency and radiation energy. More complex structures of these models can be explored if the obtained results are not promising and cannot outperform the state of the art. Nevertheless, due to the promising results reported in the literature and the simple nature of input data (1D time signal represented by several hundred sampling points, as illustrated in Figure 3.2), we expect that these models can achieve discrimination with higher performance.

The main limitation of a ML approach trained on raw signals lies in its sensitivity to the distribution of the training dataset. In the context of neutron/gamma-ray discrimination, the distribution of two datasets acquired by distinct measurement chains can vary in multiple aspects, including the input voltage range and the number of sampling points representing a signal. Consequently, for any subtle change in the components and parameters of the acquisition chain impacting the distribution of the input data, the data preparation and training of the model should be repeated, which are challenging and time consuming work. Unsupervised model adaptation can offer a solution to overcome this data

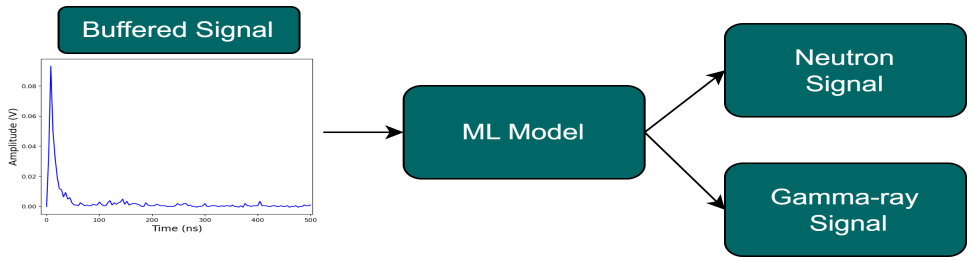


Figure 3.1: An illustration of a ML model for neutron/gamma-ray discrimination, taking as input a raw signal.

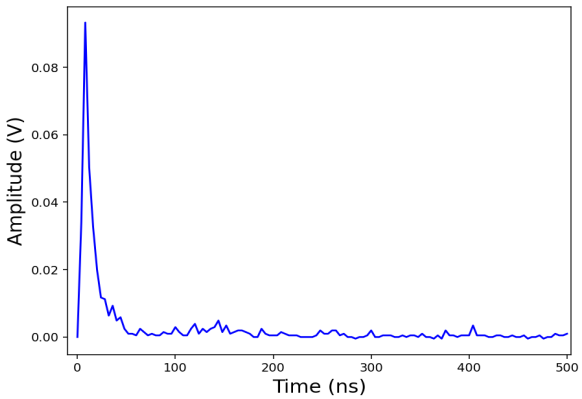


Figure 3.2: An example of a raw signal used for training MLP and 1D CNN models. The signal is obtained by EJ276 plastic scintillator and a ^{252}Cf source. The sampling frequency and signal duration are 250 MHz and 500 ns, respectively. The input length is 125 sampling points.

preparation problem by adapting a pre-trained model to a new unlabeled dataset acquired by a different measurement chain. However, a main limitation in adapting a model trained on raw signals is its dependence on the input length, determined by the signal duration, which itself relies on the characteristic of the acquisition setup. A potential solution is to perform features extraction from the signal, which can be used to train a ML model instead of using the entire raw signal, as illustrated in Figure 3.3. This approach involves reducing the data dimensionality to a manageable level that is independent of the acquisition chain and the input signal length. This is the second ML approach that will be explored to perform the discrimination in this chapter.

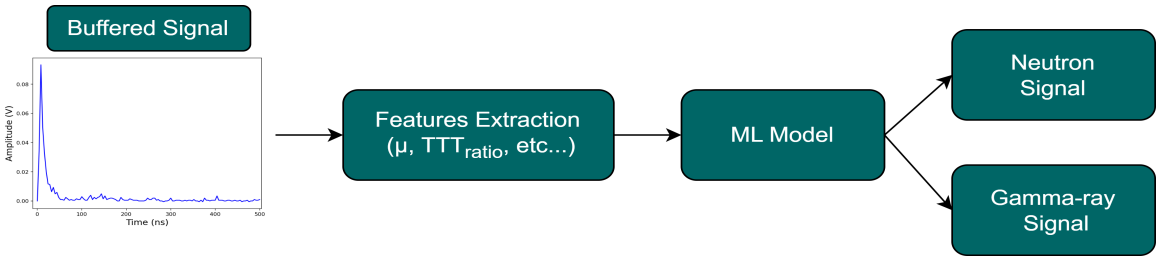


Figure 3.3: An illustration of a ML model for neutron/gamma-ray discrimination, taking as input features extracted from a raw signal.

The structure of this chapter is organized as follow. First, MLP and 1D CNN models are explained in details in sections 3.2 and 3.3. Section 3.4 presents the preparation of labeled raw signals, which are used to train and evaluate the proposed ML models. The energy calibration of these prepared datasets is presented in section 3.5. The results obtained by the trained ML models are compared to TTT_{ratio} according to the variation of sampling frequency and radiation energy in section 3.6. Section 3.7 presents the second approach, which is based on features extraction from the signals to achieve the discrimination. The results of this ML approach reveal three features, including the TTT_{ratio}, that can be used for an effective discrimination. Section 3.8 explores the consistent utilization of the other two attributes across datasets obtained from various acquisition chains, aiming to assess their suitability for discriminating between neutron and gamma-ray signals. During this study, Form Factor attribute is explored as a novel discrimination method that exhibit similar performance compared to TTT_{ratio} without the need for tuning parameters.

3.2 Multi-Layer Perceptron Model

MLP neural network is a fully connected network that tries to find the relation between predefined output and input. Regarding our application field and considering the neutron/gamma-ray discrimination problem, the input is the raw signal obtained by a scintillation measurement chain, as shown in Figure 3.2. The output is one (neutron) or zero (gamma). The model tries to find the function relating these binary output and the points representing each pulse.

Roughly speaking, MLP models are used to tackle problems with high-dimensional data. The input dimension typically ranges from several tens to several thousands. MLP models are especially valuable for solving problems that are challenging or cannot be addressed using traditional statistical approaches. With the availability of more data, ML models tend to provide better results [65, 66]. However, it is important to note that this is not always the case. There may be scenarios where the performance of ML model reaches a certain point and cannot be more improved even if a large dataset exists [67, 68].

MLP models consist of artificial neurons, where each neuron is connected to one or multiple inputs and generates a single output. These inputs can originate from other neurons within the network or from external sources such as images, or neutron/gamma-ray signal as in the presented case of this work. The neuron's output is then transmitted to various other neurons, except for the outputs of the final neurons that fulfill the specific task at hand, such as signal classification or object detection in an image.

To calculate the output of a neuron, several steps are involved. First, each input is multiplied by the corresponding weight connecting the neuron to the input source. Then, the weighted inputs are summed together. Subsequently, a bias term is added to the sum. Finally, the resulting value is passed through an activation function, as illustrated in Figure 3.4. This activation function introduces nonlinearity into the neural network, enabling it to capture complex relationships between the input data and the output. One commonly used activation function is the Rectified Linear Unit (ReLu), depicted in Figure 3.5, which has gained significant popularity in the field of ML [69].

In a MLP model, neurons are commonly arranged in layers, where each neuron in a layer is solely connected to neurons in the immediately preceding and succeeding layers. The initial layer, known

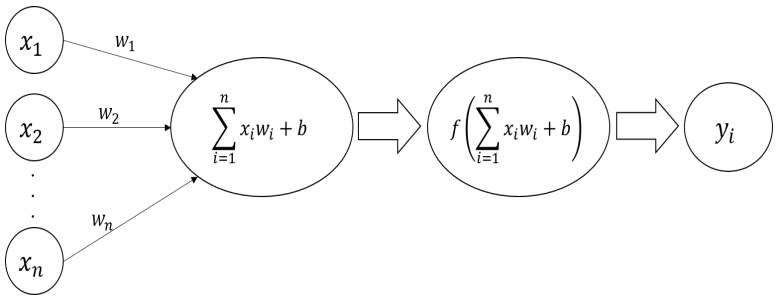


Figure 3.4: Illustration of computation steps inside a single artificial neuron. The first step is calculation the the sum of multiplications between the neuron weights and the input. Then the bias term is added to the result. The last step is calculating the neuron output through the activation function.

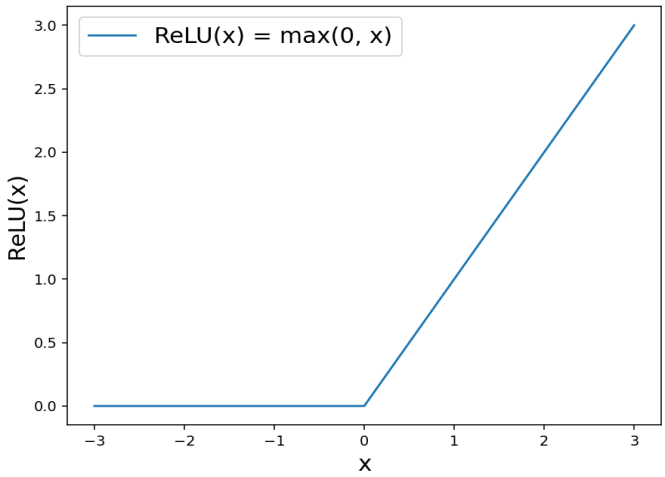


Figure 3.5: ReLU activation function used as an activation function in the hidden layers of a ML model.

as the input layer, receives the input data. The final or the output layer, generates the output that corresponds to the desired task. The choice of activation function in this layer depends on the specific task at hand. For instance, the *softmax* function is typically employed for multi-class classification problems [70], while the *sigmoid* function is commonly used for binary classification problems [71]. The layers positioned between the input and output layers are known as hidden layers.

A MLP consisting of three layers (input, hidden, and output) can be considered as a basic MLP model [39]. However, when an MLP includes more than one hidden layer, it becomes a deep learning model, as shown in Figure 3.6. The connections between two consecutive layers can either be fully connected, where each neuron in the first layer is linked to every neuron in the second layer, or non-fully connected, with some connections being omitted. The extent of dropped connections is regulated by a tuning parameter known as drop-out, which determines the percentage of connections that are dropped. Additionally, connections can be reduced, if a group of neurons in the first layer is connected to a single neuron in the second layer, thereby reducing the number of neurons in the second layer.

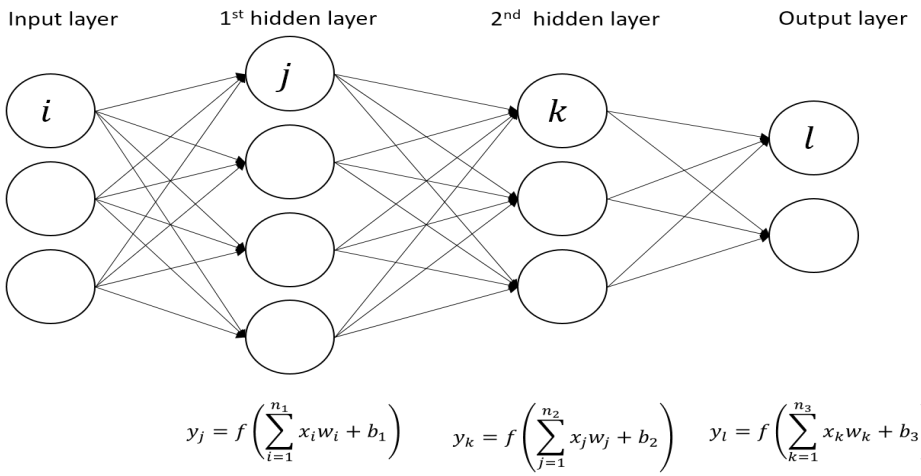


Figure 3.6: An illustration of two hidden layers MLP model. The numbers of neurons in the input, first hidden, second hidden and output layers are 3, 4, 3 and 2, respectively.

The performance of a ML model is assessed by computing the difference between the predicted and ground truth outputs. This difference, which depends on the weights of each layer in the ML model layer is called loss function. Improving the performance level requires to calculate the weights of each layer giving the minimum of the loss function. The type of the loss function depends on the targeted problem. For instance, binary-cross-entropy and categorical-cross-entropy are used for binary and multi-class classification problems, respectively. Mean-squared-error is commonly used for regression problem. Moreover, these loss functions depend on layer's activation functions where some of them are non convex and non linear, such as *sigmoid* and *softmax*.

Roughly speaking, the obtained loss function of a ML model depends on the non convex activation functions and the input/output of all the layers. In other words, we encounter a challenging optimization problem that involves minimizing a non-convex function in a high-dimensional space, where traditional methods fail to provide a solution. To address this, various approaches have been developed, including gradient descent [72], stochastic gradient descent [73], and Adam [74] algorithms. Each approach employs a distinct strategy to find the minimum and incorporates its own set of tuning parameters. These algorithms are iterative by design, meaning that they update the weight values at each iteration to converge towards the minimum. The update process depends on how the method handles the optimization problem, the values of its tuning parameters, and the outputs generated by the machine learning model for the training dataset in the previous iteration. The training process stops when the change in the loss function falls below a given threshold for a specified number of iterations (referred to as waiting iterations). If the achieved performance is deemed satisfactory, the obtained weight values can be used to predict the output for new instances.

There is another strategy used in the learning process that goes through these steps:

1. The dataset is separated into n subdatasets of equal size.

2. In each iteration, one of these subdatasets is used to update the weights for the subsequent iteration. Rather than using the full training data to predict the output and update the weights

sequentially, a subdataset is employed.

3. Once n iterations have been completed, the entire training dataset is used to update the weights for the next iteration.

4. If the variation in the loss functions has not exceeded a certain threshold for a certain number of iterations, the training stops. Otherwise, the process returns to step 2.

An epoch is defined by the steps 2, 3, and 4. The number of epoch required to converge the training depends on several factors such as the task, the dimension of input data and the size of the model. It is usually tuned during the training to achieve the desired performances. In popular ML frameworks such as keras or PyTorch, the programmer specifies the size of the subset data instead of the exact number. This parameter is referred to as the *batch-size*. If the training data has a size of m and the user selects p as the batch size, the number of subdatasets will be m/p. If m is not evenly divisible by p, the last subdataset will have the remaining data samples from the division as its size.

The tuning parameters of the chosen optimization algorithm, the loss function threshold, the waiting iterations and batch-size should be tuned by the user to optimize the training of the proposed model.

3.3 1D Convolution Neural Network

A 1D CNN is a type of neural network commonly used for processing and analyzing sequential data, such as time series data and audio signals. 1D CNNs operate on 1D or 2D input vector, where each row represents a time sequential signal, and they are therefore able to capture patterns in the sequential data more effectively.

The fundamental layer of a 1D CNN model is the One Dimension Convolution Layer (1D CL). The 1D CL applies a set of filters (also called kernels) to the input data, sliding them along the input sequence one position at a time, as illustrated in Figure 3.7. At each position, the filter computes a dot product between its weights and a segment of the input data, producing a single output value. By sliding the filter along the input sequence, the convolution layer produces a sequence of output values capturing the local patterns present in the input data.

A 1D CL has N filters, each of size K, and takes as input a tensor of shape *(C, M)*, where M is the number of channels, and C is the sequence length. The layer produces an output tensor of shape *(N, (C - K)/S + 1)*. Each row (channel) of this output tensor corresponds to the result of the convolution operation between the input of the layer and one filter. S is the stride length, which controls how many positions the filter is moved at each step. Furthermore, a CL has padding hyperparameter, which adds extra zeros around the edges to obtain an output tensor having the same number of columns than the input. After the convolution layer, an activation function is applied to the feature maps as in MLP model. This activation function introduces non-linearity into the network and helps to capture complex relationships between the input data and the output.

A 1D CNN model may contain a pooling layer, which is applied directly on the activation output. Pooling layers, such as maximum and average pooling, downsample the feature maps by summarizing

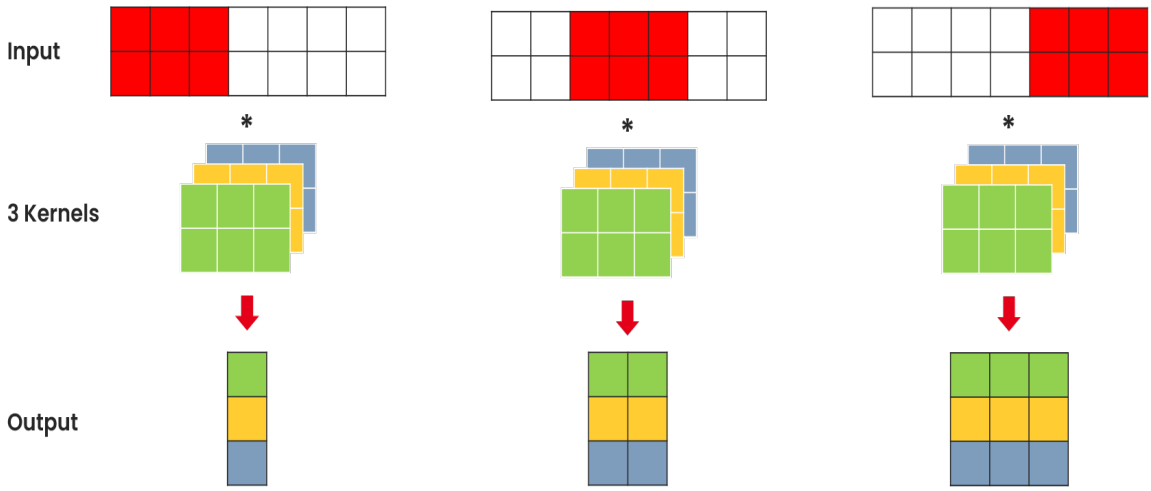


Figure 3.7: Illustration of *1D* convolution operation. The number filter, the filter width, stride, number of input channels and input length are equal to 3, 3, 2, 2 and 7, respectively.

each neighborhood of values into a single value. This reduces the size of the feature maps and helps to reduce overfitting.

After the convolution layers, there is the flatten step, where the feature maps are flattened into a one-dimensional vector. This vector is then fed into one or more fully connected layers described in section 3.2, where the dimension of the output layer and its activation function depend on the target task. The training process follows the same steps than those involved in a MLP model.

In this study, the first ML approach explored to perform the neutron/gamma-ray discrimination task is based on supervised MLP and 1D CNN models, taking as input raw signal, as illustrated in Figure 3.1. The two models are trained, evaluated and compared to TTT_{ratio} discrimination algorithm according to the variation of sampling frequency and radiation energy. Sections 3.4 and 3.5 explain the preparation and the energy calibration of the datasets used for the training and the comparison, respectively.

3.4 Data Preparation

Results obtained in section 2.1 shows that the minimum sampling rate needed to perform the discrimination between neutrons and gamma-rays by TTT_{ratio} algorithm with the implemented acquisition chain exposed in Figure 2.1, without extending the pulse duration with the Op-Amp is 250 MHz. Therefore, two datasets are created using this acquisition chain at sampling rates of 125 MHz and 250 MHz. Consequently, we can examine whether the minimum sampling rate required to discriminate raw signals with the EJ276 scintillator can be reduced by employing ML tools instead of the TTT_{ratio} method. This reduction helps to obtain an embedded discrimination system with less power consumption and complexity. Further, this makes the implementation of ML model on edge device for online application more feasible.

It should be noted that the maximum frequency component obtained by the FFT of the recorded signals in section 2.1.2 is 100 MHz. Moreover, the average rise time, the first decay time component of the EJ276 plastic scintillator and the duration needed for the discrimination are 6, 13 and 500 ns, respectively. Therefore, preparing a dataset at frequencies lower than 125 MHz results in a loss of signal shape and amplitude, both these features being crucial to carry out the discrimination. The comparison between ML methods and TTT_{ratio}, according to variation in sampling rate, cannot be achieved for a lower sampling rate.

