UNIVERSITAT DE BARCELONA

FUNDAMENTAL PRINCIPLES OF DATA SCIENCE MASTER'S THESIS

Machine Learning Particle Classifier for Water Cherenkov Detectors (Hyper-Kamiokande neutrino experiment)

Author:
Iñaki ERREGUE

Supervisors:
Dra M. Pilar CASADO
Dr Sergio ESCALERA

A thesis submitted in partial fulfillment of the requirements for the degree of MSc in Fundamental Principles of Data Science

in the

Facultat de Matemàtiques i Informàtica

June 30, 2022

UNIVERSITAT DE BARCELONA

Abstract

Facultat de Matemàtiques i Informàtica

MSc in Fundamental Principles of Data Science

Machine Learning Particle Classifier for Water Cherenkov Detectors (Hyper-Kamiokande neutrino experiment)

by Iñaki ERREGUE

Neutrinos are one of the most mysterious elementary particles known. The Hyper-Kamiokande neutrino experiment, still under construction, intends to incorporate machine learning models into its reconstruction and analysis software to improve performance and alleviate computational complexity. This thesis focuses on the development of a convolutional neural network that will provide a better distinction of electron-like events from background reactions in water Cherenkov detectors, with a 14% relative increase in the electron signal efficiency with respect to the baseline. Going further than previous investigations, in addition to the charge deposited on the photomultipliers, the relative times of detection are taken into account. Additionally, a method capable of quantifying the confidence with which the predictions are made is presented for the first time, as well as a complete study of the biases of the model, improving dependence with the different energy ranges, directions of incidence and vertex positions.

Acknowledgements

First of all, I would like to thank my supervisor, Dr M. Pilar Casado, for introducing me to this project. I would also like to express my deepest gratitude to the project advisor Dr Sergio Escalera for supervising me on the technical side and offering his critical thinking. Thanks should also go to the IFAE's neutrino team and the WatChMaL members for showing such enthusiasm for this project and helping me out. Finally, I would like to mention the support my mother has given me, not only in this project but also throughout my entire academic career.

Contents

List of Figures

List of Tables

List of Abbreviations

AUC	**A**rea **U**nder the **C**urve
CNN	**C**onvolutional **N**eural **N**etwork
HK	**H**yper-**K**amiokande
IWCD	**I**ntermediate **W**ater **C**herenkov **D**etector
MCD	**M**onte **C**arlo **D**ropout
ML	**M**achine **L**earning
mPMT	**m**ulti-**PMT** Optical Module
PMT	Photo**m**ultiplier **Tube**
T2K	**T**okai **to** **K**amioka
WCD	**W**ater **C**herenkov **D**etector

Chapter 1

Introduction

Neutrinos are a type of subatomic elementary particle that have a ridiculously small mass and no electric charge, hence their name. With these characteristics, they rarely interact with matter despite being extremely abundant. Their physical properties pose several challenges in current particle physics, both theoretically and experimentally. Currently, one of the most successful ways to detect these particles is by employing Water Cherenkov Detectors (WCDs), tanks filled with water whose walls are surrounded by light sensors. Neutrinos are detected indirectly as they pass through the tank since there is a small probability that they will interact with the matter inside. Given their small mass, neutrinos travel at almost the speed of light. Therefore, when they interact with matter, electrically charged particles are produced, propagating at a speed faster than that of light in the medium. This fact causes an effect called Cherenkov radiation, which results in the emission of a light ring that is used to reconstruct the event and obtain information about the neutrino that originated it.

Hyper-Kamiokande (HK) is one of the next-generation neutrino experiments. Its construction is expected to be completed in 2027, equipped with two WCDs. Given the size of the tanks and the large number of sensors, the HK software will feature machine learning (ML) models that will facilitate the study of neutrinos and their interactions. In particular, the WatChMaL organization (*Water Cherenkov Machine Learning Organization* 2019) focuses on the development of such algorithms. This project inherits part of the code, data and methodology used by the organization.

Currently, one of the most developed approaches used in WatChMaL to reconstruct and classify events with a single ring is based on the use of convolutional neural networks (CNNs). The signal detected by the sensors around the surface of the tank is mapped into a 2D image and then sent to the classifier. Despite being able to distinguish muon-like events from electron-like events with higher accuracy than fiTQun, the traditional reconstruction software (Missert, 2017), both methods fail to tell apart electron-like events from background reactions. In some neutrino reactions, high-energy photons capable of producing a pair of electron–positron particles are generated. Both particles in this pair production are constrained to be emitted in nearly the same direction, and if having enough energy, they can generate a couple of overlapped rings of Cerenkov radiation, considered as background.

This project[1] focuses on the distinction between electron events and background reactions (gamma events), using not only the intensity of charge deposited on the

[1] All the code developed for this project can be found in the following GitHub repository.

detectors located on the walls but also the relative times of such detections. After an exploratory analysis, it has been observed that time information is not a differential feature in this particular problem given its correlation with charge information. However, its inclusion can produce a modest improvement in the performance of the model. Different ways of encoding this temporal information have been explored, from treating it in a charge-analogous way as just another channel in the image to using it in scaling techniques or ordering chronologically the intensity of the pixels. Since the HK experiment is still under construction, no real data is available. To conduct this investigation, artificial data was generated using a simulation software called WCSim (*The WCSim GEANT4 application* 2021).

In particle physics, the association of uncertainty to the obtained results is of crucial importance. However, the estimation of uncertainty in predictions is still an open problem in the field of ML. In addition to using the time to decouple electron events from those of gamma type, this project includes a study on the epistemic uncertainty linked to the proposed model based on the Monte Carlo dropout (MCD) technique. This simple Bayesian technique employs the use of dropout in the testing phase to make approximate inference on the posterior distribution of the network weights. For this reason, an examination of the use of dropout in both training and testing has been carried out. Subsequently, the assignment of the most discriminative uncertainty measure between correct and incorrect predictions has been addressed.

Finally, the performance of the model has been compared against the default configuration used in the WatChMaL organization. The WCSim simulator also provides true particle tracking information. Therefore, in the last section a study of the biases of the model with respect to the energy of the particle, and its position and direction inside the tank is carried out.

Chapter 2

Project Fundamentals

This chapter gives a brief introduction to particle physics to understand the motivation for using Water Cherenkov Detectors. Secondly, technical information on the new detectors designed for the Hyper-Kamiokande experiment is detailed. Finally, the use of machine learning in the field of particle physics is reviewed, along with basic concepts such as neural networks, dropout and variational inference.

2.1 Particle Physics and Neutrinos

Modern physics discovered that there are more particles beyond the well-known protons, neutrons and electrons from which atoms are built. Some of these particles are considered to be fundamental, in the sense that they cannot be broken down into any smaller bits. These fundamental particles and their interactions are almost completely described by the Standard Model of particle physics. Among these, one can find the so-called neutrino.

Neutrinos belong to the lepton family, a group of particles which also includes the electron (e) and his more massive brothers, muon (μ) and tau (τ). In fact, for each electron-like particle there exists an associated neutrino: electron-neutrino (ν_e), muon-neutrino (ν_μ) and tau-neutrino (ν_τ), this property is called flavour. Unlike these three electron-like particles, neutrinos have no electric charge and their masses haven't been measured precisely by any experiment but are considered to be very small.

Of all massive particles, neutrinos are the most abundant in nature. Nonetheless, due to their lightness and null electric and color[1] charge, they only interact via weak forces and gravity, making it difficult to detect them. Furthermore, for each particle there exists an antiparticle with identical mass but opposite in every other physical property. Thus, there also exists a trio of anti-neutrinos ($\overline{\nu}_e$, $\overline{\nu}_\mu$, $\overline{\nu}_\tau$). Along with the mystery of their light masses and the possibility of being their own antiparticle, neutrinos show an even weirder behaviour, they oscillate. Neutrino oscillations refer to the transformation of a neutrino from one type of flavour when produced to another when detected.

Neutrinos are produced by many physical processes like radioactive nuclear decay, particle decay or nuclear reactions. One way of obtaining neutrino beams is using particle accelerators: protons are accelerated to high energies and then forced

[1] Color charge is a property of quarks and gluons, which are the building blocks of protons and neutrons. Only "coloured" particles interact via strong nuclear force.

to collide with a target. This collision produces among others charged pions (π), a type of particle that is unstable and commonly decays into a muon and his associated neutrino. By putting a barrier to the resulting beam, muons are absorbed and only the neutrino component survives, making it a nearly pure narrow beam of muon-neutrinos. The produced beam is directed at a distant detector and its energy can be tuned.

$$\pi^+ \rightarrow \mu^+ + \nu_\mu\,, \quad \pi^- \rightarrow \mu^- + \overline{\nu}_\mu \tag{2.1}$$

2.2 Neutrino detection

Long-baseline neutrino experiments usually incorporate at least two detectors. The obtained neutrino beam spreads out as it travels like the light produced by a torch. That is why near detectors focus on characterizing the beam and neutrino interactions with other particles. On the other hand, far detectors focus on neutrino oscillations and the interaction with matter during their propagation.

Neutrinos are detected through their interactions with matter via weak nuclear force. In most high-energy interactions of neutrinos, high energy charged particles are produced. If these charged particles surpass the speed of light in the medium where they are propagating, they emit Cherenkov radiation (bluish light) along their path. At each point light is produced in an expanding ring from its point of production, like the surface of a cone, the optical equivalent to the sound barrier.

WCDs are essentially large cylindrical tanks full of pure water, whose volume is surrounded with photomultiplier tubes[2] (PMTs) that detect light from Cherenkov radiation. Thus, each PMT can measure the number of incident photons/charge and the relative time when they were detected. By arranging the signals of the different PMTs in an image, ring patterns can be discerned. The number of rings is equal to the number of emitted particles, and their shape and thickness is determined by the type of particle and its path. The total charge detected in a ring is proportional to the energy of the particle that produced it.

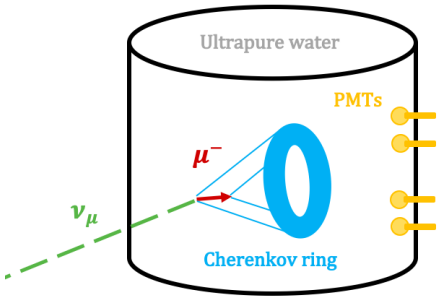

FIGURE 2.1: WCD scheme.

[2] A photomultiplier is a device that converts incident photons into an electrical signal.

