INSTITUT
POLYTECHNIQUE
DE PARIS

NNT : 2020IPPAX094

Thèse de doctorat

Algorithms for the Level-1 trigger with the HGCAL calorimeter for the CMS HL-LHC upgrade

Thèse de doctorat de l'Institut Polytechnique de Paris et de l'Université de Split
préparée à l'Ecole Polytechnique

École doctorale de l'Institut Polytechnique de Paris (ED IP Paris) n°626
Spécialité de doctorat: Physique des particules

Thèse présentée et soutenue à Split, le 18 Decembre 2020, par

MARINA PRVAN

Composition du Jury :

Ivica Puljak Professor, FESB, Split	Président du Jury
Isabelle Wingerter-Seez Research director, CNRS, Marseille	Rapporteur
Sven Loncaric Professor, FER, Zagreb	Rapporteur
Paul Dauncey Professor, Imperial College, London	Examinateur
Claude Charlot Research director, LLR, Palaiseau	Directeur de thèse
Julije Ozegovic Professor, FESB, Split	Co-directeur de thèse
Jean-Baptiste Sauvan Researcher, LLR, Palaiseau	Examinateur
Linda Vickovic Associate professor, FESB, Split	Examinateur

UNIVERSITY OF SPLIT
FACULTY OF ELECTRICAL ENGINEERING, MECHANICAL ENGINEERING
AND NAVAL ARCHITECTURE

Marina Prvan

ALGORITHMS FOR THE LEVEL-1 TRIGGER WITH THE HGCAL CALORIMETER FOR THE CMS HL-LHC UPGRADE

DOCTORAL THESIS

Split, 2020.

The research reported in this thesis was carried out at Department of electronics and computing, University of Split, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture (FESB), and Laboratory Leprince-Ringuet (LLR), Ecole Polytechnique.

Supervisors: prof. dr. sc. Julije Ozegovic, FESB, University of Split, Croatia and dr. sc. Claude Charlot, LLR, Ecole Polytechnique, France
Disertation number: 160

BIBLIOGRAPHIC INFORMATION
Keywords: CMS, HGCAL, LHC
Scientific area: Technical sciences
Scientific field: Electrical engineering
Scientific branch: Electronics
Institution of PhD completion: University of Split, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture and Institut Polytechnique de Paris, Ecole Poytechnique
Supervisors of the thesis: prof.dr.sc. Julije Ozegovic and dr.sc. Claude Charlot
Number of pages: 215
Number of figures: 158
Number of tables: 26
Number of references: 163

Summary

The Large Hadron Collider (LHC) is the world's largest and most powerful particle accelerator ever built. It is a marvel of modern particle physics, operated by the European Laboratory for Particle Physics (CERN), and designed to collide protons at extremely high energies to produce new exotic particles. The LHC is a long term project, with more than 20 years of running ahead. The detectors are constantly being upgraded, because of the to two basic reasons; the development of new technologies and the replacement of detector parts damaged by the high levels of radiation. The high luminosity HL-LHC upgrade of the multi-purpose Compact Muon Solenoid (CMS) detector is the context of this thesis. In particular, the endcaps of the electromagnetic and hadronic calorimeters are going to be replaced by new versions with a much finer spatial resolution, called the High Granularity Calorimeter (HGCAL). It is the result of modern instrumentation, which will imply to handle the exponential increase of data produced by sensors arrays. Because of the layered structure with very small silicon read-out cells, it will provide a fully three-dimensional image of particle showers (or a four dimensional image when the timing is included). In the extremely busy environment of the HL-LHC, with high energy and large pile-up (PU), it is impossible to record all collision events and, therefore, it is essential to provide a real-time decision regarding the interesting events to be kept for further analysis. This decision process, called the Level 1 (L1) trigger, has very tight time constraints as well as communication and processing limitations coming from the available hardware.

To cope with the HL-LHC requirements, the current trigger system must be upgraded, and this thesis presents the related studies that were necessary for the design of such trigger. After the first two chapters on the detector and its upgrade, the rest of the thesis is organized following the logical order of the trigger design. First, studies devoted to the CMS HGCAL design upgrade are presented as the main step towards the generation of trigger signals. It is the first time at LHC that a calorimeter like HGCAL is built, being silicon-based and highly segmented, and bringing the new paradigm of 3D calorimetry. A compact design is proposed that fully contains the showers, and a fine spatial resolution results from the large number of channels and very small silicon read-out sensors cells (SC) organized in several layers in depth. For the first time, a hexagonal geometry is used in the design, to make the detector sensitive area the most cost-effective. The main consequence of the presented research on the geometries is the design of a hexagonal sensor module (SM) for the future HGCAL, which satisfies the identified requirements and provides an optimization of the SM production cost. Various trigger cell (TC) inner-packing schemes result from the analysis and it is shown that an efficient forming of TCs inside the module is possible with a non-uniform TC packing procedure.