The raw signals at 250 MHz are first prepared. Neutron samples are obtained using a ^{252}Cf source. They are acquired, cleaned up and identified by our labeling method proposed in section 2.2. The gamma-ray dataset is directly acquired using a ^{60}Co. The t_{short} and t_{long} parameters are tuned, with the resulting optimum values being 20 ns and 500 ns, respectively. These settings are used when applying the labeling strategy and to obtain the results of section 3.6. The triggering threshold, the voltage peak to peak, and the acquisition window are all set to 15 mV, 800 mV and 1 μs, respectively. The length of each signal is 500 ns from its starting point (10% of the maximum), which makes it possible to contain the longest pulses in the observed dynamics.

Concerning the preparation of labeled datasets at 125 MHz, the proposed labeling strategy in section 2.2 cannot be used. The latter depends on TTT_{ratio} algorithm, which with the present acquisition chain is unable to differentiate the signals at this frequency level, as demonstrated in section 2.1.6. One solution could be to use the dataset prepared at 250 MHz, then downsampling the recorded signals. Downsampling consists in reducing the sampling rate (f_s) of a signal. The rate reduction by a factor N can be done in two steps [75]. First, the high frequency components of the signals are reduced with a low pass filter to avoid potential aliasing. The cutoff frequency of the filter is equal to $f_s/(2 * N)$. The second step consists in decimating the filtered signal by N. In other words, keep only every N^{th} sample. In the presented case, the neutron dataset is prepared at 125 MHz using a Butterworth digital Finite Impulse Response filter [76]. The window of the implemented filter is *Kaiser* [77] with β and length equal to 6 and $200 * f_s$, respectively. This filter's frequency response in the pass band ([0 - 62.5 MHz]) needed to produce the downsampled dataset is flat, as shown in Figure 3.8.

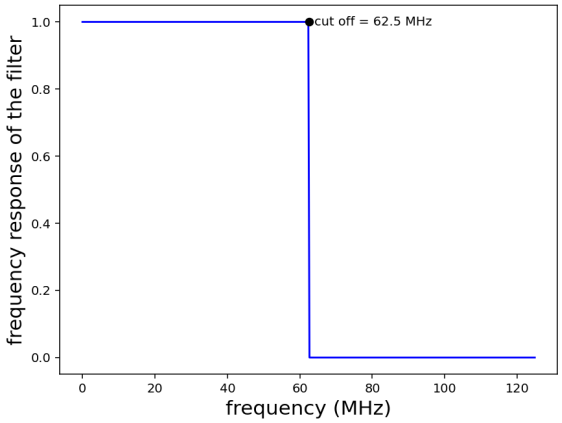


Figure 3.8: Frequency response of Butterworth digital Finite Impulse Response filter. The window is *Kaiser* with β equal to 6 and length equal to $200 * f_s$.

It should be noted that downsampled signals, though realistic, are still somewhat different to signals acquired directly at 125 MHz sampling rate in which some of the information required to perform the discrimination might be lost, while these information in the downsampled signal may be preserved. Nevertheless, obtaining a pure neutron dataset at this relatively low sampling frequency is not possible for us at the time this report was written and using downsampled version is a reasonable approach to develop our method. Next section explain the energy calibration steps of the acquisition chain implemented to acquire the prepared labeled datasets.

3.5 Energy Calibration

Pulse distribution according to the total integral charge (Q_{total}) is used to calibrate the energy of the implemented acquisition chain (examples of ^{137}Cs and ^{22}Na are shown in Figure 3.9). This calibration process is carried out according to the procedure proposed in [78]. A comparison is made by a colleague of the team in the laboratory between the measured and simulated spectrum, taking into account the degradation of energy resolution represented by the Gaussian Energy Broadening (GEB) function. The simulation is performed using the Monte Carlo N-Particle code (MCNP6.2 [79]), where the values of parameters *a, b,* and *c* are 0.02, 0.1, and -0.2953, respectively. Thereafter, the obtained Compton maximum energies of ^{137}Cs, and ^{22}Na, and the photopeaks of ^{241}Am are used, see Table 3.1. A linear energy response function is then applied for the calibration. Results presented in section 3.6 are obtained for an energy radiation higher than 100 keVee, as shown in Figure 3.10.

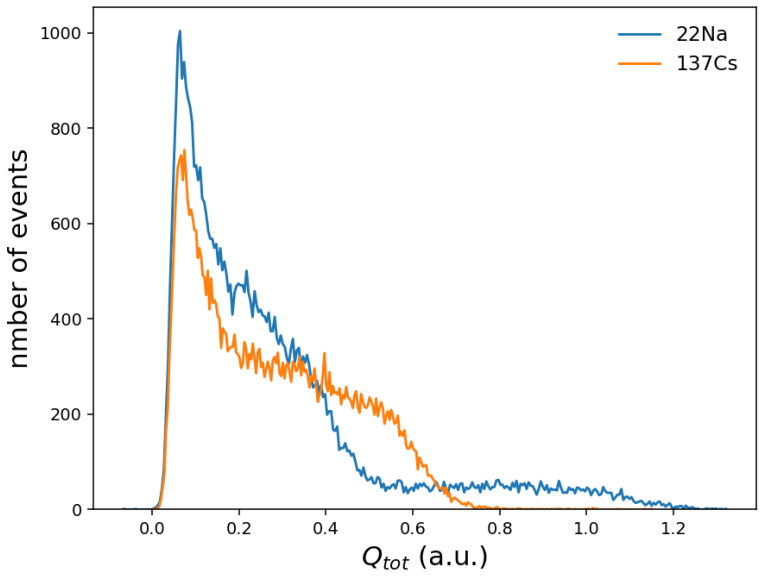


Figure 3.9: Examples of spectra obtained from ^{137}Cs and ^{22}Na sources with an EJ276 plastic scintillator and a PMT.

Thanks to this energy calibration, the comparison between the trained ML models and the TTT_{ratio} algorithm in next section can be performed according to the energy variation to evaluate its impact on the discrimination performance.

Table 3.1: The corresponding Q_{tot} for different energy levels based on the result obtained by the simulation for the energy calibration.

Source	Energy (keVee)	Q_{tot} (a.u.)
^{22}Na	1020	1.02
^{137}Cs	456	0.5
^{22}Na	323	0.24
^{241}Am	60	0.052

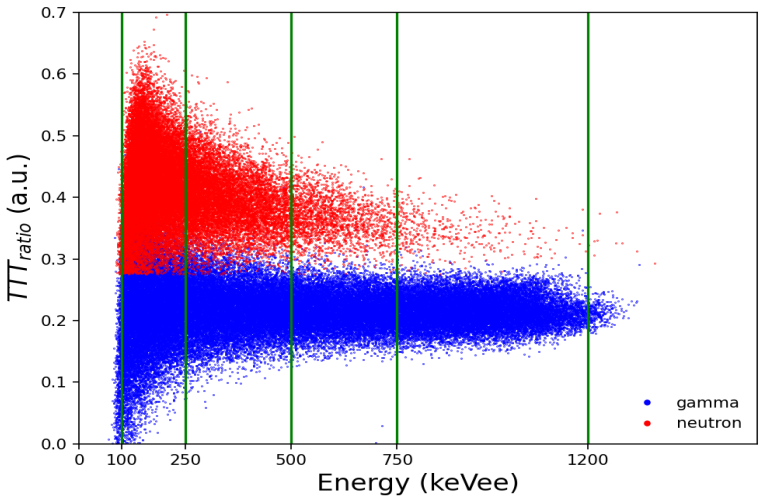


Figure 3.10: TTT_{ratio} according to the total energy integral obtained from the labeled datasets at 250 MHz. The green lines correspond to the limits of the energy range [100 keVee, 1.2 MeVee].

3.6 Experimental Results

The number of neutron and gamma-ray signals prepared at 250 MHz are 40600 and 66800, respectively. The percentage of neutron in dataset is approximately 38%. This imbalance in the data can be addressed by integrating an average weight in the loss function and the evaluation metric during training. The datasets are separated into 80% for the training and 20 % for the validation. The optimizer algorithm is Adam [74], and the loss function is binary cross-entropy. Adam is preferred to other optimization algorithms such as stochastic gradient descent [73] because it has a faster computation time and requires fewer parameters for tuning. ReLu and Sigmoid are respectively the activation functions of the hidden and last layers. The keras framework and Scikit-learn package are used for the implementation.

Using the same amount of resource, the inference time of a ML model is proportional to its size. This time is critical for real time applications where the model will be implemented on edge device, as explained in section 1.1. It can be reduced by optimizing the model size. In this work, the model size of each implemented ML model is first fine-tuned to maximize accuracy and minimize loss on the validation dataset. Thereafter, the model size is tuned to minimize it while preserving the performance previously obtained. The model parameters that need to be tuned to optimize the discrimination performance and

then reduce the size depend on the type of model.

3.6.1 MLP vs. TTT_{ratio} Discrimination Algorithm

The tuning parameters of a MLP model are the number of layers and the number of neurons in each layer. After the two adjustment steps, the obtained MLP model is one with two hidden layers of 32 neurons each. The input layer has n neurons, which is the number of points encoding a signal. The output layer is one neuron representing the probability that a signal will be a neutron or gamma-ray. In general, the classification threshold is 0.5. If the result is greater than 0.5, the radiation is considered as a neutron, otherwise as a gamma-ray. However, the user can tune this threshold to meet the performance constraints of the target application using the ROC curve, as explained in section 1.4.2.

3.6.1.1 MLP vs TTT_{ratio}: According to Incident Energy Variation

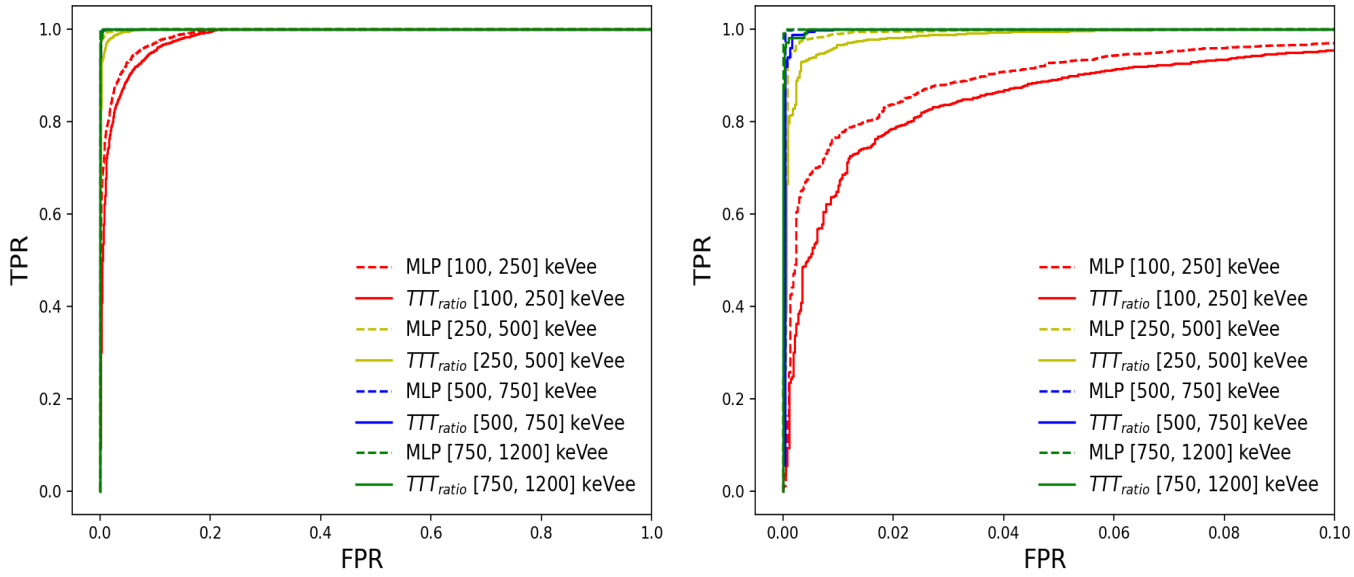
The obtained MLP model is trained and evaluated using datasets acquired and prepared at 250 MHz. The classification report of the validation dataset is shown in Table 3.2. The obtained TPR is 96%. Accordingly, 40 samples out of every 1000 neutrons are classified as gamma-rays. Furthermore, the model raises 20 false alarms for every 1000 gamma signals it classifies. In other word, the FPR is equal to 2%.

Table 3.2: Classification report of the validation data acquired at sampling rate equal to 250 MHz. Accuracy is equal to 98%.

Class	Precision	Recall	F1-score
Gamma-ray	98%	98%	98%
Neutron	96%	96%	96%

Figure 3.11 displays the ROC curves obtained by the trained MLP model and TTT_{ratio} algorithm at 250 MHz for various radiation energy ranges. The Q_{tot} for each energy level obtained according to the results of the energy calibration is shown in Table 3.3. It is clear that the discrimination performance of both approaches is degraded with the decrease of the energy levels. The trained model nonetheless maintains its superiority, especially with low energy radiations. This is seen from the sharp contrast between the ROC curves for the [100 keVee, 250 keVee] energy range. However, the discrimination performance of both methods are approximately the same for energies greater than 500 keVee, as shown in Figure 3.11.

It is important to note that the discrimination threshold that provides a certain level of FPR for the entire dataset is different from the thresholds that provide the same level of FPR for different subsets of the data. Thus, for the same level of FPR, the average TPR is not equal to the TPR of the entire dataset. The latter is equal to the former if the same discrimination threshold is used.



(a) ROC curve

(b) Zoom in ROC curve

Figure 3.11: ROC curves obtained by MLP model and TTT_{ratio} discrimination algorithm on validation data, for different energy ranges, $f_s = 250$ MHz. The ROC curves for the energy ranges [500 keVee, 750 keVee] and [750 keVee, 1.2 MeVee] are superimposed.

Table 3.3: The corresponding Q_{tot} for different energy levels based on the result obtained by the energy calibration. Column three represents the percentage of each energy range in the validation dataset ($f_s = 250$ MHz).

Energy (keVee)	Q_{tot} (a.u.)	Percentage
$[100, 250]$	$[0.06, 0.22]$	40%
$[250, 500]$	$[0.22, 0.49]$	26%
$[500, 750]$	$[0.49, 0.75]$	14%
$[750, 1200]$	$[0.75, 1.3]$	20%

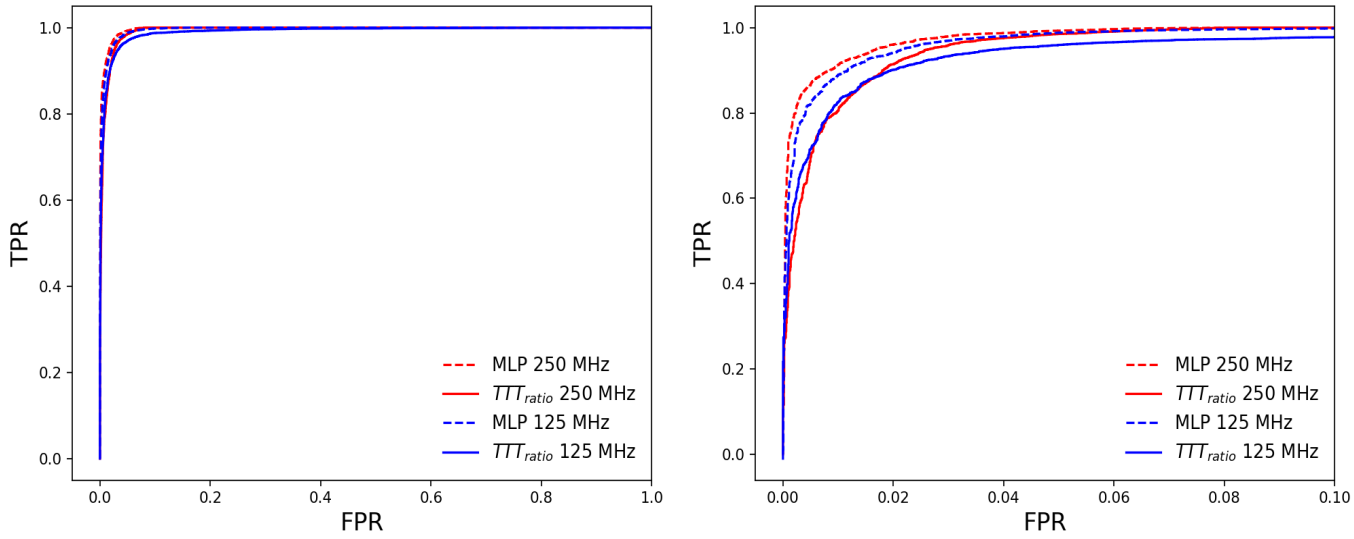
3.6.1.2 MLP vs. TTT_{ratio}: According to Sampling Frequency

Thereafter, training dataset at 125 MHz sampling rate is prepared. Then, testing dataset to assess the trained model is created. This dataset consists of downsampled neutron samples from the validation data and 15,000 signals acquired directly at 125 MHz sampling rate using the gamma-ray source ^{60}Co.

The same MLP model is trained on the prepared training dataset. The obtained results on the testing data following the training are summarized in Table 3.4. The results indicate that the model performance is proportional to the sampling frequency. Moreover, Figure 3.12 illustrates how the model outperforms the TTT_{ratio} method at 125 MHz and 250 MHz. For instance, at 125 MHz, the TPR of MLP model is 94% when the FPR is equal to 2%. The TTT_{ratio}, in comparison, provides TPRs of 90% and 91%, respectively, for the same FPR level, at both sampling rates, see Table 3.4.

Table 3.4: Obtained accuracy, TPR and FPR with MLP on validation (testing) dataset at 250 MHz (125 MHz).

		FPR = 2%	
f_s	Accuracy	TPR	TPR for TTT_{ratio}
250 MHz	97%	96%	91%
125 MHz	96%	94%	90%



(a) ROC curve

(b) Zoom in ROC curve

Figure 3.12: ROC curves obtained by MLP model and TTT_{ratio} discrimination algorithm on testing data at different sampling rates. The figure on the right is a zoom based on the ROC curve on the left.

In summary, these results show that the supervised MLP model trained on raw signals can be a potential solution to improve the performance of neutron/gamma-ray discrimination in plastic scintillator compared to the state of the art (TTT_{ratio} algorithm), especially for low energy radiations. Furthermore, it can help to reduce the complexity of an online discrimination system by reducing its sampling frequency. However, the main advantage of TTT_{ratio} method is its simplicity of computation compared to MLP models, which can make it a better option for embedded implementation.

3.6.2 MLP vs 1D CNN

In the previous section, the proposed MLP model outperforms the TTT_{ratio} algorithm in achieving neutron/gamma-ray discrimination using the EJ276 plastic scintillator. This section assesses the benefit of using 1D CNN in order to improve the discrimination performance. The parameters of the 1D CNN model that need to be adjusted to first optimize discrimination performance and then reduce model size are the number of layers, the number of filters, and their size within each layer. The resulting model from the two tuning steps is one with one hidden CL. The obtained number of filters is 4. The filter

width and stride are equal to 3 and 2, respectively. The input and output layers are the same compared to the MLP model. The model is implemented and trained using keras framework. The optimizer algorithm is Adam algorithm.

The ROC curves shown in Figure 3.13 indicate that the 1D CNN model outperforms the MLP model, mainly for low energy radiation ([100, 250] keVee). In this range, for FPR equal to 2%, the TPRs achieved by the two models are 88% and 84%, respectively, as indicated in Table 3.5. For energy levels exceeding 250 keVee, the ROC curves of both models overlap. These results validate the previous conclusion, indicating that the discrimination between neutrons and gamma-rays in the presented case is challenging when dealing with low energy radiations ([100, 250] keVee]).