2.3 Hyper-Kamiokande and IWCD

T2K is one of the most renowned neutrino experiments. It is conducted in Japan and its results have played a decisive role in the proof of neutrino oscillations, which was recognized by the Nobel committee in 2015. The T2K experiment is equipped with a near detector close to the neutrino beam source, the J-PARC accelerator (Tokai), and a long-baseline detector, the Super-Kamiokande (Kamioka), located at a distance of 295 km. The successor of the T2K experiment is the HK experiment, which will count with an upgraded system of the currently used accelerator, producing a more powerful beam, and new near detector. The new long-baseline detector, the HK, will be used in conjunction with an intermediate detector IWCD, based on the NuPrism proposal (Bhadra et al., 2014), that will minimise systematic errors in the oscillation analysis. The experiment is planned to start operation in 2027.

The IWCD will be located at a distance of 0.7–2.0 km from the J-PARC beam line (Abe et al., 2018). Unlike ordinary WCD tanks, the IWCD will consist in a 50 m deep vertical pit with a diameter of 10 m. Inside the pit, an 8 m tall structure will be equipped with inward-facing PMTs. A crane system will move the detector structure vertically along the pit to make measurements at different off-axis angles.

Instead of using ordinary PMTs as photodetection system, the IWCD will be equipped with mPMTs, which are clusters containing smaller but faster PMTs. Each module has a diameter of 50 cm and integrates 19 PMTs, each with a diameter of 7.7 cm and different orientation. The IWCD is expected to have 536 mPMTs distributed around the tank, increasing the granularity and therefore providing an enhanced event reconstruction.

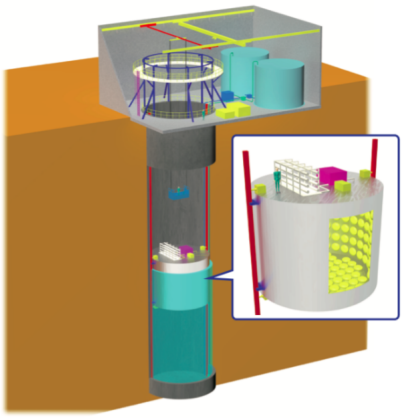

FIGURE 2.2: Schematic of the IWCD detector (*T2K and beyond* 2021).

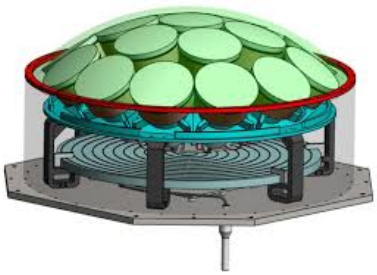

FIGURE 2.3: Multiple PMT design for the Hyper-K project (De Rosa, 2020).

2.4 Machine Learning and Particle Physics

One of the biggest challenges in particle physics nowadays is to be able to cope with the large volume of data produced by the different latest generation detectors.

Probing the Standard Model with increasing precision and the search for new particles require the identification of rare signals in immense backgrounds. In some areas, traditional software is beginning to reach the limit of achievable precision and reasonable computational complexity (Albertsson et al., 2018).

On the other hand, ML techniques have proven to have a great capacity for identifying patterns from diverse data structures and big data. Its use in particle physics research has been present since the early 90s, and ML models are already state-of-the-art in tasks such as event and particle identification or energy estimation. Moreover, once trained, these models use significantly fewer computational resources.

In plain words, ML is a branch of artificial intelligence which focuses on the use of data and algorithms to mimic the way humans learn. These algorithms are mainly trained to make classifications or predictions. We can identify 4 basic ingredients: data, model, loss and optimizer.

In supervised learning, the data used for training the algorithm has been labelled for a particular output. Thus, data can be thought of as a set of samples with some features X and labels t. The model is just a function $f(x;w)$ that given an input sample x and some internal parameters w, outputs a label y. The goal is to learn an optimal set of parameters w such that the difference between the predicted value y and the ground truth target t is minimum. The dissimilarity between the output of the model and the true value to predict is modelled by the loss function $\mathcal{L}(y,t)$. Therefore, the learning process of a ML algorithm is translated to the optimization problem in Equation 2.2. The optimizer is defined as the numerical method used to find the minimum of the loss function.

$$\min_{w} \mathcal{L}(f(x;w),t)\,. \tag{2.2}$$

In particular, WatChMaL is an organization whose main goal is to facilitate the development of ML algorithms for event reconstruction in water Cherenkov detectors, including HK's far detector and IWCD. For both experiments, initial studies have begun to explore particle type classification using various deep learning architectures such as CNNs and graph neural networks (Prouse, 2021).

2.5 Neural Networks, CNNs and ResNet

Neural Networks are the heart of deep learning algorithms and consist in as set of layers with nodes. A node is just a place where computation happens and its architecture and functionality tries to mimic the one of a neuron cell. The i-th node in the j-th layer simple combines the input data $\mathbf{x}^{(j-1)}$ with a set of internal coefficients, $\mathbf{w}_i^j$ and b_i^j, by computing the scalar product between them and sending the result through a non-linear function φ, called activation function. Thus, the neurons output is just:

$$h(\mathbf{x}^{(j-1)}) = \varphi((\mathbf{w}_i^j)^T\mathbf{x}^{(j-1)} + b_i^j) = x_i^j\,. \tag{2.3}$$

This operation is performed for all the nodes that compose the layer j in the form of matrix multiplication. Thus, the output of the j-th layer is $\mathbf{x}^j$ and serves as an input to the following layer $(j+1)$. Therefore, a neural network is just a composition of functions with some weight parameters (Figure 2.4). Moreover, the Universal

Approximation Theorem states that neural networks are universal function approximators (Hornik, Stinchcombe, and White, 1989). That is, neural networks can represent a wide variety of functions when given appropriate weights.

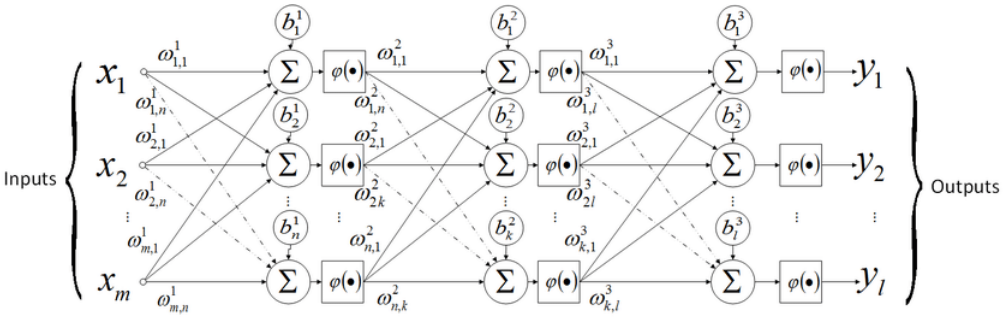

FIGURE 2.4: Generic multilayer feedforward neural network (De Oliveira et al., 2017).

At first, optimizing such a complex model might seem tedious and computationally expensive. Nonetheless, automatic differentiation deals with partial derivatives in a programatically optimum way. The idea behind this method is to create a computational graph and take advantage of this forward pass through the network to readily compute differentials as we parse the graph. Thus, computing the partial derivative of the loss function with respect to a given parameter reduces to a multiplication of the intermediate partial derivatives involved in the path to get to the correspondent node, in other words, applying the chain rule. In conclusion, the learning process can be interpreted as a forward pass of the data through the network to compute the loss function, and backpropagation to compute the gradient of the loss with respect to the weights. These gradients are used to update the value of the internal parameters.

The WatChMaL organization has placed emphasis primarily on the development of CNNs, a type of neural network has been a breakthrough in image and video processing. The prevalence of CNNs in the field of computer vision is mainly due to their performance and computational efficiency when processing 2D and 3D arrays containing pixel intensities. Inspired by the classic notions of cells in visual neuroscience, these operations take advantage of spatial relationships using a smaller number of parameters in comparison with their fully connected neural network counterpart (LeCun, Bengio, and Hinton, 2015).

A CNN typically integrates three types of layers: convolution, pooling and fully connected layers. The convolution layer performs a dot product between two matrices: the restricted portion of the receptive field (a tensor of data) and the kernel (set of learnable parameters). During the forward pass, the kernel slides across the height and width of the image, producing a new representation of that receptive region. The pooling layer simply replaces the output of the network at certain locations by computing statistics of neighbouring pixels, like the maximum, helping to reduce the size of the representation. At the end of the architecture, the image representation is flattened into a vector which is sent to a fully connected neural network.

In particular, CNNs based on residual neural networks, like ResNet, have been developed to perform event classification. The neural network depth is of crucial importance for its performance. Nonetheless, one of the consequences of stacking more and more layers, apart from the notorious problem of vanishing/exploding gradients (Bengio, Simard, and Frasconi, 1994), is the degradation of the convergence. With the network depth increasing, accuracy gets saturated and then degrades rapidly. This issue was first addressed by the use of residual neural networks (He et al., 2015) that use shortcuts to jump over some layers, providing a faster path for the gradient to pass through via additive identity transformations.

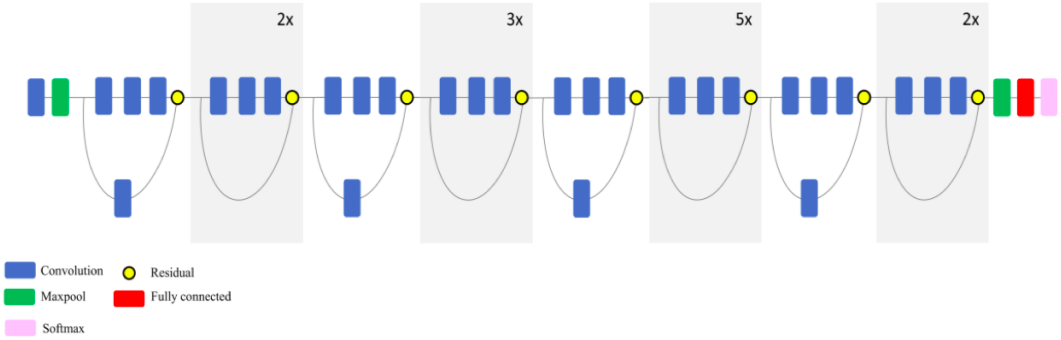

FIGURE 2.5: Schematic diagram of a ResNet model in a compressed view (Mahdianpari et al., 2018).