After the desired TCs are formed, the next part of the thesis work is devoted to the study of the selection of TCs in the FE electronics. There are two possibilities: to apply a fixed threshold on TC energies, or to select a fixed number of highest energy TCs. While a basic approach can be to sort the TCs before the selection, the difficulty is that the TC addresses must be extracted together with the energy values. This is important because once the TCs signals are received in the back-end (BE), it should be marked which module and layer they belong to. The presented research resulted in the design of an efficient maximum-finder circuit that is synchronized with the 40MHz clock cycle. Unlike a sorting algorithm, it provides a simple addressing scheme with selection bits consisting only of zeros and ones. A simple approach is used in the hardware such that the binary TC energies are compared bitwise in parallel, and an optimization is provided for the ASIC latency and area compared to existing array-based topologies. A verification of the proposed Best-Choice Topology (BCT) is done in the simulation of the trigger path, showing that it is possible to perform a selection within a 2ns time frame.

Next, a significant part of the thesis work is focused on specific strategies for the trigger reconstruction in the BE electronics. Since the upgraded HGCAL will provide not only a finer transverse spatial resolution but also a fully three-dimensional image of the particle showers, a special interest is devoted to a direct 3D clustering of TCs. It is different from the strategy considered up to now, where a 2D clustering is first performed layer-by-layer, after which clusters are linked into a larger 3D cluster. We study the main difficulties for a direct 3D clustering implementation at the L1 trigger. Architectures are proposed that provide solutions for the identified problems, and their critical points are examined. Instead of doing a direct 3D clustering in the whole detector at once, we propose to first find regions of interest (ROIs) in the detector and apply the processing only on this reduced data volume. An option of a two-step BE architecture with shower tracking followed by clustering is examined and provides more flexibility. We design a tracking algorithm (TA) that can help to "intelligently" reduce the data, especially if a shower identification mechanism is added based on the known profile of EM showers. With this knowledge included in the TA, we can reduce the number of tracks and select the signal more efficiently, reducing the required bandwidth.

Finally, we present a machine learning study based on a neural network (NNet) included in the trigger chain. While a single set of energy weights were used to encode the EM shower shape inside the TA, the idea here is similar to the concept of having a more "general set of signal weights". Namely, we have relied before on the ideal case where weights are extracted by using only the signal information but in reality, we will always have to separate between signal and PU-like data. Hence, a better discriminant is trained for binary classification, using only a few dense layers to simplify the hardware implementation. Three NNet models are compared, with different types of input ROI images: a single-channel 2D ROI image (with and without identification), a multi-channel ROI image with various numbers of layers in depth, and 3 independent 2D views of the event data. The classification accuracy is measured against the model complexity (the total number of parameters), so the best trade-off is concluded between the quality of the decision-making process and the required hardware processing power.

Résumé

Le Large hardon Collider (LHC) est le plus grand et le plus puissant accélérateur au monde. Il est en opération au Laboratoire Européen pour la Physique des Particules (CERN), et est conçu pour des collisions de protons à des énergies extrêmement élevéés afin de produire d'éventuelles nouvelles particules. Le LHC est un projet de long terme, prévu pour fonctionner encore plus de 20 ans. Les détecteurs sont constament remis à niveau afin de suivre l'évolution des technologies ainsi que pour remplacer les parties du détecteurs les plus exposées aux radiations. La mise à niveau du détecteur CMS (Compact Muon Solenoid) en vue de la phase de haute luminosité du LHC (HL-LHC) constitue le cadre de cette thèse. En particulier, les parties bouchons des calorimètres électromagnétique et hadronique vont être remplacées par un calorimètre de résolution spatiale beaucoup plus fine, le High Granularity Calorimeter (HGCAL). Le HGCAL est le résultat des développements récents en instrumentation, et devra prendre en compte l'augmentation exponentielle du volume de données produites par les éléments du détecteur. Avec sa structure en couche et ses très petites unités de lecture au silicium, il fournira une représentation tri-dimensionnelle des gerbes produites par les particules.