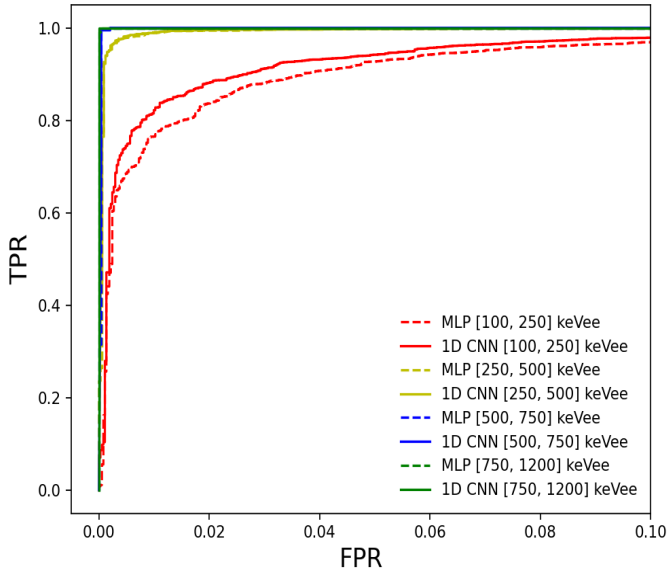


Figure 3.13: ROC curves obtained by MLP and 1D CNN models on validation dataset at 250 MHz.

Table 3.5: Obtained TPR for FPR = 2% with MLP and 1D CNN models on validation dataset at 250 MHz.

Energy range	MLP	1D CNN
[100, 250] keVee	84%	88%
[250, 500] keVee	99.42%	99.56%
[500, 750] keVee	100%	100%
[750, 1200] keVee	100%	100%

While the trained 1D CNN model outperforms the TTT_{ratio} algorithm in the presented case, a key limitation of this supervised ML model is its sensitivity to the distribution function of the trained dataset. As explained in section 3.1, there is a need to repeat the data preparation and model training for even subtle changes in the measurement chain. This process can be both challenging and time-consuming. Adapting the proposed 1D CNN model to an unlabeled dataset acquired by a different

measurement chain can be a solution to overcome this data preparation problem. However, the input dimension of this model depends on the characteristic of the acquisition setup, which can be a limitation in achieving the adaptation. Features extraction from the signal can be a potential solution to overcome this limitation, as detailed in section 3.1. The extracted attributes from raw signals can be used to train an ML model instead of using the entire raw signal. The next section provides a comprehensive explanation of the steps involved in this feature extraction approach.

3.7 Features Extraction

Supervised model adaptation involves first the training of a model on a large dataset (source dataset). Then, the trained model is adapted on a new smaller dataset (target data), having different distribution obtained from different sources. On the other hand, unsupervised model adaptation involves adapting a trained model to a new dataset (target dataset) without using labeled data. This characteristic can be of great interest when obtaining labeled data is a challenging task, as is the case for the neutron dataset in this work. Adapting a model trained on source data to perform the same task on target data is challenged by several factors, such as the different distributions of both datasets and the different dimension size that represent a signal in each dataset. An instance of unsupervised model adaptation problem is adapting the previous trained MLP or 1D CNN model to classify neutron/gamma-ray signals acquired by a different acquisition chain. In this presented case, both challenges mentioned previously are presented. Features extraction process may be a solution to overpass the difference in signal dimensions. It can reduce the dimension of the data to a manageable level independent from the acquisition chain and the length of the raw signal.

Features extraction is the process of extracting a set of relevant features from data, such as time sequence signals acquired from a sensor. These features can be more easily analyzed and processed to extract meaningful information from the data. In other words, features extraction can provide a compact and informative representation of a signal, which enables the comparison of signals with different lengths that can be useful for model adaptation. For instance, features extraction could involve identifying and extracting statistical criteria such as mean, variance, skewness, and kurtosis. These statistical criteria are simple and yet informative features that can describe the signal's shape, average deviation, and spread.

The features that can be extracted from the signals and are useful for the target application can be defined by the user or through the using of self-supervised ML tools. Self-supervised learning has emerged as a promising approach in the field of ML, offering a novel paradigm for learning useful representations that capture important aspects of the data distribution. The fundamental idea behind self-supervised learning is to design pretext tasks that guide the learning process without human annotation. These pretext tasks involve automatically generating supervisory signals from the data itself, effectively transforming unlabeled data into labeled-like data. By solving these pretext tasks, models can represent the relevant information of the dataset needed for a desired task in a lower dimension. These representations can then be used as input for another ML models to achieve different tasks, such as classification and object detection. In a wide range of applications, the relevant features learned by self-supervised learning has offered the potential to achieve the desired task with higher or similar performance, compared to using the initial dataset directly [80, 81].

While recognizing the potential of self-supervised learning for extracting meaningful data features, our study focuses solely on the definition of some features that can be useful to perform the neutron/gamma-ray discrimination task. The application of self-supervised learning to classify radioactivity signals remains a prospect for future research. Section 3.7.1 defines the various types of features that can be extracted from signals. Then, section 3.7.2 explains the process of training supervised ML model using the derived features and compares the results with those obtained by the trained MLP and 1D CNN models on raw signals in section 3.6. Finally, section 3.7.4 measures and analyses the contribution of each feature to the decision output, which can help to determine the most relevant attributes for discrimination.

3.7.1 Definitions of Extracted Features

The types of features that can be extracted rely on the signal's characteristics and the specific application. These features can be broadly classified into three categories: statistical, spectral, and time-domain features. Spectral features, such as dominant frequency, bandwidth, and harmonic ratios, are primarily derived from the frequency domain. Experimental results showed that the difference between neutrons and gamma-rays in the frequency domain (using FFT) is mainly due to their energy distribution differences. The sole disparity lies in the amplitudes of the frequency components because these amplitudes correspond to the energy of the signal in the time domain.

As presented in section 1.1, the main difference between neutron and gamma-ray signal in an organic scintillator is located in the decay phase. Therefore the main time domain and statistical attributes that may contain useful information for the discrimination (Table 3.6) are:

- Mean: the average value of the signal.
- Variance: the average of the squared differences from the mean.
- Standard Deviation: a measure of the dispersion or variability of the signal.
- Skewness: a measure of the asymmetry of the signal's distribution.
- Kurtosis: a measure of the peakedness or flatness of the signal's distribution.
- Crest Factor: the ratio of the peak value to the Root Mean Square (RMS).
- Pulse Indicator: the ratio of the peak to the average value.
- Form Factor: the ratio of the RMS value to the average value.
- TTT_{ratio}: the ratio of the tail to the total integral of the signal.

These features can represent a neutron or a gamma-ray signal in a lower dimension and it may contain the information required for the discrimination. This can be verified by using these attributes instead the raw signals to train a ML model for the discrimination task. Decision tree algorithms can be more powerful tools than MLP or 1D CNN when the purpose consists in discriminating signals using extracted features instead of raw signals [82–85]. RF [38] is one of main decision tree algorithms

Table 3.6: Equations of extracted features.

Attribute	**Equation**
Mean	$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$
Variance	$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$
Standard Deviation	$\sigma = \sqrt{\sigma^2}$
Skewness	$\gamma = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^3}{\sigma^3}$
Kurtosis	$\kappa = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^4}{\sigma^4}$
Crest Factor	$\text{Crest Factor} = \frac{\max(\{x_i\})}{\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}}$
Pulse Indicator	$\text{Pulse Indicator} = \frac{\max(\{x_i\})}{\mu}$
Form Factor	$\text{Form Factor} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}}{\mu}$
TTT_{ratio}	$\text{TTT}_{ratio} = Q_{tail}/Q_{tot}$

that have been developed to solve classification problem when the dataset is represented by convenient extracted features [86–90]. It also includes a tool for identifying the most relevant attributes considering the target application. Next section presents a brief introduction of this algorithm and its concept of work.

3.7.2 Random Forest Model for Training on Extracted Features

The RF decision tree algorithm, as described in [38], works by constructing a collection of decision trees and combining their predictions to produce a final output decision. A decision tree is based on a series of simple "if-then" rules, as shown in Figure 3.14. The decision tree classifier in this figure has 4 features. Each decision path is a conjunction of different conditions. These conditions are features and values with relations less than ($<$), greater than ($>$) and equal ($=$). For instance, if a sample has f1 greater than 0.45 and f2 greater than 1, the decision will be 'B'. Roughly speaking, algorithms relying on decision trees can be easily interpreted and understood, thanks to the simple "if-then" rules that guide the output decisions. This characteristic making them better choice for problems where the goal is to identify key features that distinguish different classes of signals. Moreover, decision tree algorithms are less prone to overfitting, which can occur when a model is too complex and performs well on the training data but poorly on new, unseen data. Roughly speaking, decision tree algorithms are a powerful and interpretable tool for discriminating between signals using extracted features, making them a popular choice in many signal processing applications [86–90].

RF decision tree algorithm operates by building a set of decision trees and aggregating their predictions to generate a final prediction, as explained previously. The process starts by randomly selecting

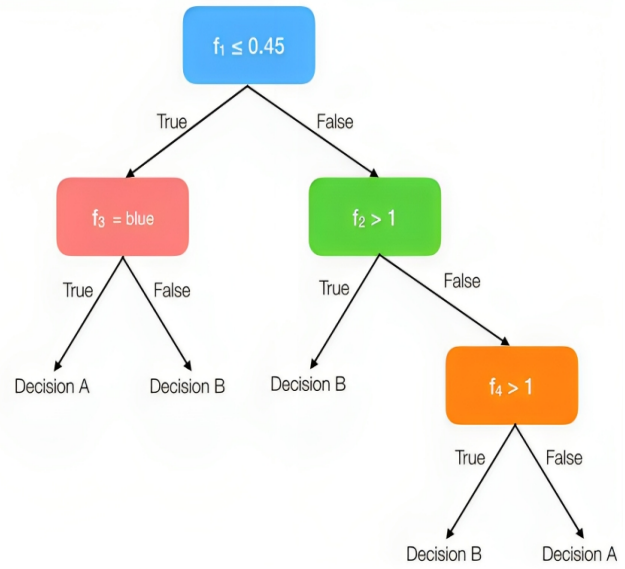


Figure 3.14: A simple decision tree classifier with 4 features [91].

a subset of the training data and a subset of features for each individual tree. Subsequently, for every tree, the algorithm recursively divides the data based on the chosen features, creating a collection of decision rules that forecast the target variable. The selection of the feature and its corresponding value for partitioning in each iteration is determined by minimizing the loss function. In the case of supervised binary classification, cross-entropy serves as the loss metric, quantifying the dissimilarity between the predicted probability and the actual class. Figure 3.15 illustrates a RF algorithm with n decision tree.

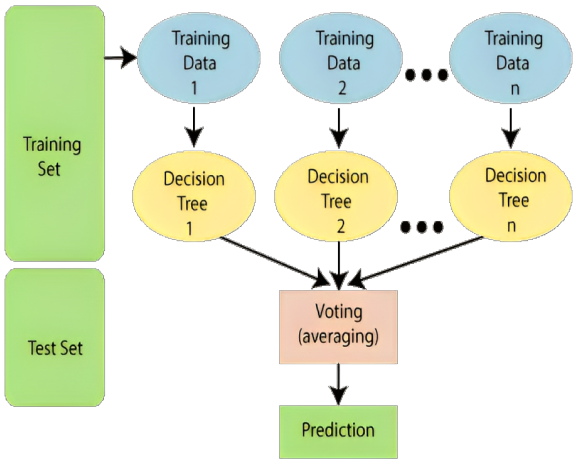


Figure 3.15: RF algorithm with n decision trees [92].

After the training, to make a prediction for a new data point, each tree in the random forest independently makes a prediction based on its decision rules. The final prediction is then determined by combining the predictions of all the trees, typically by taking a majority vote. There are several hyperparameters that should be tuned when using random forest such as the number of trees in the

forest and maximum depth of each tree. Increasing the former can improve the accuracy of the model, but also increases the computational cost and may lead to overfitting. The maximum depth controlls the largest possible length between the root to a leaf which can help prevent overfitting and improve generalization performance.

3.7.3 Performance on raw signals vs. Features Extracted

The features outlined in section 3.7.1 are derived from neutron and gamma-ray signals prepared at a frequency of 250 MHz. These labeled signals were used to train and evaluate the previously developed MLP and 1D CNN models. The results obtained from these models indicate their superior performance compared to the TTT_{ratio} algorithm, especially within the energy radiation range of [100, 250] keVee. Both the ML models and the PSD method showed similar performance for radiation energy levels exceeding 250 keVee. Consequently, we proceed to train an RF model on the complete dataset and assess its performance in comparison to the TTT_{ratio} algorithm, as well as the trained ML models on raw signals (1D CNN and MLP), within this lower energy range. The number of trees and the maximum depth parameters of RF model are tuned empirically to optimize the discrimination performance and avoid the overfitting. The obtained values are 10 and 7, respectively. Figure 3.16 illustrates a significant degradation in terms of discrimination performance when these extracted features are employed instead of the raw signals for discrimination purposes. More precisely, achieving the neutron/gamma-ray discrimination using these attributes provides a performance similar to that of the TTT_{ratio} algorithm, as shown in Table 3.7 and Figure 3.16. While the results obtained through this proposed features extraction approach are not promised, analyzing the defined attributes and assessing the contribution of each to the output decision can help explain the previously obtained results. This is the topic of the next section.

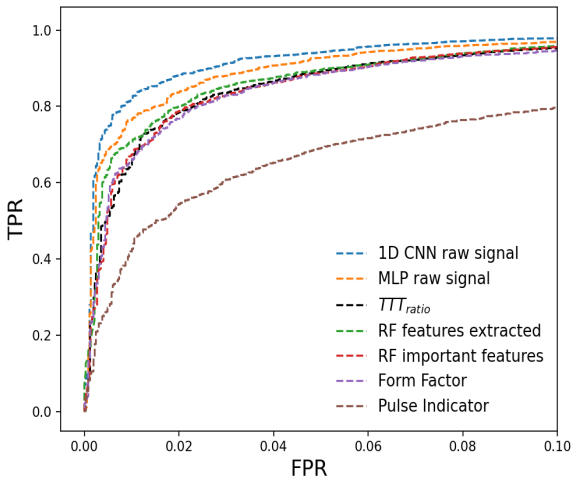


Figure 3.16: ROC curves obtained by RF, MLP and 1D CNN models on validation dataset at 250 MHz and energy range [100, 250] keVee.

Table 3.7: TPR for FPR = 2% obtained by RF, MLP and 1D CNN models on validation dataset at 250 MHz and energy range [100, 250] keVee.

Model	TPR for FPR = 2%
1D CNN	88%
MLP	84%
TTT_{ratio}	78%
RF	80%
RF important features	79%
Form Factor	77%
Pulse Indicator	54%

3.7.4 Features Selection

RF incorporates a feature selection tool capable of identifying the most relevant attributes for a given task. Figure 3.17 presents the importance score of each feature. In the presented case, Form Factor, TTT_{ratio} and Pulse Indicator emerge as the most significant features. These three attributes contain approximately 94% of the essential information required for discrimination. Therefore the training of the RF is repeated using only these three attributes. The ROC curves obtained by this trained model and the model trained using all the features on validation dataset at 250 MHz and energy range [100, 250] keVee are superimposed, as shown in Figure 3.16. This indicates that using these three relevant features for discrimination achieves approximately the same performance as when all features are used.

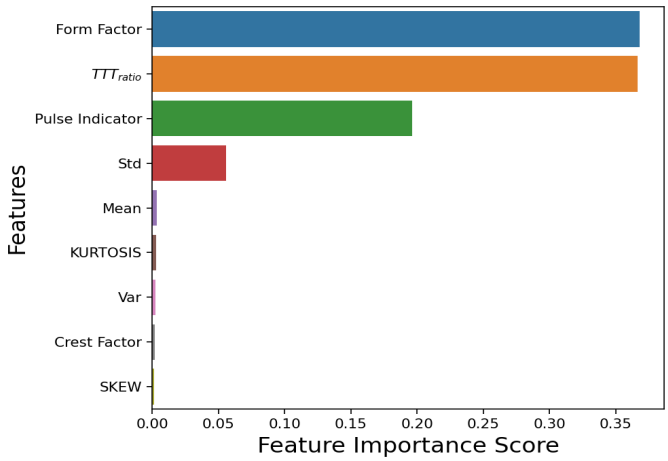
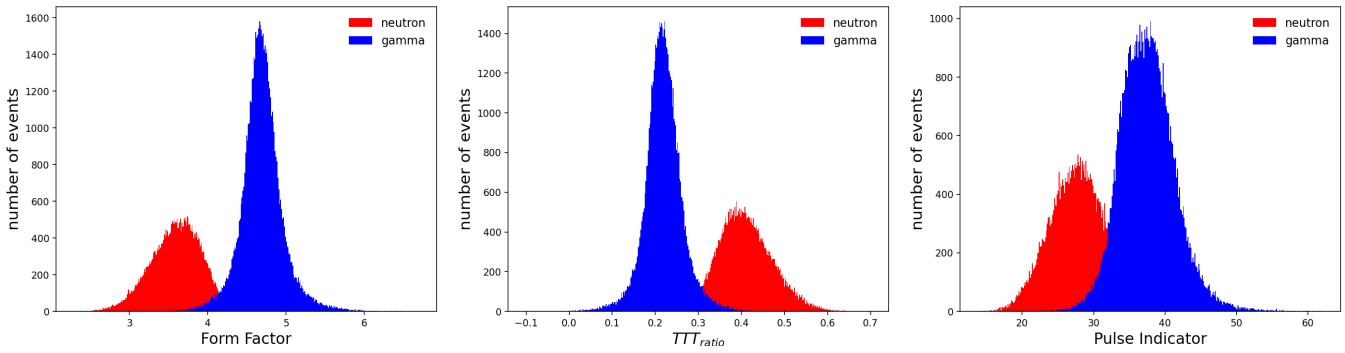


Figure 3.17: The contribution score of each defined feature to the decision output.

Figures 3.18(a) and 3.18(b) show the distributions of Form Factor and TTT_{ratio} attributes. The separation between neutron and gamma-ray classes in both distributions is roughly similar. This indicates a comparable discrimination performance, as supported by the ROC curve presented in Figure 3.16. For a FPR of 2%, both methods yield TPRs of 78% and 77%, respectively. In contrast, Figure 3.18(c) shows that the separation between neutrons and gamma-rays in the distribution of Pulse Indicator is

less pronounced compared to the separation obtained using Form Factor and TTT_{ratio}. The ROC curve using Pulse Indicator for discrimination in Figure 3.16 validates this observation, indicating a significant degradation in performance. For a 2% FPR, the TPR obtained with this attribute is 54%.

The value of the Form Factor attribute is influenced by the shape of the signal, which varies between neutron and gamma-ray signals. Therefore, this attribute holds significant information for discrimination purposes, as illustrated by its distribution in Figure 3.18(a) and validated by the results obtained in Figure 3.16. It is worth to note that there is a similar parameter called Shape Factor, which is computed as the ratio of RMS to the mean of absolute values. It is used in conjunction with other statistical features to assess the operational state of various systems like induction motors, gears, and rotating electrical machines [93, 94].



(a) Form Factor (b) TTT_{ratio} (c) Pulse Indicator

Figure 3.18: Distributions of Form Factor, TTT_{ratio} and Pulse Indicator features extracted from each class of neutron/gamma-ray labeled dataset obtained by using an EJ276 plastic scintillator (sampling frequency at 250 MHz).

The Pulse Indicator feature which is the ratio between the signal amplitude and its mean, provides an information on the agreement between the signal and an impulse response. Due to its faster decay time, a gamma-ray signal exhibits a more similar impulse response when compared to a neutron signal. This distinction appears in the higher values of the pulse indicator obtained for gamma-ray signals, Figure 3.18(c).

