Using GNNs for mass reconstruction of light pseudoscalars decaying to merged photon-pairs in the CMS ECAL endcaps

By:
Somanko Saha
Registration number:
20201163

Supervisor:
Prof. Seema Sharma

This thesis proposal was submitted in Partial Fulfillment of the requirements for the BS-MS dual degree programme

Indian Institute of Science Education and Research, Pune

March 2025

Certificate

This is to certify that the thesis on "Using GNNs for mass reconstruction of light pseudoscalars decaying to merged photon-pairs in the CMS ECAL endcaps" submitted by Somanko Saha, in Partial fulfillment of the requirements for the BS-MS dual degree is an original work carried out by him under the guidance of Prof. Seema Sharma. It is certified that the work has not been submitted anywhere else for the award of any other degree of this or any other institute. We also certify that he complied with the Plagiarism Guidelines of the institute.

Supervisor
Prof. Seema Sharma
Professor
IISER Pune

Expert
Dr. Rajdeep Mohan Chatterjee
DHEP Faculty member
TIFR, Mumbai

Student
Somanko Saha
BS-MS (20201163)
IISER Pune

Declaration

I hereby declare that the matter embodied in the report entitled *"Using GNNs for mass reconstruction of light pseudoscalars decaying to merged photon-pairs in the CMS ECAL endcaps"*, are the results of the work carried out by me at the Department of Physics, Indian Institute of Science Education and Research, Pune, under the supervision of Prof. Seema Sharma, and the same has not been submitted elsewhere for any other degree.

Somanko Saha
BS-MS (20201163)
IISER Pune

Acknowledgments

I would like to express my deepest gratitude to all those who have supported and guided me throughout the course of this research.
Firstly, I extend my heartfelt thanks to my supervisor, Professor Seema Sharma of Indian Institute of Science Education and Research (IISER) Pune, for her unwavering support, insightful guidance, and continuous encouragement. Her expertise and dedication have been instrumental in shaping this thesis.

I am deeply grateful to Dr. Rajdeep Mohan Chatterjee from the Tata Institute of Fundamental Research (TIFR), Mumbai, for his invaluable guidance and insightful feedback, which have greatly enhanced the quality of my research. Additionally, I sincerely appreciate his provision of GPU and storage resources, which were essential for the progress of my work and significantly enriched my research experience.

I acknowledge the Department of Science and Technology (DST), Government of India, for the KVPY fellowship, and the National Supercomputing Mission for providing the necessary computational resources. Their support has been crucial in the successful completion of this research.

I would like to extend my heartfelt gratitude to my seniors, Chirayu and Alpana, for their invaluable assistance in introducing me to CMSSW and machine learning models. Their guidance was instrumental in laying the foundation for my research. I also wish to acknowledge my lab members—Tanay, Chetan, and Bapi—for their insightful discussions and contributions during our group meetings, which significantly enriched my research experience.

I am profoundly grateful to my parents, Sumit Saha and Shrabani Saha, for their unwavering love and support, and to my brother, Suryanko Saha, for his constant encouragement. Their belief in me has been a steadfast source of strength throughout this journey. I also wish to acknowledge my dear friends at IISER Pune, whose camaraderie and support have been invaluable during my time here.

Table of Contents

List of Figures

Abstract

The Standard Model (SM) has been thoroughly tested in experiments conducted over the past forty-fifty years, It has successfully explained the phenomena occurring with probabilities over eight orders of magnitude. The discovery of the Higgs boson by ATLAS and CMS in 2012, has opened up new avenues for the search of Beyond Standard Model (BSM) physics. An example of such BSM scenarios is the exotic decays of the SM Higgs boson into a pair of light pseudoscalars (as), with each pseudoscalar decaying into two photons ($H \rightarrow aa \rightarrow 4\gamma$). For such pseudoscalars with low masses ($m_a < 2$ GeV), the two photons from its decay can be merged as single photon object. The project focuses on development of a novel mass reconstruction technique using Graph Neural Networks (GNNs) for such highly boosted pseudoscalars. Using GNNs allows us to train the model on low level input features from different subdetectors with varying geometries, that is sparse and irregular data, without the need for zero padding.

Chapter 1

Introduction

Following the discovery of the nucleus by Rutherford in 1911, high energy particles became a routine tool to probe the subatomic world. Particle collision experiments emerged in the mid-20th century as a way to explore the fundamental particles and their interactions. Early experiments involved accelerating particles and directing them at stationary targets (referred to as fixed target experiments). The study of cosmic showers revealed particles like muons, pions and strange particles, while stationary target experiments led to the discovery of the internal structure of the nuclei and sub nuclear regime. Various observations, like parity violation, in nuclear decays further fueled interests in understanding these particles. These findings unveiled a "zoo" of particles beyond protons, neutrons, and electrons.

This explosion of new particles highlighted the need for a framework to organize and make sense of the growing number of discovered particles. As particle physics advanced, scientists needed a comprehensive theory to explain the forces and interactions governing these particles. The Standard Model (SM) was developed on the quantum field theory framework to fulfill this need, providing a unified description of the electromagnetic, weak, and strong forces and categorizing the fundamental particles (like quarks, leptons, and bosons). It also offered predictions for particles such as the Higgs boson, which were later experimentally confirmed.

1.1 Standard Model (SM) of elementary particles

The SM of particle physics is a theoretical framework rooted in quantum field theory and built upon the principle of gauge symmetry. It encompasses three of the four fundamental forces in nature: the strong, electromagnetic, and weak interactions. The SM does not account for gravitational interactions. It is constructed on the gauge group $SU(3)_{QCD} \times SU(2)_{weak} \times U(1)_{hypercharge}$. At an energy scale of approximately 247 GeV, this symmetry undergoes spontaneous breaking, reducing the gauge group to $SU(3)_{QCD} \times U(1)_{EM}$. In this framework, all matter consists of fermions, which include quarks and leptons as shown in figure 1.1. The interactions among particles are mediated by gauge bosons, gluons for strong, γ for electromagnetic, $W^{\pm}$, and Z for weak interactions. The masses of the fundamental particles are determined by their interactions with the Higgs boson. Assuming Dirac neutrino masses, the SM contains 27 parameters: three coupling constants (g, g', and g_s), six quark masses,

three charged lepton masses, and three neutrino masses; three mixing angles and one phase for quarks; three mixing angles and one phase for leptons; the Higgs mass (m_h) and its vacuum expectation value (vev, v); the QCD vacuum angle (θ). However, not all 27 of these parameters are independent, making the model over constrained with only 19 free parameters. Despite its success in explaining a wide range of phenomena, the SM remains incomplete. It does not incorporate gravity, does not explain dark matter or dark energy, and cannot account for the matter-antimatter asymmetry observed in the universe. These open questions suggest the need for physics beyond the Standard Model (BSM), motivating theories such as supersymmetry, grand unified theories (GUTs), and higher dimensional theories including gravity.

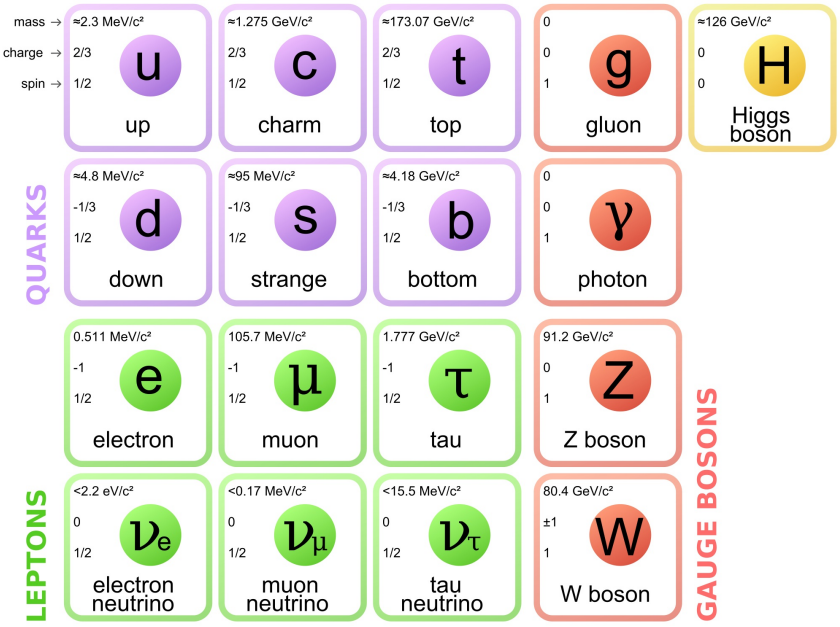


Figure 1.1: The fundamental particles of the Standard Model

1.2 Higgs discovery

The SM has been tested to very high precision, over the past forty years and has successfully explained the high energy interactions in particle physics. In 2012, the CMS and ATLAS experiments at the LHC discovered the Higgs boson, the final unobserved fundamental particle with spin 0, predicted by the SM. The discovery was made in the $H \to ZZ^* \to 4l$ and $H \to \gamma\gamma$ channels (see figure 1.2 for the diphoton invariant mass distribution relevant to the Higgs boson discovery.). Since then, numerous decay channels of the Higgs boson have been thoroughly investigated, strengthening the evidence that this newly discovered particle is indeed the Higgs boson predicted by the SM.

1.3 Higgs mechanism

The electromagnetic and weak forces are unified under the electroweak theory. However, predictions from the SM confirmed that this symmetry had to be spontaneously broken to give mass to the W and Z gauge bosons, as observed in nature. Spontaneous symmetry breaking is realized by incorporating a complex scalar SU(2) doublet with a non-zero vacuum expectation value (*vev*) into the Lagrangian. Three of the four fields of the scalar doublet are used to assign masses to the gauge bosons ($W^{\pm}$ Z and γ), leaving only one physically observable scalar field called the Higgs field. The γ remains massless even after spontaneous symmetry breaking. The Higgs boson represents the quantized excitation of the Higgs field, which interacts with fermions via Yukawa couplings to confer mass upon them.

The Higgs sector continues to be an important tool for studying BSM physics, even with the current constraints from the LHC, because it can link to hidden sectors that are neutral to the SM interactions. The SM predicts the Higgs boson to have a very narrow decay width of around 4 MeV, resulting in rare decays. Because of this, even a tiny coupling between the Higgs boson and a BSM particle can produce a measurable branching fraction into exotic decay channels, which could be detected at the LHC.

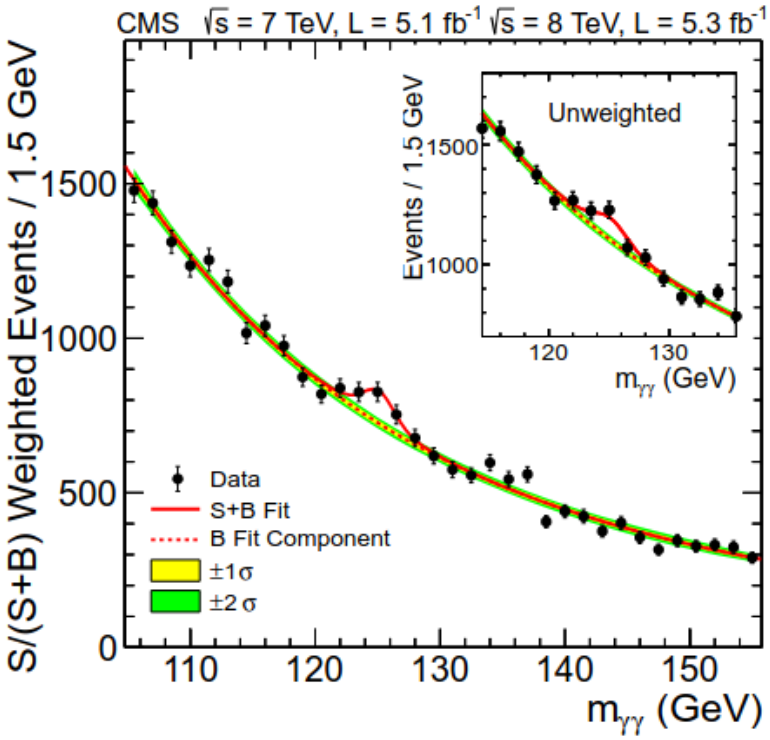


Figure 1.2: The figure displays invariant mass distribution of diphoton candidates, where each event is weighted by its S/(S + B) value. Solid curves represent the signal and background fits, and the colored bands denote the uncertainties at ±1 and ±2 standard deviations for the background estimation. An inset offers a zoomed-in view of the central region of the unweighted invariant mass distribution. [14]

1.4 BSM Higgs decays

One of the BSM scenarios is a pair of pseudoscalars that originate from exotic decays of the SM Higgs boson. Such pseudoscalar candidates are well motivated in theories like next to minimally supersymmetric extensions of SM (NMSSM), extensions involving axion like particle (ALP) production and any alternative extension of the SM that includes a hidden sector interacting with an additional scalar field. The 2HDM + singlet models (like NMSSM) predict the existence of two such pseudoscalars whose masses may not be the same.

While the signal cross sections may vary depending on the model, if the pseudoscalars have masses less than the pair production threshold for massive SM particles, then the diphoton decay mode is enhanced, as shown in figure1.3. However in such cases, the pseudoscalars from Higgs decay can be highly boosted with photons from each $a \rightarrow \gamma\gamma$ leg being sufficiently merged to be reconstructed as a single photon-like object (Γ) in the detector (figure 1.3). Such scenarios could be buried under the $H \rightarrow \gamma\gamma$ measurement like that shown in figure 1.2. These experimentally challenging topologies can now be explored given the recent advances in analyses tools like

the use of Machine Learning (ML) based methods. In the studies presented in this thesis, we develop a strategy to use a dynamic reduction network (DRN), based on a graph neural network (GNN), to reconstruct the pseudoscalars from the merged photon candidates reconstructed in the CMS electromagnetic calorimeter.

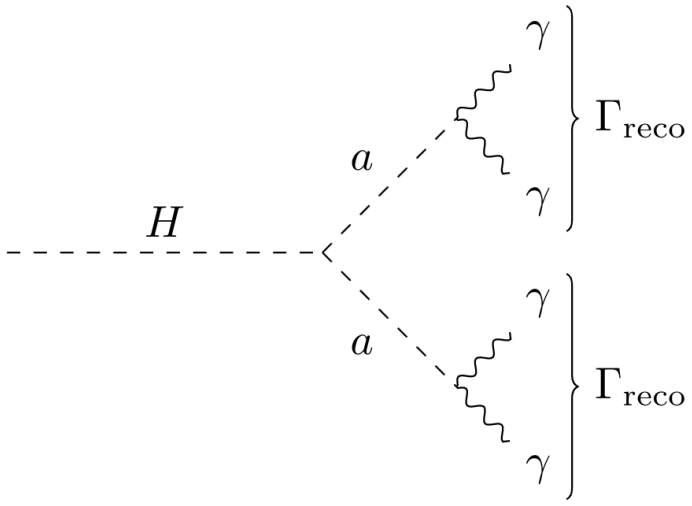
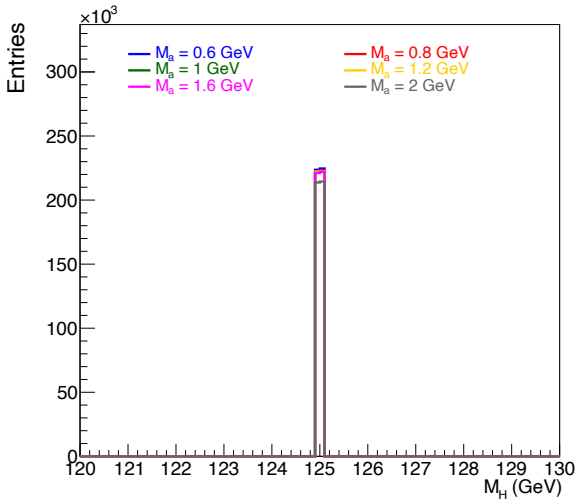


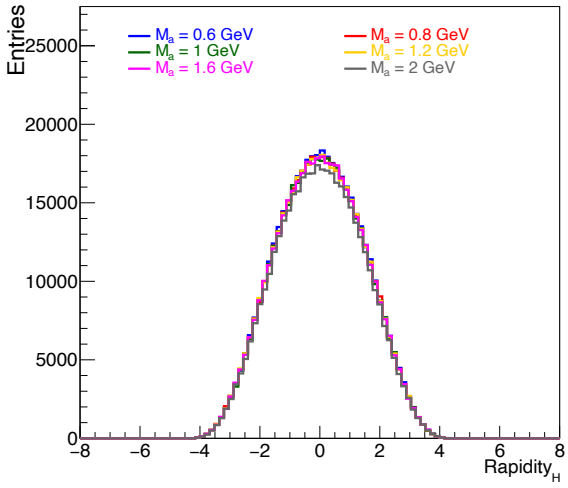
Figure 1.3: Feynman diagram for $H \to aa \to 4\gamma$ decay process

1.5 Monte Carlo generation of $H \to aa \to 4\gamma$ in pp collisions

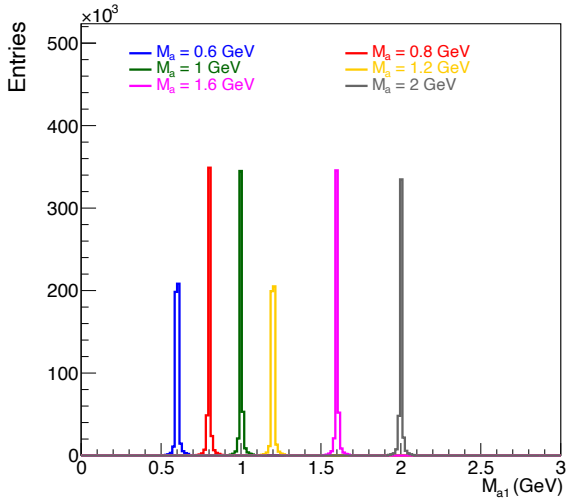
The DRN model was trained on $a \to \gamma\gamma$ samples as described in chapter 5. To avoid bias towards any mass of as while training, the distribution of kinematic variables for the $a \to \gamma\gamma$ training samples may differ from the what we may observe in the data. Hence, the DRN results are validated using the official $pp \to H \to aa \to 4\gamma$ Monte Carlo samples, generated at various mass points of the pseudoscalar a. These samples are produced under conditions that closely resemble Higgs boson production in actual proton-proton collision data. The subfigures in figure 1.4 present the kinematic distributions obtained from these official $pp \to H \to aa \to 4\gamma$ samples at different a masses.



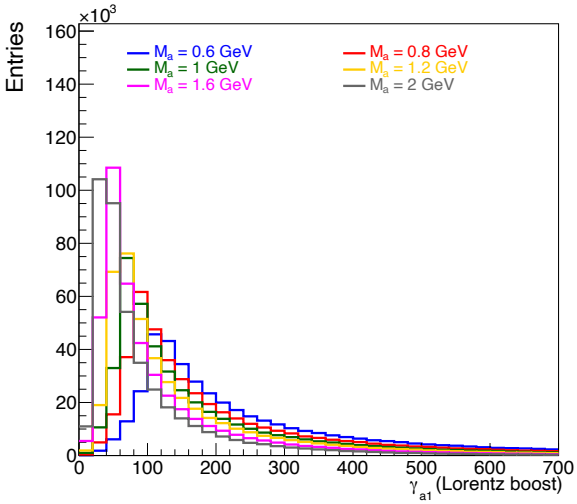
(a) Higgs mass in all the different mass samples



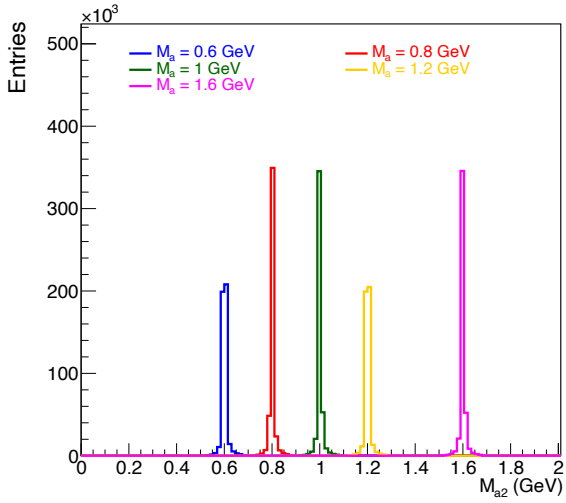
(b) Rapidity of Higgs for different mass samples



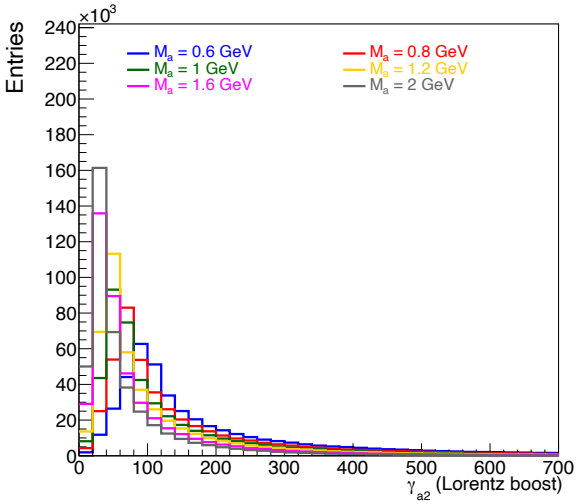
(c) Mass of the leading a



(d) Lorentz boost of the leading a



(e) Mass of the subleading a



(f) Lorentz boost of the subleading a

Figure 1.4: Kinematic distributions for the $H \to aa$ decay channel across different pseudoscalar mass samples. Shown are the Higgs boson mass (figure 1.4a and rapidity (figure 1.4b, as well as the mass (figures 1.4c 1.4e) and Lorentz boost (figures of the leading and subleading pseudoscalars.

Chapter 2

Experimental setup and event reconstruction

The key advantage of particle colliders lies in the fact that all of the colliding beam energy is available for new particle production. In stationary target experiments, much of the energy is wasted as the particles move forward after the collision. In contrast, head-on collisions in a particle collider focus all the energy into the interaction, maximizing the energy available to create new particles and study rare phenomena. This efficiency allowed colliders to probe deeper into the subatomic world, testing theories like the SM, exploring quarks, leptons, and bosons, and discovering particles like the Higgs boson.
Particle colliders became essential tools for recreating the extreme conditions of the early universe, enabling scientists to investigate how matter behaves at high energies and uncover phenomena like antimatter, dark matter candidates, and the unification of forces. Colliders like CERN's Large Hadron Collider have become indispensable for advancing our understanding of the cosmos and its origins.

2.1 Large Hadron Collider (LHC)

The LHC is a monumental scientific instrument designed to explore the fundamental properties of matter. It is located near Geneva, Switzerland and it the worlds largest hadron collider. It comprises a 27-kilometer ring of superconducting magnets designed to accelerate particles to near-light speeds for high-energy collisions. In the accelerator, the two opposite traveling high energy proton beams are presently accelerated to energy of 6.8 TeV per beam. This is achieved by employing a series of linear and circular accelerators as shown in figure 2.1 and described below.
The process starts with Linac4, a linear accelerator, that accelerates negatively charged hydrogen ions (H^-) up to an energy of 160 MeV. These ions are directed into the Proton Synchrotron Booster (PSB) where the two electrons are removed, leaving only the protons. The protons in PSB are accelerated to 2 GeV and are subsequently transferred into the Proton synchrotron (PS). In the PS, the proton beams are accelerated to 26 GeV and then passed on to the Super Proton synchrotron (SPS) to achieve an energy of 450 GeV. The protons from SPS are used to fill the beam pipe of the LHC. The beam in one beam pipe circulates clockwise while the other beam

circulates anticlockwise. It takes 4 minutes and 20 seconds to fill each ring of the LHC, and the protons require ~20 minutes to reach their maximum energy of 6.8 TeV.