Dans l'environement complexe du HL-LHC, avec une plus haute énergie et beaucoup d'empilement (PU), il est impossible d'enregistrer tous les événements issus des collisions, et, par conséquent, il est essentiel d'avoir un système de décision en temps réel permettant de sélectionner les événements interessants. Un tel système, appelé déclenchement de niveau 1, pose des contraintes fortes en termes de temps de traitement et de communication pour une implémentation matérielle. Pour satisfaire à ces contraintes, le système actuel doit être mis à niveau, et cette thèse présente les études effectuées pour la conception d'un tel système de déclenchement. Après les deux premiers chapitres qui décrivent la mise à niveau du détecteur CMS en vue du HL-LHC, le reste du document est organisé en sections qui suivent l'ordre logique de la conception du système de déclenchement.

Tout d'abord des études sur la conception du HGCAL sont présentées qui constituent une étape principale en vue de la génération des signaux pour le déclenchement. C'est la première fois qu'un tel détecteur est construit, avec des éléments de détecteur en silicium et extrêmement segmenté, et cela conduit au nouveau paradigme de calorimétrie en 3D. Un concept compact est proposé pour contenir les gerbes issues des particules de la collision, et la fine segmentation spatiale est réalisée par un grand nombre de canaux de lecture associés à de très petits éléments sensibles au silicium (SC), organisés en plusieurs couches en profondeur. Pour la première fois,

une géométrie hexagonale est proposée, afin de réduire les coûts des éléments de détecteurs. La principale conséquence du travail présenté sur la géométrie est le design d'un module hexagonal (SM) comme unité de traitement, satisfaisant aux contraintes et permettant une optimisation du coût de production. Différentes solutions pour le regroupement des cellules en cellules de déclenchement (TC) en résultent, et il est montré qu'un regroupement efficace de cellules est possible par le regroupement non-uniforme polyhex.

Une fois les TCs formées, l'étape suivante du travail de thèse est consacrée à l'étude de la sélection des TCs par l'électronique frontale. Il y a deux possibilités: appliquer un seuil fixe sur les énergies, ou bien sélectionner un nombre fixe de TCs ayant les plus grandes énergies. Alors qu'une approche simple peut être de trier les TCs en fonction de leur énergie avant la sélection, la difficulté est que les adresses des TCs doivent être extraites en même temps que les valeurs d'énergie. C'est important car une fois que les signaux des TCs sont reçus par l'électronique dorsale (BE), on doit marquer à quels module et couche correspondent les énergies. Le travail a conduit à la conception d'un circuit efficace pour la recherche de maximum, synchronisé avec le signal d'horloge à 40MHz. À la différence d'un algorithme de tri, il fourni un schéma d'adressage simple où les bits de sélection consistent seulement de zéros et de uns. Une approche simple est utilisée dans le hardware de façon à ce que les valeurs binaires des énergies soient comparées bit-à-bit en parallèle, et une optimisation est obtenue pour la latence de l'ASIC et sa surface, par rapport aux topologies existantes basées sur des tableaux. Une vérification du concept Best-Choice Technology (BCT) proposé est faite par simulation, et montre qu'il est possible de faire la sélection dans l'intervalle de temps de 2ns.

Ensuite, une part significative du travail de thèse porte sur les stratégies spécifiques pour la reconstruction des informations de déclenchement au niveau de l'électronique BE. Comme le HGCAL va fournir non seulement une résolution spatiale plus fine mais aussi une image complètement tridimensionelle des gerbes des particules, un accent particulier est mis sur une reconstruction directement en 3 dimensions (3D) des agrégats de TCs. C'est différent de la stratégie considérée jusqu'à maintenant, où un algorithme d'aggrégation 2D est appliqué couche par couche, avant que les agrégats 2D ne soient regroupés pour former des agrégats 3D plus grands. Nous étudions les principales difficultés pour l'implémentation d'une agrégation en 3D au niveau 1 du déclenchement. Des architectures sont proposées pour répondre aux problèmes identifiés, et les aspects critiques sont examinés. Au lieu d'effectuer directement une agrégation en 3D dans le détecteur complet, nous proposons d'identifier dans un premier temps des régions d'intérêt (ROIs) et d'effectuer le calcul seulement sur ce volume réduit de données. Une option d'architecture en deux étapes avec une reconstruction de trace suivie par l'agorithme d'agrégation est examinée et fournie une plus grande flexibilité pour l'algorithme d'agrégation. Nous concevons un algorithme de reconstruction de trace (TA) qui peut aider à réduire intelligement le volume de données, en particulier si un mécanisme d'identification des gerbes est ajouté, basé sur la forme connue des gerbes électromagnétiques. En utilisant cette connaissance dans le TA, il est possible de réduire le nombre de traces et de sélectionner le signal plus efficacement, réduisant ainsi la bande passante nécessaire.