The results obtained in this study show that the Form Factor and Pulse Indicator attributes can be useful to perform the discrimination. Furthermore, Form Factor exhibited similar performance compared to the state of the art PSD algorithm (TTT_{ratio}). Next section explores the consistent use of these two attributes on datasets acquired from different acquisition chains to assess their suitability for performing neutron/gamma-ray discrimination.

3.8 Form Factor: A Novel Pulse Shape Discrimination Method

3.8.1 Dataset Acquisition

In order to assess the suitability for performing neutron/gamma-ray discrimination of Form Factor and Pulse Indicator methods four different acquisition chains are employed (Table 3.8).

1. EJ276 plastic scintillator coupled to a 9821B PMT. The sampling frequency and the amplitude resolution of the digital oscilloscope (LeCroy HDO6104A-MS) are respectively 250 MHz and 12 bits. This is the same acquisition chain used to obtain the labeled dataset. In this study, the discrimination performance of each method is directly measured on the obtained raw signals without the labeling step.

2. A plastic scintillator composed of a high concentration of 2, 5-diphenyloxazole and a small quantity of 9, 10-diphenylanthracene in polystyrene. This formulation should be close to that of EJ-276 from Eljen Technology. This scintillator is connected to a R7724 PMT. The sampling frequency and the amplitude resolution of the digitizer (CAEN DT5743) are respectively 800 MHz and 12 bits.

3. The same plastic scintillator of the chain 2 is connected to a SensL ArrayC-30035-16P SiPM. The sampling frequency and the amplitude resolution of the digitizer (CAEN DT5743) are respectively 800 MHz and 14 bits.

4. BC-501A liquid scintillator connected to a R7724 PMT. The sampling frequency and the amplitude resolution of the digitizer (CAEN DT5743) are respectively 800 MHz and 12 bits.

Table 3.8: Main components and parameters of the 4 employed acquisition chains.

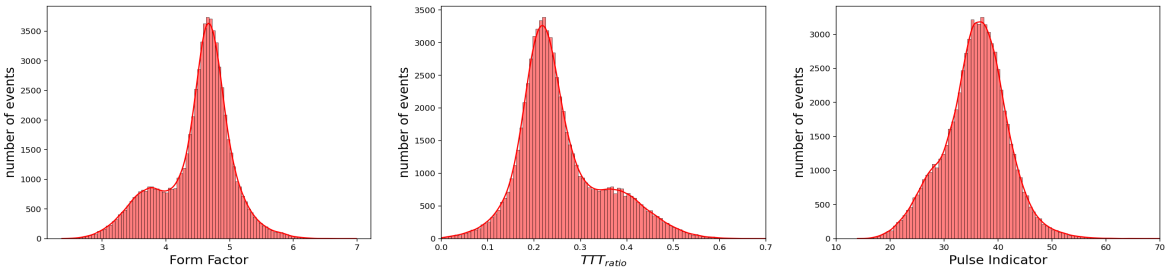
Acquisition chain	Scintillator	Photo sensor	Sampling frequency (MHz)	Amplitude resolution (bits)
1	EJ276 (plastic)	9821B PMT	250	12
2	Plastic (laboratory)	R7724 PMT	800	12
3	Plastic (laboratory)	ARRAYC3003516P SiPM	800	12
4	BC-501A (liquid)	R7724 PMT	800	12

The acquisition of the datasets is achieved using ^{252}Cf source. The values of the t_{long} and t_{short} parameters of TTT_{ratio} algorithm are fine-tuned to optimize the discrimination performance, using the method proposed in [36]. The obtained optimal values for each acquisition chain are shown in Table 3.9. The obtained values of t_{long} are used as the signal length for calculating both the Pulse Indicator and Form Factor. For each acquisition chain, the distributions of each attribute extracted from the dataset are illustrated in Figure 3.19.

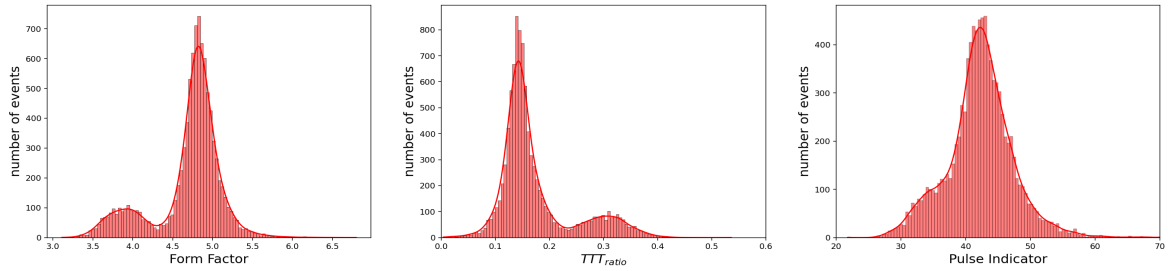
Table 3.9: Optimal values of t_{long} and t_{short} obtained on the datasets acquired by the four employed acquisition chains.

Acquisition chain	t_{long}	t_{short}
1	500	20
2	1050	75
3	1000	120
4	275	25

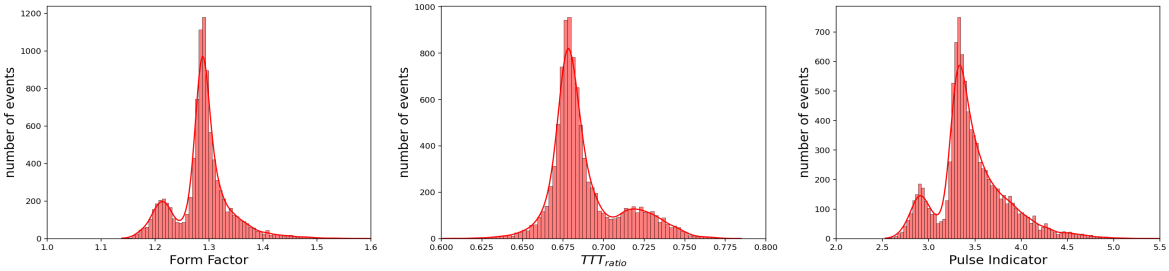
The distributions of the Pulse Indicator reveal that discrimination is not achievable when using a plastic scintillator and PMT during acquisition in our experimental setup, as indicated in Figure 3.19.



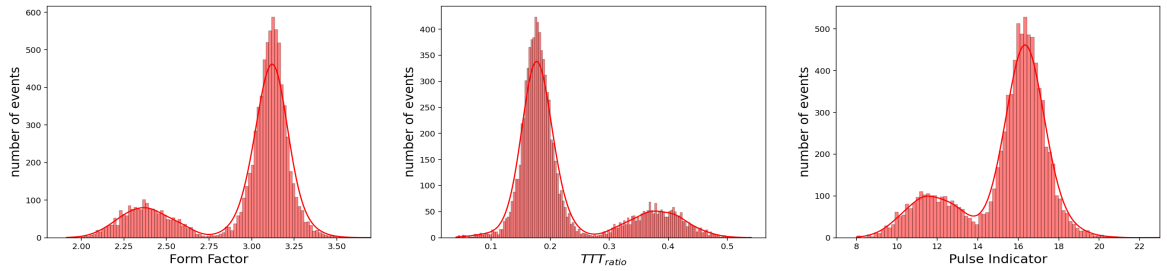
(a) EJ276 plastic - 9821 PMT



(b) Plastic (lab) - R7724 PMT



(c) Plastic (lab) - ArrayC3003516P SiPM



(d) BC501A liquid - R7724 PMT

Figure 3.19: Distributions of Form Factor, TTT_{ratio} and Pulse Indicator features extracted from neutron and gamma-ray dataset obtained using the four employed acquisition chains.

There is a notable overlap between the distributions of neutron and gamma-ray classes. However, when the plastic scintillator is replaced by a liquid one, the distribution clearly indicates the presence of two distinct classes in the dataset. Similar results are observed when the PMT is replaced by SiPM, although the distribution seems to be less Gaussian. Despite the improvement in discrimination capacity, visualizing the distributions alone is sufficient to conclude that the discrimination achieved by the

TTT_{ratio} method is better, which is not the case with the Form Factor attribute, as shown in Figure 3.19.

A quantitative measurement is necessary for a meaningful comparison between the Form Factor and TTT_{ratio} methods. These two features extracted from the signals are more prone to being derived from Gaussian distributions, as shown in Figure 3.19. Consequently, an unsupervised GMM can be an appropriate choice for this quantitative measurement making the assumption that the observations in the dataset are generated from a mixture of Gaussian distributions. Next section provides an introduction about this unsupervised ML model.

3.8.2 Gaussian Mixture Model

GMM works by fitting a specified number of Gaussian distributions to the dataset, with each distribution representing a cluster or component of the data. It assigns a probability to each observation for each component, indicating the likelihood that the observation was generated from this specific component. The GMM algorithm estimates the parameters of the Gaussian distributions, including the means and variances, using an iterative process called the Expectation-Maximization (EM) algorithm [95]. First, the algorithm randomly assigns the parameters for each Gaussian component, including mean, covariance, and mixing coefficients. In the expectation step (E-step), the algorithm calculates the probability or likelihood for each observation belonging to each Gaussian component, using the current estimated parameter. In the maximization step, the algorithm updates the assessment of the Gaussian parameters, such as means and covariances of each cluster, using the probability weights previously obtained. The E-step and M-step are alternated to refine the parameter estimates and improve the model fit. Convergence is typically determined based on a predefined criterion, such as the change in likelihood or the number of iterations.

The number of components in the GMM is a hyperparameter that needs to be specified prior to fitting the model. Selecting the appropriate number of components is important for the performance of the model, as too few components may result in underfitting and too many components may result in overfitting. There are different approaches to select the number of components, such as the Bayesian Information Criterion (BIC) [96] or the Akaike Information Criterion (AIC) [97]. In our case, we have prior knowledge that the number of components is equal to 2, corresponding respectively to the neutron and gamma-ray classes.

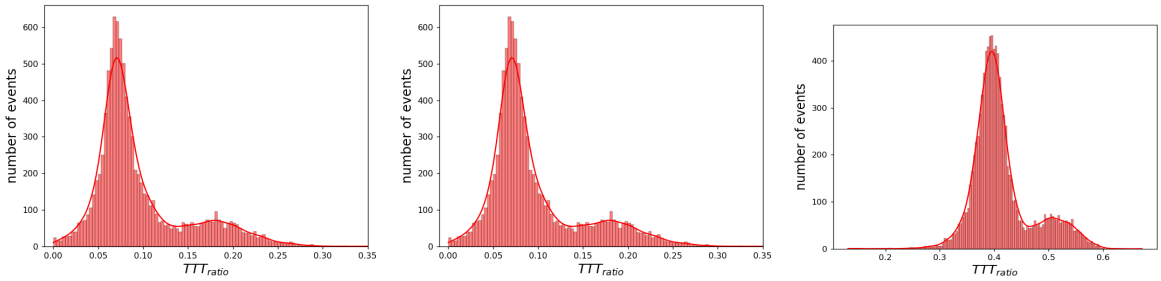
3.8.3 Experimental results

A two-component GMM is fitted to each one of these two attributes extracted from the datasets acquired by the four employed acquisition chains presented in Table 3.8. The performance of the model is evaluated by calculating the FPR on the pure gamma-ray dataset obtained from a ^{60}Co source, and the neutron percentage in the ^{252}Cf source. The latter ensures that the model does not classify all signals as gamma-rays. The obtained results are presented in Table 3.10. As it is expected, the GMM yields comparable discrimination results for both methods. However, the performance of the TTT_{ratio} method is optimized by adjusting its parameters for each acquisition chain. In contrast, the Form Factor attribute does not require any fine-tuning. It solely relies on the signal length, which can be estimated based on the characteristics of the components in the acquisition chain. Moreover, the discrimination performance of the Form Factor method is not significantly affected by variations in the signal length,

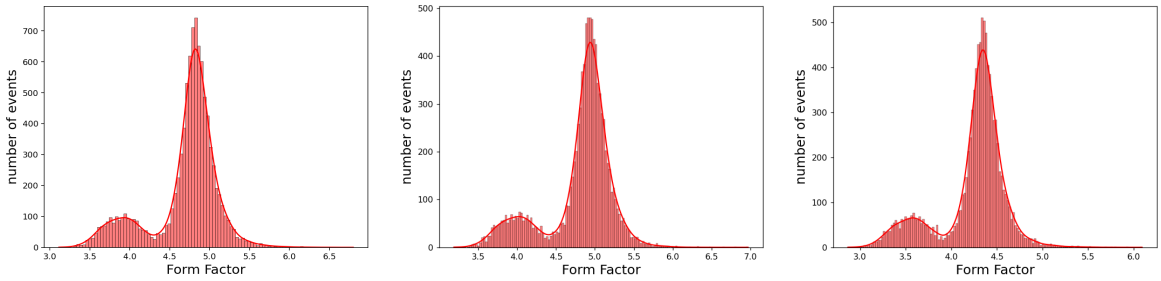
unlike the TTT_{ratio} algorithm, where the values of t_{long} and t_{short} play a crucial role, as shown in Figures 3.20 and 3.21. This consideration is quite important for practical applications.

Table 3.10: The results obtained by GMM fitted on Form Factor and TTT_{ratio} attributes.

	Form Factor		TTT_{ratio}	
Acquisition chain	% of neutron	FPR	% of neutron	FPR
1	20	1.5	19%	1.7%
2	17	0.5	17	0.6
3	20	0	20	0
4	21	0	22	0

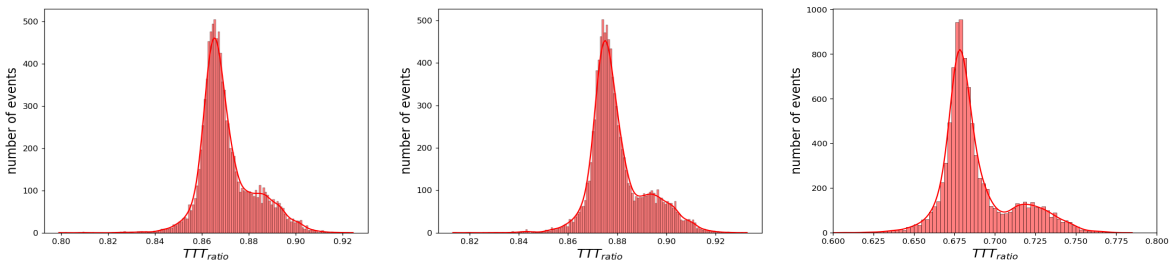


(a) t_{long} (t_{short}) is equal to 1050 (150) ns, 950 (50) ns and 800 (20) ns, respectively.

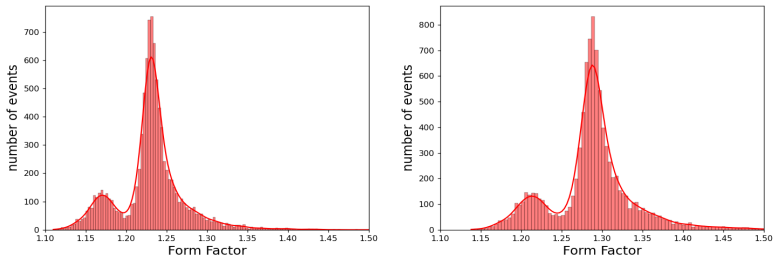


(b) Signal length is equal to 1050 ns, 950 ns and 800 ns, respectively.

Figure 3.20: Distributions of Form Factor, TTT_{ratio} obtained with the second acquisition chain, using different values t_{long}, t_{short} and signal length.



(a) t_{long} (t_{short}) is equal to 800 (50) ns, 1000 (50) ns and 1000 (120) ns, respectively.



(b) Signal length is equal to 800 ns, and 1000 ns, respectively.

Figure 3.21: Distributions of Form Factor, TTT_{ratio} obtained with the third acquisition chain, using different values t_{long}, t_{short} and signal length.

3.9 Conclusion

In this chapter, we compared the discrimination performance of neutron/gamma-ray signals recorded using an EJ276 plastic scintillator between a trained MLP and 1D CNN models and the TTT_{ratio} algorithm. The comparison was achieved according to variations in sampling frequency and radiation energy. The ML models are trained and evaluated on raw signals obtained using the labeling pipeline described in chapter 2. In the presented case, the performance of three methods decreases with the energy of the incident radiation. Nevertheless, the 1D CNN model outperforms the MLP model, which in turn, outperforms the TTT_{ratio} algorithm, especially for low energy radiations ([100, 250] keVee). Furthermore, the trained ML models outperform the TTT_{ratio} discriminating algorithm in terms of its ability to distinguish the radiations at lower sampling rates. This point can help to reduce the architecture size, power and complexity of an online discrimination system.

It is worth noting that the TTT_{ratio} PSD algorithm can provide better discrimination performance within the energy range [100, 250] keVee, using the same scintillator type (EJ276), when incorporating analog signal processing blocks, such as Op-Amp integrator, in the acquisition chain, as shown in the results obtained in Chapter 2. Nevertheless, a supervised 1D CNN model can still achieve superior discrimination for lower energy radiation compared to TTT_{ratio}, as presented in this chapter. The only necessity is to repeat the preparation of the labeled dataset using the newly proposed experimental setup and subsequently train the model. In fact, the sensitivity of the 1D CNN model to the components and parameters of the acquisition is the primary limitation of this ML based approach. Repeating the preparation of clean labeled neutron signals for any subtle change in the acquisition chain is a challenging and time consuming work.

Unsupervised domain adaptation can be a solution to overpass this significant limitation. However, the input dimension of the model mainly depends on the measurement chain, which will be a key challenge to use this approach. Features extraction can be a solution to overpass this challenge. It can reduce the dimension data to a level independently of the acquisition chain and the input signal length. The features that can be extracted from the signals and are useful for the target application can be defined by the user or through the use of self-supervised ML tools. In this chapter we explored the benefit of using the former approach for the neutron/gamma-ray discrimination task and we considered the latter for a future work. Obtained results showed that the supervised models trained on raw signals exhibit better performance than the supervised approach based on the extracted features. Lastly, our investigation of the Form Factor attribute as a discrimination method showcases its effectiveness in capturing the performance of the TTT_{ratio} algorithm without requiring any tuning parameters. This point is crucial for practical applications.

For a future work, a deeper study can be done to assess the effectiveness of unsupervised domain adaptation tools in the context of neutron/gamma-ray discrimination in organic scintillators, which is a critical point since the preparation of labeled dataset is a challenging and consuming time step.

Preserving the outperformance of the trained supervised model and respecting the time constraints are the main keys that should be considered when these models will be implemented on an embedded system to obtain an online discrimination system. Next chapter provides a detailed explanation and comparison of this implementation for the trained MLP and 1D CNN models, as well as the TTT_{ratio} and Form Factor PSD algorithms.

Chapter 4

Pulse Shape Discrimination using embedded Neural Networks: First Steps for Developing a Specific Electronic Architecture

In Chapter 3, in the context of neutron gamma-ray discrimination, using our scintillation setup, we show that the proposed 1D CNN model outperforms the MLP model, which in turn, outperforms the TTT_{ratio} discrimination algorithm. Furthermore, Form Factor PSD method exhibits a similar performance to TTT_{ratio} algorithm. In some applications, the embedded implementation of a proposed discrimination method is a critical point to achieve a classification on the fly. The main objective in this type of application is to get as close as possible to an immediate output decision after the acquisition of each signal while minimizing the percentage of missed radiation detection as explained in section 1.2.5 of chapter 1. Therefore, the comparison between different discrimination approaches should go beyond their performances. Their embedded implementations should also be compared. One of the main challenges to achieve the implementation is to preserve the same discrimination performance obtained offline. Additionally, optimizing the implementation involves balancing execution time and resource consumption.