The acceleration of these proton beams by radio frequency cavities causes the protons to form bunches in the beams. The bunches are spaced at about 7.5 m from each other along the beam (z) direction and have a root mean square (RMS) length of 7.5 cm. At the time of collision, these bunches are squeezed along the x-y plane to $16\mu m \times 16\mu m$, using quadrupole magnets, to increase the instantaneous luminosity and increase the number of hard interactions.

There are four experiments, ALICE, ATLAS, CMS and LHCb, setup at the four interaction points in the LHC ring. The beams are made to collide at these four interaction points where the out-coming particles from the collision can be detected. The center of mass energy of these collisions is 13.6 TeV. When the proton bunches collide at the interaction points, hundreds of billions of protons cross paths, causing numerous (almost) simultaneous collisions (interaction vertices). Usually only one of the interactions correspond to high energy transfer and is of interest. The rest are referred to as pileup vertices.

The pileup and the luminosity can vary depending on the experiment. For example, ALICE operates on low luminosity to prevent excessive pileup in heavy-ion collisions, as these events are more complex and result in large number of outgoing particles in each collision. On the other hand, LHCb is a forward detector that primarily focuses on the study of bottom quarks. It focuses on study of hadrons containing bottom quarks via their decay products. Hence, LHCb is optimized to run at lower luminosities to study rare events in more controlled environment and avoid contamination from pile-up interactions. The CMS and ATLAS, on the other hand, are general purpose detectors, which deal with high-luminosity proton-proton collisions targeting SM measurements and BSM searches.

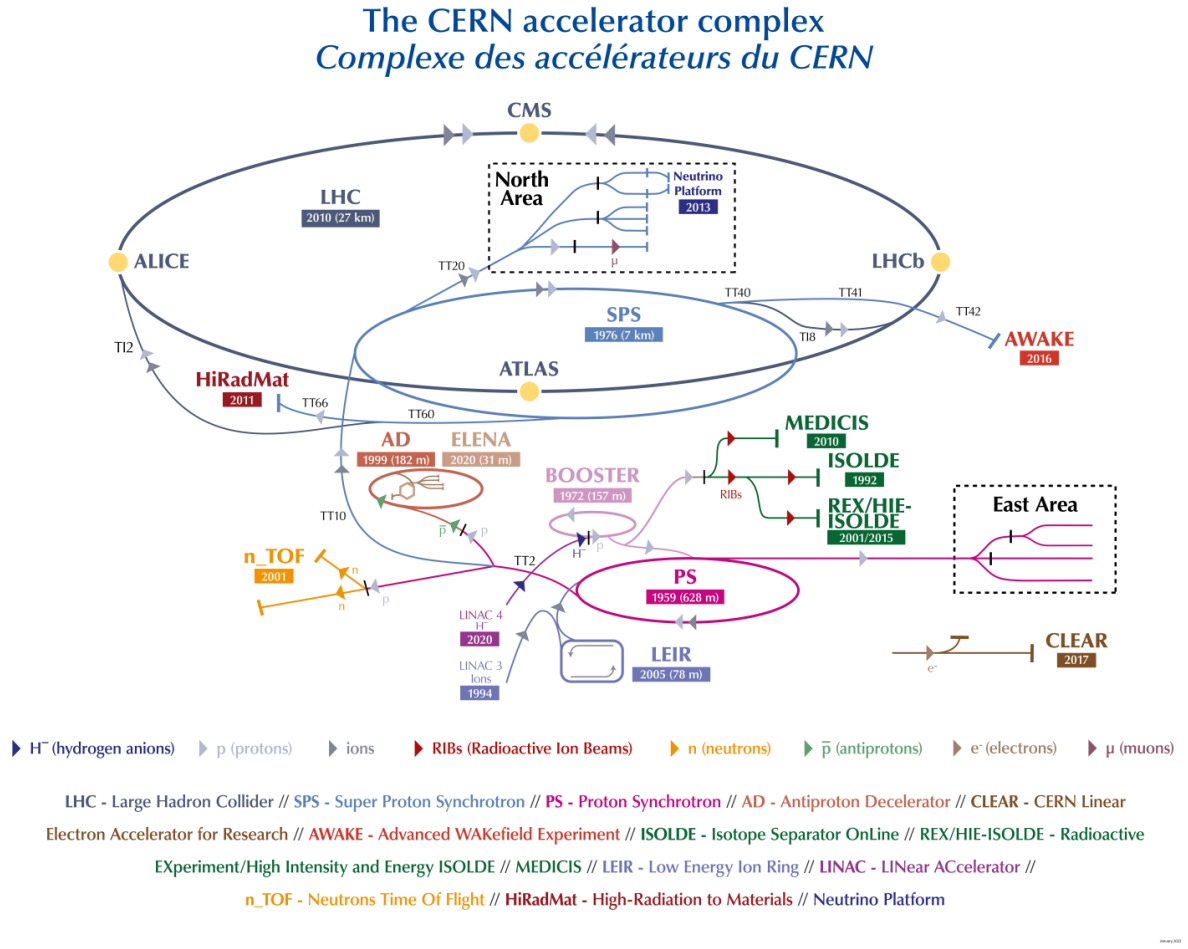


Figure 2.1: The Large Hadron Collider (LHC) serves as the culmination (represented by the dark blue ring) of an intricate ensemble of particle accelerators. The smaller accelerators operate in coordination to boost particles to their ultimate energies and supply beams for various smaller-scale experiments. (Image: CERN) [1]

2.2 Compact Muon Solenoid (CMS) detector

The Compact Muon Solenoid (CMS) is a multi-purpose detector at the LHC, designed to explore a wide array of physics phenomena, including the study of the Standard Model, the search for new particles that could explain dark matter, and the exploration of extra dimensions. The detector is equipped with a powerful solenoid magnet that generates a strong magnetic field, enabling precise measurement of charged particle momentum. The CMS detector has a cylindrical structure composed of sub detectors arranged concentrically around the collision point. Except for the muon chambers, all the sub detectors are located inside the magnetic field volume of the solenoid. The CMS detector measures approximately 21 meters in length, 15 meters in width, and 15 meters in height. The different subdetectors in CMS are briefly summarized below.

2.2.1 Tracker

Measuring the momentum of particles is essential for inferring the processes occurring at the core of a collision. In High Energy Physics (HEP), a particle's momentum is determined by measuring curvature of its trajectory through a magnetic field volume: the greater the curvature of the path, the lower the particle's momentum. The CMS tracker captures the paths of charged particles by recording their positions at multiple points along their trajectories. It uses a cylinder shaped silicon tracker with outer radius of 1.2 m and length of 5.6 m supplemented with tracking detectors in the endcaps to increase the geometric coverage. The inner-most tracker layer has radius of 4.4 cm. The barrel region ($|\eta| < 1.44$) consists of three concentric tiers of pixel detectors followed by ten successive tiers of micro-strip detectors. The endcap section is equipped with two layers of pixel-based sensors, followed by twelve layers of detectors utilizing micro-strip technology. The silicon sensor modules consist of 66 million small pixels, with each pixel having dimensions of $150\mu m \times 100\mu m$, and 9.6 million strips ranging in width from 80 to $180\mu m$. This fine detail helps to identify trajectories of particles as they move away form the collision point as well as those paths that are close together in jets.
The tracker layers, along with their associated services such as cables and supports, contribute to a substantial amount of material present before the calorimeters, as illustrated in figure 2.2. There is a significant probability of electrons, photons or hadrons interacting with the tracker material and depositing their energy. This energy deposition into the tracker material, needs to be taken into account when reconstructing the energy of the particles. This consideration impacts the reconstruction of electrons and photons, leading to the definition of shower shape variables like R9 (described in section 2.3.3), which are used to quantify these effects.

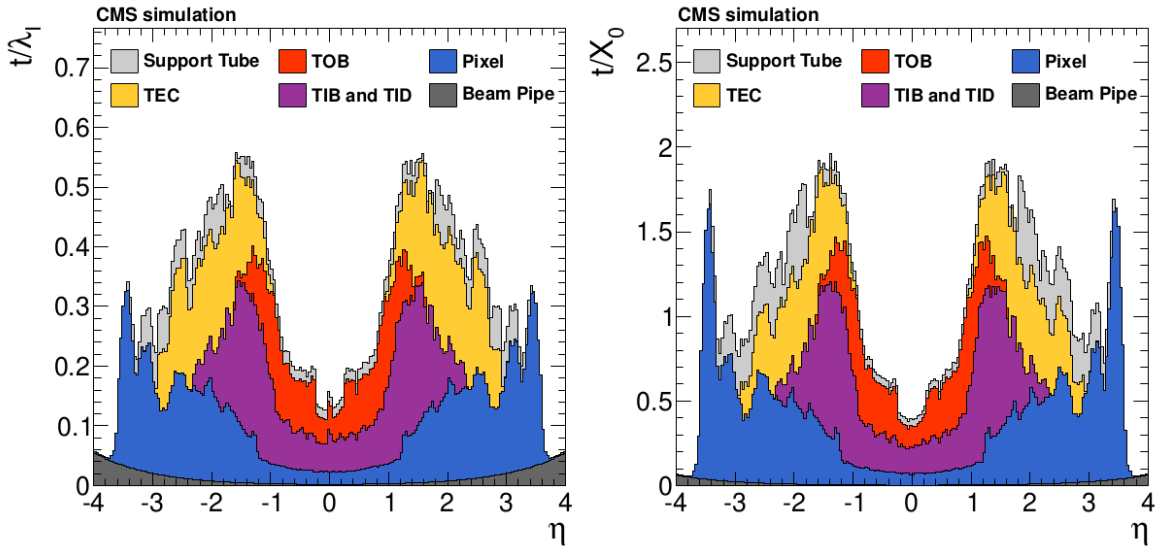


Figure 2.2: A figure illustrating the material budget before the ECAL, as a function of pseudorapidity (η). The left plot shows the material budget in terms of interaction lengths (λ_I) and the right plot depicts the same in terms of radiation length (X_0). The abbreviations TIB, TID, TOB, and TEC correspond to the "tracker inner barrel," "tracker inner disks," "tracker outer barrel," and "tracker endcaps," respectively [3].

2.2.2 Electromagnetic calorimeter (ECAL)

The CMS electromagnetic calorimeter (ECAL) is a homogeneous, hermetic, scintillation based calorimeter made of lead tungstate ($PbWO_4$) crystals. These crystals are arranged in a non-central manner in supermodules is shown in figure 2.3. These production crystals emit approximately 80% of their scintillation light within 25 ns, aligning well with the LHC's 25 ns bunch crossing interval. The blue-green colored scintillation light has a broad maximum at 420-430 nm.

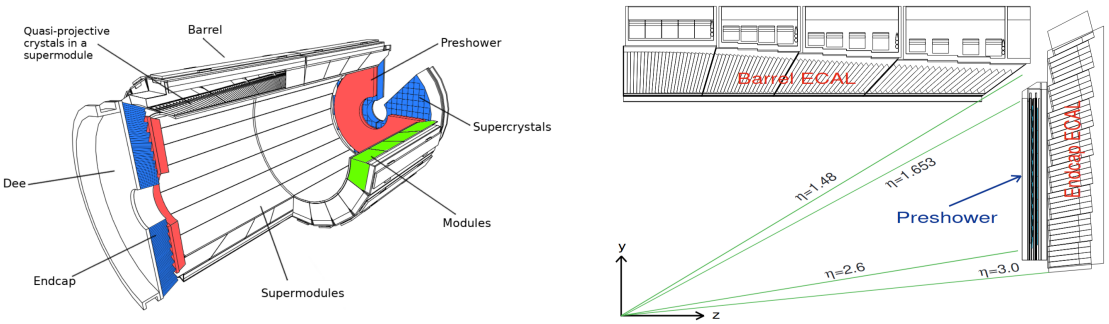
The barrel region ($|\eta| < 1.479$) consists of approximately 61,200 crystals, each measuring 2.2 cm by 2.2 cm in cross-sectional area and 23 cm in length (equivalent to 25.8 X_0). This configuration provides a fine segmentation of about 0.0174 in both pseudorapidity (η) and azimuthal angle (ϕ). The signals generated in the crystals in barrel are read out by Avalanche Photodiodes (APDs). Each endcap disk extends over the pseudorapidity region of $1.479 < |\eta| < 3.0$ and contains 7324 crystals. The endcap crystals have face area $2.86 \times 2.86\ cm^2$ and length $22cm$ (24.7 X_0). The granularity in the $\eta - \phi$ plane in the endcaps, varies from 0.0174×0.0174 to 0.05×0.05 as a function of η. Due to high particle flux in the endcaps, Vacuum Phototriodes (VPT)s are used instead of APDs to read out the signals generated in the crystals.

The Moliere radius (R_0) of lead tungstate is 2.2 cm which matches the crystal dimensions in the barrel. This high level of granularity enables precise resolution of hadron and photon energy deposits separated by as little as 5 cm. The relative energy resolution is expressed as a function

of the electron or photon energy using the following parametrization [2]:

$$\frac{\sigma}{E} = \frac{2.8\%}{\sqrt{E/GeV}} \oplus \frac{12\%}{E/GeV} \oplus 0.3\% \tag{2.1}$$

Due to the stochastic term varying as $\frac{1}{\sqrt{E}}$, a characteristic of homogeneous calorimeters, the energy measurements improve with the increasing energy of the incident particles. The electronics noise in the ECAL varies between the barrel and endcap regions. In the barrel region, the noise level is approximately 40 MeV per crystal, while in the endcaps, it is about 150 MeV per crystal.



(a) (b)

Figure 2.3: A schematic view of the CMS electromagnetic calorimeter. [12]

In front of each of the endcap disks, there is a fine grained preshower (as shown in figure 2.4). The preshower consists of two layers of lead absorbers each followed by a layer of silicon strip detectors. The lead radiators are about 2 X_0 and 1 X_0 thick, respectively. The silicon strips, with dimensions of 1.9 $\times$ 65 mm^2, are positioned to form two intersecting planes at right angles [12]. When a photon or electron traverses in lead, it initiates an electromagnetic shower. The detector's fine segmentation and the compact nature of the resulting shower enable precise localization of the shower's position. The primary goal of adding a preshower was to distinguish neutral pions ($\pi^0 s$) from prompt photons. The preshower system also aids in differentiating electrons or photons from jets by initiating the shower before the ECAL, compensating for the limited granularity in the endcaps. This enables the use of isolation variables to effectively separate prompt electrons or photons from jets. During reconstruction, the preshower energy deposits are added to the nearest ECAL clusters.

Electromagnetic showers from prompt photons or electrons or photons from π^0 or η^0 particles are well contained within the ECAL. The description of the electromagnetic showers is given in section 3.3.

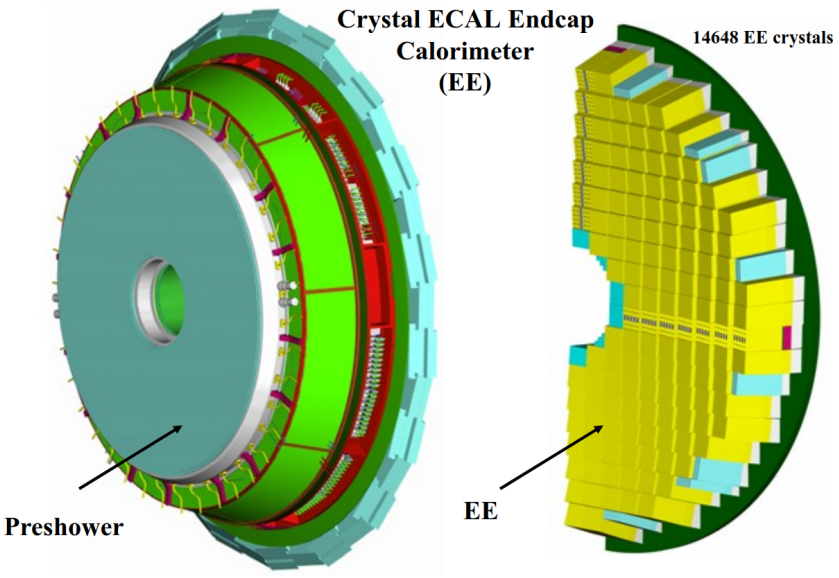


Figure 2.4: The CMS ECAL preshower and endcaps

2.2.3 Hadron calorimeter (HCAL)

The CMS hadron calorimeter is a hermetic sampling calorimeter that alternates layers of dense absorber materials (brass) with plastic scintillator tiles. This design efficiently captures and measures the energy of hadronic particles produced in high-energy collisions. The HCAL can be divided into three sections : Barrel (HB), Endcaps (HE) and Forward calorimeter (HF). The HB section encompasses a pseudorapidity range of $|\eta| < 1.3$, the HE region spans $1.3 < |\eta| < 3.0$, and the HF region extends across $3.0 < |\eta| < 5.0$.

In the barrel region, the total absorber thickness at a 90° angle is approximately 5.82 interaction lengths (λ_l). The effective thickness in the hadron barrel (HB) increases with the polar angle (θ) following a $\frac{1}{sin\theta}$ dependence. A tail catcher known as the Hadron Outer (HO) calorimeter is installed outside the solenoid to complement for this effective thickness. The HO material, equivalent to 1.4 interaction lengths at normal incidence, serves as an additional absorber. At lower pseudorapidities, the effective thickness is increased by using a 20 cm thick layer of steel. The ECAL crystals in front of the HB, contribute approximately 1.1 λ_l of material. In the barrel region of the CMS detector, the combined depth of the calorimeter system—including the Electromagnetic Calorimeter (ECAL) and the solenoid material—is at least twelve interaction lengths. In the endcap regions, this thickness is approximately ten interaction lengths. The HCAL provides a segmentation of 0.087 in pseudorapidity (η) and 0.087 in azimuthal angle (ϕ) for regions where $|\eta| < 1.6$, while for $|\eta| > 1.6$, the segmentation increases to 0.17 in both η and ϕ.

The HF is built using a steel absorber consisting of grooved plates. Radiation-hard quartz fibers are placed into these plates in alignment with the beam axis, and their signals are collected by

photomultiplier tubes (PMTs). These fibers are arranged alternately as long fibers, spanning the full thickness of the absorber (approximately 165 cm, equivalent to about ten interaction lengths), and short fibers, which begin 22 cm from the front face and cover the back portion of the absorber. Across most of the pseudorapidity range, calotowers with $\eta - \phi$ granularity of 0.175×0.175, read out the signals from the long and short fibres. These signals within each calorimeter tower are utilized to determine the hadronic and electromagnetic components of the shower.

2.2.4 Solenoid

At the heart of the CMS detector lies a superconducting solenoid magnet, a cylindrical coil of superconducting fibers that generates a powerful magnetic field when electrified. With a radius of 3.15 m and a length of 12.5 m, it is built to generate a magnetic field reaching up to 4T. The radius is large enough to fit the tracker, ECAL, and HCAL, reducing the material in front of the calorimeters. The magnet's bending power is strong enough to distinguish the energy deposits of charged and neutral hadrons.

2.2.5 Muon chambers

The magnetic field outside the solenoid is maintained at approximately 2T using an iron return yoke for detecting muons in the muon chambers. The yoke is composed of alternating steel plates and muon detector layers arranged in a multi-layered structure. The barrel region of the CMS muon system employs Drift Tube (DT) chambers, which are designed to detect muons within a pseudorapidity range of $|\eta| < 1.2$. To enhance muon detection capabilities, Resistive Plate Chambers (RPCs) are integrated, extending the coverage up to $|\eta| < 1.6$. In the endcaps, cathode strip chambers (CSCs) are used due to their fine segmentation, fast response, and radiation resistance. The CSCs cover the pseudorapidity range of $0.9 < |\eta| < 2.4$. Track information from the muon detectors and the inner tracker are integrated using a comprehensive trajectory reconstruction algorithm to reconstruct and identify muons.

2.3 Electron and photon reconstruction in the CMS experiment

2.3.1 Electromagnetic shower

When energetic photons ($E_\gamma > 10MeV$) interact with the electromagnetic field of the ECAL material, the produce an electron-positron pair. This process is called pair production. The electron (or positron), when interacting with electromagnetic fields, can emit bremsstrahlung photons, that can further pair produce, leading to a cascade of secondary electromagnetic showers. In the case of photons, the pair production continues till the energy of the photons

is less than the combined rest mass of an electron and a positron (i.e. 1.022 MeV). Beyond that, the photon loses energy by photoelectric effect or Crompton scattering. At lower energies, electrons lose energy mainly by ionization.

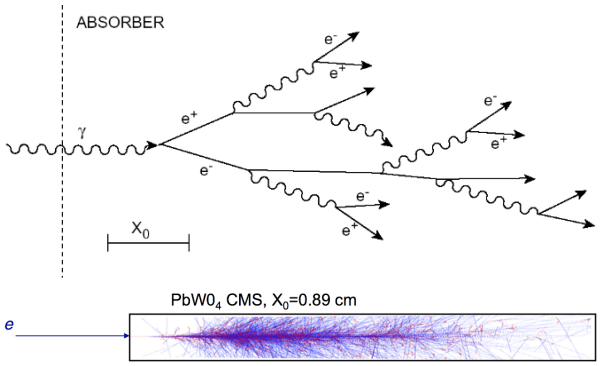


Figure 2.5: Electromagnetic shower

2.3.2 Mustache supercluster

As describe in the previous subsection, the development of electromagnetic showers results energy deposits in multiple ECAL crystals. The energy reconstruction algorithm begins by forming clusters, which are created by grouping crystals with energies above a predefined threshold (~80 MeV in barrel and ~300 MeV in endcaps). A single crystal local maxima is called a seed crystal. The seed crystal in barrel (endcaps) must have energy > 230 MeV (600 MeV). The seed crystals in endcaps must have transverse energy E_T > 150 MeV. Clusters are formed by grouping seed crystals with their neighboring crystals that pass the noise thresholds. Among the resultant clusters, the one with the most energy deposited in any region is defined as the seed cluster.

Starting from the seed cluster, additional clusters, falling into a mustache shaped geometric window in the $\Delta\eta - \Delta\phi$ plane are included to form a "mustache supercluster", as shown in figure 2.6. This is because, the magnetic field causes the electromagnetic showers to spread more in the ϕ direction than in the η direction. The size of this mustache-shaped window is dependent on the transverse energy (E_T) of the incident particle. In the figure 2.7, the two sets of green cells from mustache superclusters for electron and the bremsstrahlung photon. The mustache superclusters serve as seeds for reconstructing electrons and photons and implementing conversion-finding algorithms.