2.6 Model uncertainty

In particle physics experiments, being able to associate an uncertainty measure to the quantitative results obtained is of crucial importance for assessing their quality. Despite being able to obtain high accuracies in a wide variety of applications, uncertainty estimation in the field of ML is still an open problem. Nonetheless, Bayesian modelling provides a framework for capturing model uncertainty by putting distributions around model parameters and inferring posterior distributions given the data. Neural networks using this approach are called Bayesian neural networks. While having a valid theoretical foundation, their use is computationally prohibiting as the number of layers and the complexity of these layers increase. The MCD technique is a popular and conceptually easy approach to assess uncertainty (Gal and Ghahramani, 2016) .

2.6.1 Dropout

Before delving into MCD, it is necessary to review the concept of dropout, which refers to ignoring neurons in the training phase. Formally, some of the elements of the input tensor are randomly zeroed with probability p, using samples from a Bernoulli distribution. Especially in fully connected layers, while training, most neurons develop co-dependency amongst each other, curbing their individual power. Thus, dropout can be thought of as a regularization technique that helps the network to build better representations of the data and not just memorise it, preventing over-fitting and improving generalization (Hinton et al., 2012).

2.6.2 Variational Inference

The goal of Bayesian inference in this case is to model the posterior probability distribution of the network weights by using two factors: a prior probability distribution of the network weights $P(W)$, and a likelihood function derived from a statistical model on the observed data $P(D|W)$. By using the Bayes theorem (Equation 2.4) we update the initial knowledge of parameters W by looking at the evidence D.

$$P(W|D) = \frac{P(D|W)P(W)}{P(D)} \tag{2.4}$$

$$P(D) = \int P(D|W)P(W)dW \tag{2.5}$$

More formally, we would like to model a new prediction y^* given some new sample x^* with a given set of parameters W obtained with a dataset D, the so called posterior predictive distribution, which has the following expression:

$$P(y^*|x^*, D) = \int P(y^*|x^*, W)P(W|D)dW\,. \tag{2.6}$$

The problem lies in that the integral for $P(D)$ (Equation 2.5) is computationally feasible for very few neural network architectures (Lee et al., 2017) and consequently the integral for $P(y^*|x^*, D)$ as well.

In order to tackle this problem one of the most common approximations is variational inference. Instead of sticking with $P(W|D)$ in Equation 2.4, we approximate it with a family of distributions $q_\theta(W)$ where inference is simpler. The distribution is parametrized by θ and its values are chosen such that it resembles the original one, minizing the Kullbeck-Leibler divergence and maximizing the evidence lower bound.

2.6.3 Monte Carlo Dropout

MCD is a simple method for capturing model uncertainty based on the interpretation of dropout as an approximate inference of the posterior distribution of the network weights and can be explained as variant of variational inference (Seoh, 2020). Dropout is commonly used during training and switched off in testing. However, MCD keeps the dropout activated when predicting with the already trained model. By making multiple forward passes of a given data point, the obtained predictions will be different each pass since different nodes are zeroed every time. These results can be interpreted as samples from the posterior distribution and therefore uncertainty measurements can be modelled.

Chapter 3

Methodology

This chapter is devoted to detailing the process followed to integrate the temporal information into the model. Topics such as dataset and image formation are reviewed for a better understanding of the methodology. Next, the concept of dropout in CNNs is addressed. Finally, different measures to quantify uncertainty with the MCD method are proposed along with a method to choose the most discriminatory.

3.1 The Dataset

For both detectors IWCD and HK, construction has not yet been completed and therefore there is no real data available. However, the simulation software WCSim[1] was used to provide a consistent dataset, configuring the desired geometry of the detector as well as simulating all interactions within the detector volume, until photons hit the PMTs. Then, the response of the PMTs is simulated along with the electronics, triggering and data acquisition. The data produced by WCSim are digitised PMT hits that include the time and charge of the hit. Additionally, the software provides some true particle tracking information: the type of particle and its initial position, angle of direction and energy.

The dataset used for this project is stored in an h5 format, allowing a fast loading of samples when training. It contains a total of 2,950,284 events including both electron and photon events. The two classes are balanced, 49.9% and 50.1% of representation for electrons and photons respectively. The position of the initial particles and their direction are distributed uniformly inside the detector volume, as well as the energy, ranging up to 1 GeV. More information regarding event and hit distributions can be found in the Appendix A. The splitting selected is approximately: 85% of the events for training, 5% for validation and 10% for testing.

3.2 From Data to Images

The geometry of the detector is cylindrical and mPMTs are distributed over the entire surface (Figure 3.1). Thus, dealing with 3D images of the detector implies a huge computational inefficiency given the cost of dealing with three-dimensional kernels and the sparsity of the data (no mPMTs in the inner volume). On the contrary, the cylinder tank is unwrapped into a flat rectangular 40x29 image (Figure 3.2). Each mPMT is mapped into a pixel, the barrel section is 40x9 and contains 360 mPMTs

[1]The WCSim simulation software is based on Geant4 (Agostinelli et al., 2003), a toolkit for simulating the passage of particles through matter using Monte Carlo methods.

and caps are distributed in a circular-like shape containing 88 mPMTs each, totalling 536 meaningful pixels, the rest are zeroed. Due to the fact that each mPMT contains 19 mPMT, each pixel has 19 charge features, forming a final image of dimension 40x29x19.

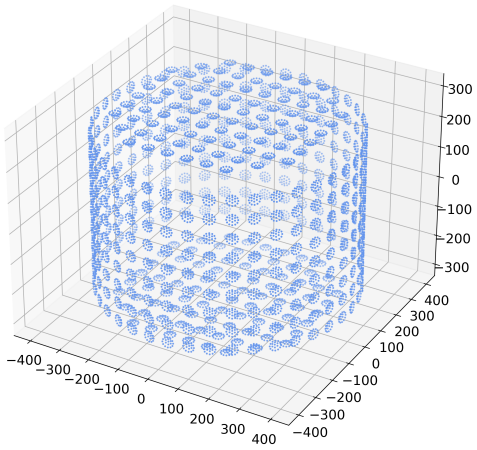

FIGURE 3.1: PMTs distributed over the detector surface.

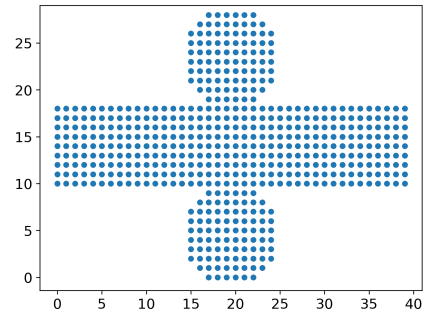

FIGURE 3.2: Position of mPMTs in the 40x29 pixel image.

Once the image is built, several transformations can be applied: horizontal flip, vertical flip and front back reflection. Originally during the training process, every sample could undergo one of these transformations at each epoch. Furthermore, events are subject to a padding algorithm that duplicates the part of the image where caps and barrel are connected in order to use the space more efficiently (Figure 3.3).

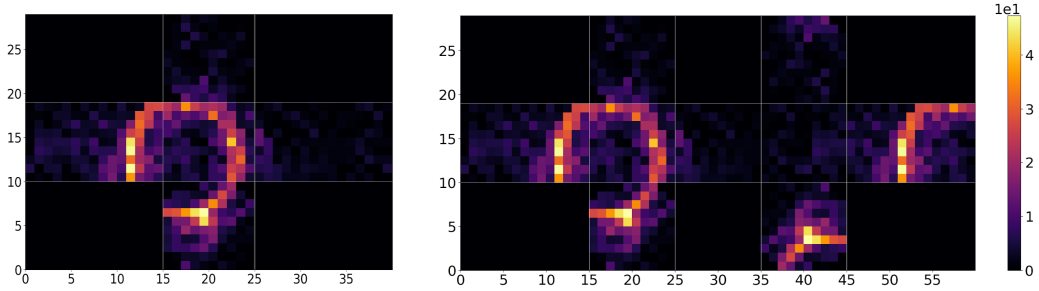

FIGURE 3.3: Sum of charge per mPMT for an electron event before (left) and after padding (right).

3.3 Time information

One of the main objectives of this project is to explore the incorporation of hit time information to boost the performance of the model. So far, this type of tasks only included charge information. Nonetheless, along with the charge intensity detected by every PMT there is information regarding the relative time of detection of the hits. This information could help to distinguish better between the event and background noise. Moreover, it could improve the classification by including the small differences in the time domain between the two types of rings. In this section

several ways of including the time information are discussed.

A small aside regarding PMTs is the fact that these detectors convert an incident photon into an electric signal. Thus, there is an associated waveform that provides more detailed information about the hit. Its proper characterization and encoding can lead to considerable improvements (Valls, Lux, and Sanchez, 2022) by denoising the detector's signals. Nonetheless, in the provided dataset these PMTs signals are characterized by just two values: the maximum amplitude (the charge) and the relative time when detected.

3.3.1 Time as a channel

A simple glance at the distributions of some time statistics representing the events (Figure A.1) is enough to notice that raw time information is not a powerful class differentiating feature. However, when using time in conjunction with spatial information of the detector, events can be visually discerned even in the time domain (Figure 3.4). Thus, the first approach considered is to include hit detection times as another pixel intensity, treating time and charge equally.

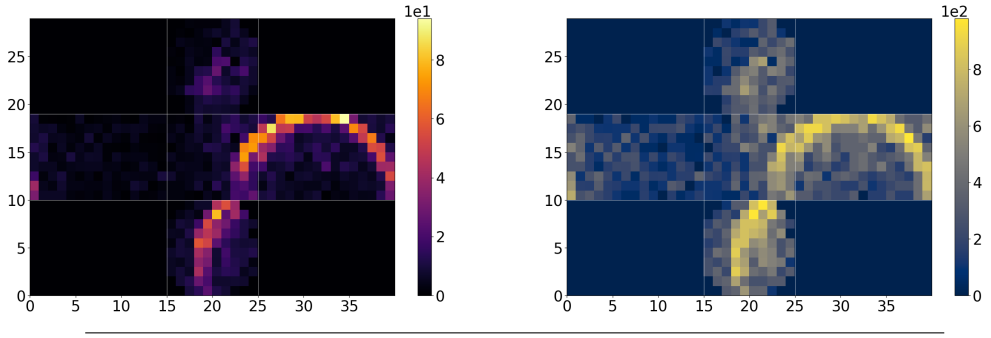

FIGURE 3.4: Sum of charge per mPMT (left) and average time detection per mPMT (right) for a gamma event.