The short duration of signals produced by a plastic scintillator (in order of several hundreds of nanoseconds) requires an acquisition at a sampling frequency on the order of several hundred Megahertz (MHz). Consequently, the embedded system should have the capability to process the digitized signal at this relatively high sampling rate without loss of information. FPGA integrated circuit can be a solution to overpass this limitation [23–26]. Furthermore, FPGA circuits have different advantages. They are reconfigurable devices with parallel processing capability, low latency and high energy efficiency.

Figure 4.1 shows the key building blocks of an online neutron/gamma-ray discrimination system. Each one of these blocks is explained in details in section 1.2.5 of chapter 1. This chapter focuses on the FPGA implementation of the discrimination method block. The chapter presents the implementation details of each proposed discrimination method during this thesis (TTT_{ratio}, Form Factor, MLP and 1D CNN). Consequently, a comparison can be made among these various discrimination approaches, considering their discrimination performance, executions time and resource consumption. The main objective is to achieve the embedded implementation while maintaining the discrimination performance

obtained offline, ensuring an execution time less than the signal duration and using the minimal amount of resources. By achieving the implementation under this time constraint, we can avoid the miss detection of some radiations, as explained in section 1.2.5. In this context, the required signal duration for discrimination is 500 ns. This duration corresponds to the average pulse duration where the longest (largest) pulse is acceptable within the ADC quantization dynamics. The work is carried out under the assumption that the discrimination block receives an already buffered pulse, as precised in section 1.2.5. The implementation of the entire system shown in Figure 4.1 will be addressed in future work.

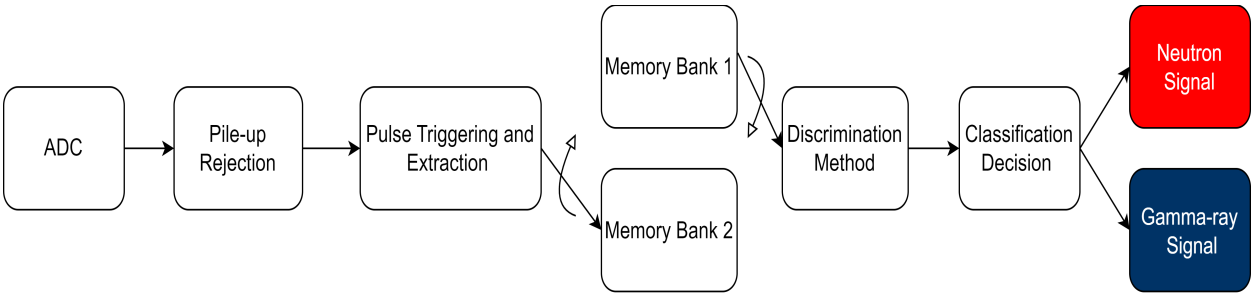


Figure 4.1: Key blocks of an online neutron/gamma-ray discrimination system.

Section 4.1 introduces the main characteristics of a FPGA and the methods used to configure this device. Section 4.2 describes the settings and process employed for the implementation. The quantization results of the four discrimination methods to reduce their sizes are presented in section 4.3. Sections 4.4 to 4.7 respectively explain the implementation details of TTT_{ratio}, Form Factor, MLP and 1D CNN. Finally, in section 4.8, the main conclusions resulting from the obtained implementation results are discussed.

4.1 Introduction to FPGA

FPGA is a semiconductor device based on a matrix of Programming Logic Blocks (PLBs) connected via reconfigurable interconnections. It is a type of integrated circuit that can be configured or reconfigured and offers a flexibility and adaptability for a wide range of digital logic applications. Therefore, it has been used in different domains, such as defense, aerospace, medical device, automation, wireless communications, and digital signal processing [98–100]. The main components of this reconfigurable device are:

1. PLBs: these are the fundamental building blocks of an FPGA. PLBs typically consist of Look-up Table (LUT) and Flip Flop (FF) components.

    - LUTs are used to implement Boolean logic functions, such as AND, OR, XOR, etc. A LUT typically has multiple inputs and a corresponding output, and its behavior is defined by a lookup table mapping input combinations to output values.
    - FFs are sequential elements that can store a single bit of data. FFs are used to build sequential circuits, such as registers, counters, and state machines. They allow for the storage and propagation of data within the FPGA design. For instance, they can be used to store the data between two consecutive layers of a ML model during the inference of a new input signal.

2. Block RAM (BRAM): this is a type of dedicated memory available in FPGA. BRAMs provide storage for large amounts of data and are often used for applications requiring data buffering, caching, or data storage. They offer high-speed access and are the components used for implementing FIFOs (First-In-First-Out) or LUTs that require fast data retrieval. For instance, they can be used to store the weights of a trained ML model or the data between two layers if its size is large and FFs are not sufficient for storage.

3. I/O Blocks: these blocks provide interfaces for communication between the FPGA and external devices. They typically include input buffers, output drivers, and configurable I/O standards to support different voltage levels and signal types. In the context of the neutron/gamma-ray discrimination task, an input block can be connected to the ADC and contain a buffer to store the acquired samples. The output block can be interfaced with a photodiode. This photodiode serves as an indicator, discerning the presence of neutrons when activated and gamma-rays when it is deactivated.

4. Digital Signal Processing (DSP) Blocks: DSP blocks are specialized components available in some FPGA that provide hardware acceleration for digital signal processing operations. They typically include dedicated Multiply–Accumulate (MAC) units and other arithmetic logic units optimized for processing digital signals. DSP blocks can significantly improve the performance and efficiency of algorithms such as filtering, modulation, convolution operation and matrix multiplication.

FPGA configuration is a process involving several steps:

1. Hardware Description of the desired circuit: this can be done using Hardware Description Language (HDL), such as VHDL or Verilog. These languages empower designers to define the behavior and functionality of intended digital circuit, while also specifying its structural details, including how components are interconnected. This level of description for a digital circuit is commonly referred to as Register Transfer Level (RTL), where designers outline how data is transferred between registers and processed by combinational logic.

2. Synthesis: the HDL code is synthesized into digital circuit made from the main components of the FPGA such as DSPs, LUTs and FFs. This circuit is represented by a logical netlist which consists of a list of the electronic components in a circuit and a list of the nodes they are connected to. In other words, the synthesis tool or software maps the high level design to specific FPGA resources.

3. Place and route: during this step, the synthesized circuit is mapped onto the actual resources of the FPGA. The placement stage determines the physical locations of the circuit elements on the FPGA, and the routing stage establishes the interconnections.

4. Bitstream generation: after the placement and routing, the design is converted into a configuration bitstream, also known as a bit file. This bit file contains the information necessary to configure the interconnections and logic blocks.

5. FPGA configuration: the generated bit file is then loaded onto the FPGA using a programming tool or hardware programmer. This process configures the FPGA to implement the desired circuit functionality.

In the past, FPGA technology could only be reconfigured by engineers with a deep understanding of digital hardware design using HDLs. The rise of High Level Synthesis (HLS) tools, however, is changing the rules of FPGA programming. HLS is a design methodology that allows designers to describe digital circuits at a higher level of abstraction using software programming languages, such as C/C++ or SystemC [101,102]. Then, these tools automatically convert these high level descriptions into synthesizable hardware descriptions, which can be used to generate the bit file for FPGA programming. The main steps involved in generating a bit file using HLS includes:

1. Algorithmic description: the designer starts by describing the desired circuit behavior and functionality using a high-level programming language like C/C++ or SystemC. This description represents the algorithm or functional level view of the design.

2. Pre-synthesis validation: the function to be synthesized should be validated before the synthesis with a test bench using C or C++ simulation. Thus, the program is ensured to correctly implement the required functionality.

3. Synthesis: the HLS tools transform the high-level description algorithm into an intermediate language allowing the modeling of data dependencies and parallelism. Then, these tools generate hardware description of the circuit (RTL level) representing the algorithm under the predefined constraints of data dependencies and parallelism. The synthesized output circuit represents the code at gate level using flip-flops, registers, multiplexers, and other low-level digital logic components. During this step, a report is generated indicating the number of components used in the circuit, the average latency time and initiation interval (II). The latency refers to the time it takes to receive the output of an input, while the II represents the number of clock cycles that must elapse before the system can accept a new input and start processing it.

4. Synthesis optimization: several optimization techniques, including resource sharing, pipelining, and loop unrolling, can be employed to enhance performance, reduce area of circuit, or meet specific design constraints. These optimizations can be incorporated into the synthesis process by instructing the HLS tool to consider the specific constraints.

5. C/RTL co-simultation: after the synthesis, the synthetized circuit performances should be validated according to the initial requirements. This step is made by comparing its output to the output obtained by the simulation in the pre-synthesis validation step.

6. Place and route, Bitstream generation and FPGA programming: these are the same steps of programming FPGA using HDLs.

Next section explains the process and the settings that we used to realize the FPGA implementations of the four discrimination methods.

4.2 Configuration and Implementation Process

The main aim of this work is to compare the FPGA implementation of TTT_{ratio}, Form Factor, MLP and 1D CNN models trained in Chapter 3 for the neutron/gamma-ray discrimination task. In order to achieve this comparison, the implementation of the four methods will be realized while respecting

the time constraint and using minimum amount of resource. The circuit synthesis of each discrimination method can be more optimized directly using HDL code instead of using the HLS approach. Nevertheless, the later can be sufficient to achieve the embedded comparison between the different discrimination approaches. It is important to note that in this work we only proceeded as far as the C/RTL co-simulation step, and the reports generated in the synthesis step used as the basis for comparing the different discrimination methods. The hardware implementation for a more deeper comparison should be considered for a future work.

The HLS software used in this work is Vitis HLS (2021.2) provided by AMD (Xilinx) [103]. For each discrimination algorithm, the code is first written and validated in C++ using fixed point arbitrary precision type instead of floating point types. Fixed point data types (ap_fixed <m, n>) represent data as a combination of integer and fraction bits, where m is the total number of bits and n is the number of bits dedicated to the integer part. By using this type, the area of the synthesized circuit can be reduced. In the presented case, discrimination is most challenging within an energy range of [100, 250] keVee. Obtaining a high TPR for a low FPR in this energy range is a difficult task, as proved in chapter 3. In this work, the number of bits for the variables in each discrimination method is adjusted to minimize the area while ensuring that the degradation in TPR is equal to or less than 2% for a 2% of FPR within this low energy, compared to the performance obtained with floating point types. Synthesizing the four proposed discrimination approaches under these extreme conditions case can enable a more valid and meaningful comparison between them. Furthermore, the input is quantized to 12 bits, where all bits are allocated to the decimal part. This quantization approach is chosen because the data is acquired through a 12-bit resolution ADC, and the maximum signal amplitude is less than one volt. A test bench created from the labeled dataset prepared at 250 MHz in chapter 3 (1000 samples) is used for validating the C++ codes and the C/RTL co-simulation step.

Following the validation and tuning of the number of bits, the synthesis process is optimized to meet the required time constraint (a latency less than 500 ns) while using minimal amount of resources. The final step involves validating the synthesized circuit through the C/RTL co-simulation. Concerning the FPGA type, this work employs a Xilinx FPGA with the part number xc7z020-clg484-3. Furthermore, the synthesis of the four discrimination algorithm is performed at a clock frequency of 200 MHz (5 ns per cycle). In our laboratory, various acquisition cards have been developed using FPGAs of a similar type, operating in the same range of clock frequency [25, 26].

Finally, it is important to note that training a ML model while taking into account the quantization of the weights and activation functions outputs that will be used for hardware implementation has proved to be more effective than quantizing the model after training [104]. In this work, Qkeras python library (version 0.9.0) is used to quantize the ML models (MLP and 1D CNN) aware training. Subsequently, the structure of each quantized model is synthesised via Vitis HLS. Figure 4.2 provides an overview of the workflow used to achieve the FPGA implementations of the four discrimination methods. The workflow of TTT_{ratio} and Form Factor methods start from the C++ implementation. Next section presents the discrimination performance obtained by the four proposed methods using floating point and fixed point types. The same architectures of MLP and 1D CNN models obtained in chapter 3 are quantized aware training. Thereafter, the obtained results are validated in C++ using Vitis HLS and the created test bench.

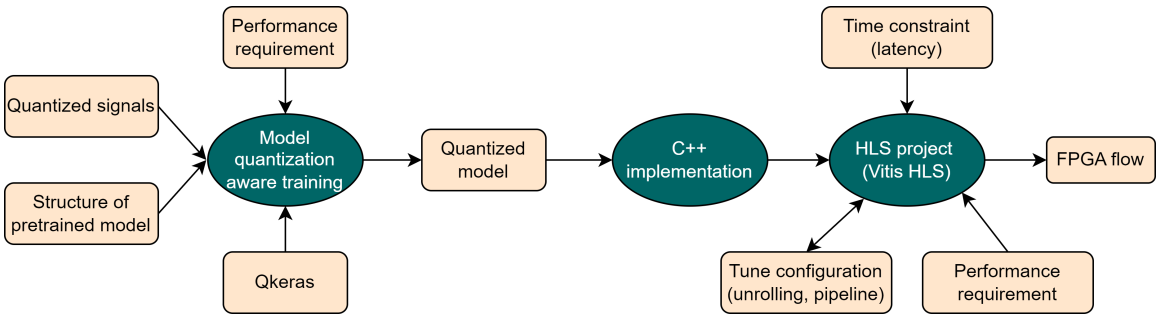


Figure 4.2: The proposed workflow to translate a ML model into a FPGA implementation using Qkeras and Vitis HLS.

4.3 Quantization of TTT_{ratio}, Form Factor, MLP and 1D CNN Discrimination Methods

In this step, the number of bits used to represent the parameters of each proposed discrimination algorithm is tuned empirically to meet the discrimination performance constraints presented in section 4.2. The algorithms of TTT_{ratio} and Form Factor methods are first coded in C++. This involves quantizing the input to 12 bits, as explained in section 4.2. Subsequently, the tuning process for the number of bits for each variable in each method is achieved. For a FPR equal to 2%, the TPR of TTT_{ratio} (Form Factor) is reduced from 78% (77%) to 77% (76%), as indicated in Table 4.1. Sections 4.4 and 4.5 provide a detailed presentation of the synthesis process for each method and the resulting number of bits for their variables.

Table 4.1: TPR for FPR = 2% obtained by TTT_{ratio} and Form Factor, MLP and 1D CNN algorithms using floating point and fixed point types on validation dataset for the energy range [100, 250] keVee.

Method	floating point	fixed point (Qkeras)	fixed point (Vitis HLS)
1D CNN (one CL)	88%	88%	79%
1D CNN (two CLs)	88%	88%	87%
MLP	84%	84%	83%
TTT_{ratio}	78%	-	77%
Form Factor	77%	-	76%

The MLP model trained in section 3.6 for neutron/gamma-ray discrimination consists of an input, an output and two hidden layers of 32 neurons each. The quantization of this MLP model is tuned for both the weights and the outputs of each layer using Qkeras framework. The minimal obtained representation is 8 bits, all of them dedicated to the decimal part. For the energy range [100, 250] keVee, the TPR is decreased from 84% to 83%, for a FPR equal to 2%, as shown in Table 4.1.

As for the MLP model, the quantization of the trained 1D CNN model in section 3.6.2 is tuned aware training for the weights and output of each layer. The optimal representation that respects the degradation constraints in the discrimination performance is 16 bits, with 4 bits to the integer part. Furthermore, the obtained results in Table 4.1 indicate that the quantized 1D CNN model outperforms the

non quantized MLP model. Nevertheless, the results obtained from testing the C++ code of this model architecture using *ap_fixed* type and Vitis HLS show that the performance of the model is significantly degraded, as indicated in Table 4.1. This may be due to the difference between the implementation of the quantization methods in Qkeras and Vitis HLS, leading to variations in quantization errors. Some information is lost during quantization by Vitis HLS due to rounding errors, which can impact the model performance as observed in the presented case. To address this problem, increasing the size of the model can be a potential solution. A larger model might be more resilient to quantization errors as it can distribute the errors across more parameters and absorb the quantization noise, leading to less performance drop.

Therefore, the size of the model is empirically tuned by quantizing various model sizes aware training and then assessing their discrimination performance through C++ simulation in Vitis HLS. The obtained model exhibits a 1% degradation in discrimination performance after its implementation in C++ using fixed types, as indicated in Table 4.1. This obtained model consists of two CLs, two ReLu, one dense and one Sigmoid operations. The obtained number of bits for quantization is 16, with 4 of them dedicated to the integer part.

4.4 TTT_{ratio} Implementation

TTT_{ratio} algorithm as defined in section 1.3.1, is the ratio between the tail and total integral of the signal. In this thesis, the optimization algorithm implemented in [36] is used to tune t_{short} and t_{long} in order to optimize the discrimination performance based on FOM evaluation metric, where the resulting optimum values are 20 ns and 500 ns, respectively. Q_{tail} and Q_{total} integrals in this optimization algorithm are calculated using trapezoidal numerical integration method with delta equal to one, which is represented by equation 4.1. The latter indicates that the computing of this numerical method involves one multiplication by a constant (0.5) and k addition operations, where k is the integral length. Thus, the main operations to calculate this ratio and achieve the synthesis for a signal of length n are:

- n addition operations for the Q_{total} integral.
- m addition operations for the Q_{tail} integral.
- The division between Q_{tail} and Q_{total} integrals.

$$\int_0^k f(t)\,dt \approx \frac{x_0 + x_k}{2} + \sum_{i=1}^{k-1} x_i \tag{4.1}$$

The algorithm of this method is first coded in C++. Then, the numbers of bits of the parameters responsible for storing the two integral calculation results and their ratio are tuned. *ap_fixed <18, 10>* is the type used for the three variables without violating the constraints of discrimination performance presented in section 4.2. Subsequently, the code is synthesized by paralyzing the calculation of the two integrals. Table 4.2 shows the obtained RTL synthesis report. A second solution can be obtained with lower latency and increased DSP unit consumption by calculating the inverse of the Q_{tot} integral and then multiplying it by the result of the Q_{tail} integral (see Table 4.2). The computation of the inverse is done using *hls::recip()* method implemented in Vitis HLS, specifically designed for fixed-point types.

ap_fixed <18, 10> type is used to store the result of the inverse operation.

Table 4.2: RTL synthesis reports for the two proposed solutions in FPGA implementation of the TTT_{ratio} discrimination algorithm.

Model	Latency (ns)	II (ns)	BRAM	DSP	FF	LUT
First solution	230	5	0	0	5256	4630
Second solution	215	5	0	9	4325	3848

Finally, it is important to note that this FPGA implementation of TTT_{ratio} algorithm assumes that the entire signal should be ready before starting the discrimination process, which is different from pipeline chains [25]. Another solution can be obtained by proposition that the execution of the discrimination algorithm can start directly after receiving the starting point of the signal. In this solution, the computation of Q_{tot} and Q_{tail} integrals are done sequentially through two steps:

1. As long as the $trigger_{tot}$ flag = 1 : $Q_{tot} = Q_{tot}$ + x(n).
2. As long as the $trigger_{tail}$ flag = 1: $Q_{tail} = Q_{tail}$ + x(n).

$trigger_{tot}$ and $trigger_{tail}$ are two boolean variables indicating the start and the end of computation of Q_{tot} and Q_{tail} integrals, respectively. Then, the division between two integrals can be performed by a floating calculation unit which can be used also for communication, such as microcontroller. The resource consumption in this approach is lower than that of the first approach, as the n (m) addition operations of Q_{tot} (Q_{tail}) integrals are executed sequentially, and the division operation is computed externally. Furthermore, the output decision can be obtained almost immediately after the acquisition end of one signal. However, for the MLP and 1D CNN models, due to the matrix multiplication and the need for optimization in implementation, the entire signal must be ready before starting the algorithm's execution. Therefore, to ensure a fair comparison among the four proposed discrimination approaches, their embedded implementations are performed under the assumption that the algorithm block receives an already buffered signal.