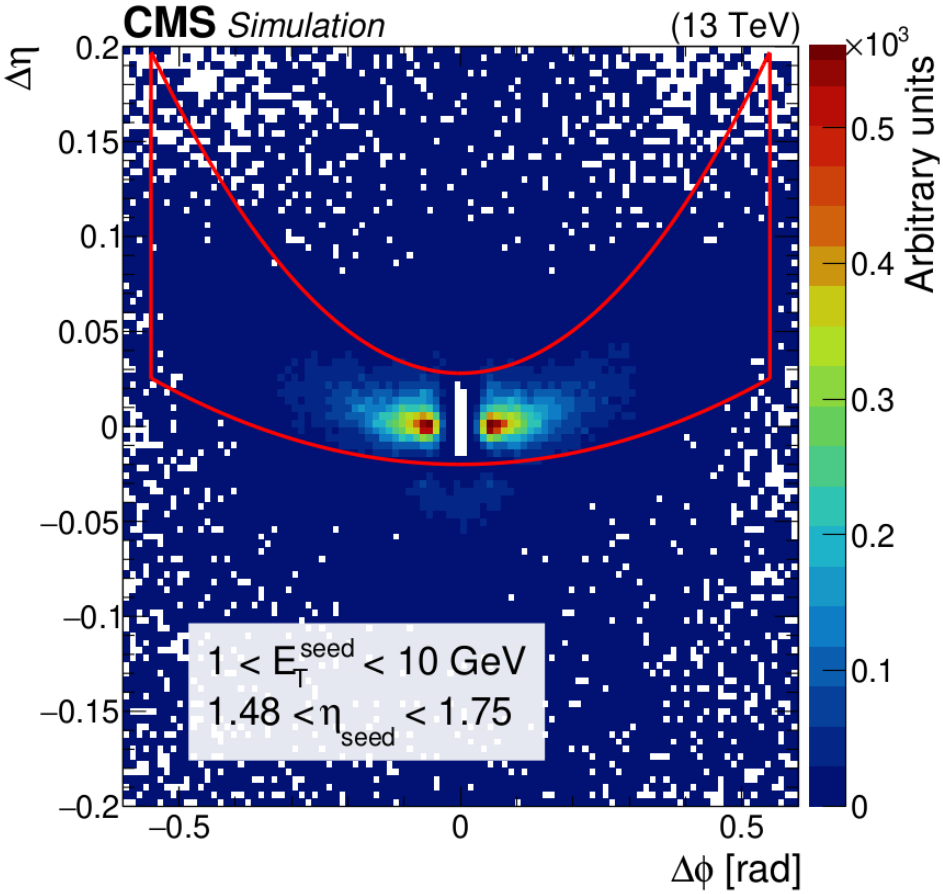


Figure 2.6: The plot illustrates the distribution of $\Delta\eta = \eta_{seed\ cluster} - \eta_{cluster}$ versus $\Delta\phi = \phi_{seed\ cluster} - \phi_{cluster}$ for simulated electrons within the range $1 < E_T^{seed} < 10$ GeV and $1.48 < \eta_{seed} < 1.75$. The red contour outlines the region of clusters recognized by the mustache algorithm. The central white area depicts the $\eta - \phi$ footprint of the seed cluster, highlighting the core region where most of the seed cluster's energy is concentrated.

2.3.3 Shower shape variables

Shower shape variables are used to identify electromagnetic objects and distinguish them from jets. Some shower shape variables used in the CMS experiment are briefly discussed below.

SigmaIEtaIEta ($\sigma_{i\eta i\eta}$)

SigmaIEtaIEta is the root mean square of the shower's log-energy distribution, measured in units of crystals. It is defined as follows :

$$\sigma_{i\eta i\eta} = \sqrt{\frac{\Sigma_i^{5\times5} w_i(\eta_i - \bar{\eta}_{5\times5})^2}{\Sigma_i^{5\times5} w_i}} \tag{2.2}$$

Here $w_i = -4.7 + ln\frac{E_i}{E_{5\times5}}$ is the weight, E_i is the energy of the i-th crystal, $E_{5\times5}$ is the energy of the 5×5 supercluster and η is in units of crystals.
This effectively reduces the noise by including only those rechits that have more than 0.9% of the energy of the 5×5 supercluster.

R9

The R9 variable is a measure of spatial energy spread of converted photons i.e. photons which have undergone e^+e^- pair production. It is defined as the ratio of the energy in the 3×3 grid around the seed ($E_{3\times3}$) to the energy of the raw supercluster ($E_{5\times5}$).

$$R9 = \frac{E_{3\times3}}{E_{5\times5}} \tag{2.3}$$

A low R9 value typically indicates that the photon has undergone conversion (i.e., pair production) in the tracker. The smaller the R9, the earlier the photon is likely to have converted. Both R9 and $\sigma_{i\eta i\eta}$) are transverse shower shape variables.

H/E

The H/E is a longitudinal shower shape variable that is defined as the ratio of the hadronic energy deposit to the electromagnetic energy deposit within a single HCAL tower or a certain cone of ΔR. One HCAL tower maps to twenty five ECAL crystals (a 5×5 grid of crystals).

$$H/E = \frac{E_{Had}}{E_{EM}} \tag{2.4}$$

Here the E_{Had} is energy deposited in the HCAL within the specified cone of ΔR = 0.15 or single tower. The E_{EM} is the energy deposited in the ECAL.

2.3.4 Tracking and track cluster association

The Gaussian Sum Filter (GSF) tracking algorithm is used for electrons to account for radiative losses caused by bremsstrahlung. The electron track reconstruction starts with identifying a pattern of hits that could correspond to an electron trajectory, a process known as "seeding". Electron trajectory seeds can originate from two methods: "ECAL-driven" and "tracker-driven." The ECAL-driven seeding starts by reconstructing superclusters in the ECAL and takes advantage of the fact that the supercluster position and energy can be used to estimate the trajectory of the electron in the first layers of the tracker. The combined collection of ECAL-driven and tracker-driven seeds is then utilized to initiate the electron track reconstruction process. A BDT is employed to determine whether a GSF track should be associated with

an ECAL cluster. The BDT integrates track data, supercluster characteristics, and variables associated with track-cluster matching. The track data encompasses kinematic features as well as quality-related metrics such as $\sigma_{i\eta i\eta}$, $R9$, and $|\Delta\eta| = |\eta_{SC} - \eta_{trk\ in}|$.

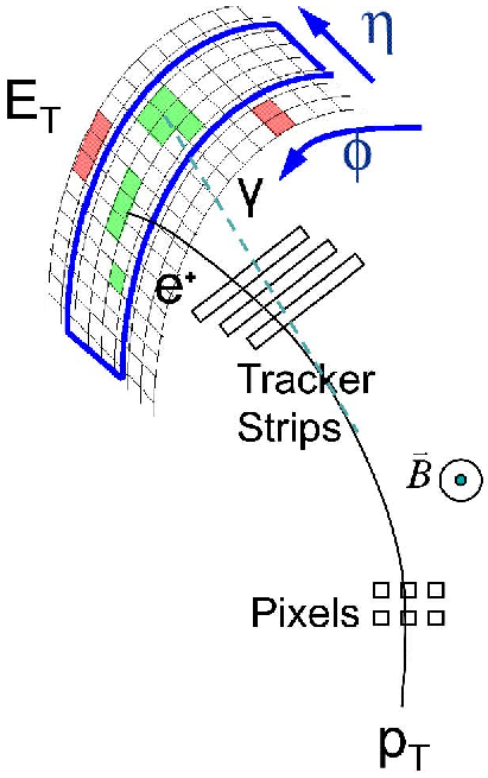


Figure 2.7: Electron reconstruction in CMS, showing the arc of tracks in magnetic field and the ECAL clusters. The ECAL clusters, representing the electron and its bremsstrahlung photons, are depicted as green squares, while the supercluster is enclosed in a blue box.[13]

2.3.5 Supercluster refinement

To enhance the precision of mustache superclusters (SCs), data from detector subsystems beyond the ECAL crystals and preshower detectors is incorporated. This process, known as supercluster refinement, involves utilizing information from the tracker to recover additional clusters resulting from photon conversions and bremsstrahlung. Specifically, tangents to the Gaussian Sum Filter (GSF) track's trajectory are projected from each tracker layer toward the ECAL to identify clusters produced by bremsstrahlung photons. Algorithms designed for conversion-finding are used to identify tracks originating from photon conversions. A boosted decision tree (BDT) is used to identify these tracks when only one one leg has been reconstructed. The Particle Flow (PF) algorithm constructs e/γ objects using inputs such as primary generic tracks, GSF tracks, mustache SCs, ECAL clusters and conversion-flagged tracks. ECAL clusters linked to bremsstrahlung tangents and secondary conversion tracks are incorporated into the refined supercluster to ensure the total energy aligns better with the GSF track momentum at the

innermost layer.

Chapter 3

Deep learning and Dynamic Reduction Network (DRN)

This chapter primarily focuses on the machine learning techniques and model architectures utilized for the project. The following sections provide a concise overview of the mathematical operations involved in training the ML model used in this study. Additionally, the architecture of the employed ML model and its internal operations are discussed briefly.

3.1 Deep learning

Machine learning, a subfield of artificial intelligence, is centered on developing algorithms enabling computers to analyze data, recognize patterns, and make decisions autonomously without requiring direct programming. The first idea of mathematical models of neural networks was given by Walter Pitts and Warren McCulloch in 1943. The term machine learning was first introduced in the 1950s by Arthur Samuel, when he built a model to calculate the winning chances of a player in a game of checkers. The perceptron was introduced in 1957, by Frank Rosenblatt, as an extension of the McCulloch-Pitts model [4].
Deep learning is a specialized area within machine learning that employs neural networks to to perform tasks such as regression, classification, and representation learning. Modern deep learning algorithms employ multi-layered perceptrons (MLPs), which utilize a weighted sum of inputs, biases, and activation functions to transform inputs into nonlinear functions. These components enable MLPs to learn complex patterns and relationships in data. This is made possible due to the Universal Approximation Theorem [5, 6]. The learning procedure for deep learning models has two processes : forward propagation and backward propagation. The forward propagation step generates the output based on the existing weights, while the backward propagation step updates the weights using stochastic gradient descent, as described in Section 3.2.

3.2 Stochastic gradient descent

Gradient descent is an optimization algorithm that iteratively adjusts parameters to minimize a function's value. It is one of the most used methods for changing a model's parameters (weights)

in order to minimize the loss. The gradient represents a vector indicating the direction of the steepest ascent of the objective function at a given point. The algorithm iteratively moves in the opposite direction of the gradient, progressively descending toward lower values of the function until it converges to a minimum. This convergence depends on the loss function. In out case, we use the mean-squared loss (MSE). Stochastic gradient descent is a variant of the Gradient Descent algorithm that uses a randomly selected small batch from the training sample to calculate the gradient and update the weights. This provides computational efficiency when dealing with a large dataset. Figure 3.1 shows how gradient descent can help to achieve the minima of a loss function.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.1}$$

During the backpropagation step involves:

- Calculation of gradient of the loss function (L) with respect to the weights (w_i) : $\frac{\partial L}{\partial w_i}$
- The weights are updated iteratively, according to the optimizer, in direction opposite to the gradient to minimize the loss.
- For a simple gradient descent, the weights are updated as follows:

$$w_{t+1,i} = w_{t,i} - \eta \frac{\partial L}{\partial w_i} \tag{3.2}$$

  where $w_{t+1,i}$ is the i-th weight at t+1-th iteration, $w_{t,i}$ is the i-th weight at t-th iteration and η is the learning rate that can be chosen to be large, small or dynamic depending on the problem.

- The Adam optimizer updates model parameters using gradients computed from batches of data. At each step, it calculates the mean gradient over the current batch, then updates exponentially decaying averages of past gradients (first moment) and their squares (second moment). These moment estimates are used to adaptively adjust effective learning rates for each parameter, enabling efficient and stable convergence during training. The mathematical formula for updating the weights is show in equation 3.

$$\begin{aligned} v_t &= \beta_1 v_{t-1} - (1-\beta_1) g_t \\ s_t &= \beta_2 s_{t-1} - (1-\beta_2) g_t^2 \\ w_{t+1} &= w_t - \eta \frac{v_t}{\sqrt{s_t + \varepsilon}} g_t \end{aligned} \tag{3.3}$$

  where g_t is gradient at time t along the weight, v_t is exponential average of gradients along the weight, s_t is average of square of gradients along the weight and β_1 and β_2 are two hyper parameters.

- The model goes through the dataset multiple times (epochs) to enhance its performance. With each iteration, the weights are updated progressively, improving the model's ability to generalize to new data.
- The learning process is tracked by evaluating the model on a separate validation dataset, which helps prevent overfitting and ensures reliable generalization to unseen data.

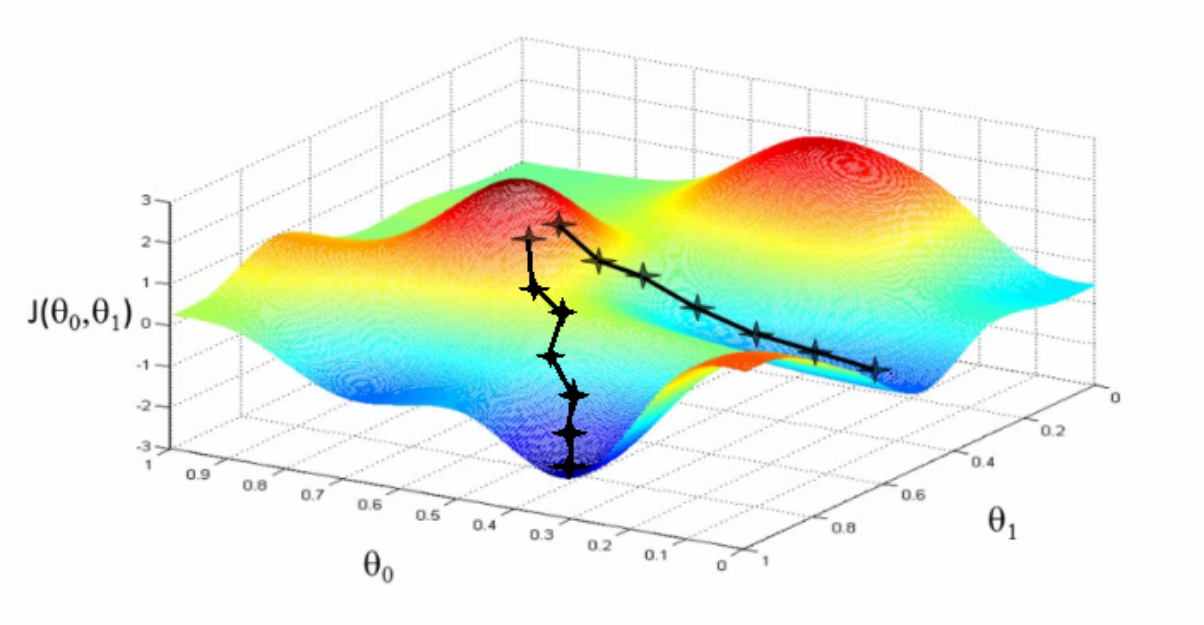


Figure 3.1: Example of a loss function (J) represented on the z-axis, depicting how it varies with different combinations of weights θ_0 and θ_1. The black line illustrates the path of gradient descent as it moves toward the global minimum. (Credit: Howie Choset)

3.3 Graph Neural Networks (GNN)

Graphs are a type of data structure that represents a collection of objects and their relationship through nodes and edges. Graph structures are well-suited for capturing complex relationships and dependencies between objects, making them essential for accurately representing physical data. Unlike other data representations such as vector-like or grid-like structures, graphs can naturally accommodate variable sized data and hence, eliminating the need for zero padding or the loss of information as required by fixed size images.

Building on the success of convolution neural networks (CNNs) for structured grid-based data, such as image pixels, graph neural networks (GNNs) are designed to extend these powerful methodologies to handle irregular, graph-like data structures.

Most graph-based learning algorithms in HEP are dedicated to reconstruction (clustering), identification (classification) and regression tasks such as mass and energy prediction. While

these tasks have fundamentally distinct objectives, their associated GNN-based workflows share structural similarities. In most cases, the inputs are a set of unordered data points like the position of hits and amplitude of the signals in a detector. During graph construction, these points are represented as graph nodes, with edges connecting them to convey relational information. GNNs generate predictions at the node, edge, or graph level, which can then be utilized by a subsequent post-processing algorithm.

3.4 Dynamic Reduction Network

For our studies, we have used a Dynamic Reduction Network (DRN) that has the following architecture :

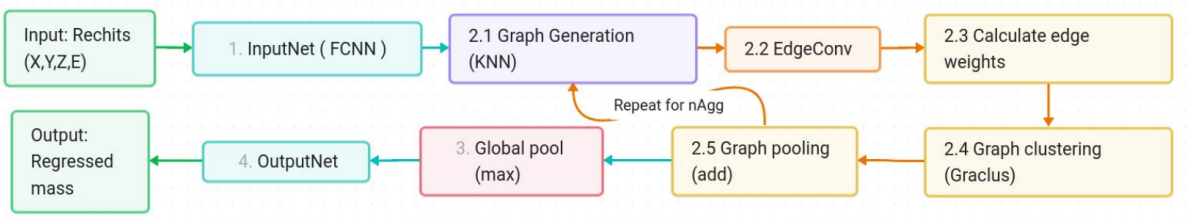


Figure 3.2: A figure showing the architecture of the Dynamic Reduction Network.

The details of all the steps involved in the DRN are given in the following sections.

3.4.1 InputNet

A point cloud of the rechit coordinates and energies along with a subdetector flag is given as input to the DRN. The inputs are mapped to a higher dimensional latent space using a Fully Connected Neural Network (FCNN) where more complex and informative features could be constructed from the non linear combinations of low level rechit information. The number of such features to be constructed is dependent on the dimensionality of latent space (in our case, it is 64).

3.4.2 Graph generation

In our case, the rechits serve as nodes for constructing the graphs. To construct a graph, the k-nearest neighbors of each node are connected in a higher-dimensional feature space. Formally, given a point cloud consisting of n points, each represented by a vector in $\mathbb{R}^F$, denoted as $X = x_1, x_2, \ldots, x_n$, a directed graph $G = (V, E)$ is constructed to capture the local structure of the point cloud. Here, $V = \{1, \ldots, n\}$ represents the vertices, and $E \subseteq V \times V$ represents the edges. The graph G is initially formed as an undirected k-nearest neighbor (k-NN) graph of the dataset

X, embedded in a F dimensional latent space. It includes connections where each node is also linked to itself (self loops). This process generates a set of associated nodes for each point in the graph.

3.4.3 Graph Convolution [9]

Once the graph is constructed, the edge features (e_{ij}) between pairs of nodes are defined using the EdgeConv operation as follows:

$$e_{ij} = h_\theta(x_i, x_j) \tag{3.4}$$

where h is a non-linear function with a set of learnable parameters (θ)

$$h : R^F \times R^F \rightarrow R^F \times R^F \tag{3.5}$$

Since, we do not know the form of the function h, a FCNN with a set of learnable parameters is used to mimic it [6], as shown in figure 3.3.The objective is to design a function that effectively captures all the information from the connected nodes. This information is then aggregated using the 'EdgeConv' operation and combined with the central node's own features, enabling the node to learn about its surrounding nodes.

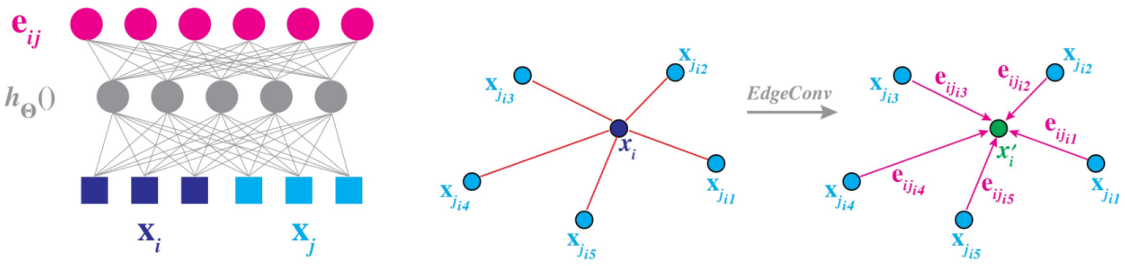


Figure 3.3: An example of a function processing the features of two nodes, x_i and x_j. In this case, the feature representation resulting from their interaction has a dimensionality of 6, which is twice the size of the original feature space ($F = 3$). [9]

The EdgeConv operation applies a symmetric aggregation operation $\square$ to the channel-wise edge features of all edges emerging from each vertex.

$$x'_i = \underset{j:i,j\in E}{\square} h_\Theta(x_i, x_j) \tag{3.6}$$

Similar to how convolution operates on data grids, we consider x_i as the reference point and $x_j : (i, j) \in E$ as the surrounding points. This approach enables the i-th node to be updated using information from its connected neighbors. In our case, the edge features for each message

passing layer are defined as follows :

$$e'_{ijm} = ReLU(\theta_m.(x_i - x_j) + \phi_m.x_i) \tag{3.7}$$

where we have used the rectified linear unit ($ReLU$) activation function and $\Theta = (\theta_1, \theta_2, ..., \theta_m, \phi_1, \phi_2, ..., \phi_m)$ is the set of trainable parameters and each $\theta_m \in R^F$. Here m represents the output dimensions. Increasing the number of message passing layers increases the complexity of the edge features. Finally, the edge features of a node are updated by combining the edge features of its neighboring nodes.

$$x'_{im} = \max_{j:ij \in E} e'_{ijm} \tag{3.8}$$

The output represents an M-dimensional point cloud that maintains the same number of nodes as the original graph. The operator exhibits a partial translation invariance property, as the chosen edge functions explicitly highlight the translation-dependent component, which can be optionally disabled if needed. Consider a translation T applied to x_i and x_j, then the edge features will transform as follows :

$$\begin{aligned} e'_{ijm} &= ReLU(\theta_m.(x_i + T - x_j - T) + \phi_m.x_i + T) \\ e'_{ijm} &= ReLU(\theta_m.(x_i - x_j) + \phi_m.x_i + T) \end{aligned} \tag{3.9}$$

Taking $\phi_m = 0$ can make the operator completely translation invariant. Keeping the model partially translation invariant allows it to take into account the local geometry while maintaining the global shape information. Before proceeding to clustering, the normalized edge weights are calculated as follows :

$$E_{ij} = e_{ij}(\frac{1}{deg(i)} + \frac{1}{deg(j)}) \tag{3.10}$$

In this context, E_{ij} denotes the computed edge weight between the nodes i and j, while e_{ij} represents the distance between these nodes in the feature space. Additionally, $deg(i)$ and $deg(j)$ indicate the connections associated with nodes i and j, respectively.

3.4.4 Graph Clustering [10]

After applying graph convolution, the graph is clustered and the nodes within individual clusters are pooled using the edge weights. This approach allows relevant local information to be incorporated into broader, intermediate feature representations. Graph clustering serves as an effective method for organizing nodes. Recent techniques have been designed to effectively manage complex, non-linearly separable data. In this work, we use Graclus, a multilevel algorithm that can perform clustering without the need to compute the eigenvectors.

It is an effective clustering tool designed for partitioning unlabeled data through graphical representations.
Graclus involves three primary stages. First, Graph Coarsening is performed, where the graph is iteratively reduced to smaller representations. At each level, nodes are merged to form supernodes in the next level, with edge weights being summed from the original graph.
Next, Base Clustering is applied to this simplified graph. Clusters are formed by selecting random nodes and expanding regions around them. Each node is either connected to a randomly chosen neighbor or the neighbor with the highest edge weight. If a node has no unclustered neighbors, it forms a standalone cluster. This process typically reduces the number of nodes by around fifty percent, though the reduction can be less if isolated nodes form their own clusters. This adaptive node reduction is why the architecture is referred to as a Dynamic Reduction Network.
Finally, Graph Refinement is performed as the graph is progressively uncoarsened. The clustering is further improved using an iterative approach inspired by the weighted kernel k-means algorithm to optimize the partitioning. The algorithm incorporates these objectives into a weighted kernel k-means framework, employing an iterative, k-means-like approach to optimize the clustering process.

3.4.5 Graph Pooling and OutputNet

The number of nodes in the final graph is not fixed and can vary based on the number of rechits and the number of iterations in the graph generation and graph convolution steps. Each node l_i (where i=1,2,...), in the final graph, is associated with F features $x_1.x_2,\ldots,x_F$. A global max pooling operation is performed by selecting the maximum value of each feature x_j (for $j = 1,2,\ldots,F$) across all nodes. The final output is a single node with F features, where each feature corresponds to the maximum value observed across the variable set of nodes. Finally this node's features are then passed through a FCNN known as the output network, which combines the F different features to generate the final result of the analysis as a single numerical value.

Application

The DRN framework described in this chapter has been utilized for reconstructing the mass of light pseudoscalars decaying into photon pairs in the CMS ECAL, as presented in Chapter 4, and for electron and photon energy regression, as detailed in Chapter 5.