Since we are considering two different sources of information with unlike ranges of numerical values when building the channels of the image, it is common sense to perform feature scaling. A traditional approach when working with RGB images is considering maximum pixel intensity normalization $\tilde{x} = x/x_{max}$, since the maximum is known to be 255, scaling the input to $[0, 1]$. However, our input data are not RGB images and the maximum pixel intensity in the dataset is found at the end of a heavy-tailed distribution, which would cause the majority of values to be concentrated in zero. Another approach is performing hit standardization $\tilde{x}_i = (x_i - \mu_i)/\sigma_i$, for $i = \{\text{charge, time}\}$, which centers the input values around 0 with unitary standard deviation (Figure 3.5). Therefore, when the input data is multiplied by weight values, their activation remains on scale 1 so they do not saturate fast, avoiding near zero gradients in early stages.

3.3.2 Alternatives

An alternative for encoding both temporal and charge data is to consider an aggregated representation at the mPMT level. Instead of using 19 channels per each type of information, we can use only 2: the mean of the mPMT intensities and its standard deviation (Figure 3.6).

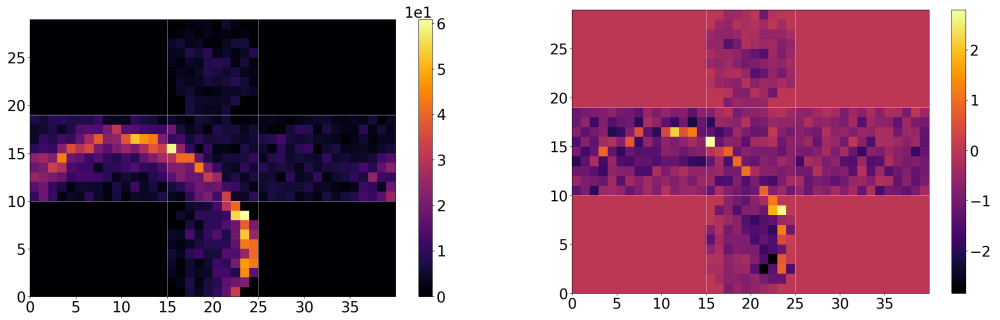

FIGURE 3.5: Sum of charge per mPMT before (left) and after hit standardization (right) for a gamma event.

Furthermore, detection time can also be used to order the pixel charge intensities chronologically (e.g a mPMT with 4 PMTs records charges $q = (2, 8, 5, 1)$ at relative time $t = (0.2, 0.9, 0.5, 0.8)$, the pixel features chronologically ordered are: $\tilde{q} = (2, 5, 1, 8)$). Another approach to incorporate time is the use of a custom charge hit normalization that takes into account hit recording time explicitly:

$$\tilde{q} = \frac{q}{1 + \frac{|t-\mu_t|}{\sigma_t}} \,. \tag{3.1}$$

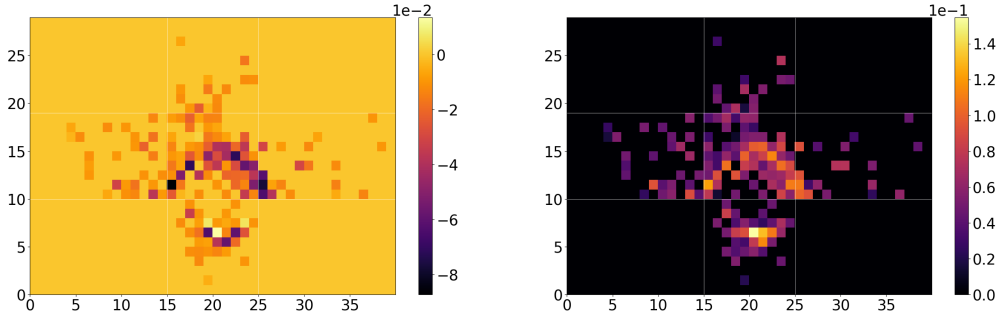

FIGURE 3.6: Mean of normalized charge per mPMT (left) and the corresponding standard deviation (right) for a gamma event.

3.4 Dropout

In the original model used by the WatChMaL organization, dropout is not implemented. However, it is a key element in the estimation of epistemic uncertainty using the MCD method. When it comes to implementing dropout in a ResNet-like architecture, several options can be considered. Drop-neuron is the standard dropout, applied to each input neuron. Despite being a good regularizer in dense layers, this method fails in CNNs due to the spatial correlations of pixels. On the other hand, drop-channel refers to randomly zeroing out an entire channel with probability p, promoting independence between feature maps (Tompson et al., 2014). The conventional use of drop-neuron and drop-channel generally fails to produce a performance improvement in CNNs due to a stochasticity conflict with the following layer, batch normalization. To mitigate this issue, (Cai et al., 2019)

propose placing dropout operations right before the convolution layer instead (Figure 3.7). Before delving into the MCD method and assessing uncertainty, different dropout options have been explored in training.

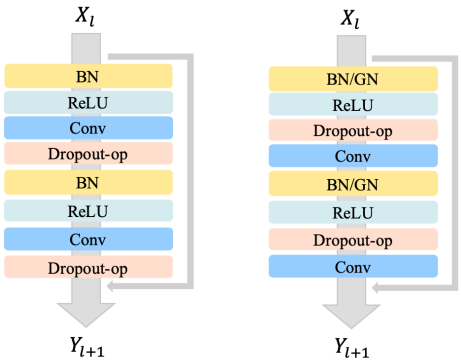

FIGURE 3.7: Traditional (left) and alternative (right) dropout layer position in a ResNet basic block (Tompson et al., 2014).

3.5 Uncertainty

Once the model is trained, we can estimate the uncertainty in the predictions for the test set. After performing T forward passes of a given event, we obtain a set of probabilities $\{p_t\}_{t=1}^{T}$, where p indicates the probability of being an electron event. These values can be interpreted as samples from the posterior predictive distribution. The model estimation for the event can be computed as the mean of all predictions performed, p^*. To measure how reliable a model makes predictions, several uncertainty measures can be used to quantify it apart from the variance (Milanés-Hermosilla et al., 2021).

The predictive entropy (Equation 3.2) captures the average of the amount of information contained in the predictive distribution and attains its maximum value, $H = \log 2$, when both classes are predicted to have an equal uniform probability and therefore a more uncertain prediction.

$$H = p^* \log(p^*) + (1 - p^*) \log(1 - p^*) \tag{3.2}$$

The mutual information (Equation 3.3) measures mutual dependence between samples, capturing the model's confidence from its output.

$$I = H - \frac{1}{T} \sum_{t=1}^{T} [p_t \log(p_t) + (1 - p_t) \log(1 - p_t)] \tag{3.3}$$

The margin of confidence (Equation 3.4) is probably the most intuitive measurement proposed yet. It is computed taking into account the difference in probabilities between the two most probable classes. Since our problem is binary, d_t is just the difference between the two outputted probabilities for a single forward pass t.

$$M = \frac{1}{T} \sum_{t=1}^{T} d_t \tag{3.4}$$

The objective of this section is to assign to the model's predictions that measure of uncertainty or confidence that best discriminates between correct and incorrect classifications. In order to do so, the considered similarity measurement is the Bhattacharyya distance (Equation 3.5), which computes the amount of overlap between two statistical populations, $\mathbf{p} = \{p_i\}$ and $\mathbf{q} = \{q_i\}$.

$$D_B(\mathbf{p}, \mathbf{q}) = -\ln\left(\sum_i \sqrt{p_i q_i}\right) \tag{3.5}$$

Chapter 4

Results and Discussion

This chapter presents the main results obtained from the inclusion of the temporal information to the evaluation of the biases in the final model, including data augmentation and dropout experiments. All experiments were carried out in a machine with a single GPU, NVIDIA GeForce RTX 2080 Ti. Configuration is handled by Hydra, a framework that enables to compose and override configuration from the command line and configuration files, helping in the development process. Some of the performance metrics used are the accuracy, the F1-score and the area under the curve (AUC). The electron signal efficiency of the results, namely the true positive rate, has also been studied as well as the background rejection, which is closely related to the false positive rate. In the final stages, not only performance metrics are taken into account, but also the biases concerning the metadata and the uncertainty associated with the predictions.

4.1 Time information

All the models presented in this section were trained for 20 epochs using a batch size of 512 events. The selected optimizer was Adam, with a learning rate of 1×10^{-4} and null weight decay. Despite setting the same seed for all models, only one possible transformation per sample has been allowed at the beginning of each epoch in order to avoid overfitting. The first set of experiments was focused on identifying the best way to include temporal information as channel in the model. Different model combinations have been explored concerning the type of data included (charge and/or time), the scaling technique (hit standardization), and the mPMT aggregation. For 20 epochs, the training process took about 9 hours per model. Results in Table 4.1 show that including time as a channel generally boosts the performance of the model.

Given the similarity in all the metrics for the top three models, it was decided to repeat these experiments two more times with different seeds, and thus reducing the stochasticity of the results (see Table 4.2). The best model, selected by combining all metrics[1] and its low variance, uses a hybrid combination of 19 channels per charge and a mPMT aggregation for the hit time data. It is interesting to note that when combining charge and time with this dataset, the architecture seems to prefer fully aggregated data over the standard 19-channel configuration, and also hit standardization over unscaled data.

[1]The four metrics used: loss, accuracy, F_1 and AUC, are normalized to the 0-1 scale. To select the best overall model a combination of all measures above was used: adding the scoring metrics and subtracting the normalized loss.

Model	Loss	Accuracy	F_1 score	AUC
Q+Ts	0.6047	0.6696	0.6613	0.7327
Qs+Ts	0.6048	0.6712	0.6542	0.7317
Q+T	0.6105	0.6638	0.6488	0.7249
Qs+T	0.6133	0.6617	0.6422	0.7215
T	0.6304	0.6428	0.6308	0.6962
Qu	0.6273	0.6426	0.6201	0.7003
Q	0.6358	0.6313	0.6142	0.6862
Qs	0.6466	0.6186	0.6006	0.6680
Qu+Tu	0.6646	0.5906	0.5801	0.6295

TABLE 4.1: Loss function along with some performance metrics. The letters *Q* and *T* in the model's name indicate the use of charge and/or time respectively in the model. On the other hand, *s* indicates if the correspondent type of data was used mPMT aggregated, and *u* if indicates that hit standardization has not been applied. Thus, model *Qu* represents the original WatChMaL configuration.

Model	Loss	Accuracy	F_1 score	AUC	σ_{AUC}
Q+Ts	0.6037	0.6713	0.6613	0.7332	0.0009
Qs+Ts	0.6042	0.6705	0.6601	0.7328	0.0014
Q+T	0.6106	0.6642	0.6481	0.7254	0.0013

TABLE 4.2: Average value for the loss function along with some performance metrics and the standard deviation in the AUC result.