Finallement, nous présentons une étude d'algorithmes à apprentissage basée sur l'utilisation de réseaux de neurones dans le système de déclenchement. Alors qu'un unique ensemble de poids pour les énergies était utilisé pour coder la forme d'une gerbe électromagnétique dans le TA, l'idée ici est d'avoir un ensemble de poids plus général. Spécifiquement, nous avons considiré jusqu'ici sur le cas idéal dans lequel les poids sont extraits à partir de l'information du signal, mais il est nécessaire de prendre en compte que nous aurons à séparer le signal du bruit de fond venant de l'empilement. Ainsi, un meilleur discriminant est entrainé pour une classification binaire, en utilisant seulement quelques couches denses pour simplifier l'implémentation matérielle. Trois modèles d'architectures neuronales sont comparés, avec differents types d'images en entrée: une image simple-canal en 2D (avec et sans identification), une image multi-canaux avec différent nombres de couches en profondeur (incluant en 3D), et une architecture utilisant 3 vues indépendantes en 2D des données de l'événement. Les performances (classification, précision) sont mesurées en fonction de la complexité du modèle (nombre total de paramètres). Ainsi le meilleur compromis entre la qualité de la décision et la puissance de calcul nécessaire est trouvé.

Sažetak

Veliki hadronski sudarač (eng. Large Hadron Collider, LHC) je najveći i najsnažniji akcelerator na svijetu ikad napravljen. Radi se o čudu tehnologije u modernoj fizici čestica, kojim upravlja Europski laboratorij za fiziku čestica (CERN), a dizajniran je za sudaranje protona pri izuzetno visokim energijama kako bi se stvorili uvjeti za stvaranje novih egzotičnih čestica. LHC je dugoročan projekt namijenjen za rad u periodu od idućih 20 godina. Detektori se neprestano nadograđuju iz dva temeljna razloga; razvoj novih tehnologija i zamjena dijelova detektora oštećenih visokom razinom zračenja. Temeljni kontekst u okviru kojeg je izrađena ova disertacija je nadogradnja višenamjenskog kompaktnog muonskog solenoida (eng. Compact Muon Solenoid, CMS) za LHC period visokog luminoziteta (High Luminosity LHC, HL-LHC). U toj fazi će završni poklopci elektromagnetskog i hadronskog kalorimetra biti zamijenjeni novim inačicama s mnogo finijom prostornom razlučivosti, koje će se realizirati tzv. kalorimetrom visoke granularnosti (eng. High Granularity Calorimeter, HGCAL). Rezultat je to moderne instrumentacije, koja će se trebati nositi s eksponencijalnim povećanjem količine podataka dobivene očitanjem sa niza senzora. Zbog svoje slojevite strukture s vrlo malim silicijskim senzorskim ćelijama (eng. sensor cells, SC) za očitavanje energija, HGCAL će pružiti potpuno trodimenzionalnu sliku pljuska čestica (eng. particle shower).

U izuzetno zahtjevnom okruženju kakvo će pružiti HL-LHC, s visokim energijama i velikim učinkom nagomilavanja pozadinskih ne-signalnih podataka (eng. pile-up, PU), nemoguće je zabilježiti sve događaje sudara i stoga je neophodno u realnom vremenu donijeti odluku o tome koji su događaji dovoljno zanimljivi da budu zadržani za daljnju analizu. Ovaj postupak donošenja odluke, nazvan okidač prve razine (eng. level 1 trigger, L1), ima vrlo zahtjevna vremenska ograničenja, kao i ograničenja u mogućem prijenosu podataka i obrade koju je moguće obaviti na temelju trenutno dostupnog hardvera. Kako bi se mogao nositi sa HL-LHC zahtjevima, trenutni sustav okidača mora se nadograditi, a ova disertacija upravo predstavlja provedena istraživanja i studije koje su bile potrebne za dizajn takvog okidača. Nakon prva dva poglavlja koja opisuju nadogradnju CMS detektora za HL-LHC, ostatak doktorskog rada organiziran je prateći logičan slijed dizajna dijelova sustava okidača.