Chapter 4

Merged photon reconstruction in CMS Electromagnetic Calorimeter (ECAL)

Reconstruction of mass of highly boosted light pseudoscalars (a) decaying to two photons is limited by the transverse granularity of the present CMS electromagnetic calorimeter (ECAL). For as that have Lorentz boost (γ_a) between 50 and 250, the showers start overlapping laterally. This is called shower merging. To account for photon conversions, the PF algorithm reconstructs the shower-merged photons as single photon candidates. For pseudoscalars with Lorentz boosts beyond 250, the photons deposit their energy into the same ECAL crystals and hence such scenarios are indistinguishable from single photon showers. This is called instrumental merging. The opening angle (θ) between the two photons is given by the following equation:

$$cos(\theta) = 1 - \frac{m_a^2}{2E_{\gamma 1}E_{\gamma 2}} \tag{4.1}$$

Using the property of arithmetic mean $\geq$ geometric mean, the upper bound on cosine of the opening angle between the photons is given as :

$$\frac{E_{\gamma 1} + E_{\gamma 2}}{2} \geq \sqrt{E_{\gamma 1}E_{\gamma 2}} \tag{4.2}$$

$$\because E_{\gamma 1} + E_{\gamma 2} = E_a \tag{4.3}$$

$$\implies 4E_{\gamma 1}E_{\gamma 2} \geq E_a^2 \tag{4.4}$$

$$\implies cos(\theta) < 1 - \frac{2}{\gamma_a^2} \tag{4.5}$$

where γ_a is the Lorentz boost of a, defined as $\frac{E_a}{m_a}$. Figure 4.1 shows the distribution of opening angles between the photons as a function of mass of as for a sample generated uniformly distributed in p_T and η (flat samples).

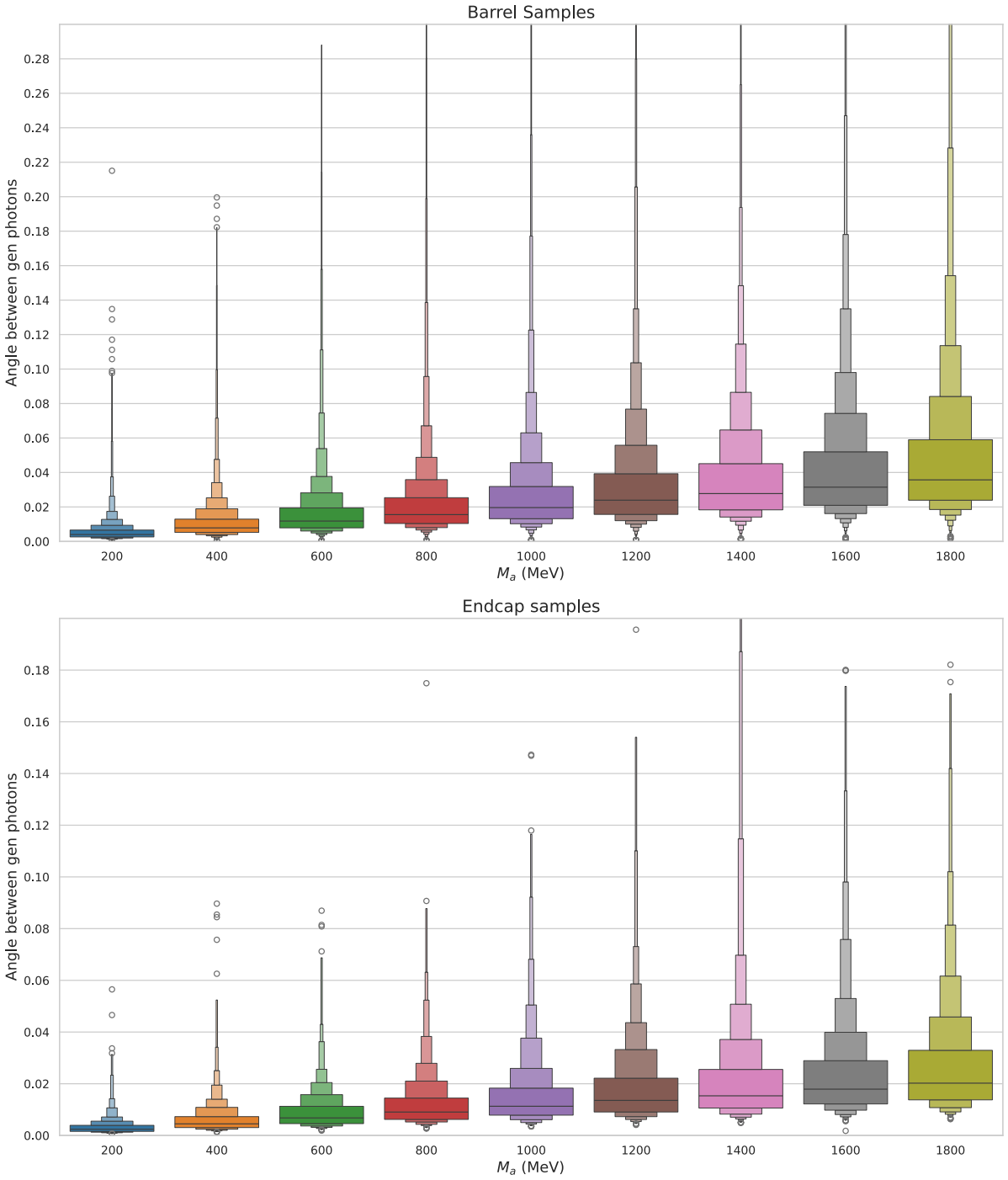


Figure 4.1: Boxen plot showing opening angles (y-axis) between the two photons in $a \to \gamma\gamma$ as a function of mass of a for the barrel (top) and endcap (bottom) samples. Each box illustrates the interquartile range (IQR) of angles, which is the range between the first (25th percentile) and third (75th percentile) quartiles, with the line inside the box representing the median.

It is worth noting that the opening angle between the two photons is significantly smaller in the

endcaps compared to the barrel. This is because, for a given transverse momentum (p_T), the longitudinal momentum increases with pseudorapidity (η), as $p_T = p/\cosh(\eta)$, resulting in more boosted pseudoscalars in the endcap region. Consequently, the decay photons are more collimated in the endcaps, leading to a higher degree of merging. This distinction is particularly important since both triggers and analysis object selections are typically based on p_T and not on the total momentum p.

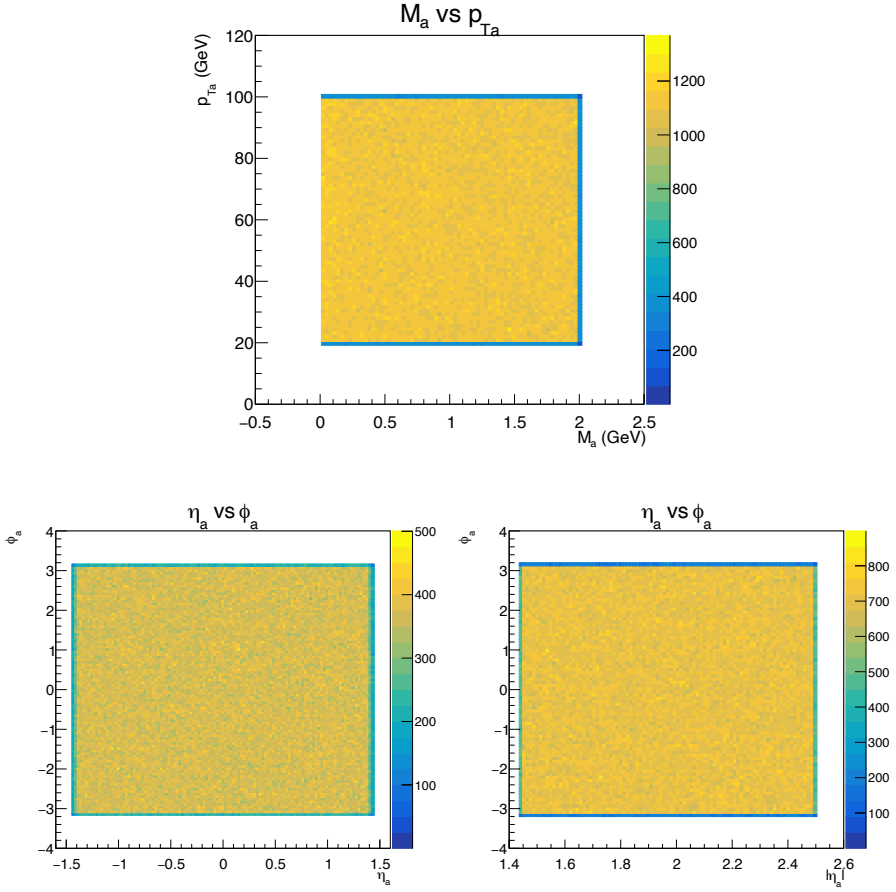
We are using a dynamic reduction network (DRN) trained on low level rechit information, to reconstruct the mass of the pseudoscalar. Two different DRN models are used - one for ECAL barrel and one for ECAL endcaps.

4.1 Sample production

To train the model, $a \to \gamma\gamma$ particle gun samples are generated in the following kinematic phase space (shown in figure 4.2) using the CMS software framework (CMSSW).The sample events are overlaid with additional interactions (pileup) to replicate conditions expected in real proton-proton collisions. The sample production steps are briefly discussed in the following subsections. The mass of these pseudoscalars is uniformly varied from 0 to 2 GeV. The kinematic ranges of the samples used for are as follows :

Mass of a (m_a): 0 GeV $< m_a <$ 2 GeV
Transverse momentum of a (p_{T_a}): 20 GeV $< p_{T_a} <$ 100 GeV
Pseudorapidity of a (η_a): For barrel: $|\eta_a| < 1.44$; For endcaps: $1.44 < |\eta_a| < 2.5$
Phi of a (ϕ_a) : $|\phi_a| < \pi$

Figure 4.2 presents 2D histograms illustrating mass versus transverse momentum and η versus ϕ. Flat mass samples are used for training the models to ensure that the model learns uniformly, preventing biased predictions toward specific mass points.



(a) Barrel samples (b) Endcap samples

Figure 4.2: 2D histograms showing the distribution of generator level kinematic variables of the $a \to \gamma\gamma$ samples. There are no overflow or underflow of events in these plots.

4.1.1 Generation and Simulation (GEN-SIM)

The PYTHIA8 p_T gun is used to simulate the $a \to \gamma\gamma$ events within the specified kinematic ranges as discussed in section 4.1. Pileup events are simulated to account for multiple physics interactions occurring within a single beam crossing due to the high particle density in a bunch, replicating conditions observed in real pp collisions. The generated stable final state particles are passed through GEANT4 for detailed simulation of response of various sub-detectors in CMS.

4.1.2 Digitization (DIGI)

The simulation process generates data represented by signals distributed across various parts of the detectors.. Neutrino samples from the following dataset were overlaid with the simulated samples to simulate the low energy min bias events:

`/Neutrino_E-10_gun/RunIISummer20ULPrePremix-UL18_106X_upgrade2018_realistic_v11_L1v1-v2/PREMIX`

The digitization step models the response of detector readout electronics. The effects of

electronic signal spills from the previous bunch crossing into the current one, can be accounted for by the pileup simulations. This usually happens when the duration of the electronic signal in the subdetector is longer than the LHC bunch spacing of 25 *ns*.

4.1.3 Reconstruction (RECO)

The reconstruction step starts with reading out the digitized samples. The signal amplitude for each detector channel or cell is inferred from the digitized samples after applying relevant calibrations. A detector cell with its signal amplitude and position coordinates is called a "rechit". The information from the rechits is used to reconstruct physics objects like electrons, photons, muons, jets and missing transverse energy (MET). The CMS particle flow (PF) algorithm integrates the information reconstructed from multiple subdetectors to achieve optimal particle reconstruction with superior resolution and efficiency, as outlined in section 2.3

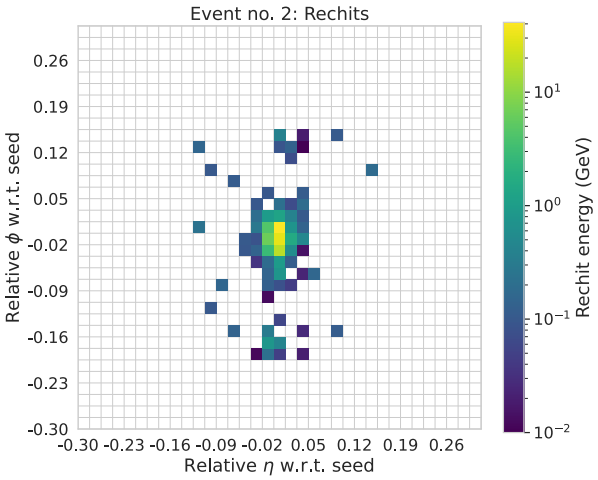
4.2 Event selection

To obtain a robust sample of $a \rightarrow \gamma\gamma$ events for training the model, the following selection criteria have been established:
• The events must have two photons at generator level.
• The reconstructed photon(s) must have transverse momentum (p_T) > 10 GeV.

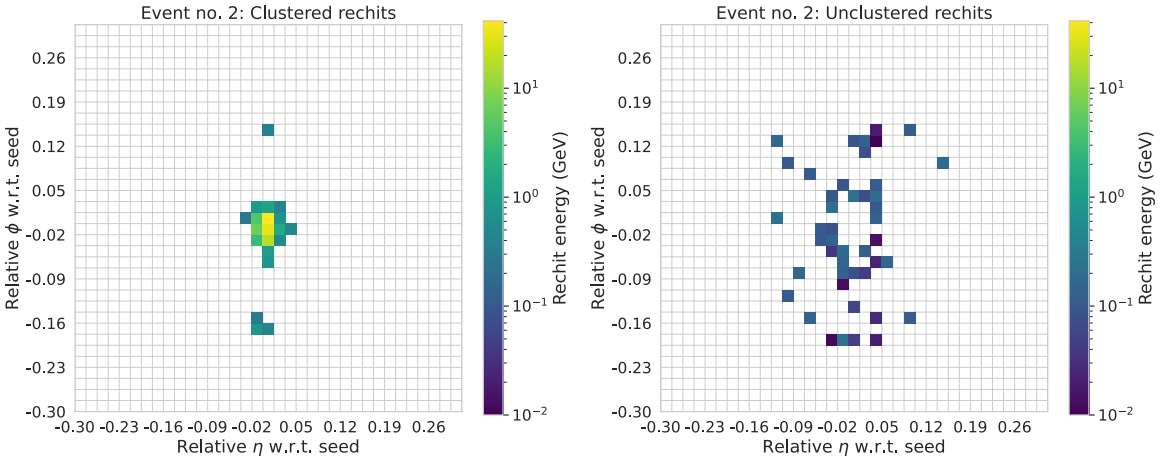
These selections result in a rejection of approximately 3.5% of the total events.
Finally, the cartesian coordinates (x, y, z) and energies of the rechits associated with the photon objects are stored as tensors in pickle files, making them easily accessible for the ML model. In the endcaps, and additional flag is given to the rechits from the preshower. For our final training the longitudinal and transverse components of the energy were given as separate inputs (refer to section 4.4.9.

4.3 Model inputs

It was observed that the unclustered rechits contain some information about the mass of the pseudoscalar. Therefore rechits from the photon supercluster in the ECAL, along with unclustered rechits within a dR of 0.3 around the supercluster seed, are used for training the model.



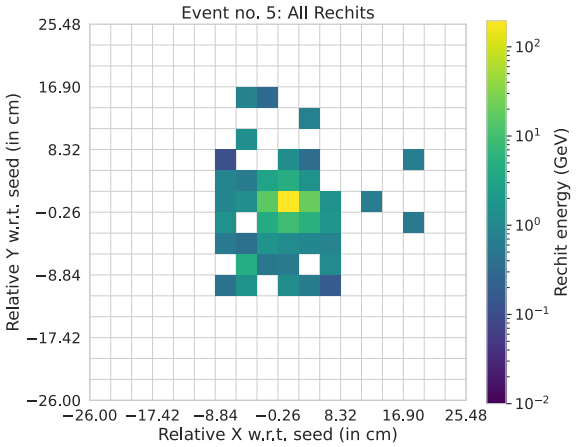
(a) All rechits



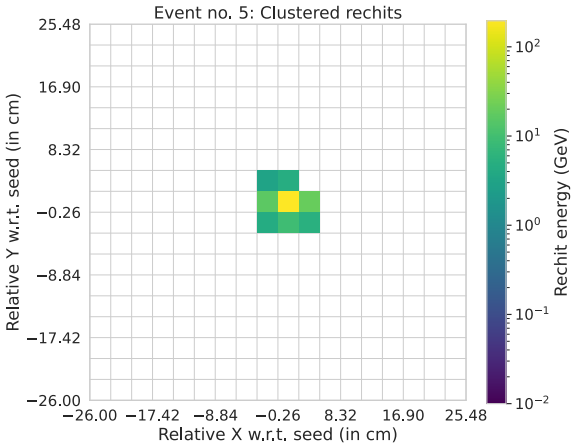
(b) Clustered rechits

(c) Unclustered rechits

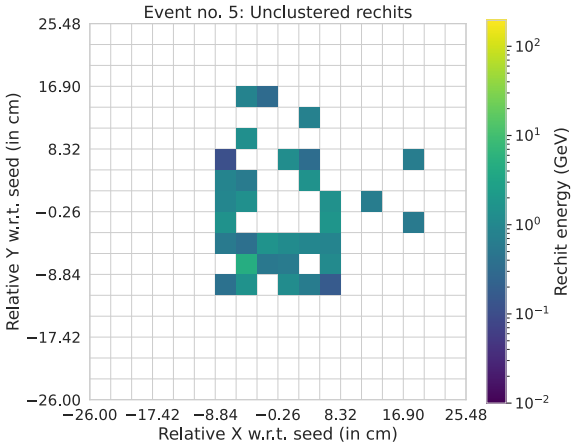
Figure 4.3: Distributions energy deposits in the $\eta - \phi$ plane, by the photons in the barrel region. All rechits contained within a cone of $\Delta R = 0.3$ are displayed in the figure 4.3a .The bottom left figure presents the clustered rechits (4.3b), whereas the bottom right figure illustrates the unclustered rechits (4.3c). Each box in the grid represents the front face of an individual ECAL crystal in the barrel.



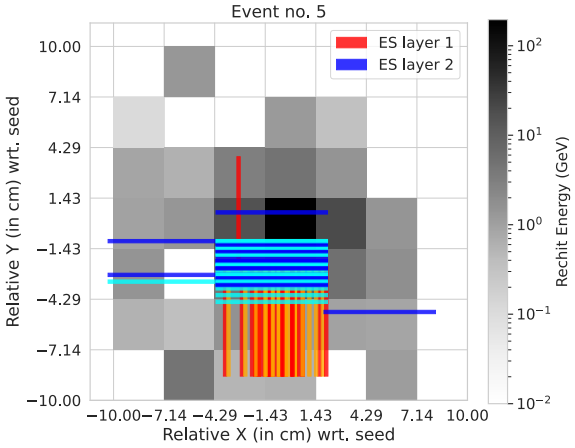
(a) All rechits



(b) Clustered rechits



(c) Unclustered rechits



(d) EE and ES granularitites

Figure 4.4: Distributions of energy deposits in the x-y plane by the photons in the endcaps. All the rechits within a ΔR cone of 0.3 are shown in figure 4.4a . Figure 4.4b displays the clustered rechits, while figure 4.4c shows the unclustered rechits. Figure 4.4d compares the granularity of the preshower and the ECAL endcap crystals. The red and orange lines represent the strips in the first layer of the ES while the light and dark blue lines represent the strips in the second layer of the ES. Each box in the grid represents the front face of an individual ECAL crystal in the endcaps. Note: for the preshower, only the strips where some energy is deposited, are shown in the figure 4.4d

From figure 4.4d, we observe that the preshower is a fine grained detector with a granularity of 1.9 mm in one direction and 6.5 cm in the other. The length of a preshower strip is ~1cm longer than two ECAL crystals but its width is less than $\frac{1}{10}$th of the width of ECAL crystals. The energy deposited in the preshower is negligible compared to the energy deposited in the ECAL crystals. The main purpose of the preshower is to initiate the showers before the ECAL crystals to make up for the lower granularity in the endcaps.

From figures 4.3 and 4.4, it is evident that the energy deposits from photon showers are more spatially spread out in the barrel region of the detector. This wider distribution leads to energy being deposited over a larger number of crystals, which in turn results in a greater number of recorded rechits in the barrel. In contrast, in the endcap region, the energy deposits tend to be more localized due to higher boosts of particles, lower granularity of the detector and higher noise levels, leading to fewer but more concentrated rechits.

We use the rechit coordinates (x,y and z) and energies as inputs to the model. For the endcaps, we give the preshower rechits a separate flag. The rechit coordinates are rescaled between 0 to 1. The ECAL rechit energies are rescaled between 0 to 100. The preshower rechit energies are rescaled separately among themselves between 0 to 100.

4.4 Model Optimizations

The hyper parameters of the Dynamic Reduction Network were tuned to achieve optimal performance in terms of accuracy, computational efficiency and generalization capabilities. While models with a large number of parameters may achieve better performance, they typically require extensive datasets for training and significantly more time to train. Additionally, when such models are trained on smaller datasets, they are prone to over-fitting, which compromises their ability to generalize effectively. Our optimization efforts (described in the following subsections) focused on minimizing the number of parameters, ensuring that the model could learn and make accurate predictions while being trainable within a reasonable time frame, even with limited data.
For training the model, the entire dataset was split into a training sample (80 %) and validation sample (20%). The following hyperparameters were kept constant for all the variations of the model :

- Number of input layers : 3
- Number of output layers : 2
- Batch size (Training and validation): 200
- Learning rate : 0.0001
- Loss function : Mean square loss
- Pooling scheme : Max
- Optimizer : AdamW

The above hyperparameters were kept constant over all the model variants that had been tried in this project. The hyperparameters that were changed were :

- Number of message passing layers
- Number of aggregation layers
- Number of hidden dimensions

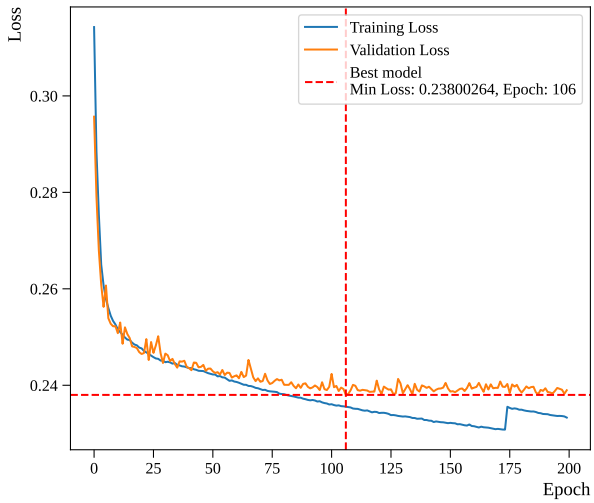
To evaluate the performance of the models, we consider two key diagnostics: the two-dimensional distribution of the predicted mass of the pseudoscalar aa versus its true mass, as shown in figure 4.5b, and the training and validation loss curves, as shown in figure 4.5a. In the 2D mass distribution, the y-axis represents the predicted mass and the x-axis the true mass, with the color scale indicating the number of entries per bin. Ideally, the highest bin densities should lie along the diagonal $y = x$ (highlighted by the pink line), indicating accurate predictions. Deviations from this diagonal signify discrepancies between the predicted and true masses. The loss curves, plotted as a function of training epochs, are used to monitor model convergence, compare final training and validation losses, and check for signs of overfitting.