Next section explains how the implementation on FPGA of the proposed Form Factor discrimination method can be optimized. The synthesis results are compared to those obtained by the FPGA implementation of TTT_{ratio} algorithm.

4.5 Form Factor Implementation

Form Factor PSD method is the ratio between the root mean square and the mean of the signal as defined in section 3.7.1. Thus, the main operations to calculate this factor and achieve the synthesis for a signal of length n are:

1. n addition operations to compute the mean of the signal.
2. n MAC to calculate the mean square.
3. The RMS.

4. The division between the mean and RMS.

After developing the algorithm in C++, the number of bits for the variables storing the results of mean, mean square, root mean square and Form Factor are tuned to meet the degradation constraints in the discrimination performance presented in section 4.2. The obtained type is *ap_fixed<32, 3>*. Thereafter, during the synthesis process, the calculation of the mean and the mean square of the signal are fully paralyzed. The root mean square is then computed using the dedicated hls::sqrt() method implemented in Vitis HLS, specifically designed for fixed-point types. Following this calculation, the inverse of the mean is calculated and then multiplied by the RMS. Experimental results in Table 4.3 indicate that performing direct division violates the time constraint, which is a latency less than 500 ns. The type used to store the result of the inverse operation is *ap_fixed<32, 25>*.

Table 4.3: Comparing the RTL synthesis reports for the solutions proposed in the FPGA implementation of TTT_{ratio} and Form Factor neutron/gamma-ray discrimination algorithms.

Model	Latency (ns)	II (ns)	BRAM	DSP	FF	LUT
TTT_{ratio} (first solution)	230	5	0	0	5256	4630
TTT_{ratio} (second solution)	215	5	0	9	4325	3848
Form Factor (direct division)	585	5	0	132	22680	14022
Form Factor	290	5	0	143	16526	10344

The obtained synthesis report indicates that the total latency achieved is 290 ns, using 143 DSP, 16526 FFs and 10344 LUTs. Furthermore, as it was expected, Table 4.3 indicates a high superiority of TTT_{ratio} algorithm concerning the embedded implementation, consuming less amount of resources and execution time. Nonetheless, the Form Factor algorithm has the advantage of being independent of tuning parameters, providing the same discrimination performance.

4.6 MLP implementation

Different approaches have been proposed to optimize the inference of MLP on FPGA [105–107]. The works by [105] and [107] explain the implementation of two MLP models trained for medical diagnosis and digital recognition tasks, respectively. The authors in [106] develop a general MLP flow that can take arbitrary datasets as input and automatically produce optimized neural network architectures and hardware designs based on a set of constraints and fitness functions such as the accuracy, latency and throughput. They show that, in general, executing a trained MLP model on FPGA is faster than on GPU. Their framework is based on evolutionary optimization algorithms, OpenCL framework and 2D systolic array configuration. In the presented case we work to optimize the embedded implementation of the trained MLP model in section 3.6 for the neutron/gamma-ray discrimination task, while respecting the time constraint. Building an automatic framework is not an objective of this study.

The trained MLP model in section 4.3 has three main components (Figure 4.3). The first one is the fundamental component of a MLP model, the dense layer. The computation of this layer is done via two nested *for* loops. Algorithm 1 indicates that these two loops iterate over the number of neurons (m) in the layer and the input length (n) respectively. In other words, there are $n*m$ MAC operations for one layer. The two loops of each layer can be unrolled to optimize the synthesis, where the unrolling factor

of a loop is a divisor of its number of iterations. Figure 4.4 shows that the parallel execution of the first loop requires to copy the data k times, where k is the unrolling factor of the loop. Therefore, parallel execution of the second loop of all layers has first priority. However, this is hampered by a memory dependency problem. In algorithm 1, the next iteration reads the variable *output[i]*, while the previous iteration writes data to *output[i]*. Therefore, the subsequent iteration cannot start until the prior one is completed. An adder tree structure can be a solution to this problem, as illustrated in Figure 4.5. In this solution, all n multiplications from the second loop are first computed and loaded into a *temp* array of length n. Then, the sum of the elements in *temp* is calculated using the adder tree function, as shown in algorithm 2. This approach reduces the calculation complexity from O(n) to O(log(n)).

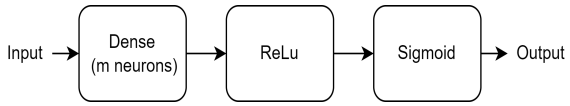


Figure 4.3: The three main operations in the trained MLP model.

Algorithm 1 Dense algorithm

input array *input* with length n weight matrix w with dimensions $m \times n$
output array *output* with length n

```
0: function Dense(input, weight)
1: for i ← 0 to m do
2:     output[i] ← b[i]
3:     for j ← 0 to n do
4:         output[i] += input[j] · w[i][j]
5:     end for
6: end for
7: return output
```

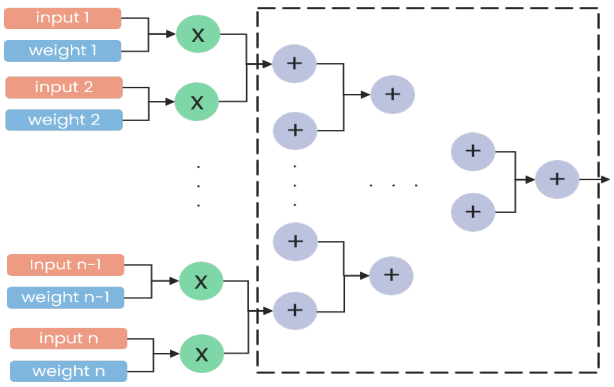


Figure 4.5: Adder tree structure

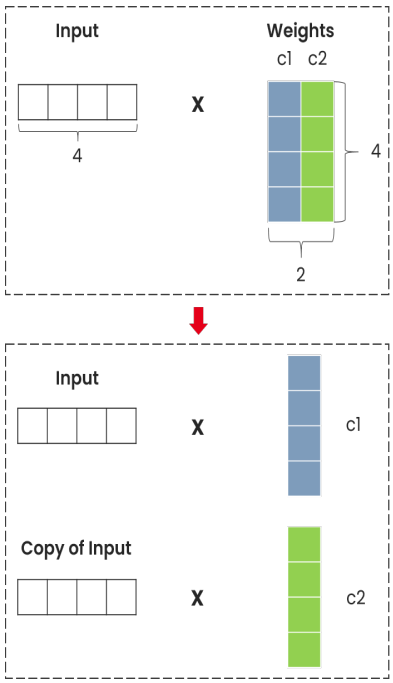


Figure 4.4: Illustration of the first loop unrolling in dense layer.

Algorithm 2 Optimized dense algorithm

input array *input* with length n weight matrix w with dimensions $m \times n$
output array *output* with length n

```
0: function Dense(input, weight)
1: for i ← 0 to m do
2:     temp[n]
3:     for j ← 0 to n do
4:         temp[i] = input[j] · w[i][j]
5:     end for
6:     output[i] = adder_tree(temp)
7:     output[i] += b[i]
8: end for
9: return output
```

The second component of a MLP model is ReLu activation function of the two hidden layers which is shown in algorithm 3. In this work, the calculation of this function is fully unrolled for all layers.

The third component is the Sigmoid activation function of the output layer which is represented by the equation $1/(1+exp^{-x})$. In the presented case, x represents the output resulting from the last dense layer. Calculating the exponential component of this function can be computationally challenging for hardware implementation. Therefore, to speed up the hardware computation, Sigmoid function can be approximated using piecewise linear approach and a lookup table:

Algorithm 3 ReLU function algorithm for one layer

input array *input* with length n
output array *output* with length n

```
0: function ReLu(input)
1: for i ← 0 to n do
2:     if input[i] ≥ 0 then
           output[i] ← input[i]
3:     end if
4: end for
5: return output
```

1. A lookup table of Sigmoid values is created, dividing the input range into N segments and calculating Sigmoid values for each segment using the standard Sigmoid function.

2. The lookup table is stored in BRAM during the synthesis.

3. The segment index of x is calculated based on the input range.

4. The obtained index in step 3 corresponds to the position of the element in the created lookup table that represents the Sigmoid value of x.

5. The input range and the number of segments N should be tuned to preserve a similar precision of the standard Sigmoid function.

The trained quantized MLP model for the discrimination task between neutrons and gamma-rays consists of three dense, two ReLu and one Sigmoid operations (Figure 4.6). The dimensions of the weight matrices (loop parameters) for these layers are (32, 126), (32, 32), and (1, 32) respectively. The structure of this trained quantized MLP model is implemented in C++. Subsequently, the synthesis is optimized to meet the time constraint by empirically tuning the unrolling factors for each layer. This adjustment can be achieved by using for each loop in each layer the *pragma HLS unroll factor*. This Vitis HLS pragma is dedicated to parallel loop execution. The obtained *unrolling factors* for the loop iterations in the three layers are (1, 126), (1, 32), and (1, 32) respectively, with corresponding latency of 39, 37, and 6 cycles. The Sigmoid function uses 1 BRAM for lookup table and takes 2 cycles for execution, with input range (-8, 8) and 1024 segments. The first layer consumes 126 DSPs for MAC operations, while the MAC operations of the second and third layers are computed by LUTs. This feature in Vitis HLS can only be applied when the multiplication operation involves numbers represented by a maximum of 8 bits each. Weight storage in FFs and LUTs is preferred over BRAMs to avoid latency increase due to read and write operations. The obtained synthesis report indicates a total latency of 490 ns (98 cycles) using 126 DSP units, 46,059 LUTs, 17,657 FFs, and 1 BRAM.

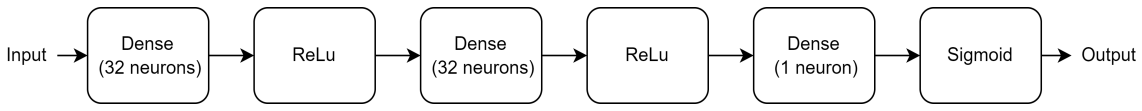


Figure 4.6: The inference flow of the trained MLP model.

Another synthesis approach can yield a latency value of 350 ns, as shown in Table 4.4. In this proposed solution, the unrolling factors of the three layers are (32, 3), (32, 4) and (1, 32). This solution enables the parallel computation of 256 multiplications. The synthesis report shown in Table 4.4 indicates that only 96 DSP units are consumed for the multiplications, with the remaining 160 multiplications being executed by LUTs. These 96 DSP units are assigned to the first layer. In this solution the weights of first and second layers and 32 copies of the input signal are stored in BRAMs. Therefore, the number of BRAMs is increased to 220 (Table 4.4). In summary, a trade-off exists between the latency and resources of both solutions. The choice between them depends on the specific requirements of target application.

Table 4.4: RTL synthesis reports for the two proposed solutions in the FPGA implementation of the trained MLP model.

Model	Latency (ns)	II (ns)	BRAM	DSP	FF	LUT
MLP (first solution)	490	235	1	126	38364	46059
MLP (second solution)	350	240	220	96	20580	51177

4.7 1D CNN implementation

Many methods for incorporating machine learning into FPGA devices primarily focus on 2D-CNNs for image recognition tasks [108–116]. Different studies have addressed the implementation of 1D-CNN on FPGA for 1D signal applications [117, 118]. In [117], the authors optimized the implementation of a specific trained 1D-CNN designed for underwater target spectrum recognition. The authors used three main techniques: quantization, loop unrolling and tilling. Moreover, they assessed the impact of these techniques on the resource consumption of their 1D-CNN model implementation. The authors in [118] propose a pyramid layer-folding pipeline structure to implement a 1D-CNN model trained for speaker recognition task. The primary goal of this work is to optimize the embedded implementation of the 1D CNN model from chapter 3 while meeting the time constraint. Building an automatic framework or a novel approach to optimize the embedded implementation is not an objective of this work.

The obtained 1D CNN model for the discrimination task in section 4.3 has four components: 1D CL, dense layer, ReLu and Sigmoid activation functions, as illustrated in Figure 4.7. The calculation in the 1D CL goes through four nested *for* loops. They iterate over the output length (out_size), number of filters (nf), number of input channels (nc), and kernel width (k), respectively (algorithm 4). Thus, the total number of MAC operations for a layer is $out\_size * nf * nc * k$. To reduce the latency of the FPGA implementation and meet the time constraints of our application, it may be necessary to parallelize the execution of these operations for each layer.

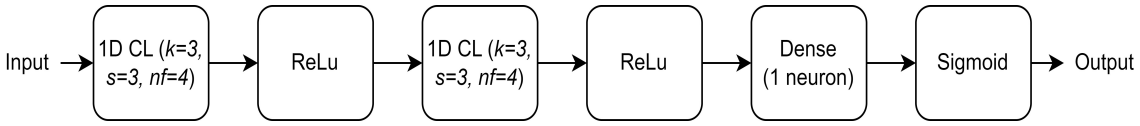


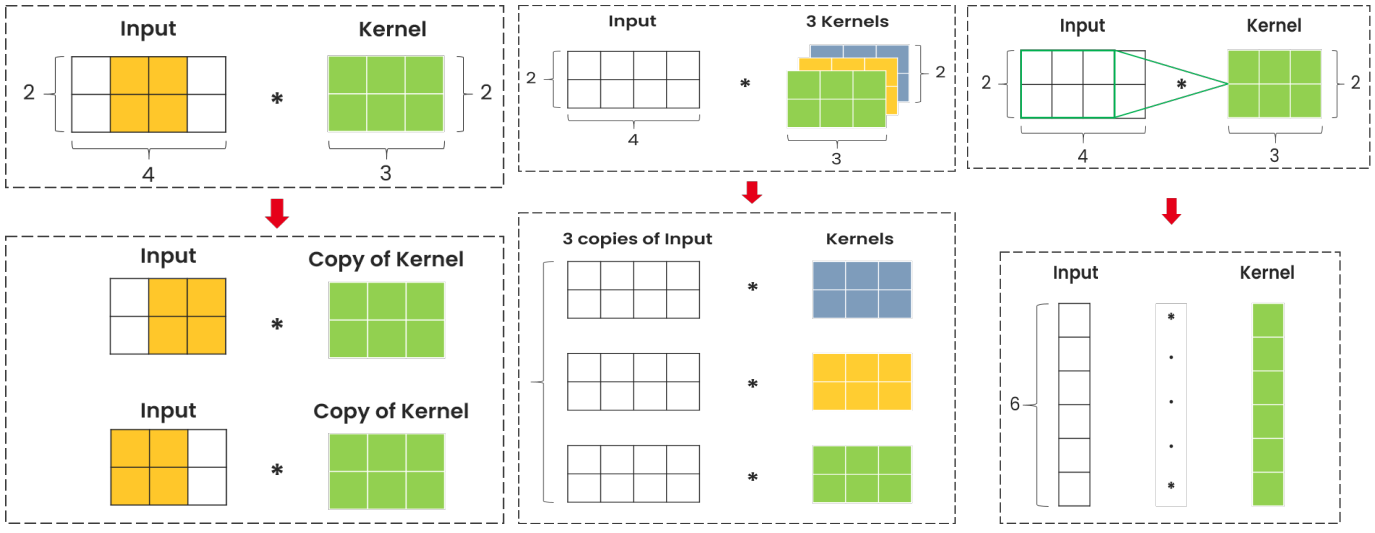
Figure 4.7: The inference flow of the trained 1D CNN model.

Algorithm 4 1D Convolution layer algorithm

input and weight (w) matrices with dimensions $nc \times n$ and $nc \times k \times nf$
output array $output$ with dimensions $out\_size \times nf$

0: **function** $1DCNN\_Layer(input, weight)$
1: **for** $i \leftarrow 0$ to out_size **do**
2: **for** $j \leftarrow 0$ to nf **do**
3: $output[j, i] \leftarrow b[j]$
4: **for** $c \leftarrow 0$ to nc **do**
5: **for** $p \leftarrow 0$ to k **do**
6: $output[j, i] \mathrel{+}= input[c, s * (p - 1) + k] \cdot w[c][p][j]$
7: **end for**
8: **end for**
9: **end for**
10: **end for**
11: **return** $output$

The parallel execution of the first loop involves copying *(k-s) * nc* columns of the input data and the kernel matrix of the layer m times, where m and s are unrolling factor of the loop and the stride, respectively (Figure 4.8(a)). On the other hand, the parallel execution of the second loop requires duplicating the input data m times (Figure 4.8(b)), while the unrolling of third and fourth loops does not impact the data size (Figure 4.8(c)). In our approach, the primary focus is on the parallel execution of the third and fourth loops across all layers. Subsequently, if these unrolling steps do not help to meet the time constraint, we empirically tune the unrolling factor of the first and second loops.



(a) Unrolling of first loop, s=1, k=3, and nc=2

(b) Unrolling of second loop, nc=2, nf=3

(c) Unrolling of third and fourth loops

Figure 4.8: Illustration of 1D CL loops unrolling

The dense layer is the output layer comprising a single neuron. The number of multiplications performed in this layer is determined by multiplying the output length of the last CL by its number of channels. To optimize the implementation, it is necessary to tune the unrolling factor, which should be a divisor of this multiplication result, along with other unrolling parameters of the CLs. The Sigmoid and ReLu activation functions can be implemented and synthesized in the same manner than in the case of MLP model.

Table 4.5 displays the configuration of the two CLs and the dense layer of the quantized 1D CNN model obtained in section 4.3. It also presents the dimensions of the loops for each layer. The dimension of the dense layer loop is calculated by multiplying the output length of the second CL (14) by its number of filters. The structure of this model is implemented in C++ using *ap_fixed* type. Thereafter, empirical tuning is made to unrolling factors for each CL layer, number of segments, and input range of the Sigmoid function to optimize the synthesis and respect the time constraint. Table 4.5 shows the obtained value for each one of these parameters. The total latency achieved is 490 ns, consuming 228 DSPs, 156,833 LUTs, 193,625 FFs and 2 BRAMs. The 2 BRAMs are dedicated for the lookup table of the Sigmoid.

Table 4.5: Obtained unrolling factors for each layer in the trained 1D CNN model.

Layers	Loops dimensions	Unrolling factors
First CL (k=3, s=3, nf=4)	(42, 4, 1, 3)	(42, 1, 1, 3)
Second CL (k=3, s=3, nf=4)	(14, 4, 4, 3)	(1, 4, 4, 3)
Dense layer (one neuron)	(56)	(56)
Sigmoid	number of segments = 2048, input range = (-32 , 32)	

These results show that, in the presented case, the MLP model can be effectively implemented on a FPGA for online classification with fewer resources compared to the 1D CNN model, while achieving the same latency (Table 4.6). However, it is important to note that the discrimination performance of the 1D CNN model is higher than that of the MLP model. Furthermore, Table 4.6 indicates that both the latency and resource usage of TTT_{ratio} algorithm are considerably lower than those of ML models. Nevertheless, for relatively low energy range ([100, 250] keVee), this discrimination method is significantly less efficient than ML models (Table 4.6). Therefore, in applications involving classification of low energy radiations in real time, the higher resources consumed by MLP and 1D CNN models can be justified. In contrast, for the discrimination of higher energy radiations, TTT_{ratio} algorithm is more advantageous as it achieves the same discrimination performance as ML models with lower latency and resource consumption. Moreover, the synthesised reports in Table 4.6 shows a superiority of TTT_{ratio} algorithm compared to Form Factor method, concerning their FPGA implementations. The second solution of the former uses only 6% of the DSP units, 26% of the FFs and 37% of LUTs compared to those used by the latter. Nevertheless, the FPGA implementation of Form Factor provides similar discrimination performance and its independent of tuning parameters, which can be an advantage point for practical applications.

Table 4.6: RTL synthesis reports for the solutions proposed in the FPGA implementation of 1D CNN, MLP, TTT_{ratio} and Form Factor neutron/gamma-ray discrimination methods.