Alternatives considering time as a scaling constant obtained a negligible performance improvement over the baseline configuration, not comparable with channel results (see Table 4.3). On the other hand, the chronological ordering of charge intensity at pixel level worsens the performance.

Model	Loss	Accuracy	F_1 score	AUC
T scale	0.6272	0.6424	0.6320	0.7002
Qu	0.6273	0.6426	0.6201	0.7003
T order	0.6358	0.6362	0.6372	0.6902
Tu order	0.6463	0.6192	0.6043	0.6679

TABLE 4.3: Loss function along with some performance metrics. *T scale* refers to the charge scaling described in Equation 3.1. *T order* models has pixel intensities chronologically ordered and *u* indicates no scaling. Model *Qu* represents the original configuration.

4.2 Transformation depth

Once the time information has been included in the model, data augmentation is assessed. As mentioned in the previous chapter, there are three possible transformations to be applied before each epoch. Until now only one of these transformations was applied. In this section, we explore the depth of the transformation, i.e. to apply more than one of these transformations simultaneously

and randomly chosen with replacement. These experiments were carried keeping the same hyperparameters as in Section 4.1. In order to reduce the stochasticity of the results, each configuration has been trained three times (see Table B.1).

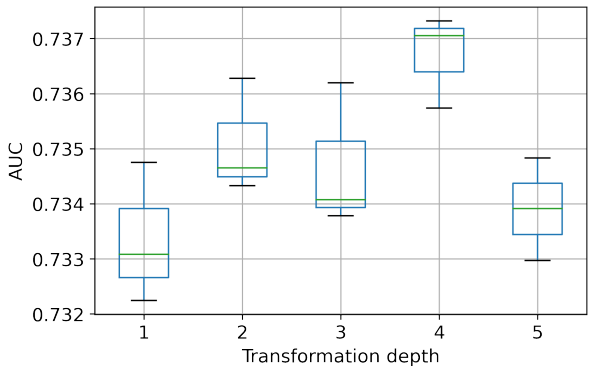

FIGURE 4.1: Box plot of the AUC values for different transformation depths.

To visually understand these results, the AUC obtained for the different configurations has been plotted in Figure 4.1. Showing how using more than a single transformation slightly improves performance. Although there is a small overlap, the best result is obtained when considering four possible transformations. The drastic result when moving to a fifth one is because the population to be sampled has dimension four, the three transformations mentioned above plus the possibility of not undergoing any of them. This inevitably implies that some member of the population must be repeated, and if two identical transformations are applied consecutively, their net effect is null.

4.3 Dropout

Since the MCD method for estimating model uncertainty employs dropout in the testing phase, the use of this technique during training has been explored, using different dropout rates for both weight (p_w) and channel (p_c) dropout, exploring all possible combinations and changing the learning rates of the optimizer.

The obtained results are displayed in Figure 4.2, showing that the use of dropout in training worsens the performance of the model in general. However, having trained with dropout enabled may be more beneficial when applying it during testing. Therefore, three models will be selected for the next experimental phase: the best model without dropout ($lr = 3 \times 10^{-4}$), the best model with drop-neuron ($lr = 5 \times 10^{-4}$, $p_w = 0.1$) and the best model with drop-channel and drop-neuron ($lr = 5 \times 10^{-4}$, $p_w = 0.1$, $p_c = 0.05$). In addition, better results have been obtained when increasing the learning rate of the optimizer. Other heatmaps plotting the accuracy and the loss function can be found in the Appendix B.2.

The three models mentioned above have been re-trained, keeping their configurations, for 30 epochs. In order to use the MCD method, the dropout must be active during the testing phase. Thus, rates must be assigned to both types of dropouts in the evaluation phase. The last set of experiments is focused on

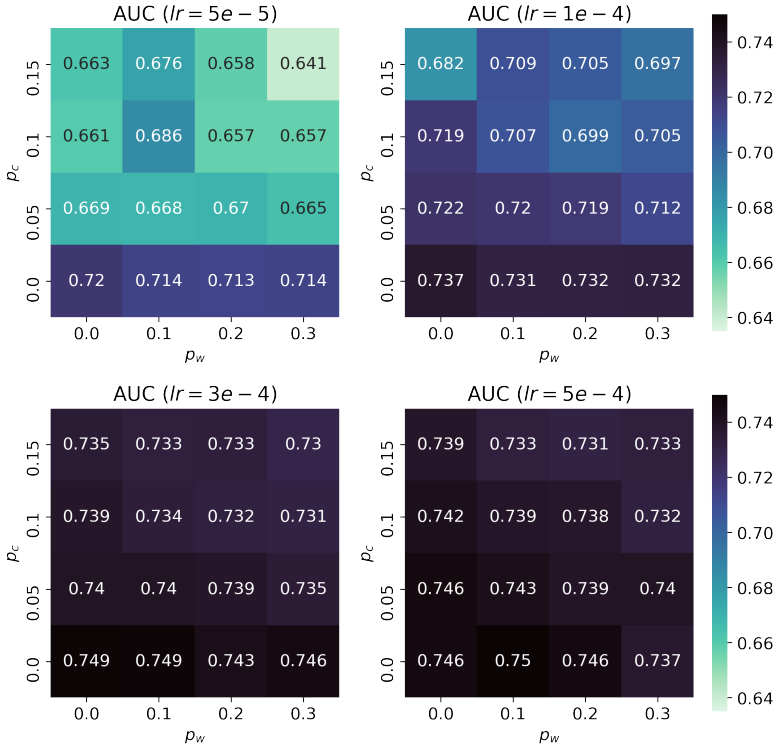

FIGURE 4.2: AUC values for different drop-neuron/channel dropout rates and learning rates.

investigating the best combination of dropout rates in the evaluation phase, applying the MCD method in the validation set with 50 forward passes per sample for the three models considered. Performance results are displayed in Figure 4.3, which shows the negative impact of using channel dropout in models that haven't been trained with it. On the other hand, weight dropout applied in the last fully connected layers of the model seems to keep the same level of performance. Furthermore, the model that has been trained with both types of dropout, achieves its best results when the same rates are used. Other heatmaps plotting the accuracy and the loss function can be found in Appendix B.3.

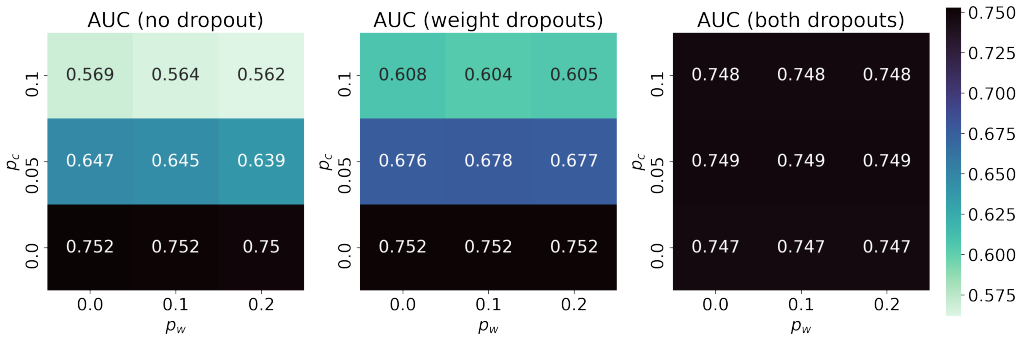

FIGURE 4.3: AUC values for different drop-neuron/channel dropout rates in validation.

4.4 Uncertainty

In the last experiment of the previous section concerning the dropout rates in the evaluation phase, the different measures of uncertainty have also been computed, as well as the Bhattacharyya distance between correct and incorrect predictions for every uncertainty measure, model and dropout configuration (Figure 4.4).

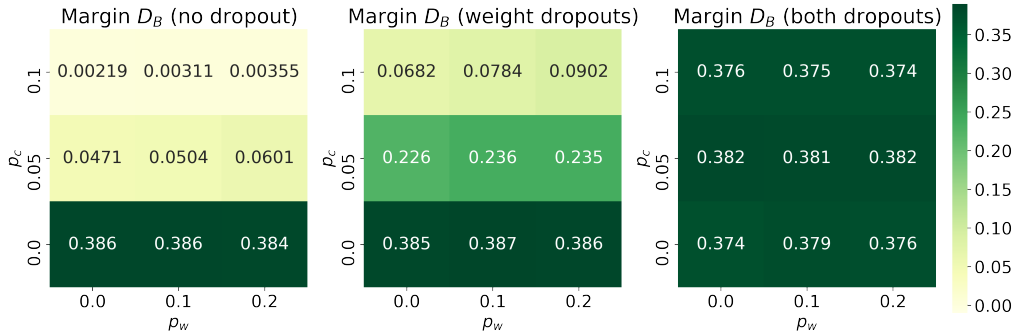

FIGURE 4.4: Bhattacharyya distance for different models and drop-neuron/channel dropout rates using the margin of confidence as uncertainty measure.

Comparing Figure 4.3 and the one above, there is a clear correlation between the performance of the model and how discriminative any uncertainty measure can be. Considering these two characteristics, the overall winner of this set of experiments is the model that is trained only with drop-neuron using $p_w = 1$, and that keeps the same dropout rate during MCD. A heatmap displaying the Bhattacharyya using the variance as an uncertainty measure can be found in Appendix B.3.

Figure 4.5 depicts the distributions of both correct and incorrect predictions using several uncertainty measures for the selected model. As it can be found on the title of each histogram, the most discriminative measure is the margin of confidence, achieving a Bhattacharyya distance (Equation 3.5) of $D_B = 0.3870$. Moreover, for this measure, the distribution of correct predictions is shifted towards 1 with respect to the distribution of incorrect predictions. This fact shows that, as one would expect, a prediction in which the margin of confidence between classes is greater has a higher probability of being correctly classified. On the other hand, it is interesting to note that the variance also has a comparable discriminative power. However, it is observed that incorrect predictions have a lower variance than correct ones. About entropy, incorrect predictions show a higher value, indicating higher randomness in the model predictions, i.e. lack of confidence. In addition, mutual information is higher among correct predictions.