4.4.1 Training on clustered rechits only

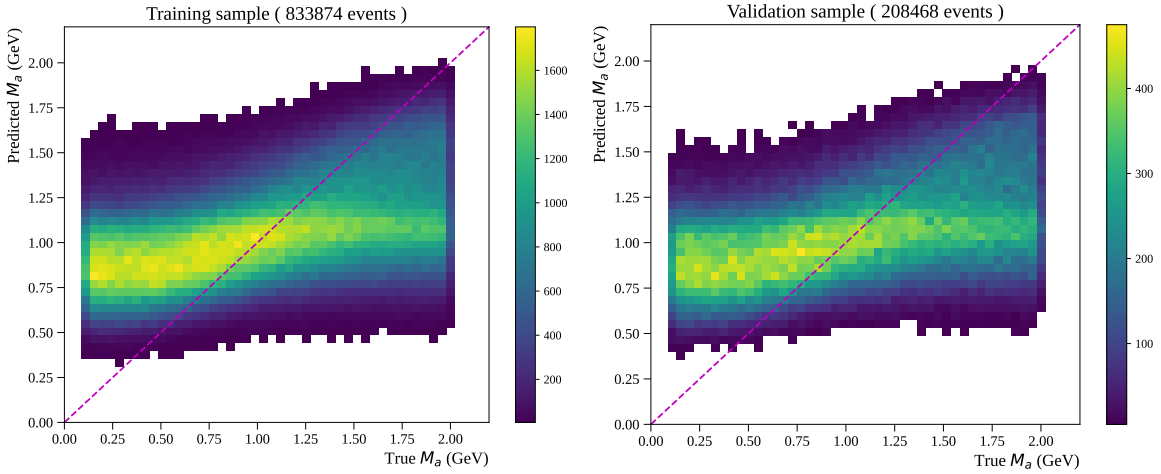
We started with a complex model with about five hundred thousand parameters. We initially started with training on the clustered rechits only. The exact hyperparameters (except the ones listed above) for the model are as follows :

- Number of message passing layers : 4
- Number of aggregation layers : 3
- Number of hidden dimensions : 128
- Total number of parameters : 512297
- Sample size : 1.074 M
- Sample mass range : $0.1 < M_a < 2$ GeV
- Region : Endcaps

Results



(a)



(b) (c)

Figure 4.5: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.5b) and the validation dataset (figure 4.5c). Additionally, the loss versus epoch plot (figure 4.5a) illustrates the model's optimal performance, corresponding to the epoch with the lowest validation loss.

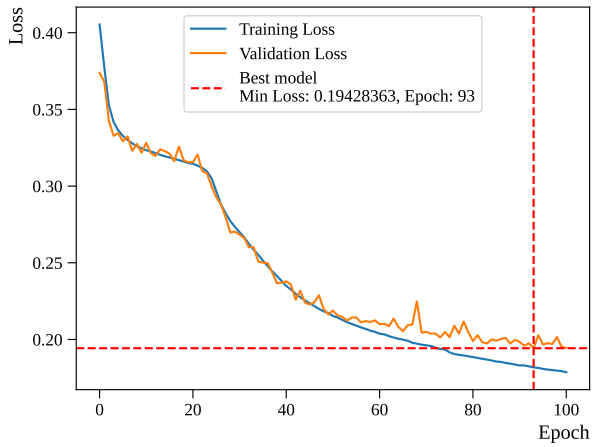
From figures 4.5b and 4.5c, it is evident that the model deviates significantly from the ideal $y = x$ line, with a clear tendency to underestimate the true mass, particularly at higher values. Despite being trained on a uniformly distributed mass spectrum, the model consistently predicts masses within the narrow range of 0.75–1.0 GeV, indicating that it fails to reproduce the full range of target values. This behavior suggests that the model tends to predict values closer to the center of the training distribution, rather than adapting to variations across the full spectrum. Such a trend points to limited learning capacity, possibly due to insufficient or uninformative

input features.

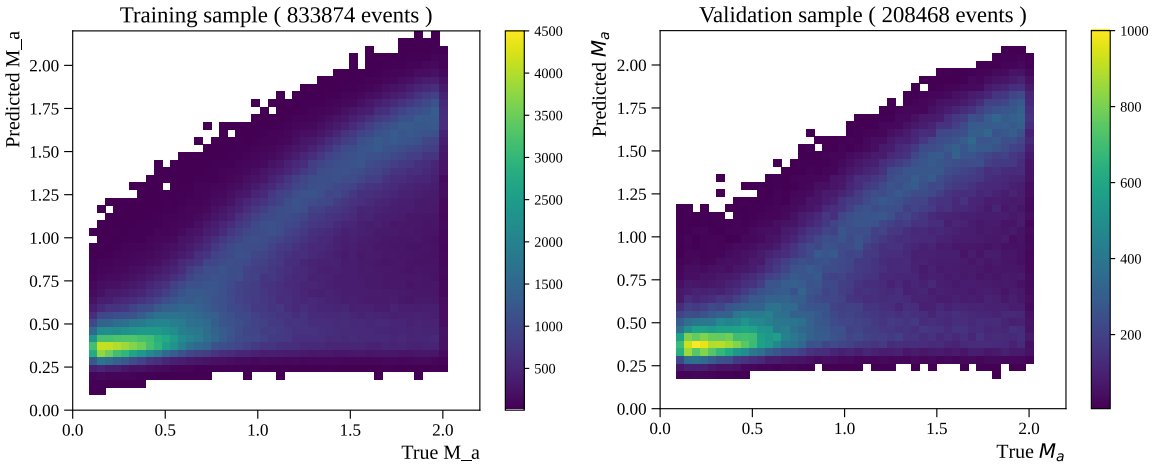
4.4.2 Training on supercluster + dR rechits

All rechits from the supercluster and the unclustered rechits within a ΔR cone of 0.3 were used to train the model. The model hyperparameters were same as in the previous section (section 4.4.1).

Results



(a)



(b) (c)

Figure 4.6: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.6b) and the validation dataset (figure 4.6c). Additionally, the loss versus epoch plot (figure 4.6a) illustrates the model's performance at its best epoch, corresponding to the lowest validation loss.

Compared to the previous results in figure 4.5 , the current model output (figure 4.6) exhibits a more noticeable diagonal trend along the $y = x$ line, indicating improved correlation between the predicted and true masses of the pseudoscalar a. While the predictions still show some bias and spread—particularly at low masses—the presence of a visible diagonal structure suggests that the model has learned to better capture the mass dependence, in contrast to the earlier model which predominantly predicted values clustered in a narrow band around 0.75–1.0 GeV with little variation across the true mass range.

4.4.3 Inclusion of high level input features

We incorporated high-level input features such as the supercluster pseudorapidity (η_{SC}), the event energy density (ρ), and the shower shape variable $\sigma_{i\eta i\eta}$, as described in section 2.3.3, to provide the model with additional contextual and shape-related information that could aid in improving its predictive performance.

Results

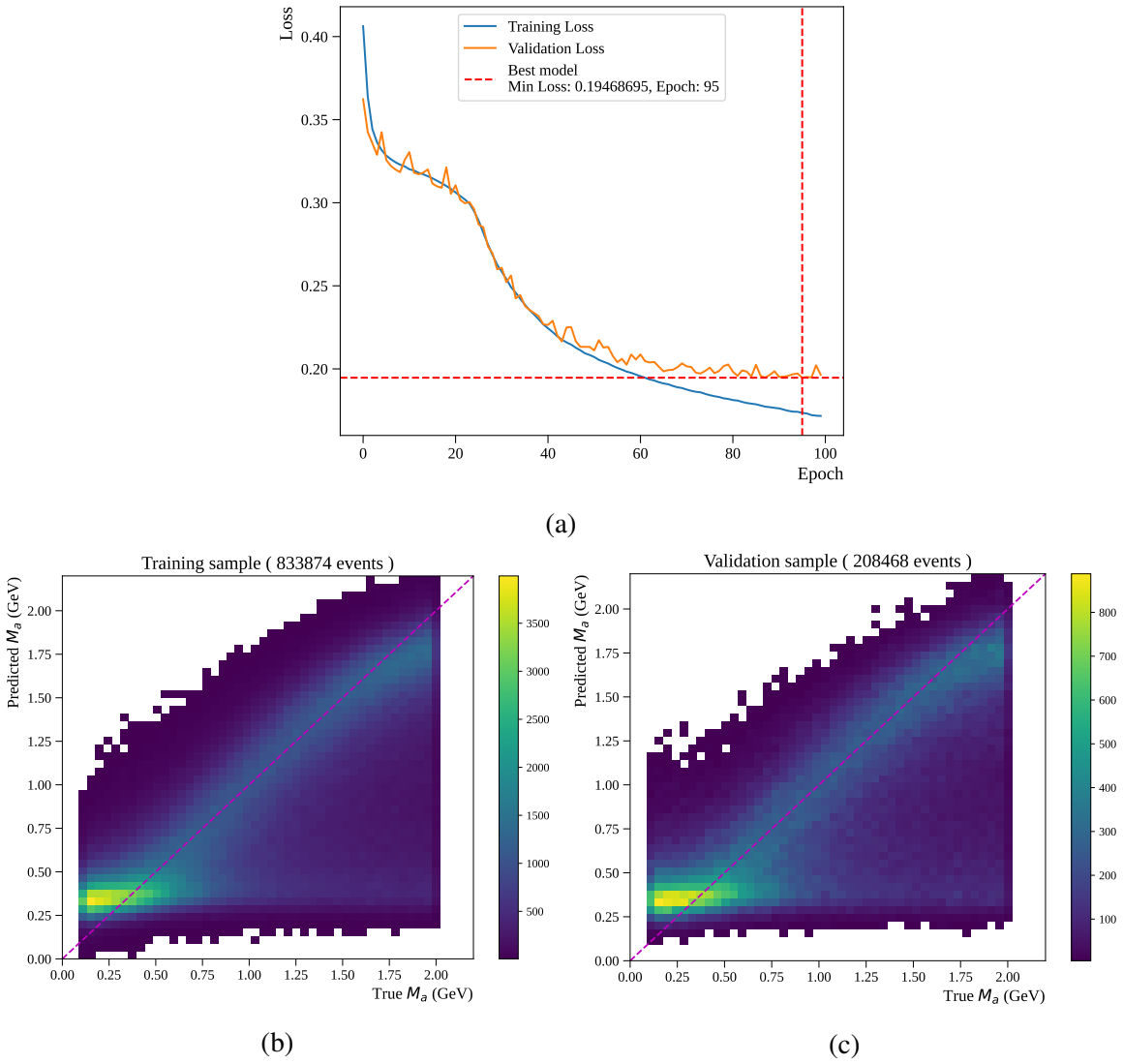


Figure 4.7: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.7b) and the validation dataset (figure 4.7c). Additionally, the loss versus epoch plot (figure 4.7a) illustrates the model's performance at its best epoch, corresponding to the lowest validation loss.

Although the 2D mass distributions of predicted vs true mass of a for this model (figures 4.7b and 4.7c) appear qualitatively similar to ones described in subsection 4.4.2 (figures 4.6b and 4.6c) —showing a diagonal trend that indicates some correlation between predicted and true masses—the slightly higher loss suggests that the overall prediction errors are marginally larger. This may be because the DRN performs a dimensional reduction in every pooling operation. Including unnecessary or repetitive features affects this pooling step as the unimportant features may also get some non-zero weights. This may be avoided by training the DRN longer and over

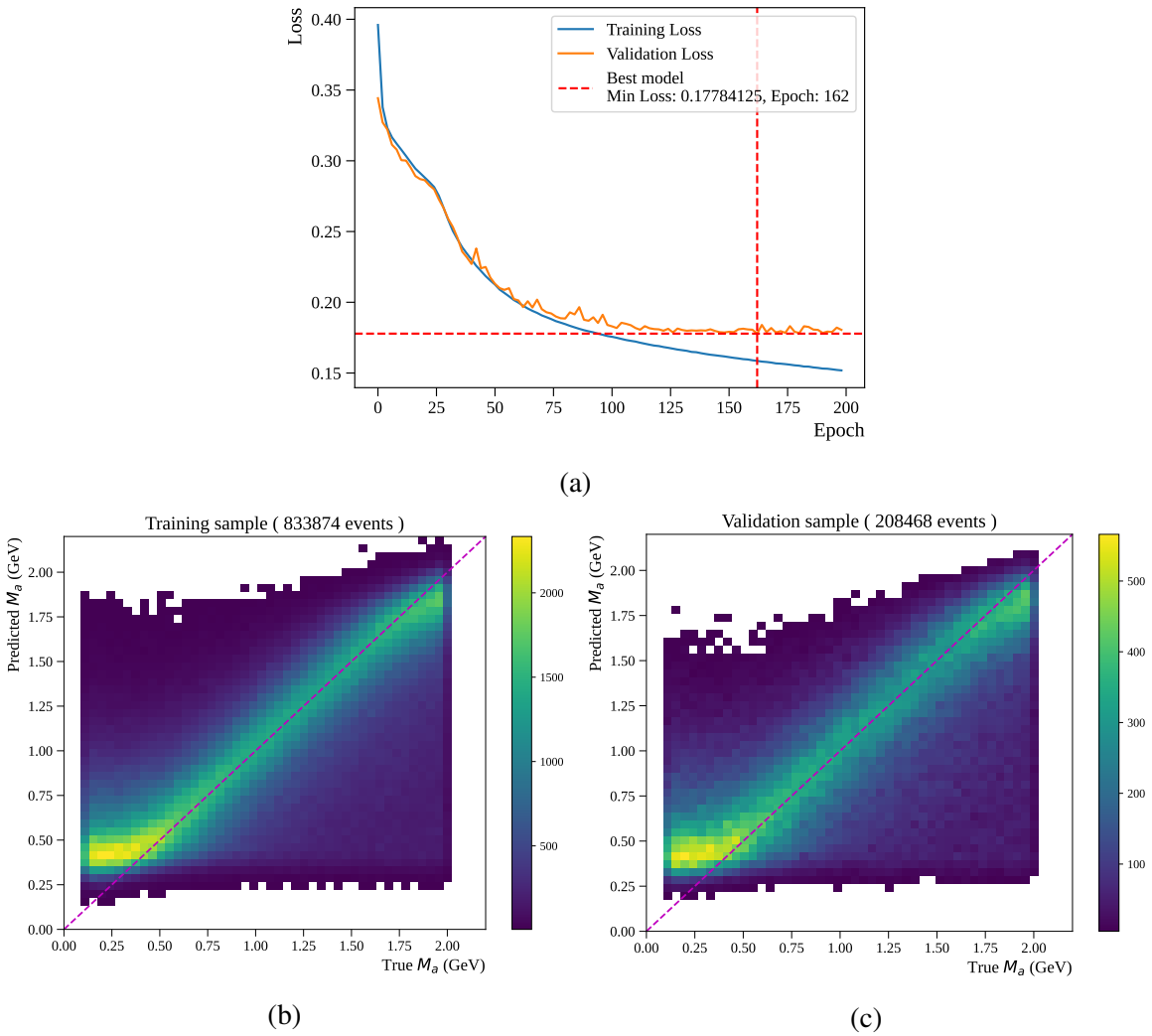
a much larger dataset, but that may be computationally expensive.

4.4.4 Changing number of hidden dimensions

Based on the previous results, we concluded that the model had too many parameters relative to the size of the data, making it difficult to optimize effectively. Hence, we tried reducing the number of hidden dimensions. :

- Number of message passing layers : 4
- Number of aggregation layers : 3
- Number of hidden dimensions : 64
- Total number of parameters : 186574
- Sample size : 1.074 M
- Sample mass range : $0.1 < M_a < 2$ GeV

Results



(a)

(b)

(c)

Figure 4.8: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.8b) and the validation dataset (figure 4.8c). Additionally, the loss versus epoch plot (figure 4.8a) illustrates the model's optimal performance, corresponding to the epoch with the lowest validation loss.

A comparison between the 2D mass distributions of this model (Figures 4.8b and 4.8c) and those presented in section 4.4.2 (Figures 4.6b and 4.6c) suggests that reducing the number of hidden dimensions from 128 to 64 enhances the overall diagonal trend. This indicates that the model captures the correlation between predicted and true masses more effectively. Nevertheless, a plateau-like region remains for true masses below 0.6 GeV, implying that the model continues to underperform in this low-mass regime. A direct comparison of the loss vs. epoch plots for the two models further supports this improvement, showing that the best model loss decreases by nearly 0.2 when the number of hidden dimensions is reduced.

This improvement can likely be attributed to a reduction in model complexity, which helps mitigate overfitting—especially when the training dataset is not large enough to fully constrain a higher-dimensional representation. By reducing the number of hidden dimensions from 128 to 64, the model is encouraged to learn more generalizable features rather than memorizing noise or spurious correlations in the training data.

4.4.5 Exclusion of preshower rechits

To check if the preshower rechits are providing any important information, we try to train the model with same hyperparameters as in section 4.4.4 but only on the ECAL endcap rechits.

Results

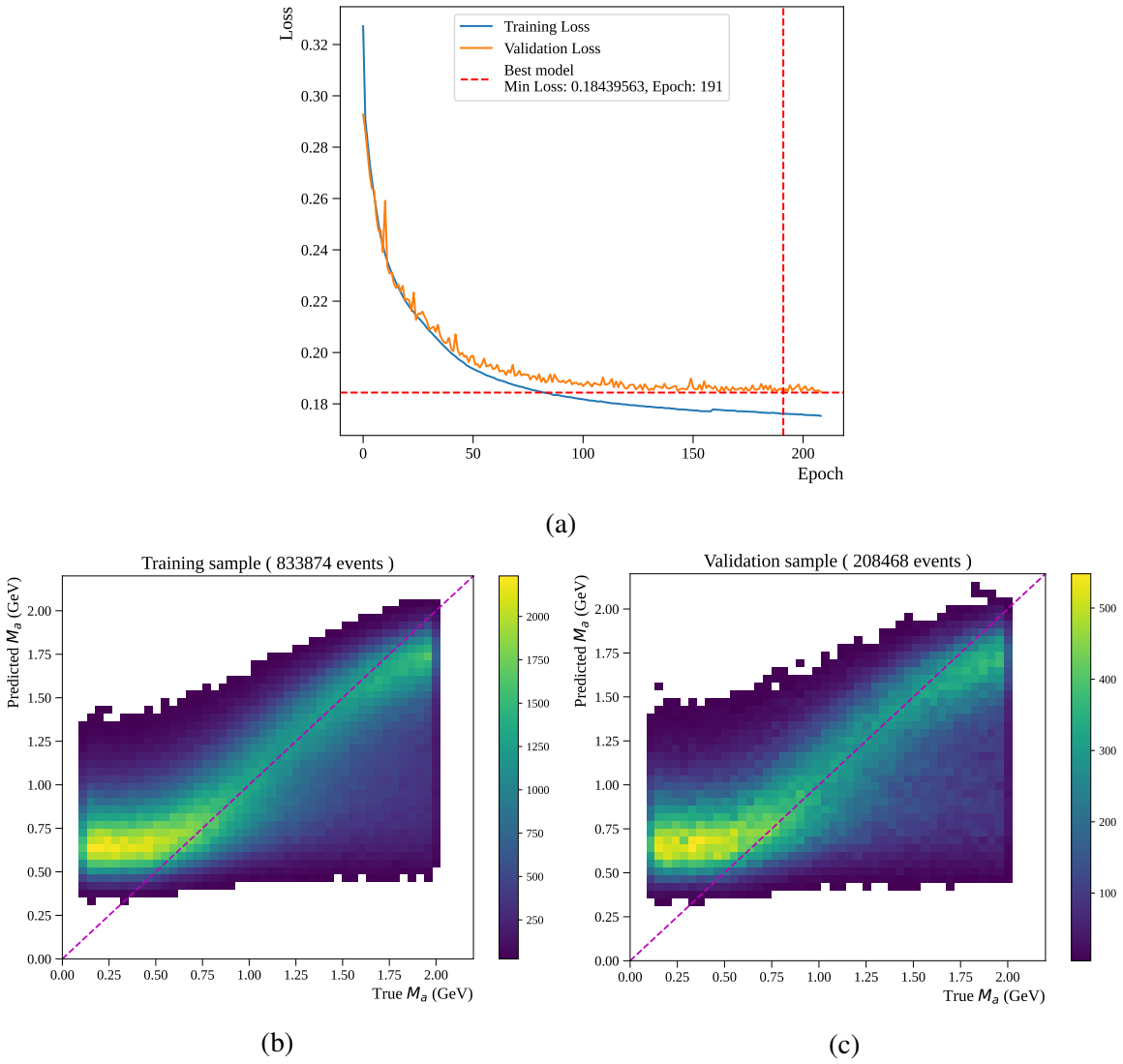


Figure 4.9: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.9b) and the validation dataset (figure 4.9c). Additionally, the loss versus epoch plot (figure 4.9a) illustrates the model's performance at its best epoch, corresponding to the lowest validation loss.

A comparison of the 2D distributions of predicted vs. true mass of a for this model (figures 4.9b and 4.9c) and the model described in section 4.4.4 (figures 4.8b and 4.8c) reveals that the exclusion of preshower rechits leads to a degradation in resolution. Additionally, for true masses below approximately 0.6 GeV, the model output saturates, predicting nearly constant values regardless of the input. This behavior suggests a reduced sensitivity and impaired learning in the low-mass regime.

A comparison of the loss vs. epoch curves for the two models (figures 4.9a and 4.8a) clearly

demonstrates that including preshower rechits significantly improves the model's performance. Specifically, the model trained with preshower information achieves a noticeably lower loss, highlighting the crucial role of preshower rechits in accurately reconstructing the mass of pseudoscalars.
A direct comparison of the response, standard deviation,and relative resolution for the ResNet-18 (a CNN) and DRN models—with and without the inclusion of the preshower—is presented in section 4.4.7.

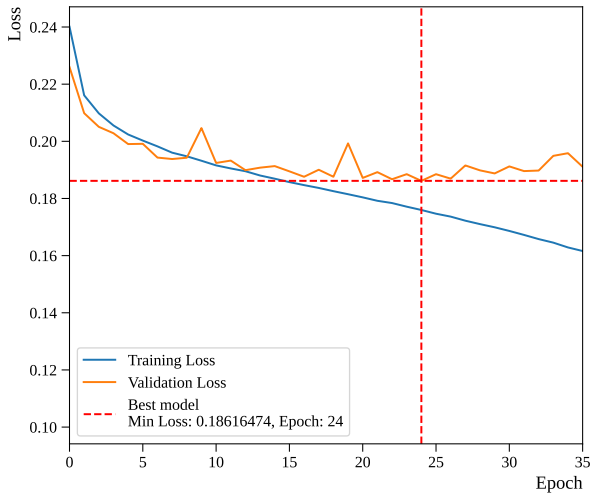
4.4.6 Using convolution neural network (CNN)

Based on the refrence [19], we also tried training a CNN, ResNet18 , for mass regression. The CNN was trained on only the ECAL rechits in the endcaps. The hyperparameters used in ResNet18 are as follows :

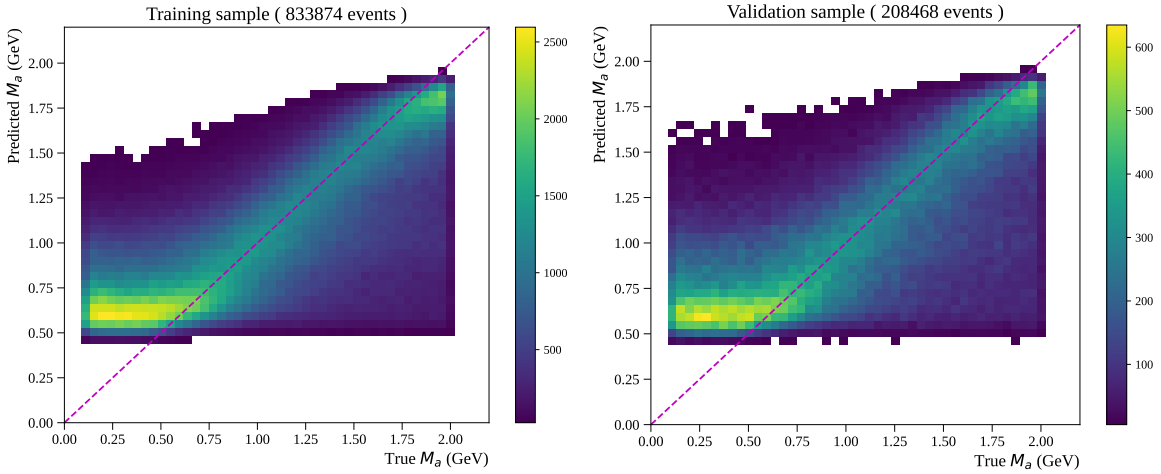
- Block used : Basic block (two 3×3 convolution layers)
- Activation function: ReLU
- Total number of parameters : 11173889

The full details about ResNet and its implementation in Pytorch can be found in Ref. [7] and Ref.[8].