Model	Latency (ns)	II (ns)	BRAM	DSP	FF	LUT	TPR (FPR=2%)
TTT_{ratio} (first solution)	230	5	0	0	5256	4630	77%
TTT_{ratio} (second solution)	215	5	0	9	4325	3848	77%
Form Factor	290	5	0	143	16526	10344	76%
MLP (first solution)	490	235	1	126	38364	46059	83%
MLP (second solution)	350	240	220	96	20580	51177	83%
1D CNN	490	185	2	228	193625	156833	87%

4.8 Conclusion

This study compares the FPGA implementations using HLS tools of the TTT_{ratio} and Form Factor pulse shape discrimination algorithms, MLP, and 1D-CNN models for neutron/gamma-ray discrimination using EJ276 plastic scintillator. The main objective is to achieve the implementation of each method, aiming for a latency lower than the signal duration while minimizing resource consumption. This constraint is important in some applications to perform a discrimination on the fly and minimizing the percentage of missed radiation detection, especially in a high count scenario. Based on discrimination performance as the evaluation metric, the 1D CNN model outperforms the MLP model, which, in turn, outperforms the TTT_{ratio} and Form Factor algorithms, especially for low energy radiations ([100 keVee, 250 keVee]). Furthermore, both the TTT_{ratio} and Form Factor methods exhibit similar performance, with the latter having the advantage of being independent of any tuning parameter. In contrast, when the comparison is based on latency and resource consumption, the order of the four methods is completely inverted. For this reason, the choice of the method depends on the target application and the available resources for implementation.

For a more accurate comparison between the four methods, future work could involve a hardware implementation of each method and the complete discrimination system, incorporating different tasks such as the BLR and the classification decision.

Chapter 5

Conclusions and Future Developments

In the field of nuclear instrumentation, there exist various types of radiation detectors, including gas ionization, semiconductor, and scintillator detectors. While some detectors are sensitive to specific types of radiation, others can simultaneously detect multiple types of radiation. Consequently, there is a need for a discrimination method to classify the different detected radiation types. An example of a multi-radiation detector is the plastic scintillator, which can detect neutrons and gamma-rays. This organic detector has different advantage points for radiation detection. It can be easily shaped, produced in large volumes and has a relatively low-cost. In addition, it has increased durability, non-toxicity, and non-flammability characteristics. However, this detector exhibits relatively lower discrimination capability when compared to other types of detectors, such as liquid or stilbene scintillators. This is mainly due to the slight difference between the signals produced by neutron and gamma-ray interactions in a plastic scintillator. State of the art methods using this type of detector have limitations in terms of performances, particularly for classifying low energy radiations. Furthermore, the absence of a ground truth neutron dataset adds a significant challenge for evaluating any proposed discrimination approach. The acquisition of this dataset is primarily challenged by the influence of gamma-ray emissions in neutron sources. Additionally, the short duration of signals generated by a plastic scintillator, lasting only a few hundred nanoseconds, presents an additional challenge for the implementation of an online discrimination system. In some applications, the embedded implementation should be achieved with an inference time less than the signal duration to minimize the percentage of missed radiation detection. These difficulties make the discrimination problem in this detector one of the most challenging classification problems in the field of nuclear instrumentation.

The main objective of this thesis is to examine the potential of ML techniques in enhancing the classification of radioactivity signals for online applications. Due to the relatively high complexity of neutron/gamma-ray discrimination in plastic scintillators, as explained in the previous paragraph, the work of this thesis was dedicated to address this problem using ML tools. The identified effective solution could be extended to classify different types of radiation detected by varying detectors. The approach adopted in the course of this work was organized as follows.

First, we introduced the topic of classifying radioactivity signals with a focus on the discrimination of neutrons and gamma-rays in plastic scintillators. Then, we presented the various components of a typical scintillation acquisition system. Subsequently, we analyzed and discussed the main limitations of state of the art regarding the neutron/gamma-ray discrimination in plastic scintillator. This study

showed that the evaluation of discrimination methods remains a challenging task, mainly due to the effects of gamma-ray emissions in neutron sources. Furthermore, the cost and size of an embedded discrimination system are proportional to its sampling frequency. Therefore, it is important to determine the minimum sampling frequency needed for discrimination to optimize the embedded implementation. This point is key in the framework of this study for embedding ML approaches on edge devices.

Therefore, we studied how the discrimination performance of an organic scintillator varies concerning SNR and vertical resolution across different sampling frequencies. The study results showed that a trade off between the sampling frequency and SNR is needed to optimize the discrimination performance. Increasing SNR can lead to a reduction in the minimum required sampling rate for effective discrimination. Furthermore, experimental results indicated that, when using the same sampling rate, a higher SNR can provide superior discrimination performance. In other words, the performance of a discrimination method depends on the parameters and components of the acquisition chain. Even subtle modifications in acquisition conditions and environmental factors can have a significant impact on the discrimination performance. Consequently, to obtain the best discrimination performance for a predefined application, it is essential to carefully design and optimize the measurement setup. The comparison of different discrimination approaches across different studies can be challenging due to the complexities of ensuring consistent acquisition conditions and environments. A detailed information about the SNR is needed to ensure a fair comparison.

The study of acquisition conditions served as a basis for proposing a method for generating clean and labeled neutron/gamma-ray datasets with an organic scintillator. A primary limitation of the proposed signal labeling approach is its reliance on the TTT_{ratio} discrimination algorithm. Another limitation is its dependence on both the sampling frequency and the energy range of the incident radiations. Furthermore, in comparison to the traditional method, the implementation of the ToF measurement chain is more complicated, and the acquisition process takes a longer time due to the necessity for coincidence detection. In contrast, the presented labeling process can identify and remove mislabeled samples across the entire ToF spectrum, including those contributing to peak values. Moreover, the proposed process is capable of producing labeled datasets even in situations where the discrimination capability of the employed acquisition chain is limited.

By using the proposed labeling pipeline and by optimizing the acquisition chain components and parameters, labeled datasets of neutron/gamma events were obtained with an EJ276 plastic scintillator at a minimum sampling rate of 250 MHz. Supervised MLP and 1D CNN models were trained and assessed using these datasets to discriminate the prepared neutron and gamma-ray events. We compared the performance of these models against that of the state of the art TTT_{ratio} algorithm, while taking into account variations in sampling frequency and radiation energy levels. The 1D CNN model showed superior performance compared to the MLP model, which, in turn, outperformed the TTT_{ratio} algorithm. The most notable performance difference was observed for low energy radiations ([100, 250] keVee), highlighting the interest of using ML models for applications necessitating discrimination of relatively low energy neutron/gamma-ray events.

It is crucial to emphasize that the TTT_{ratio} PSD algorithm can exhibit enhanced discrimination performance for the energy range [100, 250] keVee, using the same scintillator type (EJ276), when

employing a dedicated analog signal processing blocks, such as Op-Amp integrator in the acquisition chain, as shown in the results presented in Chapter 2. Nevertheless, a supervised ML model trained on the acquired 1D signals can still achieve better discrimination performance for lower energy radiation compared to TTT_{ratio}, as shown by the obtained results in this thesis. The only requirement is to repeat the preparation of the labeled dataset using the newly proposed experimental setup and subsequently train the model. In fact, the main limitation of this supervised ML approach is its sensitivity to the components and parameters of the measurement chain. The repetition of the preparation of clean labeled neutron signals for any subtle change in the acquisition chain presents a challenging and time consuming work.

Unsupervised domain adaptation can be a solution to overpass this limitation. However, a critical challenge to use this approach arises from the input dimension of the model, which depends on the characteristics of the measurement chain. To address this problem, features extraction process emerges as a potential solution. This method can reduce the dimension data to a certain level regardless of the acquisition chain and input signal length. In this study, several features were defined for extraction from the signals taking into account the target application. The obtained results indicated that supervised models trained on raw signals provide superior performance compared to the supervised approach relying on the extracted features. Our exploration of the features extraction approach shows that Form Factor attribute can be used as a discrimination method, reaching the performance of the TTT_{ratio} algorithm without requiring any parameter adjustments. This independence from tuning parameters can be crucial in some applications.

For real time applications, the comparison between the proposed discrimination approaches should not only be limited to the discrimination performance. Their embedded implementations should also be compared. This is the last part of the work presented in this manuscript, where the implementations were performed on FPGA. The decision to employ FPGA was motivated by its parallel processing property and its capacity to process the digitized signal at a sampling frequency on the order of several hundred megahertz (MHz), which is a key requirement in this work due to the short duration of neutron and gamma-ray signals generated by a plastic scintillator. The objective was to optimize the implementation of each method in order to achieve a latency less than the signal duration while minimizing the amount of consumed resources. Based on discrimination performance as the evaluation metric, the 1D CNN model outperforms the MLP model, which, in turn, outperforms the TTT_{ratio} and Form Factor algorithms, especially for low energy radiations ([100, 250] keVee). Furthermore, both the TTT_{ratio} and Form Factor methods exhibit comparable performance, with the latter having the advantage of not relying on any tuning parameters. However, when assessing based on latency and resource consumption, the ranking of the four methods is completely reversed. Consequently, the choice of method depends on the target application and the available implementation resources. For instance, in applications involving classification of low energy radiations, the higher resources usage of MLP and 1D CNN models can be justified. In contrast, for the discrimination of higher energy radiations, Form Factor and TTT_{ratio} algorithms are more advantageous as they achieve the same discrimination performance as ML models with lower latency and resource consumption.

Future Developments

The work presented in this thesis reveals the interest of using embedded ML in the context of radioactivity signals classification, mainly for low energy radiations. Furthermore, the work opens different paths that can be considered for future work:

1. The results obtained by this work can be validated by a hardware implementation of a prototype for an online discrimination system with a plastic scintillator based on ML approach. This is significant in applications where the accurate classification of low energy radiations is a requirement. This step involves the hardware implementation of the complete discrimination system, incorporating different tasks such as the BLR and the classification decision.

2. The work process of this thesis can be expanded to classify various radiation types (neutron, gamma, beta, and alpha) detected by different types of detectors. For example, EJ276 and other scintillators can be used to create neutron, beta and gamma-ray datasets. The neutron dataset can be generated using the labeling method proposed in this thesis, while gamma-ray and beta datasets can be obtained using pure gamma-ray (^{137}Cs) and beta-ray (^{36}Cl) sources. Additionally, synthetic pile-up events can be generated and incorporated into the labeled datasets as a fourth class. Therafter, a 1D CNN model can be trained on these prepared datasets to classify the four classes simultaneously.

3. A study could be realized to assess the effectiveness of unsupervised domain adaptation tools in the context of neutron/gamma-ray discrimination in organic scintillators, which is a critical point since the preparation of labeled dataset is a challenging and time consuming step. Furthermore, self-supervised ML tool can be explored as a solution to overcome the challenge raised by the varying input lengths of ML models due to changes in the measurement chain. This exploration could lead to the identification of invariant features capable of representing the signal and containing relevant information for discrimination purposes. In order to carry out this study, first a neutron/gamma-ray dataset should be acquired by a different acquisition chain than the one used in this study with EJ276 plastic scintillator. This dataset can be cleaned and labeled using the labeling approach proposed in this thesis. Subsequently, different unsupervised domain adaptation tools can be explored to adapt the models trained in this thesis for classifying the newly acquired dataset. The labeling of the second dataset is essential for accurately evaluating the different proposed adaptation methods.

Personal Bibliography

Journals

A. Hachem, Y. Moline, G. Corre, B. Ouni, M. Trocme, A. Elayeb, and F. Carrel, "Labeling strategy to improve neutron/gamma discrimination with organic scintillator," *Nuclear Engineering and Technology*, 2023

A. Hachem, Y. Moline, G. Corre, J. Gauthier, and F. Carrel, "Multilayer Perceptron Model vs Charge Comparison Method for Neutron/Gamma Discrimination in Plastic Scintillator According to Sampling Frequency and Energy Radiation," *IEEE Transactions on Nuclear Science*, pp. 1–1, 2023

Conferences

A. Hachem, A. Kanj, Y. Moline, G. Corre, C. Lynde and F. Carrel, "Neutron/Gamma Discrimination Performance with Plastic Scintillator According to SNR, Vertical Resolution and Sampling Frequency," in *2022 IEEE Nuclear Science Symposium (NSS), Medical Imaging Conference (MIC) and Room Temperature Semiconductor Detector (RTSD) Conference*, forthcoming

A. Hachem, I. Belalchheb, Y. Moline, F. Carrel, and G. Corre, "FPGA Implementation of MLP, 1D-CNN and TTTratio algorithms for Neutron/Gamma-ray Discrimination using Plastic Scintillator," in *2023 IEEE Nordic Circuits and Systems Conference (NorCAS)*, pp. 1–7, 2023

Patents

A. Hachem, Y. Moline, G. Corre, "Form Factor Pulse Discrimination Method in Organic/Inorganic Scintillator," Forth coming (2024)

Bibliography

[1] G. F. Knoll, *Radiation detection and measurement*. John Wiley & Sons, 2010.

[2] J. Julin, "Development of a high energy resolution gas ionization detector for a recoil spectrometer," 2011.

[3] T. R. Gentile, G. L. Jones, A. K. Thompson, J. Barker, C. Glinka, B. Hammouda, and J. Lynn, "Sans polarization analysis with nuclear spin-polarized 3he," *Journal of applied crystallography*, vol. 33, no. 3, pp. 771–774, 2000.

[4] W. Chen, R. Erwin, J. Mciver Iii, S. Watson, C. Fu, T. Gentile, J. Borchers, J. Lynn, and G. L. Jones, "Applications of 3He neutron spin filters at the NCNR," *Physica B: Condensed Matter*, vol. 404, no. 17, pp. 2663–2666, 2009.

[5] M. J. Cieślak, K. A. Gamage, and R. Glover, "Critical review of scintillating crystals for neutron detection," *Crystals*, vol. 9, no. 9, p. 480, 2019.

[6] T. Laplace *et al.*, "Comparative scintillation performance of EJ-309, EJ-276, and a novel organic glass," *Journal of Instrumentation*, vol. 15, no. 11, p. P11020, 2020.

[7] F. Ferrulli, N. Dinar, L. G. Manzano, M. Lablme, and M. Silari, "Characterisation of stilbene and EJ-276 scintillators coupled with a large area sipm array for a fast neutron dose rate detector," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, p. 165566, 2021.

[8] M. Grodzicka-Kobylka, T. Szczesniak, M. Moszyński, K. Brylew, L. Swiderski, J. Valiente-Dobón, P. Schotanus, K. Grodzicki, and H. Trzaskowska, "Fast neutron and gamma ray pulse shape discrimination in EJ-276 and EJ-276G plastic scintillators," *Journal of Instrumentation*, vol. 15, no. 03, p. P03030, 2020.

[9] J. B. Birks, "Scintillations from organic crystals: specific fluorescence and relative response to different radiations," *Proceedings of the Physical Society. Section A*, vol. 64, no. 10, p. 874, 1951.

[10] R. Voltz and G. Laustriat, "Radioluminescence des milieux organiques I. Étude cinétique," *Journal de Physique*, vol. 29, no. 2-3, pp. 159–166, 1968.

[11] C. Dujardin and M. Hamel, "Introduction—Overview on Plastic and Inorganic Scintillators," *Plastic Scintillators: Chemistry and Applications*, pp. 3–33, 2021.

[12] H. Arahmane, E.-M. Hamzaoui, and R. Moursli, "Improving Neutron-Gamma Discrimination with Stilbene Organic Scintillation Detector Using Blind Nonnegative Matrix and Tensor Factorization Methods," *Journal of Spectroscopy*, vol. 2019, pp. 1–9, 05 2019.

[13] C. Fu, A. Di Fulvio, S. Clarke, D. Wentzloff, S. Pozzi, and H. Kim, "Artificial neural network algorithms for pulse shape discrimination and recovery of piled-up pulses in organic scintillators," *Annals of Nuclear Energy*, vol. 120, pp. 410–421, 2018.

[14] L. M. Simms, B. Blair, J. Ruz, R. Wurtz, A. D. Kaplan, and A. Glenn, "Pulse discrimination with a Gaussian mixture model on an FPGA," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 900, pp. 1–7, 2018.

[15] W. Zhang, W. Tongyu, B. Zheng, L. Shiping, Y. Zhang, and Y. Zejie, "A real-time neutron-gamma discriminator based on the support vector machine method for the time-of-flight neutron spectrometer," *Plasma Science and Technology*, vol. 20, no. 4, p. 045601, 2018.

[16] G. Liu, M. Aspinall, X. Ma, and M. Joyce, "An investigation of the digital discrimination of neutrons and γ rays with organic scintillation detectors using an artificial neural network," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 607, no. 3, pp. 620–628, 2009.

[17] K. Zhao, C. Feng, S. Wang, Z. Shen, K. Zhang, and S. Liu, "n Y discrimination for CLYC detector using a one-dimensional Convolutional Neural Network," *Journal of Instrumentation*, vol. 18, no. 01, p. P01021, 2023.

[18] J. Griffiths, S. Kleinegesse, D. Saunders, R. Taylor, and A. Vacheret, "Pulse shape discrimination and exploration of scintillation signals using convolutional neural networks," *Machine Learning: Science and Technology*, vol. 1, no. 4, p. 045022, 2020.

[19] S. Yoon, C. Lee, H. Seo, and H.-D. Kim, "Improved fast neutron detection using CNN-based pulse shape discrimination," *Nuclear Engineering and Technology*, vol. 55, no. 11, pp. 3925–3934, 2023.

[20] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, 1949.

[21] J. B. Simões, J. Cardoso, N. Cruz, and C. M. Correia, "A pc104 multiprocessor dsp system for radiation spectroscopy applications," *P-3004516. Portugal: Instrumentation Center, Physics Dep. University of Coimbra*, 1994.

[22] Y. Moline, M. Thevenin, G. Corre, and M. Paindavoine, "Auto-Adaptive Trigger and Pulse Extraction for Digital Processing in Nuclear Instrumentation," *IEEE Transactions on Nuclear Science*, vol. 62, no. 2, pp. 480–486, 2015.

[23] X. Zhu, C. Feng, Q. Li, Z. Shen, S. Liu, and Q. An, "FPGA-Based Real-Time n/Y Discrimination With Liquid Scintillator," *IEEE Transactions on Nuclear Science*, vol. 65, no. 12, pp. 2877–2882, 2018.

[24] R. T. Schiffer, M. Flaska, S. A. Pozzi, S. Carney, and D. D. Wentzloff, "A scalable FPGA-based digitizing platform for radiation data acquisition," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 652, no. 1, pp. 491–493, 2011.

[25] S. Normand, V. Kondrasov, G. Corre, and C. Passard, "PING : A new approach for nuclear fuel cycle instrumentation," in *2009 1st International Conference on Advancements in Nuclear Instrumentation, Measurement Methods and their Applications*, pp. 1–4, 2009.

[26] L. Barbot, D. Fourmentel, G. De Izarra, C. Destouches, J. Villard, Y. Moline, and H. Hamrita, "MONACO v2: multipurpose and integrated data acquisition system for on-line neutron and gamma measurements," in *EPJ Web of Conferences*, vol. 225, p. 03012, EDP Sciences, 2020.

[27] B. D'Mellow, M. Aspinall, R. Mackin, M. J. Joyce, and A. Peyton, "Digital discrimination of neutrons and γ-rays in liquid scintillators using pulse gradient analysis," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 578, no. 1, pp. 191–197, 2007.

[28] J. Adams and G. White, "A versatile pulse shape discriminator for charged particle separation and its application to fast neutron time-of-flight spectroscopy," *Nuclear Instruments and Methods*, vol. 156, no. 3, pp. 459–476, 1978.