4.5 Bias Analysis and Benchmarking

In the previous section we defined the final model as well as the best dropout configuration and the best measure that allows us to quantify the confidence in the predictions. The next step is to evaluate this model in the test set together with the configuration originally used by WatChMaL. These results are shown in Table 4.4, obtaining an improvement in all the performance metrics used and in particular, increasing accuracy by almost a 7% with respect to the baseline. On the other hand,

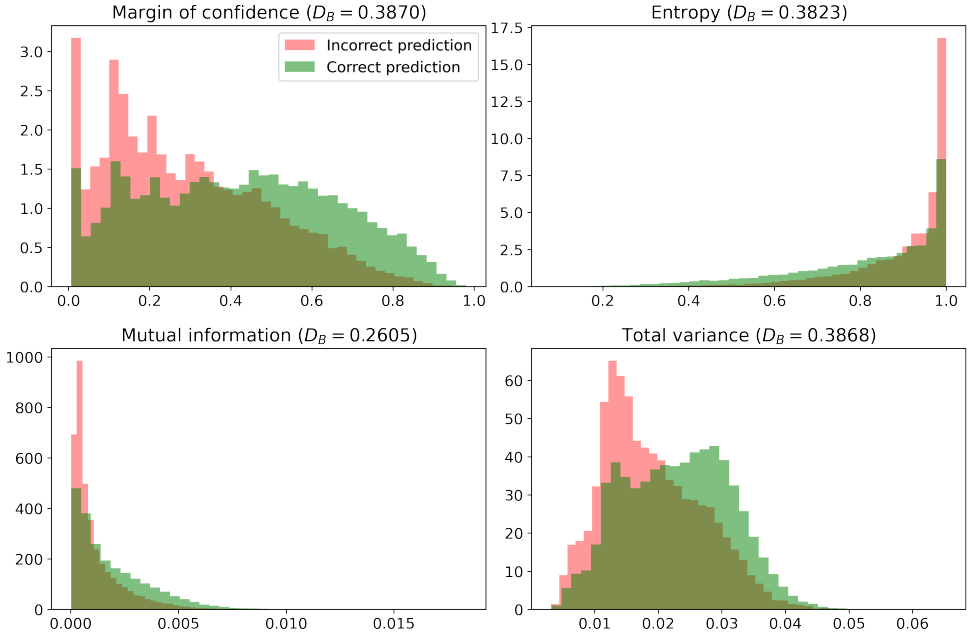

FIGURE 4.5: Histogram of correct and incorrect predictions for several uncertainty measures and their discriminating power using the Bhattacharyya distance.

in the Figure 4.6, the confusion matrix for both models, normalized along the true labels, can be found. The true negative rate has slightly increased, resulting in a preservation of the backgorund rejection. Nonetheless, the number of electrons correctly predicted has increased significantly, leading to a remarkable improvement of 14% in the electron signal efficiency. Moreover, the proposed model has a mean confidence of 38% on its predictions. In particular, the mean confidence in the classification of true electron events is of 39,1% and of 37% for gamma events. The closeness between these two values is an indicator that the model is well calibrated and does not show any significant bias towards any of the classes. Furthermore, it is striking how the model predicts gamma-type events more accurately but, on the contrary, offers higher confidence in electron-type predictions.

Model	Loss	Accuracy	F_1 score	AUC
Proposed	0.5856	0.6864	0.6795	0.7550
Original	0.6271	0.6427	0.6203	0.7007

TABLE 4.4: Loss function evaluated in the test set along with some metrics for the original configuration and the proposed model.

Next, both models are compared with respect to the different energy ranges of the particles (Figure 4.7). The proposed model shows higher accuracy specially in high energy ranges, plus an increasing tendency. Moreover, it outperforms the original configuration in signal efficiency except for low energies ranges (0-100 MeV) and shows a steadier behaviour. In reference to the confidence in the model predictions, a general trend is observed: the more energetic the particle, the higher the confidence. This fact is to be expected in all displayed metrics, since the higher

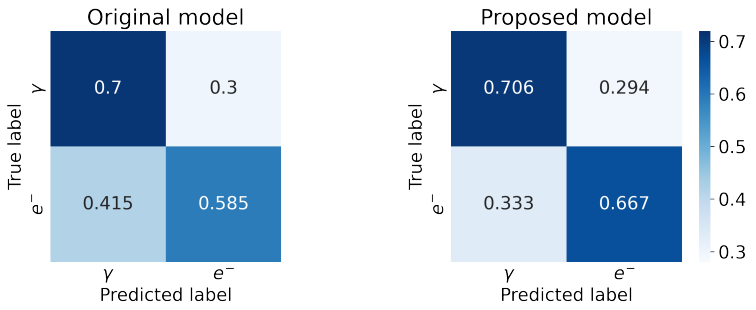

FIGURE 4.6: Row normalized confusion matrix for both original and proposed models.

the energy, the more PMTs are activated, detecting also higher charge values that allow for a better ring identification. In particular, the predictions for electrons show slightly higher confidence than for gamma events.

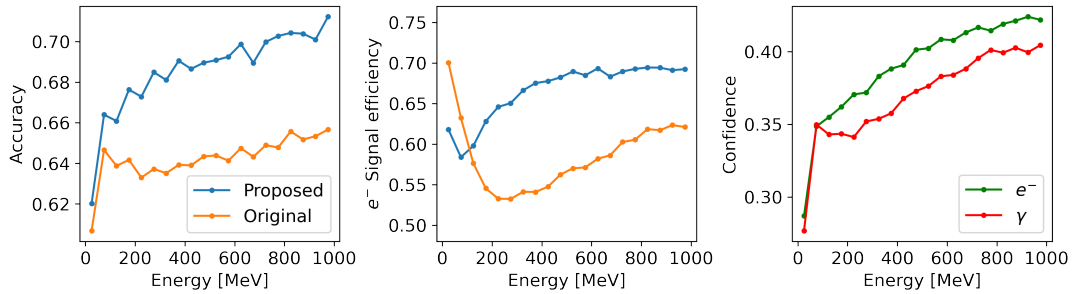

FIGURE 4.7: Both proposed and original models compared in terms of accuracy and electron signal efficiency for different energy ranges. The right picture depicts the confidence in the predictions of the proposed model for both classes in different energy ranges.

Performance has also been analyzed as a function of the position of the particle, i.e., the vertex from which it starts to radiate. In particular, Figure 4.8 shows the performance of both models for different positions. The distance to the wall is defined as the distance to the nearest PMT following the direction of the particle. The proposed model shows an improvement in accuracy in all ranges. However, it is not able to correct the negative trend: those events whose ring is too large in the image are predicted less accurately. This fact is confirmed by the behavior of the confidence in the model predictions, which is also slightly higher for electron-type events. This tendency can be explained by the fact that the average energy radiated by the particle is the same, however, its ring is larger, so the density of photons detected per PMT is lower, resulting in lower charge values which can be faded out with background noise. In addition, the radiated photons that compose the ring travel longer through the volume and thus undergo higher scattering interactions with the medium, producing a more diffuse ring and a worse identification. Regarding electron signal efficiency, the proposed model remains stable at around 60%, while the original model shows a stronger negative trend. This first fact together with the negative trend in the accuracy implies that the background rejection worsens significantly over large distances. Similar plots can be found in Appendix B.4 with respect to the shortest distance of the vertex to the wall.

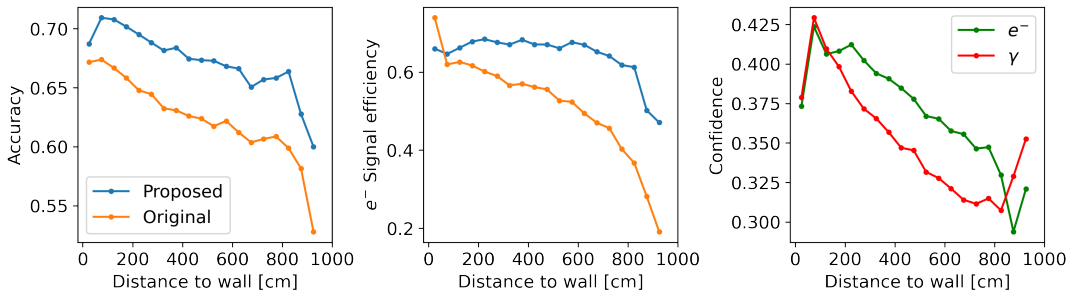

FIGURE 4.8: Both proposed and original models compared in terms of accuracy and electron signal efficiency for different positions. The right picture depicts the confidence in the predictions of the proposed model for both classes in these ranges.

Following this analysis with respect to the geometry of the detector, a similar study has also been performed for the zenith angle of incidence of the particle. The black lines represent those angles at which the incidence changes from the cylinder barrel to the caps. Again, the proposed model shows an improvement with respect to the baseline, in addition to a more stable behavior for all directions. This pattern is also reflected in the electron signal efficiency and in the predictions' confidence. However, the model presents higher confidence for gamma events in the barrel zone than in the caps. This very same analysis has been carried out with respect to the azimuth angle of incidence and ratifies the improvement over the baseline model, the lack of biases in the direction of the particle and the null negative effect of unwrapping the surface of the tank in a 2D image (see Appendix B.4).

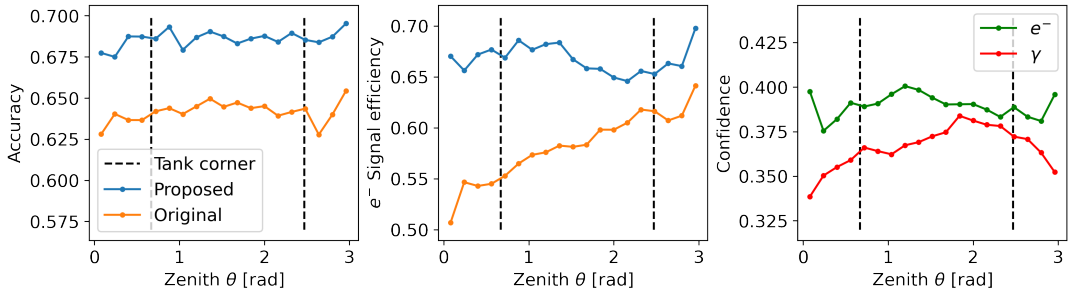

FIGURE 4.9: Both proposed and original models compared in terms of accuracy and electron signal efficiency for different zenith angles. The right picture depicts the confidence in the predictions of the proposed model for both classes in these ranges.

Chapter 5

Conclusions and Future Work

As seen in this project, the use of ML algorithms in particle physics offers a very attractive alternative to deal with the large amount of data produced by current detectors. In particular, for WCDs the WatChMaL organization has managed to improve both performance and computational time over its traditional counterpart fiTQun in both reconstruction and identification tasks, to provide the HK experiment with software superior to the one of its predecessor. However, the distinction between electron-like events and background reactions produced by highly energetic photons is still an open problem. In the present thesis, several modifications to the original pipeline used by the organization have been proposed in order to improve this particle identification.