Results



(a)



(b) (c)

Figure 4.10: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.10b) and the validation dataset (figure 4.10c). Additionally, the loss versus epoch plot (figure 4.10a) illustrates the model's optimal performance, corresponding to the epoch with the lowest validation loss.

The loss vs epoch plot for ResNet18 (figure 4.10a) shows that the model can get overtrained very easily as the number of trainable parameters is more than the number of training data points. Therefore, we terminate the training at 35 epochs to prevent overtraining.

The loss vs. epoch plots for ResNet-18 (Figure 4.10a) and the DRN model without preshower information (Figure 4.9a) show that the DRN consistently achieves a lower loss, indicating superior performance.

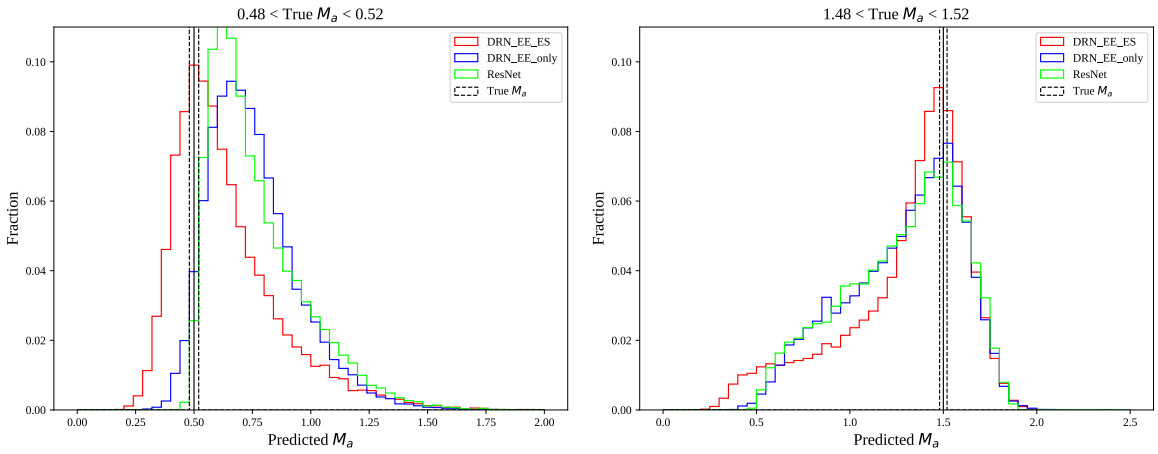
The 2D distributions of predicted vs. true mass of a (Figures 4.10b and 4.10c) reveal that

the CNN also fails to reconstruct masses below 0.6 GeV accurately, pointing to a persistent challenge in this low-mass region across different model architectures.

4.4.7 Comparing the models

We try to compare the performance of the DRN with and without preshower to that of ResNet18. We plot the response and relative resolution, on the validation dataset, of all the three models (DRN with preshower, DRN without preshower and ResNet18). The figures 4.11 and 4.12 below compares the 1D distribution, response, standard deviation and relative resolution of the DRN and ResNet18.

For each mass point, predicted masses corresponding to data points within a 40 MeV window of the true mass, are collected and plotted as a histogram. A Gaussian function is then fitted around the peak of each histogram, with the mean of the fit (μ_{fit}) representing the model's response at that mass point (Figure 4.12a). The standard deviation of the fit (σ_{fit}) is taken as the model's resolution and is shown in Figure 4.12b. The ratio of the standard deviation to the mean ($\sigma_{\text{fit}}/\mu_{\text{fit}}$), representing the relative resolution, is plotted in figure 4.12c. The 1D distributions of the predicted masses—binned in 40 MeV intervals—for the final models described in section 4.5, along with the corresponding fits, are presented in figures 4.16 and 4.18, for the barrel and endcap regions, respectively.



(a) (b)

Figure 4.11: Comparing the 1D mass prediction at 0.5 GeV and 1.5 GeV (in 40 MeV bins)

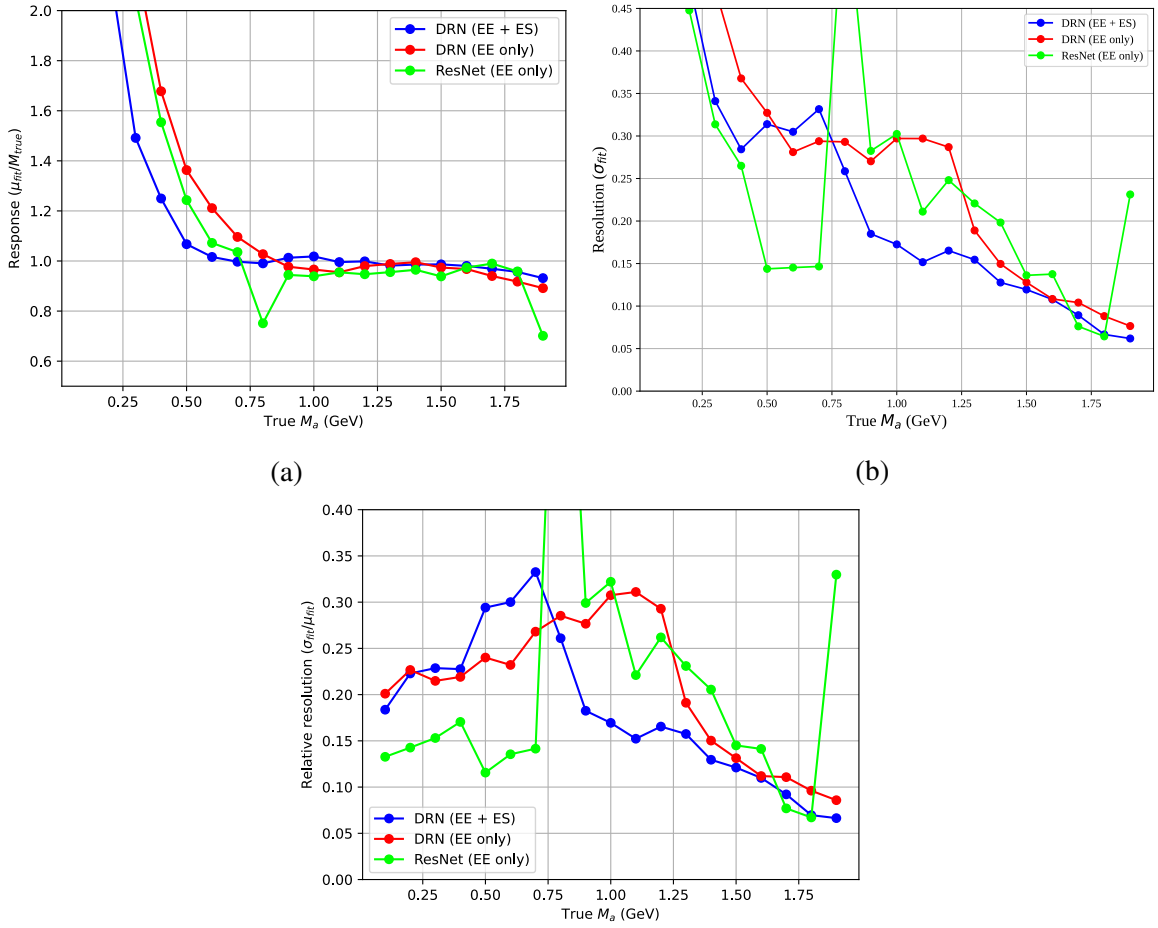


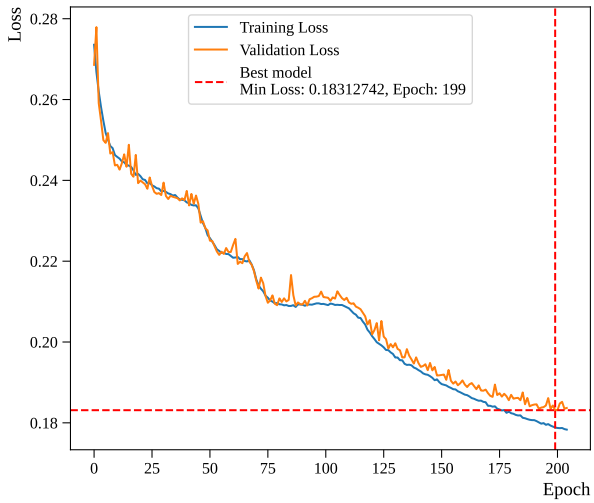
Figure 4.12: Comparing the response 4.12a, standard deviation 4.12b and relative resolution 4.12c of the models.

The figure 4.12 shows that the DRN performs better than ResNet18 in the endcaps. Although, it appears as if ResNet18 has slightly better resolution than DRN for lower masses ($m_a < 0.6GeV$), we see that its response is bad as ResNet tends to overpredict the mass. Finally, the DRN performs best when it is given the information about preshower rechits.

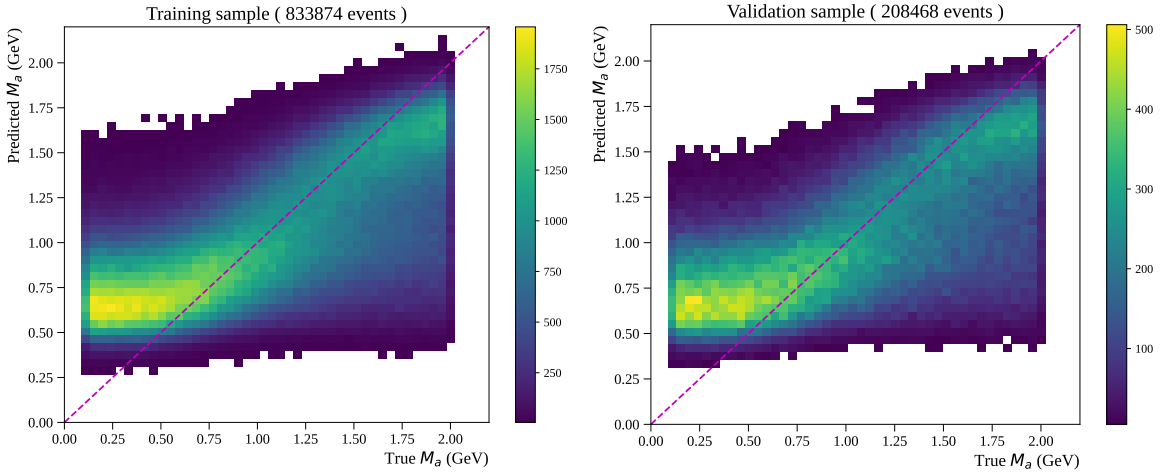
4.4.8 Giving noise as additional inputs

Seeing that the unclustered and preshower rechits improved the DRN training results, we also tried using the noise pedestal as an additional input feature to the DRN. The hyperparameters remain the same, as stated in section 4.4.4.

Results



(a)



(b) (c)

Figure 4.13: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.13b) and the validation dataset (figure 4.13c). Additionally, the loss versus epoch plot (figure 4.13a) illustrates the model's optimal performance, corresponding to the epoch with the lowest validation loss.

The 2D mass distributions of predicted vs. true mass of a for the DRN model with noise pedestals as input (figures 4.13b and 4.13c) exhibit degraded resolution and noticeable deviations from the ideal diagonal trend compared to the DRN model without noise pedestals described in section 4.4.4 (figures 4.8b and 4.8c). This suggests that incorporating noise pedestals adversely affects the model's ability to reconstruct the mass of a accurately.

4.4.9 Giving E_T and E_z as separate inputs

We attempted to provide the transverse (E_T) and longitudinal (E_z) energy components as separate inputs to capture information about the azimuthal angle (θ). The other hyperparameters for the model, remain the same as stated in section 4.4.4.

Results

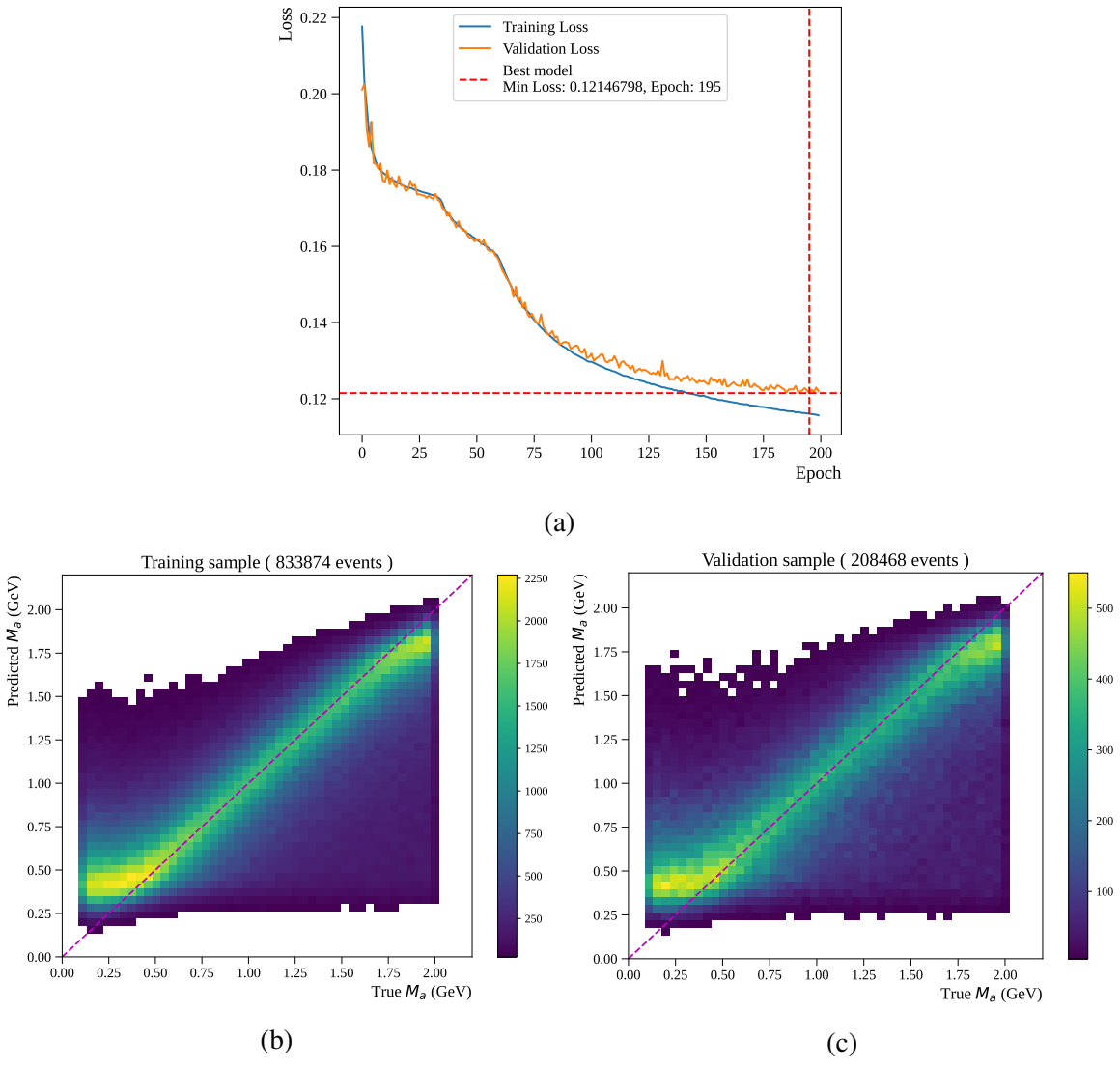


Figure 4.14: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.14b) and the validation dataset (figure 4.14c). Additionally, the loss versus epoch plot (figure 4.14a) illustrates the model's optimal performance, corresponding to the epoch with the lowest validation loss.

The 2D mass distributions of predicted vs. true mass of a for this model (figures 4.14b and 4.14c) exhibit improved resolution compared to the model described in section 4.4.4 (figures 4.8b and

4.8c). Despite this improvement, the model continues to struggle in accurately reconstructing masses below 0.6 GeV, indicating persistent limitations in the low-mass regime.
The loss vs. epoch plot for this model (figure 4.14a) demonstrates a significantly lower loss compared to all the models previously discussed in this section (sections 4.4.1-4.4.8) , indicating better overall performance and more effective learning.

4.5 Final model training and results

Based on the observed optimizations, the final models were trained on larger datasets (4.3M for the barrel and 5M for the endcaps) over 200 epochs, using a similar training and validation split. The epoch with the lowest validation loss was selected for predictions. The final version of the Dynamic Reduction Network, optimized for merged photon reconstruction, was configured with the following hyperparameters:

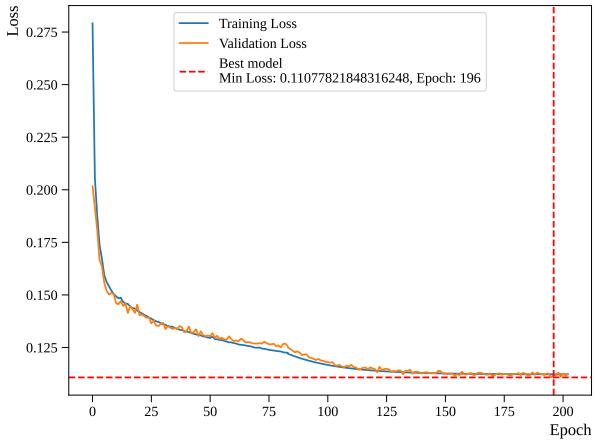
- Input layers : 3
- Output layers: 2
- Hidden dimensions: 64
- Aggregation layers : 3
- Message passing layers : 4
- Batch size (Training and validation): 200
- Learning rate : 0.0001 (Constant)
- Loss function: Mean square loss
- Pooling scheme : Max
- Optimizer : AdamW

The input features (discussed in 4.3) for the final model are the cartesian coordinates (x,y,z) of rechits, the transverse and longitudinal component of energies (E_T and E_z) and a subdetector flag (1 for preshower rechit and 0 for ECAL rechit). For the barrel model, no preshower flags are used.

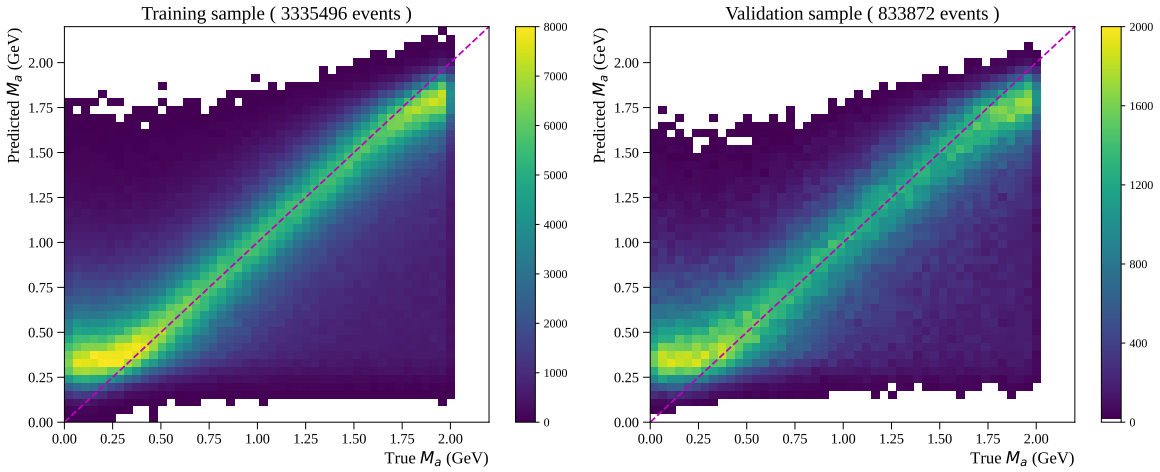
4.6 Mass prediction on $a \to \gamma\gamma$ particle gun sample

4.6.1 Barrel region $|\eta| < 1.44$

Training sample size : 4.3 M



(a)



(b) (c)

Figure 4.15: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.15b) and the validation dataset (figure 4.15c). Additionally, the loss versus epoch plot (figure 4.15a) illustrates the model's performance at its optimal epoch, which corresponds to the lowest validation loss.

To validate our results, we have plotted the 1D distribution of predictions using 40 MeV bins centered around specific mass points spaced at intervals of 100 MeV. A Gaussian is iteratively fitted over these 1D distributions of predicted masses, and the mean (μ_{fit}) and standard deviation (σ_{fit}) of the Gaussian are determined. The figures 4.16 and 4.18 show the 1D distribution of predicted masses at four different mass points for the barrel and endcap samples respectively. The figures 4.19 display the response (μ_{fit}/m_{true}) and resolution (σ_{fit}/μ_{fit}) for the barrel and

endcaps region. For true values near the edges, a secondary peak emerges close to the center of the range due to edge effects as seen in the figures 4.16 and 4.18.

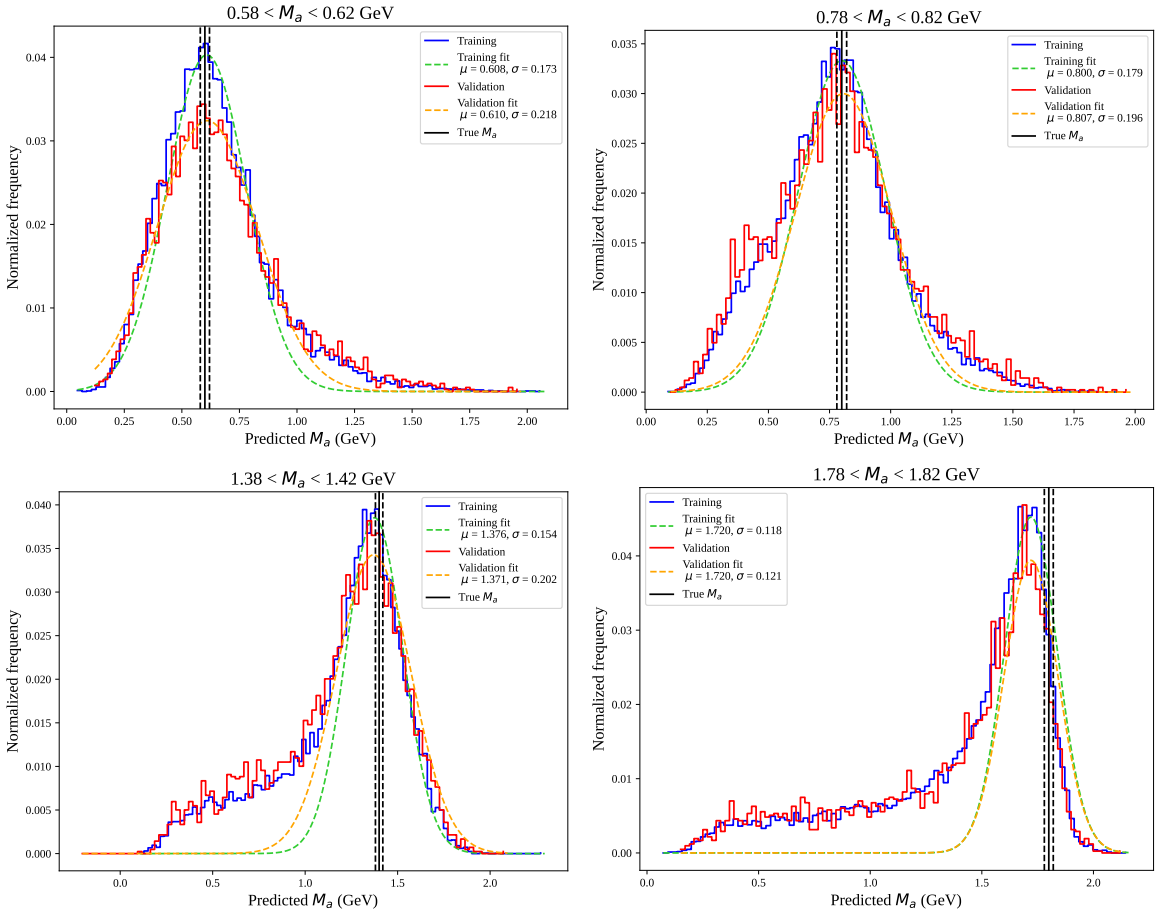
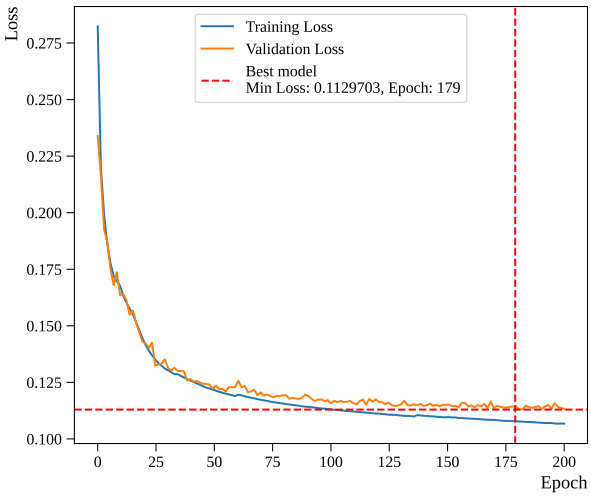


Figure 4.16: 1D histograms of mass predictions from the training dataset in 40 MeV bins around different mass points.