Prvo su opisane studije posvećene mehaničkom dizajnu novog CMS HGCAL detektora, koje predstavljaju glavni korak prema stvaranju okidačkih signala. Ovo je prvi put u LHC-u da se izrađuje kalorimetar kao što je novi HGCAL, dakle temeljen na bazi silicija, otporan na zračenje i visoko segmentiran, koji donsi novu paradigmu na području 3D kalorimetrije. Nadogradnja se sastoji od zamjene postojećih završnih poklopaca (eng. endcaps) kalorimetra koji

su u potpunosti redizajnirani. Predložen je kompaktni dizajn koji sadrži pljusak čestica, a fina prostorna razlučivost rezultat je velikog broja kanala i vrlo malih silicijskih senzorskih ćelija za očitavanje podataka, koje su organizirane nekoliko slojeva u dubinu. Po prvi puta se u dizajnu koristi šesterokutna geometrija senzora kako bi dizajn osjetljivog područja detektora bilo najisplativiji. Glavna posljedica provedenog istraživanja na području geometrije je dizajn šesterokutnog senzorskog modula (eng. sensor module, SM) za budući HGCAL, koji udovoljava utvrđenim zahtjevima i osigurava optimizaciju troškova proizvodnje. Analiza je pokazala da su mogući razni načini kako pakirati ćelije okidača (eng. trigger cells, TC) unutar senzorskog modula te da je moguće učinkovito formiranje simetričnih grupa od četiri senzorske ćelije korištenjem neujednačene (eng. nonuniform) procedure grupiranja (eng. clustering) šesterokuta.

Nakon formiranja željenih TC-ova, ljedeći dio doktorskog rada posvećen je istraživanju dizajna algoritma odabira ili selekcije dijela energije TC-ova u tzv. detektorskoj elektronici prednjeg kraja (Front-End, FE) . Dvije su mogućnosti: primijeniti fiksni prag (eng. threshold) na energije ili odabrati fiksni broj energija s najvećim vrijednostima. Iako se selekcija može ostvariti pristupom kao što je sortiranje energija prije samog odabira, problem je u tome što se moraju slati TC adrese zajedno sa samim vrijednostima energija. To je važno da bi se mogla izvršiti uspješna rekonstrukcija podataka kad se TC signali prime na ulazu u idućem dijelu arhitekture okidača kojeg nazivamo detektorska elektronika stražnjeg kraja (eng. Back-End, BE), gdje treba znati točno kojem modulu i detektorskom sloju svaka energija pripada. Provedeno istraživanje je rezultiralo dizajnom učinkovitog digitalnog sklopa za traženje maksimalnog elementa (eng. maximum-finder circuit) čiji rad je sinkroniziran sa taktom od 40MHz. Za razliku od algoritma za sortiranje, on pruža jednostavnu shemu adresiranja s indikatorskim bitovima za selekciju koji se sastoje od nula i jedinica. U hardveru se koristi jednostavan pristup, tako da se binarne energije TC-ova uspoređuju u paraleli bit po bit, a postignuta je i optimizacija kašnjenja i površine sklopa realiziranim u ASIC tehnologiji u usporedbi s postojećim algoritmima. Verifikacija predloženog dizajna nazvanog topologija najboljeg izbora (eng. Best-Choice Topology, BCT) prikazana je simulacijom unutar okidačkog lanca obrade podataka, pokazujući da je moguće izvršiti odabir u zadanom vremenskom okviru od 2ns.

Nadalje, značajan dio rada usmjeren je na definiranje strategija za rekonstrukciju okidača u BE elektronici. Budući da će nadograđeni HGCAL pružiti ne samo finiju poprečnu prostornu razlučivost već i potpunu trodimenzionalnu sliku pljuska čestica, poseban interes posvećen je izravnom 3D grupiranju TC-ova. Ono se razlikuje od tadašnje strategije, gdje se prvo izvodi 2D grupiranje podataka sloj po sloj, nakon čega se novonastale grupe povezuju u veći 3D klaster (eng. cluster). Provedeno istraživanje otkriva glavne poteškoće za implementaciju izravnog 3D grupiranja unutar algoritma L1 okidača. Predložene su arhitekture kao rješenja za identificirane probleme i ispitane su njihove kritične točke koje utječu na performanse. Umjesto da radimo izravno 3D klasteriranje u cijelom detektoru odjednom, predlaže se da se u detektoru prvo pronađu područja od interesa (region of interest, ROI) te se primijeni obrada samo na ovom reduciranom volumenu podataka. Ispitana je mogućnost dizajna BE arhitekture u dva sloja, s praćenjem pljuska čestica (eng. shower tracking) u prvom sloju, nakon čega obavlja grupiranje u drugom sloju, što