Due to the high correlation between the temporal and charge data collected by PMTs in events with a single ring and vertex, the addition of temporal information does not pose an outstanding improvement in this particular particle identification task. Nonetheless, it has been demonstrated that its proper use can enhance the performance of the model while correcting biases. It also provides a good starting point for more complex studies addressing multi-ring events.

Several ways of including temporal information in the ResNet18 type architecture have been evaluated. The most successful approach handles this data in a charge-analogous way, as additional channels in the input image. Given the different numerical ranges of the two data sources, a hit-level standardization has been performed to ease the convergence of the gradient descent algorithm. It should also be noted that due to the similar spatial correlation of time and charge, the winning model incorporates the time in an aggregated version, thus being represented by just two channels. One represents the mean detection time of the mPMT module and the other corresponds to its standard deviation, providing a different representation to that given by the charge alone. Regarding data augmentation, given the relatively small size of the dataset, employing several image transformations with replacement at the beginning of each epoch has had a positive impact on the performance.

Currently, these results are being validated with a much larger dataset, with about 20 million events, that ensures smaller binomial errors in the test set. Early tests show that this hybrid combination continues to be the best option to include temporal information. However, the intention is also to explore deeper and more powerful models, such as ResNet50, to check if this choice is due to the lack of parameters in the architecture to deal with a fully disaggregated version. Other types of architectures such as PointNet or a move towards transformers and

self-attention mechanisms should also be considered in future investigations.

The association of uncertainty to experimental results in the field of particle physics is of crucial importance to validate their quality. For this reason, and given the difficult task of classifying these two types of events, an alternative method has been proposed when using the model in the evaluation phase. This method, called MCD, employs the dropout to perform inference on the posterior distribution of the network parameters, enabling the computation of several statistics to quantify uncertainty and confidence. An analysis of the use of dropout in the CNN has concluded that using it only in the fully connected part of the network is the most beneficial approach in both training and evaluation phases. In addition, the confidence margin has been postulated as the measure that best discriminates correct from incorrect predictions. Thus, the model is able not only to classify an event but also to provide the confidence with which it has done so.

Finally, a comparison of the proposed model with the original configuration shows an overall improvement in all performance indicators, maintaining the background rejection but increasing the electron signal efficiency. Thanks to the confidence measures, it has been observed that the model is properly calibrated and not biased towards any class. In addition, with the metadata available for each event, it is evidenced that the proposed model also presents a more stable behaviour in the different energy ranges, directions of incidence and vertex positions.

The results shown in this thesis have been weekly reported to the WatChMaL organization and the findings have been well received. It is envisaged that these ML models will be deployed in the WCTE (Barbi et al., 2019), a small WCD built at CERN (*European Organization for Nuclear Research*), whose purpose is to test the technology that will be used in the HK experiment. This detector is expected to be operational in less than two years.

Appendix A

Data and Metadata distributions

A.1 Event distributions

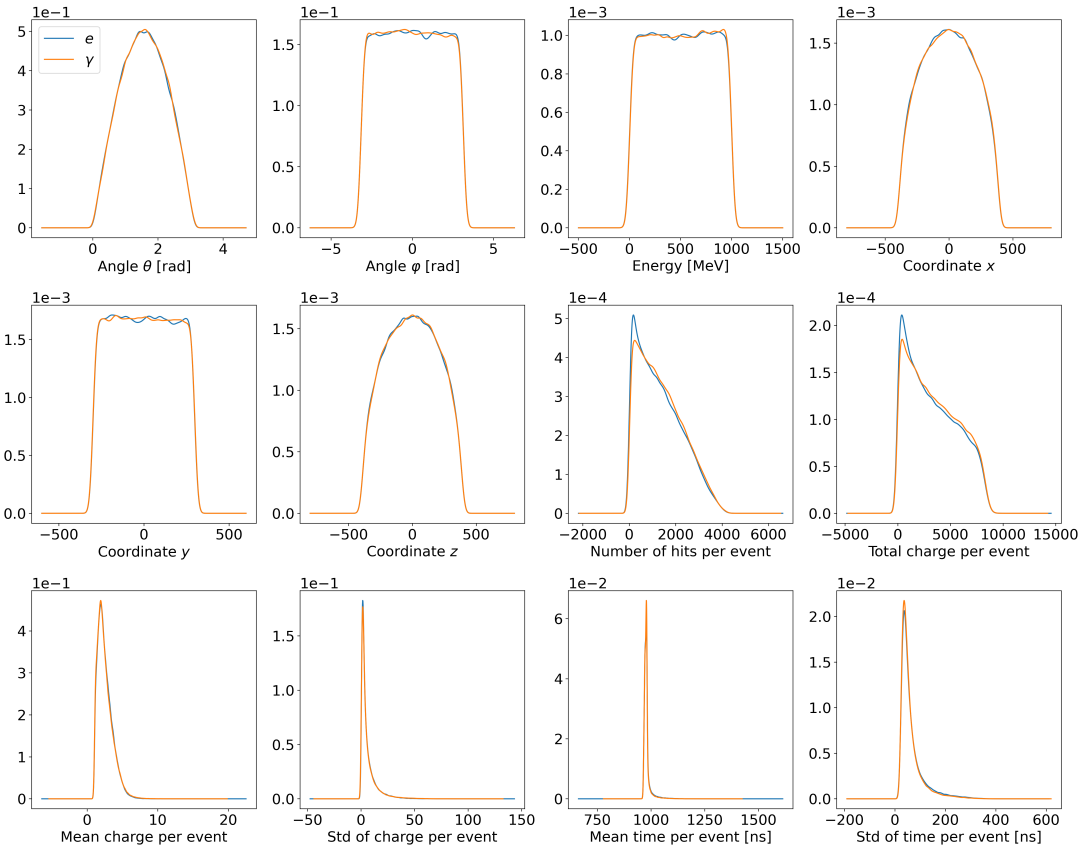

FIGURE A.1: Distributions of the position, direction, energy, hits per event and charge and time statistics for both classes (electron and gamma) using a subset of 236,091 randomly selected events from the training dataset A. "Std" stands for standard deviation. Charge is in units of photoelectrons, normalised so that so that charge of 1 corresponds to the average charge produced by one photon hitting the PMT and producing one photoelectron at its photocatode.

Appendix B

Experiment Results

B.1 Transformation depth

Transf. depth	Loss	Accuracy	F_1 score	AUC	σ_{AUC}
4	0.6018	0.6728	0.6538	0.7367	0.0008
2	0.6030	0.6712	0.6571	0.7351	0.0010
3	0.6042	0.6708	0.6609	0.7347	0.0013
5	0.6047	0.6699	0.6561	0.7339	0.0009
1	0.6050	0.6702	0.6550	0.7334	0.0013
0	0.6224	0.6539	0.6468	0.7114	0.0028

TABLE B.1: Average loss function along with some performance metrics and standard deviations. Transformation depth equals 1 is the original configuration for data augmentation used in WatChMaL.

B.2 Train Dropout

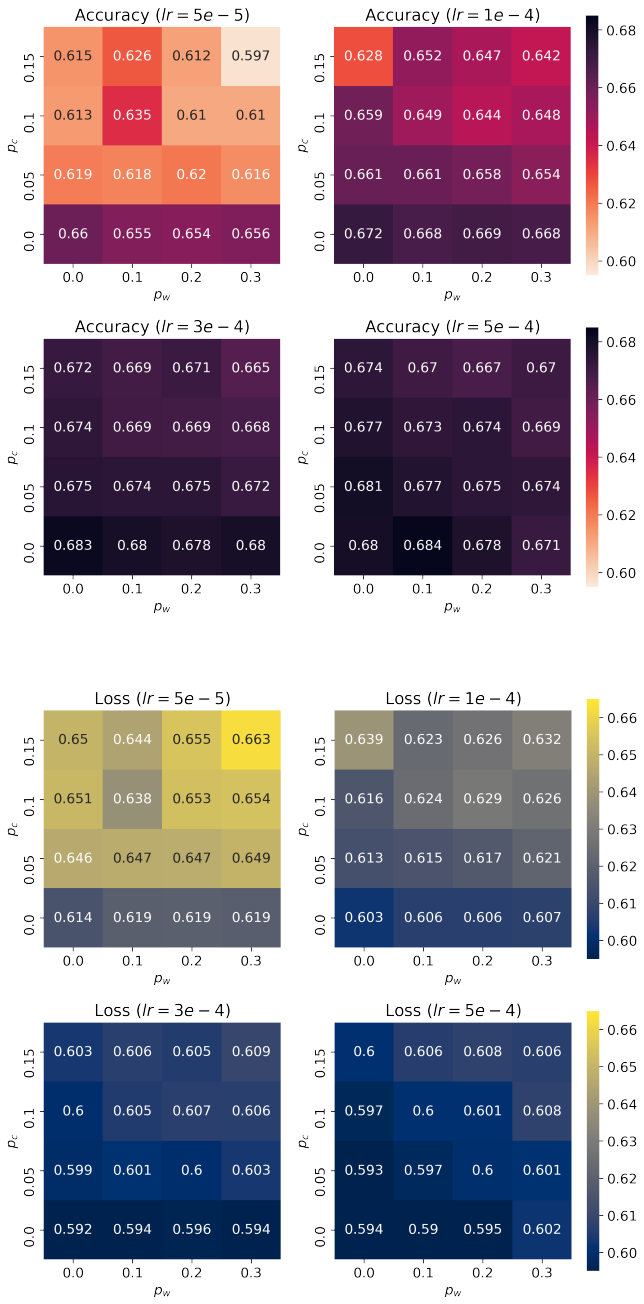

FIGURE B.1: Accuracy and loss function results for different drop-neuron/channel dropout rates and learning rates.

B.3 Evaluation Dropout

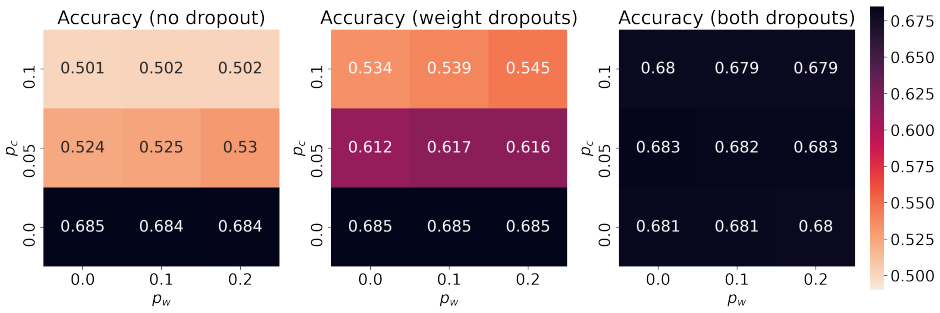

FIGURE B.2: Accuracy values for different drop-neuron/channel dropout rates in validation.