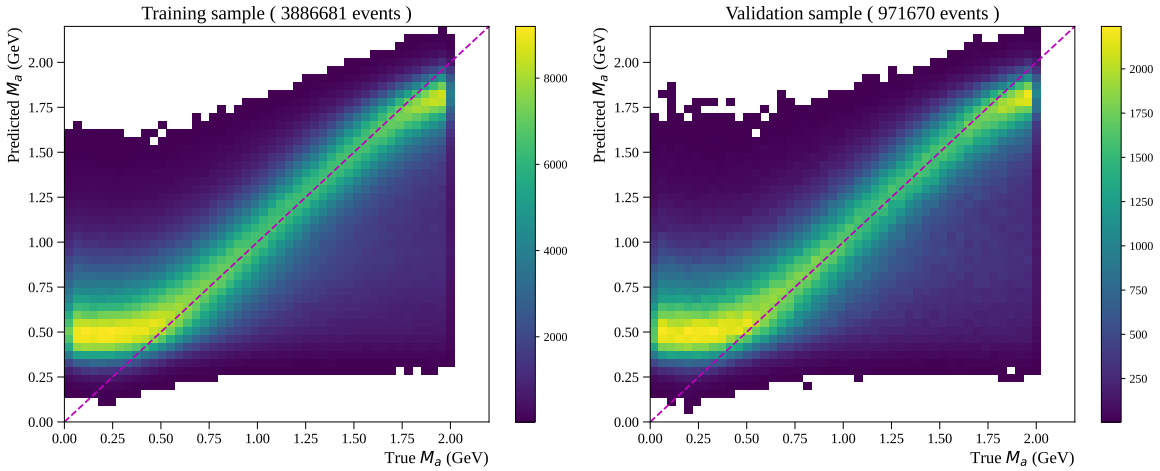
The points around the peaks of histograms in figure 4.16 are iteratively fitted with a Gaussian and the mean (μ_{fit}) is taken as the response. The relative resolution is defined as the ratio of the Gaussian standard deviation to its mean (σ_{fit}/μ_{fit}).

4.6.2 Endcaps region

Training sample size : 5 M



(a)



(b) (c)

Figure 4.17: The plots display the predicted mass versus the true mass of a for both the training dataset (figure 4.17b) and the validation dataset (figure 4.17c). Additionally, the loss versus epoch plot (figure 4.17a) illustrates the model's optimal performance, corresponding to the epoch with the lowest validation loss.

The 1D distribution of predicted mass, similar to the one show for barrel in figure 4.16 is plotted for the endcaps (figure 4.18).

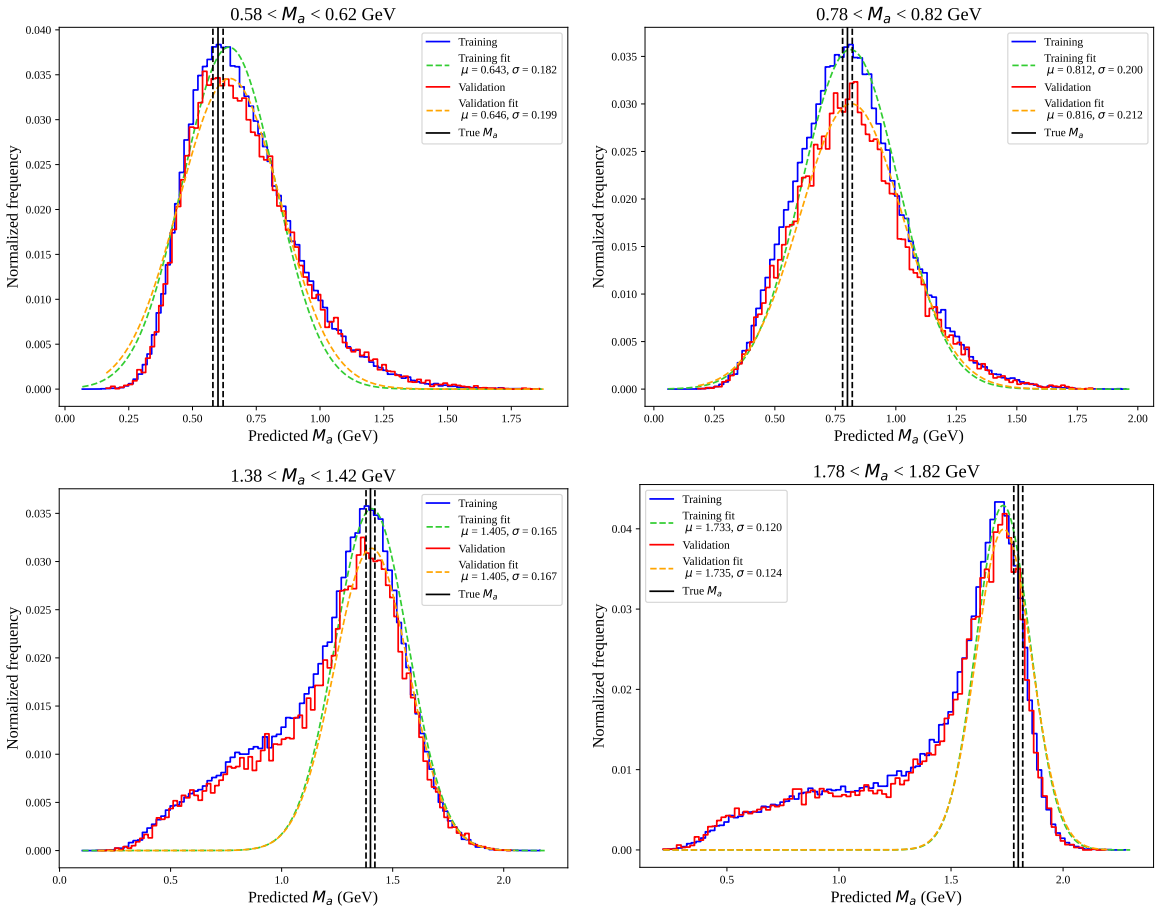
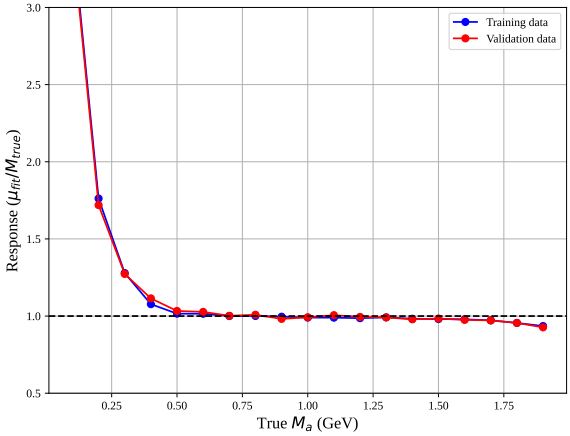
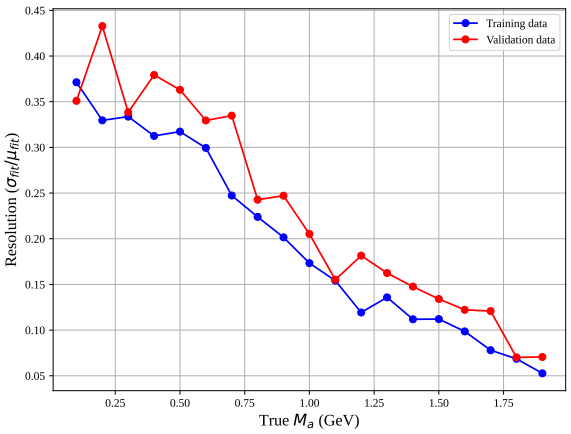


Figure 4.18: 1D histograms of mass predictions from the training dataset in 40 MeV bins around different mass points.

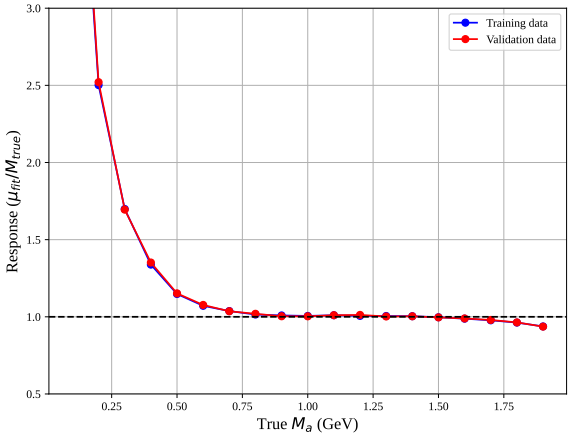
The points around the peaks of histograms in figure 4.18 are iteratively fitted with a Gaussian and the mean (μ_{fit}) is taken as the response. The relative resolution is defined as the ratio of the Gaussian standard deviation to its mean (σ_{fit}/μ_{fit}).



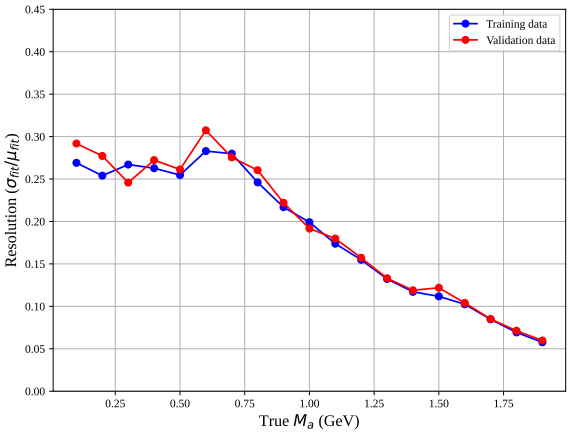
(a) Response in barrel



(b) Relative resolution in barrel



(c) Response in endcaps



(d) Relative resolution in endcaps

Figure 4.19: The above figures show the response (figure 4.19a and 4.19c) and resolution (figure 4.19b and 4.19d) of the DRN for the barrel and endcaps training respectively.

The relative resolution plots for the barrel (figure 4.19b) and endcap (figure 4.19d) models indicate that, for a masses below 0.6 GeV, the endcap model exhibits a lower relative resolution compared to the barrel. However, this should not be interpreted as superior performance by the DRN in the endcaps. The apparent improvement arises from the endcap model's tendency to overpredict the mass in this low-mass regime, leading to a higher response. Since relative resolution is defined as the ratio $\sigma_{\text{fit}}/\mu_{\text{fit}}$, an inflated response artificially lowers the relative resolution, masking the underlying inaccuracy.

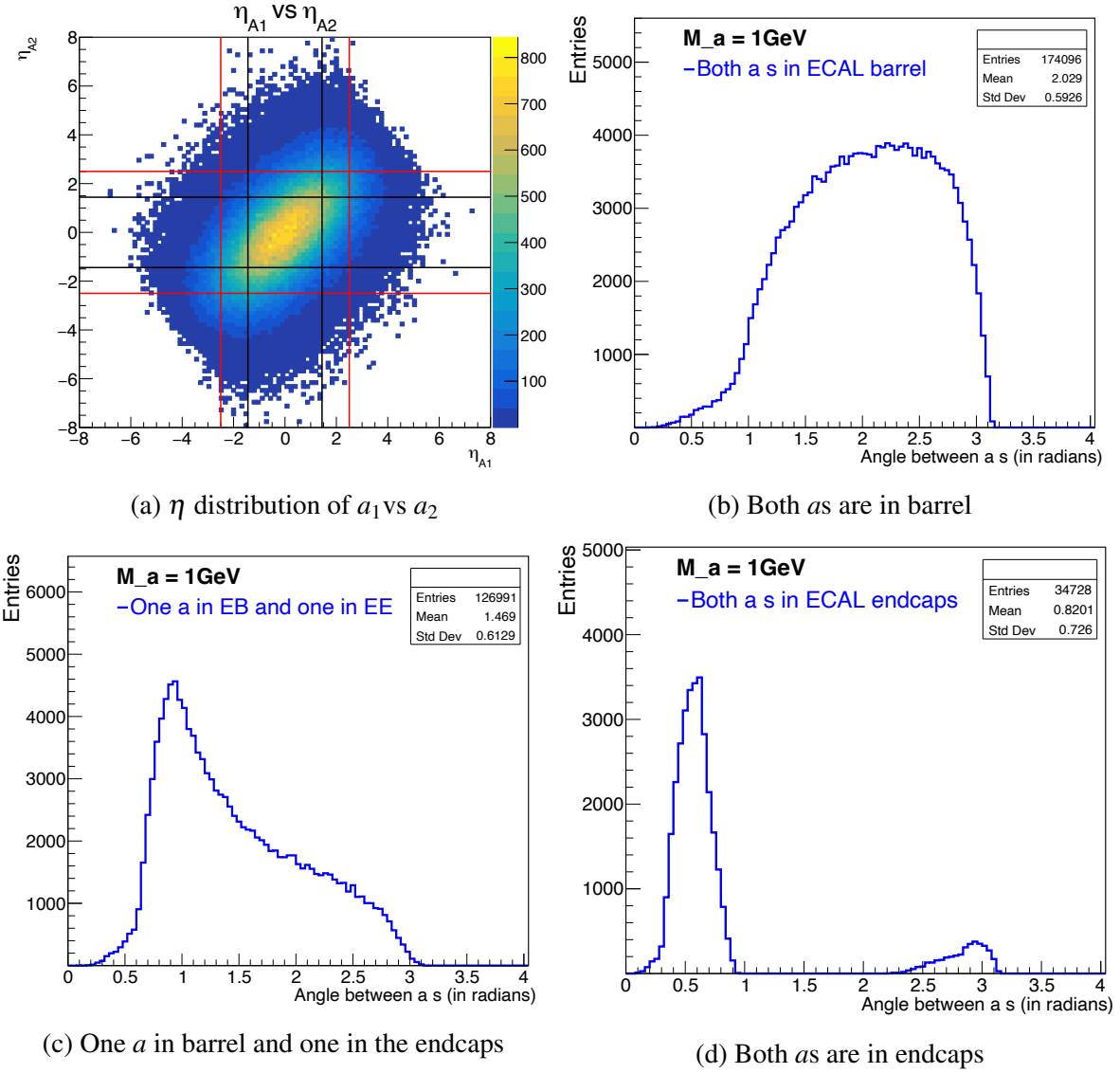
4.7 Validation on $H \to aa \to 4\gamma$ samples from simulated pp collisions

The $H \to aa \to 4\gamma$ samples are generated for various a masses, specifically m_a = 0.1, 0.2, 0.4, 0.6, 0.8, 1.2, 1.6 and 2 GeV. However, for this project, we validated our models only for m_a= 0.6, 0.8, 1.2, 1.6 and 2 GeV. Since, these $H \to aa \to 4\gamma$ events come from pp collisions, the

photons from each a may either be in the barrel or in the endcaps. Thus we can categorize the samples into three categories :

1. When both the as are in the barrel
2. When one of the as is in barrel and the other is in the endcaps.
3. When both the as are in the endcaps.

The following subfigures in figure 4.20 show the distribution of η_{a1} vs η_{a2} and the opening angle between the a s for the above listed cases for the sample with m_a= 1GeV.



(a) η distribution of $a_1$vs a_2

(b) Both as are in barrel

(c) One a in barrel and one in the endcaps

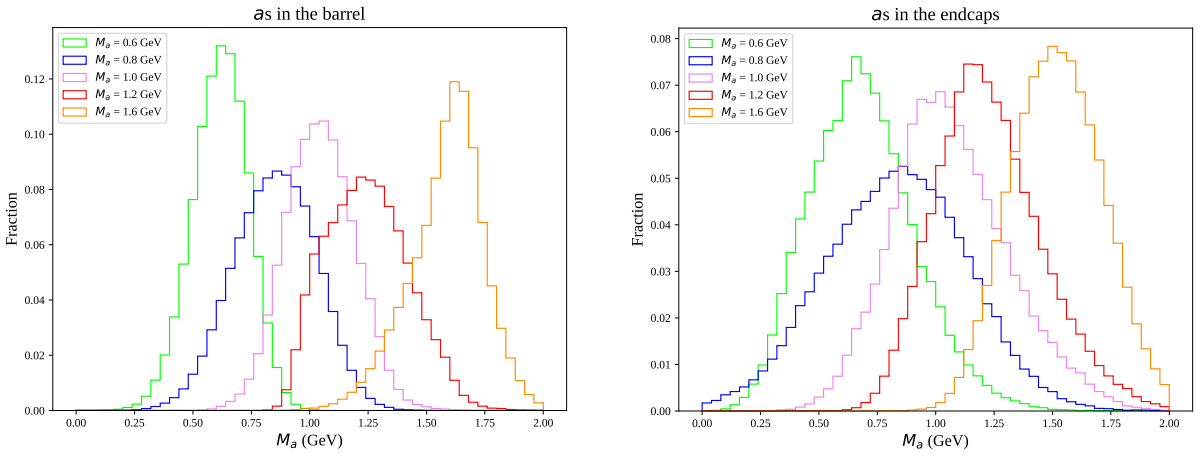
(d) Both as are in endcaps

Figure 4.20: The figure 4.20a shows the correlation of η_{a1} and η_{a2}. The black lines represent the end of the barrel region (in η coordinate) and the red lines represent the ends of the endcap region. Figures 4.20b, 4.20c and 4.20d represent the opening angle between the as from the $H \rightarrow aa$ decays.

The black and red lines in figure 4.20a show the ends of the ECAL barrel and endcap regions, respectively. We can only reconstruct the photons that fall within this geometric coverage area i.e. only photons (and hence as) whose η lies within the area enclosed by the red lines, will be reconstructed.
For the cases, when both the as are in the endcaps, we see a two distinct peaks. The peak on the lower side of the x-axis occurs when the produced Higgs boson is boosted, causing both the a particles to travel in the same direction as the boosted Higgs in the lab frame. The second peak occurs when the as from the Higgs decays are back to back in the lab frame. Hence the opening angle between them, in such cases, is π.

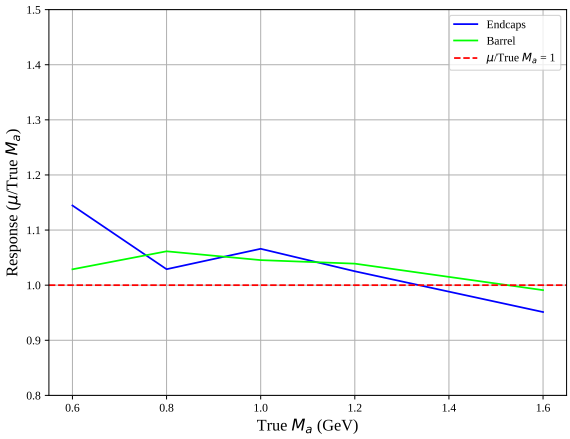
4.7.1 Results

The barrel and endcap models were validated on these samples. We used the η coordinate of the reconstructed PF photon supercluster to determine whether to apply the barrel model or the endcaps model. The figures 4.21a and 4.21b show the performance of the barrel and endcap models on the $pp \to H \to aa \to 4\gamma$ samples.

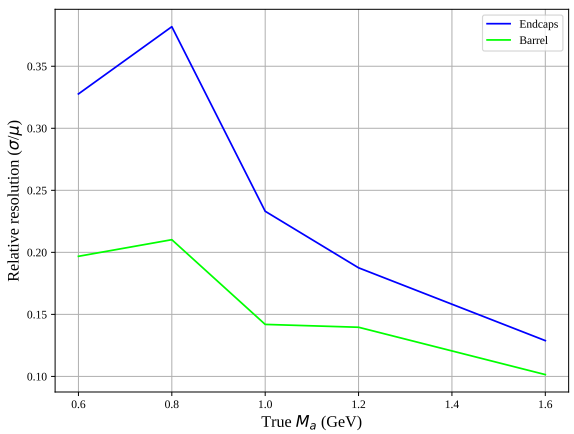


(a) Barrel model

(b) Endcaps model



(c) Response



(d) Relative resolution

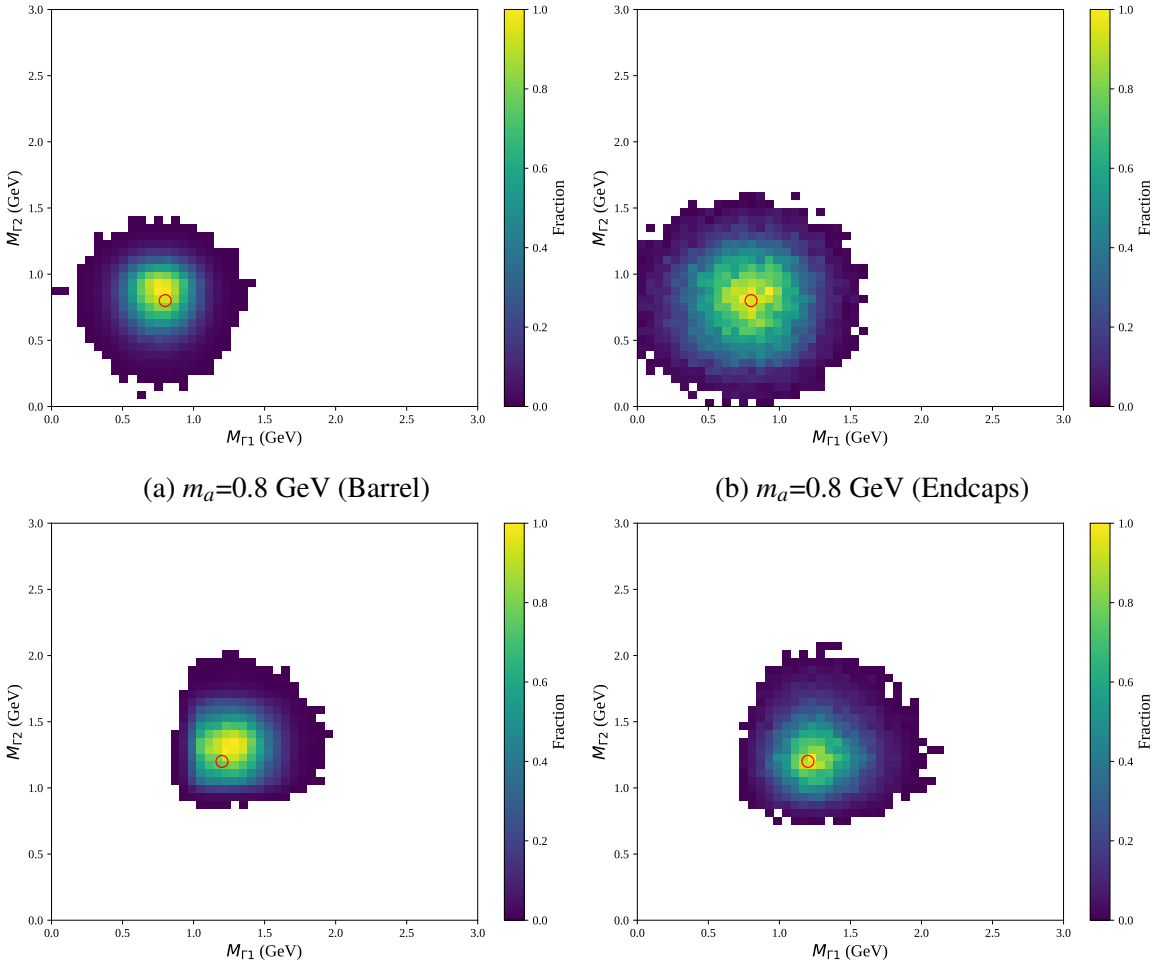
Figure 4.21: This figures 4.21a and 4.21b show the 1D mass predictions for the barrel and endcaps model on the $H \to aa \to 4\gamma$ sample. Figures 4.21c and 4.21d compare the response and resolution of the barrel and endcap models. The red line in figure 4.21c shows the y=1 line.