[29] S. Marrone, D. Cano-Ott, N. Colonna, C. Domingo, F. Gramegna, E. Gonzalez, F. Gunsing, M. Heil, F. Käppeler, P. Mastinu, *et al.*, "Pulse shape analysis of liquid scintillators for neutron studies," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 490, no. 1-2, pp. 299–307, 2002.

[30] A. Tomanin, J. Paepen, P. Schillebeeckx, R. Wynants, R. Nolte, and A. Lavietes, "Characterization of a cubic EJ-309 liquid scintillator detector," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 756, pp. 45–54, 2014.

[31] O. McCormack, L. Giacomelli, G. Croci, A. Muraro, G. Gorini, G. Grosso, R. Pasqualotto, E. P. Cippo, M. Rebai, D. Rigamonti, *et al.*, "Characterization and operational stability of EJ276 plastic scintillator-based detector for neutron spectroscopy," *Journal of Instrumentation*, vol. 16, no. 10, p. P10002, 2021.

[32] E. Ryabeva, I. Urupa, E. Lupar, V. Kadilin, A. Skotnikova, Y. Kokorev, and R. Ibragimov, "Calibration of EJ-276 plastic scintillator for neutron–gamma pulse shape discrimination experiments," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 1010, p. 165495, 2021.

[33] M. Grodzicka-Kobylka, T. Szczesniak, M. Moszyński, K. Brylew, L. Swiderski, J. Valiente-Dobón, P. Schotanus, K. Grodzicki, and H. Trzaskowska, "Fast neutron and gamma ray pulse shape discrimination in EJ-276 and EJ-276G plastic scintillators," *Journal of Instrumentation*, vol. 15, no. 03, p. P03030, 2020.

[34] N. Zaitseva, A. Glenn, A. Mabe, M. Carman, C. Hurlbut, J. Inman, and S. Payne, "Recent developments in plastic scintillators with pulse shape discrimination," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 889, pp. 97–104, 2018.

[35] R. W. Engstrom, *Photomultiplier handbook*. RCA Solid State Division. Electro Optics and Devices, 1980.

[36] C. Lynde, E. Montbarbon, M. Hamel, A. Grabowski, C. Frangville, G. H. Bertrand, G. Galli, F. Carrel, V. Schoepff, and Z. El Bitar, "Optimization of the Charge Comparison Method for Multiradiation Field using Various Measurement Systems," *IEEE Transactions on Nuclear Science*, vol. 67, no. 4, pp. 679–687, 2020.

[37] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[38] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1, pp. 278–282, IEEE, 1995.

[39] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.

[40] R. Venkatesan and B. Li, *Convolutional neural networks in visual computing: a concise guide*. CRC Press, 2017.

[41] J. Sen, S. Mehtab, and A. Engelbrecht, *Machine Learning: Algorithms, Models and Applications*. BoD–Books on Demand, 2021.

[42] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN computer science*, vol. 2, no. 3, p. 160, 2021.

[43] Z. Ghahramani, *Unsupervised Learning*, pp. 72–112. Berlin, Heidelberg: Springer Berlin Heidelberg.

[44] X. J. Zhu, "Semi-supervised learning literature survey," tech. rep., University of Wisconsin-Madison Department of Computer Sciences, 2005.

[45] K. P. Lennox, P. Rosenfield, B. Blair, A. Kaplan, J. Ruz, A. Glenn, and R. Wurtz, "Assessing and minimizing contamination in time of flight based validation data," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 870, pp. 30–36, 2017.

[46] R. Winyard, J. Lutkin, and G. McBeth, "Pulse shape discrimination in inorganic and organic scintillators. i," *Nuclear Instruments and Methods*, vol. 95, no. 1, pp. 141–153, 1971.

[47] Wikipedia, "Receiver operating characteristic," 2023. Figure 3: The ROC space for a better and worse classifier.

[48] M. Flaska, M. Faisal, D. D. Wentzloff, and S. A. Pozzi, "Influence of sampling properties of fast-waveform digitizers on neutron- gamma-ray, pulse-shape discrimination for organic scintillation detectors," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 729, pp. 456–462, 2013.

[49] F. Belli, B. Esposito, D. Marocco, and M. Riva, "A study on the pulse height resolution of organic scintillator digitized pulses," *Fusion Engineering and Design*, vol. 88, no. 6-8, pp. 1271–1275, 2013.

[50] E. TECHNOLOGY, "PSD PLASTIC SCINTILATOR EJ-276 and EJ-276G," 2022.

[51] E. Enterprises, "9821b series data sheet," 2022.

[52] B. Carter, "Chapter 10: Op Amp Noise Theory and Applications," *SLOA082*, 2008.

[53] Wikipedia, "Op amp integrator," 2023. Figure 1: Ideal circuit.

[54] CANBERRA, "Model 2111 timing filter amplifier," 2007.

[55] D. Fobar, L. Phillips, A. Wilhelm, and P. Chapman, "Considerations for Training an Artificial Neural Network for Particle Type Identification," *IEEE Transactions on Nuclear Science*, vol. 68, no. 9, pp. 2350–2357, 2021.

[56] M. Aspinall, B. D'Mellow, R. Mackin, M. Joyce, N. Hawkes, D. Thomas, Z. Jarrah, A. Peyton, P. Nolan, and A. Boston, "Verification of the digital discrimination of neutrons and γ rays using pulse gradient analysis by digital measurement of time of flight," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 583, no. 2-3, pp. 432–438, 2007.

[57] A. D. Kaplan, B. Blair, C. Chen, A. Glenn, J. Ruz, and R. Wurtz, "A neutron-gamma pulse shape discrimination method based on pure and mixed sources," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 919, pp. 36–41, 2019.

[58] V. T. Jordanov, "Deconvolution of pulses from a detector-amplifier configuration," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 351, no. 2-3, pp. 592–594, 1994.

[59] X. Luo, V. Modamio, J. Nyberg, J. Valiente-Dobón, Q. Nishada, G. De Angelis, J. Agramunt, F. Egea, M. Erduran, S. Ertürk, *et al.*, "Pulse pile-up identification and reconstruction for liquid scintillator based neutron detectors," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 897, pp. 59–65, 2018.

[60] M. Nakhostin, Z. Podolyak, P. Regan, and P. Walker, "A digital method for separation and reconstruction of pile-up events in germanium detectors," *Review of Scientific Instruments*, vol. 81, no. 10, p. 103507, 2010.

[61] S. Marrone, D. Cano-Ott, N. Colonna, C. Domingo, F. Gramegna, E. Gonzalez, F. Gunsing, M. Heil, F. Käppeler, P. Mastinu, *et al.*, "Pulse shape analysis of liquid scintillators for neutron studies," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 490, no. 1-2, pp. 299–307, 2002.

[62] S. Lee, B. Park, Y. Kim, and H. Myung, "Peak Detection with Pile-Up Rejection Using Multiple-Template Cross-Correlation for MWD (Measurement While Drilling)," in *Robot Intelligence Technology and Applications 3*, pp. 753–758, Springer, 2015.

[63] J. F. Dicello, W. Gross, and U. Kraljevic, "Radiation Quality of Californium-252," *Physics in Medicine and Biology*, vol. 17, pp. 345–355, may 1972.

[64] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, pp. 281–297, Oakland, CA, USA, 1967.

[65] J. G. A. Barbedo, "Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification," *Computers and electronics in agriculture*, vol. 153, pp. 46–53, 2018.

[66] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proceedings of the IEEE international conference on computer vision*, pp. 843–852, 2017.

[67] A. Bailly, C. Blanc, É. Francis, T. Guillotin, F. Jamal, B. Wakim, and P. Roy, "Effects of dataset size and interactions on the prediction performance of logistic regression and deep learning models," *Computer Methods and Programs in Biomedicine*, vol. 213, p. 106504, 2022.

[68] J. Cho, K. Lee, E. Shin, G. Choy, and S. Do, "How much data is needed to train a medical image deep learning system to achieve necessary high accuracy?," *arXiv preprint arXiv:1511.06348*, 2015.

[69] A. F. Agarap, "Deep learning using rectified linear units (relu)," *arXiv preprint arXiv:1803.08375*, 2018.

[70] J. S. Bridle, "Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition," in *Neurocomputing: Algorithms, architectures and applications*, pp. 227–236, Springer, 1990.

[71] J. Han and C. Moraga, "The influence of the sigmoid function parameters on the speed of backpropagation learning," in *International workshop on artificial neural networks*, pp. 195–201, Springer, 1995.

[72] C. Lemaréchal, "Cauchy and the gradient method," *Doc Math Extra*, vol. 251, no. 254, p. 10, 2012.

[73] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[74] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[75] F. J. Harris, *Multirate signal processing for communication systems*. River Publishers, 2021.

[76] S. Butterworth *et al.*, "On the theory of filter amplifiers," *Wireless Engineer*, vol. 7, no. 6, pp. 536–541, 1930.

[77] J. Kaiser and R. Schafer, "On the use of the I0-sinh window for spectrum analysis," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 1, pp. 105–107, 1980.

[78] C. Kim, Y. Kim, M. Moon, and G. Cho, "Iterative Monte Carlo simulation with the Compton kinematics-based GEB in a plastic scintillation detector," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 795, pp. 298–304, 2015.

[79] C. J. Werner, J. S. Bull, C. J. Solomon, F. B. Brown, G. W. McKinney, M. E. Rising, D. A. Dixon, R. L. Martz, H. G. Hughes, L. J. Cox, A. J. Zukaitis, J. C. Armstrong, R. A. Forster, and L. Casswell, "MCNP version 6.2 release notes,"

[80] A. Saeed, T. Ozcelebi, and J. Lukkien, "Multi-task self-supervised learning for human activity detection," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 2, pp. 1–30, 2019.

[81] P. Sarkar and A. Etemad, "Self-supervised learning for ecg-based emotion recognition," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3217–3221, IEEE, 2020.

[82] L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on tabular data?," *arXiv preprint arXiv:2207.08815*, 2022.

[83] A. Ahmad, A. Mustapha, E. D. Zahadi, N. Masah, and N. Y. Yahaya, "Comparison between neural networks against decision tree in improving prediction accuracy for diabetes mellitus," in *Digital Information Processing and Communications: International Conference, ICDIPC 2011, Ostrava, Czech Republic, July 7-9, 2011, Proceedings, Part I*, pp. 537–545, Springer, 2011.

[84] V.-H. Nhu, A. Shirzadi, H. Shahabi, S. K. Singh, N. Al-Ansari, J. J. Clague, A. Jaafari, W. Chen, S. Miraki, J. Dou, *et al.*, "Shallow landslide susceptibility mapping: A comparison between logistic model tree, logistic regression, naïve bayes tree, artificial neural network, and support vector machine algorithms," *International journal of environmental research and public health*, vol. 17, no. 8, p. 2749, 2020.

[85] R. Shwartz-Ziv and A. Armon, "Tabular data: Deep learning is not all you need," *Information Fusion*, vol. 81, pp. 84–90, 2022.

[86] N. Morizet, N. Godin, J. Tang, E. Maillet, M. Fregonese, and B. Normand, "Classification of acoustic emission signals using wavelets and Random Forests: Application to localized corrosion," *Mechanical Systems and Signal Processing*, vol. 70, pp. 1026–1037, 2016.

[87] F. Saki and N. Kehtarnavaz, "Background noise classification using random forest tree classifier for cochlear implant applications," in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 3591–3595, IEEE, 2014.

[88] S. Bose, V. Rama, N. Warangal, and C. R. Rao, "EEG signal analysis for Seizure detection using Discrete Wavelet Transform and Random Forest," in *2017 International Conference on Computer and Applications (ICCA)*, pp. 369–378, IEEE, 2017.

[89] M. Cerrada, G. Zurita, D. Cabrera, R.-V. Sánchez, M. Artés, and C. Li, "Fault diagnosis in spur gears based on genetic algorithm and random forest," *Mechanical Systems and Signal Processing*, vol. 70, pp. 87–103, 2016.

[90] C. Li, R.-V. Sanchez, G. Zurita, M. Cerrada, D. Cabrera, and R. E. Vásquez, "Gearbox fault diagnosis based on deep random forest fusion of acoustic and vibratory signals," *Mechanical Systems and Signal Processing*, vol. 76, pp. 283–293, 2016.

[91] I. Mollas, N. Bassiliades, and G. Tsoumakas, "Conclusive local interpretation rules for random forests," *Data Mining and Knowledge Discovery*, vol. 36, no. 4, pp. 1521–1574, 2022.

[92] javatpoint, 2023. Figure 1: Random Forest Algorithm.

[93] J. C. Quiroz, N. Mariun, M. R. Mehrjou, M. Izadi, N. Misron, and M. A. M. Radzi, "Fault detection of broken rotor bar in LS-PMSM using random forests," *Measurement*, vol. 116, pp. 273–280, 2018.

[94] B.-S. Yang and K. J. Kim, "Application of Dempster–Shafer theory in fault diagnosis of induction motors using vibration and current signals," *Mechanical Systems and Signal Processing*, vol. 20, no. 2, pp. 403–420, 2006.

[95] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal processing magazine*, vol. 13, no. 6, pp. 47–60, 1996.

[96] G. Schwars, "Estimating the dimension of a model," *Annals of statistics*, vol. 6, pp. 461–464, 1978.

[97] H. Bozdogan, "Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions," *Psychometrika*, vol. 52, no. 3, pp. 345–370, 1987.

[98] J. J. Rodríguez-Andina, M. D. Valdés-Peña, and M. J. Moure, "Advanced Features and Industrial Applications of FPGAs—A Review," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 4, pp. 853–864, 2015.

[99] N. Mehendale, A. Aurobindo, and S. Kakatkar, "A Review on Applications of FPGAs," *Available at SSRN 4094931*, 2022.

[100] G. Lacey, G. W. Taylor, and S. Areibi, "Deep learning on FPGAs: Past, present, and future," *arXiv preprint arXiv:1602.04283*, 2016.

[101] M. C. McFarland, A. C. Parker, and R. Camposano, "The high-level synthesis of digital systems," *Proceedings of the IEEE*, vol. 78, no. 2, pp. 301–318, 1990.

[102] G. Martin and G. Smith, "High-level synthesis: Past, present, and future," *IEEE Design & Test of Computers*, vol. 26, no. 4, pp. 18–25, 2009.

[103] AMD, "Vitis hls 2021.2 documentation," 2021.

[104] M. Nagel, M. Fournarakis, R. A. Amjad, Y. Bondarenko, M. Van Baalen, and T. Blankevoort, "A white paper on neural network quantization," *arXiv preprint arXiv:2106.08295*, 2021.

[105] A. Sanaullah, C. Yang, Y. Alexeev, K. Yoshii, and M. C. Herbordt, "Real-time data analysis for medical diagnosis using FPGA-accelerated neural networks," *BMC bioinformatics*, vol. 19, pp. 19–31, 2018.

[106] P. Colangelo, O. Segal, A. Speicher, and M. Margala, "Automl for multilayer perceptron and FPGA co-design," in *2020 IEEE 33rd International System-on-Chip Conference (SOCC)*, pp. 265–266, IEEE, 2020.

[107] I. Westby, X. Yang, T. Liu, and H. Xu, "FPGA acceleration on a multi-layer perceptron neural network for digit recognition," *The Journal of Supercomputing*, vol. 77, no. 12, pp. 14356–14373, 2021.

[108] C. Zhang, G. Sun, Z. Fang, P. Zhou, P. Pan, and J. Cong, "Caffeine: Toward uniformed representation and acceleration for deep convolutional neural networks," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 38, no. 11, pp. 2072–2085, 2018.

[109] K. Guo, L. Sui, J. Qiu, J. Yu, J. Wang, S. Yao, S. Han, Y. Wang, and H. Yang, "Angel-eye: A complete design flow for mapping CNN onto embedded FPGA," *IEEE transactions on computer-aided design of integrated circuits and systems*, vol. 37, no. 1, pp. 35–47, 2017.

[110] X. Wei, C. H. Yu, P. Zhang, Y. Chen, Y. Wang, H. Hu, Y. Liang, and J. Cong, "Automated systolic array architecture synthesis for high throughput CNN inference on FPGAs," in *Proceedings of the 54th Annual Design Automation Conference 2017*, pp. 1–6, 2017.

[111] Y. Shen, M. Ferdman, and P. Milder, "Maximizing CNN accelerator efficiency through resource partitioning," *ACM SIGARCH Computer Architecture News*, vol. 45, no. 2, pp. 535–547, 2017.

[112] A. Aimar, H. Mostafa, E. Calabrese, A. Rios-Navarro, R. Tapiador-Morales, I.-A. Lungu, M. B. Milde, F. Corradi, A. Linares-Barranco, S.-C. Liu, *et al.*, "NullHop: A flexible convolutional neural network accelerator based on sparse representations of feature maps," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 3, pp. 644–656, 2018.

[113] X. Zhang, J. Wang, C. Zhu, Y. Lin, J. Xiong, W.-m. Hwu, and D. Chen, "DNNBuilder: An automated tool for building high-performance DNN hardware accelerators for FPGAs," in *2018 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 1–8, IEEE, 2018.

[114] Y. Guan, H. Liang, N. Xu, W. Wang, S. Shi, X. Chen, G. Sun, W. Zhang, and J. Cong, "FP-DNN: An automated framework for mapping deep neural networks onto FPGAs with RTL-HLS hybrid templates," in *2017 IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*, pp. 152–159, IEEE, 2017.

[115] C. Hao, X. Zhang, Y. Li, S. Huang, J. Xiong, K. Rupnow, W.-m. Hwu, and D. Chen, "FPGA/DNN co-design: An efficient design methodology for IoT intelligence on the edge," in *Proceedings of the 56th Annual Design Automation Conference 2019*, pp. 1–6, 2019.

[116] Y. Liang, L. Lu, Q. Xiao, and S. Yan, "Evaluating fast algorithms for convolutional neural networks on FPGAs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 4, pp. 857–870, 2019.

[117] W. Wang, X. Zhao, and D. Liu, "Design and Optimization of 1D-CNN for Spectrum Recognition of Underwater Targets," *Integrated Ferroelectrics*, vol. 218, no. 1, pp. 164–179, 2021.

[118] J. Xu, S. Li, J. Jiang, and Y. Dou, "A simplified speaker recognition system based on FPGA platform," *IEEE Access*, vol. 8, pp. 1507–1516, 2019.

[119] A. Hachem, Y. Moline, G. Corre, B. Ouni, M. Trocme, A. Elayeb, and F. Carrel, "Labeling strategy to improve neutron/gamma discrimination with organic scintillator," *Nuclear Engineering and Technology*, 2023.

[120] A. Hachem, Y. Moline, G. Corre, J. Gauthier, and F. Carrel, "Multilayer Perceptron Model vs Charge Comparison Method for Neutron/Gamma Discrimination in Plastic Scintillator According to Sampling Frequency and Energy Radiation," *IEEE Transactions on Nuclear Science*, pp. 1–1, 2023.

[121] A. Hachem, A. Kanj, Y. Moline, G. Corre, C. Lynde and F. Carrel, "Neutron/Gamma Discrimination Performance with Plastic Scintillator According to SNR, Vertical Resolution and Sampling Frequency," in *2022 IEEE Nuclear Science Symposium (NSS), Medical Imaging Conference (MIC) and Room Temperature Semiconductor Detector (RTSD) Conference*, forthcoming.

[122] A. Hachem, I. Belalchheb, Y. Moline, F. Carrel, and G. Corre, "FPGA Implementation of MLP, 1D-CNN and TTTratio algorithms for Neutron/Gamma-ray Discrimination using Plastic Scintillator," in *2023 IEEE Nordic Circuits and Systems Conference (NorCAS)*, pp. 1–7, 2023.

[123] A. Hachem, Y. Moline, G. Corre, "Form Factor Pulse Discrimination Method in Organic/Inorganic Scintillator," Forth coming (2024).