omogućava veću fleksibilnost grupiranja. Dizajniran je algoritam praćenja (eng. tracking algorithm, TA) koji pomaže u inteligentnom reduciranju podataka, posebno kada se uključi mehanizam za identifikaciju na temelju poznatog profila elektromagnetskog (eng. electromagnetic, EM) pljuska. Zahvaljujući ovom mehanizmu uključenom u TA, možemo smanjiti broj potencijalnih tragova (eng. shower tracks) i učinkovitije odabrati signal, smanjujući pritom količinu podataka koju treba prenijeti.

Konačno, predstavljeno je istraživanje koje se temelji na mogućnosti primjene neuronske mreže (eng. neural network, NNet) unutar okidačkog lanca obrade podataka. Dok je prethodno unutar TA korišten jedinstveni skup težina za kodiranje oblika EM pljuska, ovdje je ideja koristiti „općenitiji skup signalnih težina“. Naime, u prethodnoj strategiji identifikacije smo se oslanjali na idealnu situaciju da se primijenjene težine temelje samo na informacijama o signalu. Međutim, treba uzeti u obzir realniji slučaj kada ćemo imati signal pomiješan sa pozadinskim ne-signalnim PU podacima te ćemo uvijek morati razdvajati signalne EM-like i PU-like podatke. Stoga je ostvaren klasifikator osposobljen za binarnu klasifikaciju ulaznih ROI slika u dvije zasebne klase, koristeći samo nekoliko gustih slojeva neuralne mreže kako bi se pojednostavnila hardverska implementacija. Uspoređena su tri modela NNet arhitektura, s različitim vrstama ulaznih ROI slika koje se koriste za prikaz podataka događaja u detektoru: arhitektura koja razlikuje jednokanalne 2D slike (sa i bez uključene EM identifikacije), arhitektura koja se temelji na višekanalnim ROI slikama, pri čemu je broj kanala parametriziran (uključuje i 3D slike) te arhitektura s tri nezavisna 2D prikaza. Izvedba (točnost klasifikacije) mjeri se prema složenosti modela koji je izražen kao ukupan broj parametara. Na kraju je zaključeno koja arhitektura pruža najbolji kompromis između kvalitetnog donošenja odluke u kompromisu s potrebnom količinom hardverske obrade.

Contents

Introduction **4**

1 The CMS detector at the Large Hadron Collider **7**

1.1 The Large Hadron Collider 7
  1.1.1 The LHC experiment design 8
  1.1.2 The LHC performance parameters 9
  1.1.3 Detectors 11
  1.1.4 The LHC operations 11
1.2 The CMS experiment 12
  1.2.1 Detector design 13
  1.2.2 Coordinate system 17
  1.2.3 Trigger system 18
1.3 Physics event reconstruction 19
1.4 The CMS upgrade in HL-LHC 19

2 The new Phase-2 CMS endcap HGCAL calorimeter **21**

2.1 Detector longitudinal structure and granularity 21
2.2 Engineering design 23
  2.2.1 Silicon sensors 24
  2.2.2 Sensor modules and layout 25
2.3 The HGCAL trigger readout architecture 26
  2.3.1 On-detector electronics 27
  2.3.2 Trigger primitive generator 29
2.4 Simulation software and samples used in the trigger studies 30

3 HGCAL detector geometry studies **33**

3.1 Survey on embedding hexagons in circular and hexagonal ROI 34

3.1.1 The differences between the state-of-the-art models 38
3.1.2 Discussion and evaluation 38
3.2 HGCAL detector sensor module design and sensing layer model 39
3.2.1 Problem formulation and mechanical constraints 40
3.2.2 Study on general hexagonal architectures 40
3.2.3 Using aligned architectures to design SM from a regular hexagon 42
3.2.4 Using rotated architectures to design a SM from a regular hexagon 45
3.2.5 SM designed from an irregular hexagon 47
3.2.6 Evaluation of architectures 48
3.2.7 Study on sensor module production cost 52
3.3 Front-end data reduction based on geometry 57
3.3.1 Trigger cell definition and study on TC regularity 58
3.3.2 Discussion and evaluation 62
3.4 Proposed SM geometry design for HGCAL 64
3.4.1 Uniform TC4 formation with H(D) 64
3.4.2 Non-uniform TC4 clustering with H(D) 65
3.4.3 Discussion and evaluation 66