The relative resolution in the endcaps is worse than that in barrel due to lower granularity, high particle flux and higher noise thresholds (figure 4.21d).

Since the $H \to \gamma\gamma$ process is a Standard Model (SM) process that serves as a background for the $H \to aa \to 4\gamma$ signal (refer to figure 1.3), we aim to identify signatures of the $H \to aa \to 4\gamma$ process by examining the 2D distributions of masses predicted by the DRN models. The underlying idea is that if the rechits originate from a true photon (rather than a merged diphoton object resulting from the decay of a), the DRN should predict very low masses. Such events should produce a 2D distribution where most points are clustered near the axes.

Conversely, if the photon objects are actually merged diphotons from the decay of a, the DRN is expected to correctly predict their masses in most cases, resulting in a distinct blob in the 2D mass distribution. The figure 4.22 shows the 2D mass distributions ($m_{\Gamma 1}$ vs. $m_{\Gamma 2}$) for various

masses of a, where m_Γ represents the regressed mass of the photon object Γ.



(a) m_a=0.8 GeV (Barrel)

(b) m_a=0.8 GeV (Endcaps)

(c) m_a=1.2 GeV (Barrel)

(d) m_a=1.2 GeV (Endcaps)

Figure 4.22: The red dot marks the true mass of the pseudoscalar and the z-axis (color scale) represents the normalized frequency of the particular bin in the 2D mass distribution.

The 2D distributions of predicted masses shown in figure 4.22 were generated to examine the shape of the signal template, following an analysis strategy similar to that employed in Ref. [19] for distinguishing $H \to aa \to 4\gamma$ signals from the $H \to \gamma\gamma$ peak (figure 1.2). While the techniques developed in this chapter demonstrate the ability to reconstruct pseudoscalars with masses above 0.6 GeV, it is essential to evaluate the model performance and the corresponding 2D mass distributions for background processes that could mimic the $H \to aa \to 4\gamma$ signal. These aspects, along with the broader implications of the results presented here, are discussed in detail in chapter 5.

Chapter 5

Summary and outlook

This project presents a novel end-to-end approach for reconstructing highly boosted particles that decay into two photons, which are often detected as a single photon object. This innovative framework provides a versatile method for analyzing such particles with improved precision. Utilizing Graph Neural Network (GNN) models, the approach eliminates the dependency on particle-flow objects by directly processing raw detector-level data. This streamlined method allows for the efficient extraction of relevant particle features with minimal preprocessing. This technique is applicable not only for reconstructing pseudoscalars like as but also for other low-mass particles that decay into diphotons. It offers a powerful tool for uncovering hidden peaks of $H \to aa \to 4\gamma$ within the Standard Model $H \to \gamma\gamma$ signal.

This project marks the first application of machine learning techniques to studying such challenges in the CMS ECAL endcap detectors. Previous studies focused on the barrel region ($|\eta|$<1.4), utilizing convolution neural networks (CNNs) for pseudoscalar masses between 0 and 1.2 GeV [19] and graph neural networks (GNNs) for masses ranging from 0.6 to 2 GeV [18]. By extending the study to the endcaps, this work demonstrates that incorporating the preshower significantly enhances mass predictions (as shown in figure 4.12).

Furthermore, as indicated by the stat boxes in figures 4.20b, 4.20c, and 4.20d, approximately 40–50% of reconstructible $H \to aa \to 4\gamma$ events involve at least one of the as decaying in the endcaps. This underscores the importance of expanding machine learning-based approaches beyond the barrel region, as it significantly broadens the accessible phase space in η (area under the red box in figure 4.20a) and enhances the potential for new physics discoveries.

This technique enables us to accurately regress the mass of pseudoscalars up to 600 MeV. However, for masses below 600 MeV, reconstruction becomes increasingly challenging as the merged diphoton showers resemble single photon showers, making them difficult to distinguish. To enhance reconstruction performance, incorporating tracker information as an additional input to the model and adding additional embedding layers for rechit from different subdetectors could be beneficial. Furthermore, with the upcoming High Granularity Calorimeter (HGCAL) upgrade for the endcaps, we anticipate significantly improved performance based on our findings after including the preshower (see section 4.4.7). Furthermore, to enhance the robustness of the reconstruction technique across a wider mass range and mitigate edge effects, composite network architectures—such as those proposed in Ref. [20] and implemented in Ref. [21]—could be explored as a potential improvement.

The next steps could involve integrating these ML techniques directly into the CMSSW framework using Services for Optimized Network Inference on Coprocessors (SONIC), enabling the direct application of these reconstruction techniques within CMSSW. We are currently carrying out a related preliminary study by incorporating DRN models, trained for electron and photon energy regression, into CMSSW and performing inference on particle gun samples with varying noise levels and rechit thresholds to evaluate their impact on energy regression. Further details on this study are provided in appendix 5.

References

[1] CERN, "Accelerator complex." https://www.home.cern/science/accelerators/accelerator-complex, 2024. Accessed: 2024-12-24.

[2] CMS collaboration, "Particle-flow reconstruction and global event description with the CMS detector," *Journal of Instrumentation*, vol. 12, no. 10, p. P10003, 2017.

[3] CMS Collaboration, "Description and performance of track and primary-vertex reconstruction with the cms tracker," *Journal of Instrumentation*, vol. 9, p. P10009–P10009, Oct. 2014.

[4] "The perceptron: A probabilistic model for information storage and orga> author=Rosenblatt, Frank, journal=Psychological Review, volume=65, number=6, pages=386–408, year=1958, publisher=American Psychological Association, url=https://www.ling.upenn.edu/courses/cogs501/Rosenblatt1958.pdf,"

[5] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989.

[6] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken, "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," *Neural Networks*, vol. 6, no. 6, pp. 861–867, 1993.

[7] P. Contributors, "Resnet models in torchvision." https://pytorch.org/vision/main/models/resnet.html, 2025.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.

[9] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," 2019.

[10] R. H. Bisseling and D. Meijerink, "Graph coarsening and clustering on parallel computers," *Scientific Articles Webspace*, 2012.

[11] CMS Collaboration, "The CMS Experiment at the CERN LHC," *JINST*, vol. 3, p. S08004, 2008.

[12] E. Tournefier, "The preshower detector of CMS at LHC," *Nuclear Instruments & Methods in Physics Research Section A-accelerators Spectrometers Detectors and Associated Equipment*, vol. 461, pp. 355–360, 2001.

[13] M. Shoaib, *Single top-quark production cross-section measurement in association with Z boson and performance studies of the RPCs with the CMS detector using 8 TeV pp collisions at the LHC*. PhD thesis, Quaid-i-Azam U., 2017.

[14] CMS Collaboration, "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC," *Physics Letters B*, vol. 716, pp. 30–61, 2012.

[15] M. D. Schwartz, *Quantum Field Theory and the Standard Model*. Cambridge: Cambridge University Press, 2014.

[16] "Standard model." https://en.wikipedia.org/wiki/Standard_Model, 2025.

[17] "Mathematical formulation of the standard model." https://en.wikipedia.org/wiki/Mathematical_formulation_of_the_Standard_Model, 2025.

[18] C. Gupta, "Novel deep learning techniques for reconstruction of paricles decaying to merged photons and energy calibration of charged hadrons," 2024. http://dr.iiserpune.ac.in:8080/xmlui/handle/123456789/8902.

[19] CMS collaboration *et al.*, "Reconstruction of decays to merged photons using end-to-end deep learning with domain continuation in the CMS detector," *arXiv preprint arXiv:2204.12313*, 2022.

[20] D. Kim, K. Kong, K. T. Matchev, M. Park, and P. Shyamsundar, "Deep-learned event variables for collider phenomenology," 2021. https://arxiv.org/abs/2105.10126.

[21] K. Pedro and P. Shyamsundar, "Optimal mass variables for semivisible jets," 2023. =https://arxiv.org/abs/2303.16253.

[22] CMS Collaboration, "Performance of photon energy corrections in the CMS ECAL using graph neural networks," 2022. https://cds.cern.ch/record/2814001.

[23] CMS Collaboration, "Performance of electron energy calibration in the CMS ECAL using graph neural networks," 2022. https://cds.cern.ch/record/2809560.

Appendix A

In this section, we explore the incorporation of machine learning techniques within the CMSSW framework through the Services for Optimized Network Inference on Coprocessors (SONIC) Triton service. Specifically, we have attempted to utilize a pre-trained DRN model [22] [22], for energy regression, to perform direct inference on particle gun electron samples within CMSSW. The following sections provide a concise overview of the importance of energy regression and detail our efforts in implementing it within the CMSSW framework.

A.1 Motivation

Electrons and photons play a critical role in numerous analyses conducted by the CMS Collaboration, such as measuring the mass of the W boson and studying Higgs boson properties and new physics searches. Since many of these analyses are statistically limited, achieving high precision in reconstruction techniques is essential for enhancing their sensitivity. Consequently, improving the reconstruction of electrons and photons is of paramount importance.

A.2 Energy regression for photons and electrons

The reconstruction of electrons and photons, as outlined in Section 2.3, involves applying various corrections to accurately determine their kinematic properties. One such correction at the object level is energy regression, which is applied to electron or photon objects. The need for energy regression arises due to several factors:

- Electromagnetic showers can leak longitudinally.
- Energy can escape transversely through intermodular gaps.
- Energy loss may occur in the tracker.
- Additional energy can be contributed by pileup.
- Energy loss can occur in dead crystals within the detectors.

Currently, energy regression is performed using a BDT that takes approximately 30 correlated observables, such as R9 and isolation, as input to predict the ratio $y = E_{True}/E_{Raw}$. Here, E_{True} represents the true energy of the particle, while E_{Raw} denotes the raw energy of the particle prior to regression. For data, the corrected energy is obtained by multiplying this ratio y with the raw energy.
The BDT predictions are sensitive to detector and run conditions, which necessitates retraining the model whenever the noise level or rechit threshold is altered.

A.3 Energy regression using DRN

A dynamic reduction network, similar to the one described in section 3.4 was used by for energy regression of electrons and photons. The architecture of the DRN used for energy regression, is shown in figure A.1 It was observed that the DRN had better response and resolution than the BDT (as shown in the figures A.2 and A.3). Further details regarding the training and validation samples, strategies and physics motivation of the choices made for training the DRN can be found in the references [23] and [22]. The datasets used for training the DRN models are as follows :

- **Electron regression training:**
  - `/DoubleElectron_Pt-1To300-gun/RunIISummer20UL18MiniAODv2-FlatPU0to70EdalIdealGT_EdalIdealGT_106X_upgrade2018_realistic_v16_L1v1-v2/MINIAODSIM`
  - `/DoubleElectron_FlatPt-1To300/RunIISummer19UL18MiniAODv2-FlatPU0to70RAW_106X_upgrade2018_realistic_v16_L1v1_ext2-v1/MINIAODSIM`
- **Photon regression training:**
  - `/DoublePhoton_Pt-5To300-gun/RunIISummer20UL18RECO-FlatPU0to70EdalIdealGT_EdalIdealGT_106X_upgrade2018_realistic_v11_L1v1_EcalIdealIC-v2/AODSIM`
  - `/DoublePhoton_FlatPt-5To300/RunIISummer19UL18MiniAODv2-FlatPU0to70RAW_106X_upgrade2018_realistic_v16_L1v1_ext1-v1/MINIAODSIM`

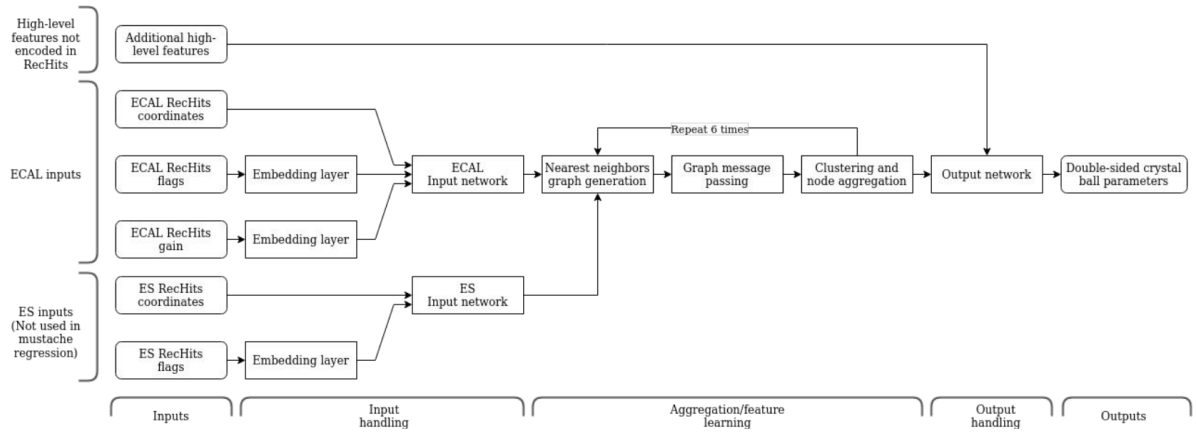
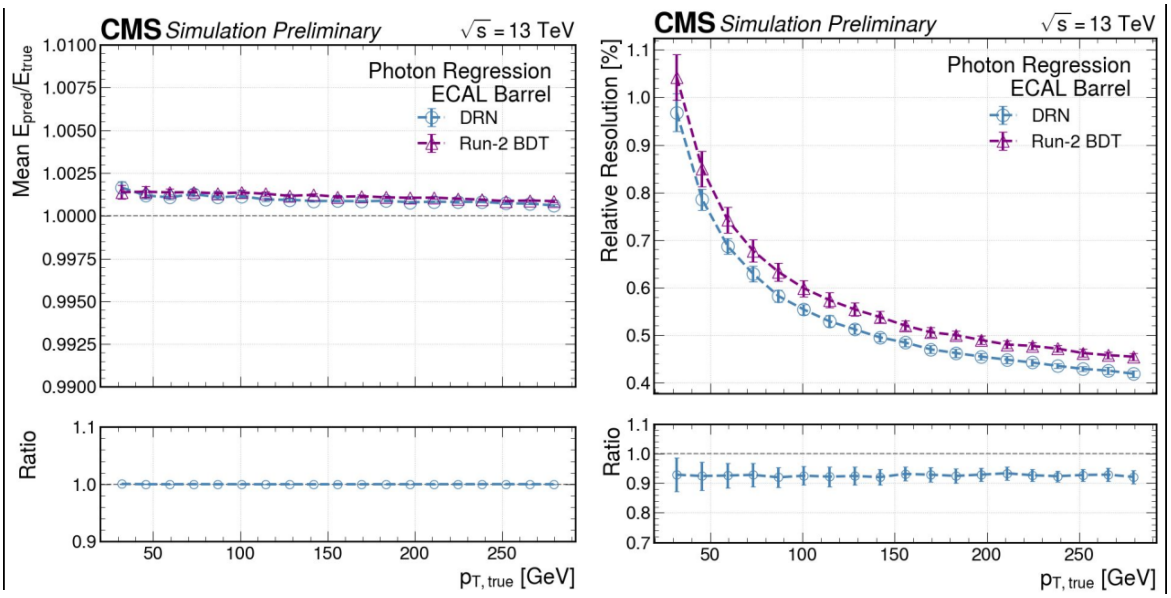
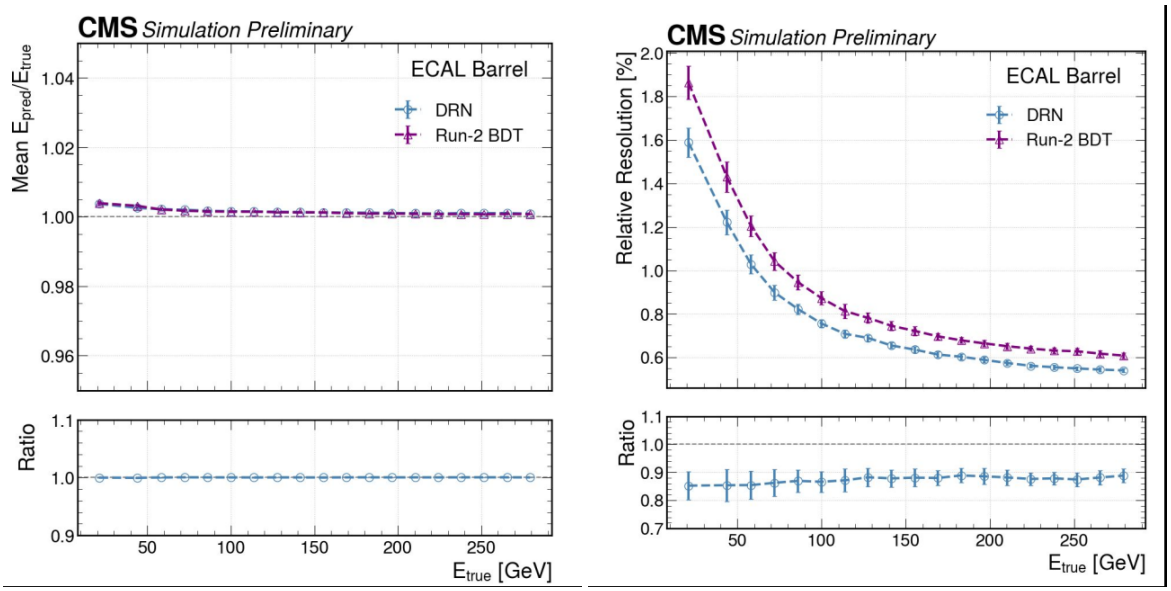


Figure A.1: Architecture of DRN used for energy regression



(a) (b)

Figure A.2: Figures comparing the response and relative resolution of the DRN with Run 2 BDT on photons from $H \to \gamma\gamma$ events, in the barrel.[22]



(a) (b)

Figure A.3: Figures comparing the response and relative resolution of the DRN with Run 2 BDT on electrons (or positrons) from $Z \to e^+e^-$ events, in the barrel.[23]

A.4 Using DRN in CMSSW

The DRN models for energy regression are integrated into the CMSSW framework. Although the model weights are not commited to the cms-data repository, inference can still be performed

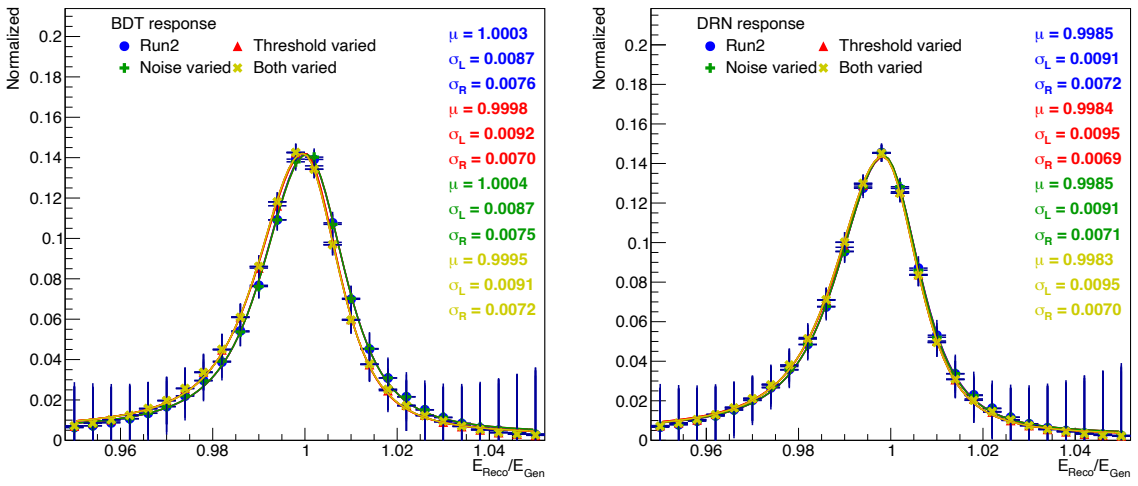
using model weights from a local are using SONIC Triton servers.
To run the DRN, a local Triton server must be set up with access to the model repository containing the model weights. The CMSSW module, equipped with the Triton Client, can then interact with the local Triton server to perform inference.

A.5 Variation of energy corrections with noise levels and rechit thresholds

To assess whether the Dynamic Reduction Network (DRN) requires retraining, similar to the Boosted Decision Tree (BDT), when noise levels and rechit thresholds are altered, we performed DRN energy regression on four electron particle gun samples with distinct run conditions. The conditions of the samples are as follows :

1. **Run 2 UL18 sample**: Serves as the control.
2. **Noise-varied sample**: Evaluates the impact of modified noise levels.
3. **Rechit threshold-varied sample**: Assesses the effect of adjusted rechit thresholds.
4. **Combined noise and rechit threshold-varied sample**: Examines the influence of simultaneous variations in both parameters.

In all samples, the four electrons are generated with a transverse momentum (p_T) uniformly distributed between 1 and 500 GeV, and a pseudorapidity (η) uniformly distributed between -3 and 3. Figure A.4 compares how the responses of the BDT and DRN models vary under different reconstruction conditions. The histograms are fitted with a Cruijff function and the mean of the fit should be taken as a response.



(a) (b)

Figure A.4: Figures displaying the $E_{Corrected}/E_{True}$ ratio for both DRN and BDT across all four samples are presented. A Cruijff function (given in equation A.1) is fitted to each histogram, with the mean (μ) and the left and right standard deviations (σ_L and σ_R) indicated on the plots.

The Cruijff function is given by:

$$f(x) = A\exp\left(-\frac{(x-\mu)^2}{2\sigma_L^2 + 2\alpha_L(x-\mu)}\right), \quad \text{if} \quad x < \mu$$
$$f(x) = A\exp\left(-\frac{(x-\mu)^2}{2\sigma_R^2 + 2\alpha_R(x-\mu)}\right), \quad \text{if} \quad x \geq \mu \tag{5.A.1}$$

Where: - A is the normalization factor. - μ is the peak position (mean). - σ_L and σ_R are the Gaussian widths on the left and right sides of the peak, respectively. - α_L and α_R are the tail parameters on the left and right sides, respectively.

Analysis of the plots indicates that variations in the rechit threshold have a more pronounced impact on the BDT performance compared to changes in noise levels.
Although the plots indicate that the DRN response is mostly unaffected by variations in noise levels and rechit thresholds, as evidenced by the nearly unchanged mean across all cases, further analysis is needed to draw a definitive conclusion.