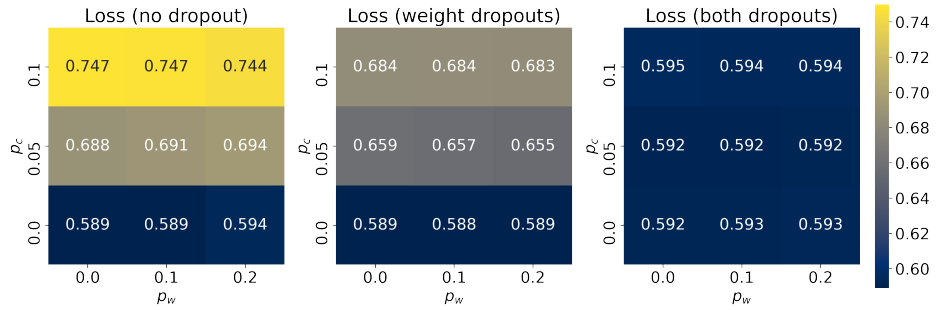

FIGURE B.3: Loss function values for different drop-neuron/channel dropout rates in validation.

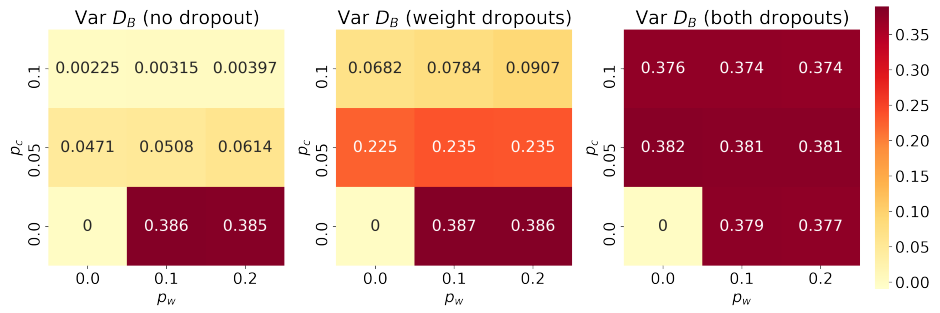

FIGURE B.4: Bhattacharyya distance for different models and drop-neuron/channel dropout rates with variance as uncertainty measure.

B.4 Bias Analysis

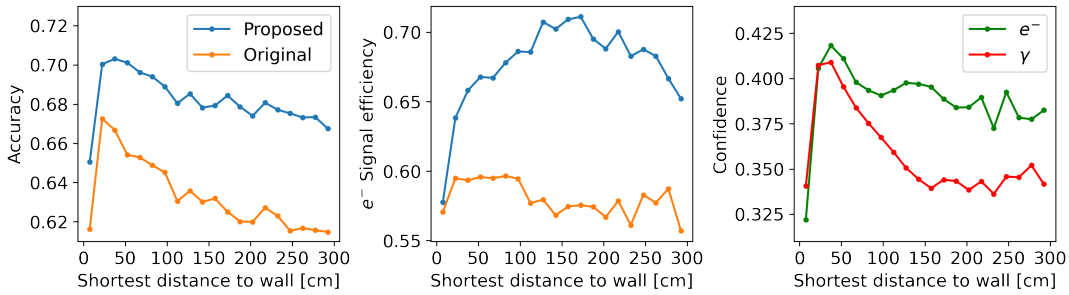

FIGURE B.5: Both proposed and original models compared in terms of accuracy and electron signal efficiency for different positions. The right picture depicts the confidence in the predictions of the proposed model for both classes in these ranges.

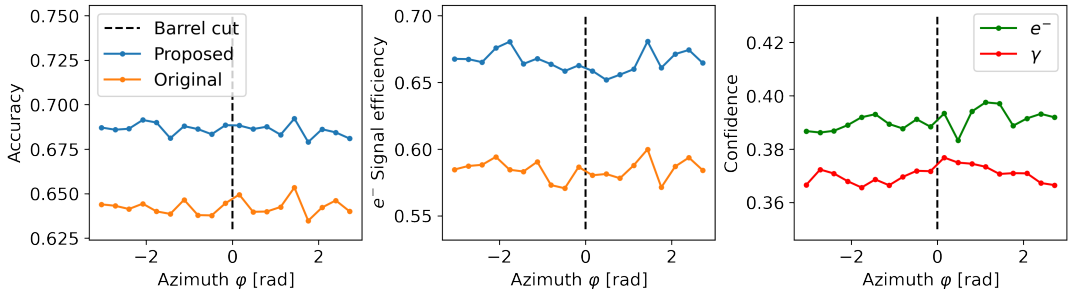

FIGURE B.6: Both proposed and original models compared in terms of accuracy and electron signal efficiency for different azimuth angles. The right picture depicts the confidence in the predictions of the proposed model for both classes in these ranges.

Bibliography

Abe, K. et al. (May 2018). "Hyper-Kamiokande Design Report". In: arXiv: 1805.04163 [physics.ins-det].

Agostinelli, S. et al. (2003). "GEANT4–a simulation toolkit". In: *Nucl. Instrum. Meth. A* 506, pp. 250–303. DOI: 10.1016/S0168-9002(03)01368-8.

Albertsson, Kim et al. (2018). "Machine Learning in High Energy Physics Community White Paper". In: *J. Phys. Conf. Ser.* 1085.2, p. 022008. DOI: 10.1088/1742-6596/1085/2/022008. arXiv: 1807.02876 [physics.comp-ph].

Barbi, M et al. (2019). *A Water Cherenkov Test Beam Experiment for Hyper-Kamiokande and Future Large-scale Water-based Detectors*. Tech. rep. Geneva: CERN. URL: https://cds.cern.ch/record/2692463.

Bengio, Y., P. Simard, and P. Frasconi (1994). "Learning long-term dependencies with gradient descent is difficult". In: *IEEE Transactions on Neural Networks* 5.2, pp. 157–166. DOI: 10.1109/72.279181.

Bhadra, S. et al. (2014). *Letter of Intent to Construct a nuPRISM Detector in the J-PARC Neutrino Beamline*. DOI: 10.48550/ARXIV.1412.3086. URL: https://arxiv.org/abs/1412.3086.

Cai, Shaofeng et al. (2019). *Effective and Efficient Dropout for Deep Convolutional Neural Networks*. DOI: 10.48550/ARXIV.1904.03392. URL: https://arxiv.org/abs/1904.03392.

De Oliveira, Rodrigo et al. (Sept. 2017). "A System Based on Artificial Neural Networks for Automatic Classification of Hydro-generator Stator Windings Partial Discharges". In: *Journal of Microwaves, Optoelectronics and Electromagnetic Applications* 16, pp. 628–645. DOI: 10.1590/2179-10742017v16i3854.

De Rosa, Gianfranca (2020). "A multi-PMT photodetector system for the Hyper-Kamiokande experiment". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 958. Proceedings of the Vienna Conference on Instrumentation 2019, p. 163033. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2019.163033. URL: https://www.sciencedirect.com/science/article/pii/S0168900219313968.

Gal, Yarin and Zoubin Ghahramani (2016). *Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning*. arXiv: 1506.02142 [stat.ML].

He, Kaiming et al. (2015). "Deep Residual Learning for Image Recognition". In: *CoRR* abs/1512.03385. arXiv: 1512.03385. URL: http://arxiv.org/abs/1512.03385.

Hinton, Geoffrey E. et al. (2012). *Improving neural networks by preventing co-adaptation of feature detectors*. DOI: 10.48550/ARXIV.1207.0580. URL: https://arxiv.org/abs/1207.0580.

Hornik, Kurt, Maxwell B. Stinchcombe, and Halbert L. White (1989). "Multilayer feedforward networks are universal approximators". In: *Neural Networks* 2, pp. 359–366.

LeCun, Yann, Y. Bengio, and Geoffrey Hinton (May 2015). "Deep Learning". In: *Nature* 521, pp. 436–44. DOI: 10.1038/nature14539.

Lee, Jaehoon et al. (2017). *Deep Neural Networks as Gaussian Processes*. DOI: 10.48550/ARXIV.1711.00165. URL: https://arxiv.org/abs/1711.00165.

Mahdianpari, Masoud et al. (2018). "Very Deep Convolutional Neural Networks for Complex Land Cover Mapping Using Multispectral Remote Sensing Imagery". In: *Remote Sensing* 10.7. ISSN: 2072-4292. DOI: 10.3390/rs10071119. URL: https://www.mdpi.com/2072-4292/10/7/1119.

Milanés-Hermosilla, Daily et al. (2021). "Monte Carlo Dropout for Uncertainty Estimation and Motor Imagery Classification". In: *Sensors* 21.21. ISSN: 1424-8220. DOI: 10.3390/s21217241. URL: https://www.mdpi.com/1424-8220/21/21/7241.

Missert, Andrew D. (2017). "Improving the T2K Oscillation Analysis With fiTQun: A New Maximum-Likelihood Event Reconstruction for Super-Kamiokande". In: *Journal of Physics: Conference Series* 888, p. 012066. DOI: 10.1088/1742-6596/888/1/012066. URL: https://doi.org/10.1088/1742-6596/888/1/012066.

Prouse, Nick (2021). "Advances in simulation and reconstruction for Hyper-Kamiokande". In: *PoS* ICHEP2020, p. 919. DOI: 10.22323/1.390.0919.

Seoh, Ronald (2020). *Qualitative Analysis of Monte Carlo Dropout*. DOI: 10.48550/ARXIV.2007.01720. URL: https://arxiv.org/abs/2007.01720.

T2K and beyond (2021). Accessed: 2022-05-30. URL: https://t2k-experiment.org/beyond-t2k/.

The WCSim GEANT4 application (2021). https://github.com/WCSim/WCSim.

Tompson, Jonathan et al. (2014). *Efficient Object Localization Using Convolutional Networks*. DOI: 10.48550/ARXIV.1411.4280. URL: https://arxiv.org/abs/1411.4280.

Valls, César, Thorsten Lux, and F. Sanchez (Mar. 2022). *Data-driven detector signal characterization with constrained bottleneck autoencoders*.

Water Cherenkov Machine Learning Organization (2019). Accessed: 2022-07-30. URL: https://www.watchmal.org.