4 Front-end data selection and the TPG architecture design 69

4.1 Data reduction mechanisms in HGCAL 70
4.2 Trigger data selection algorithm 72
4.2.1 Background and related work 72
4.2.2 The proposed maximum-finder algorithm design 76
4.2.3 Verification of the algorithm functionality for the trigger 79
4.2.4 The sorting network implementation in hardware 84
4.2.5 Verification of the SN selection used for the trigger 85
4.2.6 Discussion and conclusion 86
4.3 TPG architectures for a 3D clustering at the L1 trigger 88
4.3.1 3D clustering advantages and implementation difficulties 89
4.3.2 Two-step 3D architecture design 92
4.3.3 Study on data bandwidth between TPG stages in the two-step architecture 93
4.3.4 Discussion and evaluation 97

5 Reconstruction of trigger signals 99

5.1 L1 trigger data reconstruction studies 99

5.1.1 Motivation 100
5.1.2 Background and related work 100
5.1.3 Basic assumptions for particle energy track finding at the L1 trigger 103
5.2 Tracking algorithm design 105
5.2.1 Projective binning with Thales projection and Hough transform 105
5.2.2 Reconstruction of the shower track initiated by the seed 107
5.2.3 EM shower parametrization 108
5.2.4 Basic TA algorithm verification 109
5.2.5 Shower identification by using the EM longitudinal profile 111
5.2.6 Studies on selecting the optimal TA parameters 114
5.2.7 Discussion and evaluation 128

6 Shower data classification with machine learning 131

6.1 Theoretical background on the neural networks 132
6.1.1 The terminology 132
6.1.2 The hardware implementation challenges 133
6.2 State of the art and open research directions 135
6.2.1 Image-based shower data representation 137
6.2.2 Image processing and pre-processing techniques 140
6.2.3 Main research challenges and opportunities 141
6.2.4 Existing multi-channel ML classification models 143
6.2.5 Main research challenges and opportunities 145
6.2.6 Discussion and evaluation 146
6.3 Study on EM shower classification 147
6.3.1 Signal and background multi-channel ROI images generation 147
6.3.2 ROI visualization 150
6.4 Model verification and ML classification results 151
6.4.1 The neural network architecture and training parameters 151
6.4.2 Impact of the ROI image resolution on classification performance 159
6.4.3 Impact of the ROI size on the classification performance 161
6.4.4 Impact of the ROI depth on the classification performance 168
6.5 Classification with pre-processed single-channel 2D ROI images 175
6.6 The classification with the three separate views of the detector data 177
6.6.1 Visualization of the ROI data projections 178

6.6.2 Classification with ROI images for 3 projections 179
6.7 Discussion and evaluation 183
6.7.1 The summarized research findings and the comparison with prior work 183
6.7.2 Limitations of the study and future work 186

7 Outlook and perspectives **189**
7.1 Current baseline architecture for the direct 3D algorithm 189
7.2 The hardware implementation potential of the selected ML model 191

Conclusion **199**

Introduction

Modern instrumentation in high energy particle physics is facing the exponential growth of data provided by the sensors arrays. The detectors are constantly evolving, and their upgrades are associated with an exponential increase of the output data volume. In particular silicon-based calorimeters, the next generation of calorimeters, provide not only finer transverse spatial resolution but also a fully three-dimensional image of particle showers. The High Granularity Calorimeter (HGCAL) project, which is part of the upgrade of the Compact Muon Solenoid (CMS) detector for the High-Luminosity Large Hadron Collider (HL-LHC), will be the first such calorimeter at an hadron collider. The detector working environment will be very challenging, with the increased overall energy and luminosity of the machine. In such a busy environment, where the probability of an interesting event is low, it is impossible to record all collision events and it is essential to provide a real-time decision of high quality on whether to read-out the event data or not. This decision process, called the trigger, has very tight time constraints as well as communication and processing limitations from the available hardware. The communication bottleneck forces the use of partial and compressed data transfer, while the processing bottleneck implies the implementation of extremely efficient decision making algorithms. This thesis presents the work done on the trigger, and summarizes the studies that were needed along the full trigger level 1 design, from the detector sensor geometry and partial data selection, to the back-end data processing.

The thesis is structured as follows. An introduction to the LHC is given in Chapter 1, together with its parameters and operations in the series of upgrades. The future performance in the upgraded high luminosity era motivates the presented thesis work. The LHC detectors are briefly described here, with a special attention devoted to the CMS detector design. Some of its sub-detectors are described, such as the tracker, the electromagnetic calorimeter (ECAL) and the hadronic calorimeter (HCAL). The event reconstruction is briefly introduced, as well as the standardized coordinate system adopted in the CMS detector to describe the reconstructed particles. Also, the trigger system that is applied for the decision-making on the detector readout data is briefly described. Next, in Chapter 2, the upgrade of the CMS detector to the HL-LHC phase is summarized, emphasizing the trigger design details that are relevant for the thesis work. The upgraded HGCAL detector longitudinal sampling concept is presented, as well as the engineering construction with its small silicon readout sensors and the trigger readout architecture. A more detailed description is provided on the on-detector electronics and the trigger primitive generator (TPG) algorithm.

Chapter 3 presents the geometry studies devoted to the new CMS HGCAL design, which represent the main step towards the generation of trigger signals. We concentrate on the HGCAL design, with the silicon readout cells organized in several layers in depth. Also, various sensor cell groupings are studied in order to form the trigger cells (TC) inside the module, and to reduce the data at the earliest trigger stage. In Chapter 4, the selection of data is studied, with the design of an efficient maximum-finder circuit that is synchronized with the 40MHz event clock cycle. The implemented algorithm selects a fixed number of TCs received from the detector. The advantages and disadvantages towards alternative selection approaches are discussed, such as data sorting prior to selection, or applying a fixed threshold on the input data.

We analyze possible architectures for the TPG design that would allow the implementation of a direct 3D clustering instead of the contemporary 2D layer-by-layer followed by 3D. Also, the bandwidth used to transfer the reduced detector data is studied between the stages of the TPG design. The reconstruction of the TCs is described in Chapter 5, being part of the TPG architecture and enabling the direct 3D processing. We have proposed a tracking algorithm and studied the algorithm parameters in order to provide the desired reconstruction optimization. In Chapter 6, we provide a study on the classification between signal and background images generated from the selection of the detector events data. The goal is to examine whether the machine learning (ML) techniques can be used in the trigger, whereas the accuracy of the model is crucial, and needs to be balanced with the model complexity. Our main guideline is a possible ML implementation in trigger, so the network is kept as simple as possible to reduce the hardware requirements. Chapter 7 provides the outlook and the perspectives from the presented studies, followed by a general conclusion of the thesis and the references used.

Chapter 1

The CMS detector at the Large Hadron Collider

The Large Hadron Collider (LHC) is the most famous and the most powerful collider in the world. It belongs to the particle accelerator complex of the European Laboratory for Particle Physics (CERN), and is designed to try to provide answers to the most fundamental physics questions. The LHC task is to collide protons at extremely high energies and to produce events to be further studied. The LHC is built inside a 27 km long tunnel that holds a ring of superconducting magnets and a number of accelerating structures to boost the energy of the particles throughout the ring. Four main experiments using the modern tools to collect and analyze the collision data are installed at the collision points.

The mechanical apparatus of the LHC is briefly described in Section 1.1. Next, Section 1.2 is devoted to the Compact Muon Solenoid (CMS) detector and Section 1.3 briefly describes the principles of particle detection. Finally, the CMS requirements for the future high-luminosity HL-LHC operational phase are described in Section 1.4.

1.1 The Large Hadron Collider

The LHC collides two beams of protons organized in bunches and accelerated to travel at close to the speed of light at the maximal center-of-mass-energy of 6.5TeV. These bunches are dense, each containing one hundred billion of protons, leading to multiple collisions of pairs of protons in each bunch crossing (BX). The data resulting from a specific BX make up a physical event. An example of CMS event is given on Figure 1.1.

The primary goal of the LHC is to study proton-proton (pp) collisions with a nominal center-of-mass collision energies of 14TeV [2]. However, the target energy of the LHC acceleration stages was changed over the years, to be more progressive, as shown on Figure 1.2. We are currently within the second long shutdown (LS2) where a

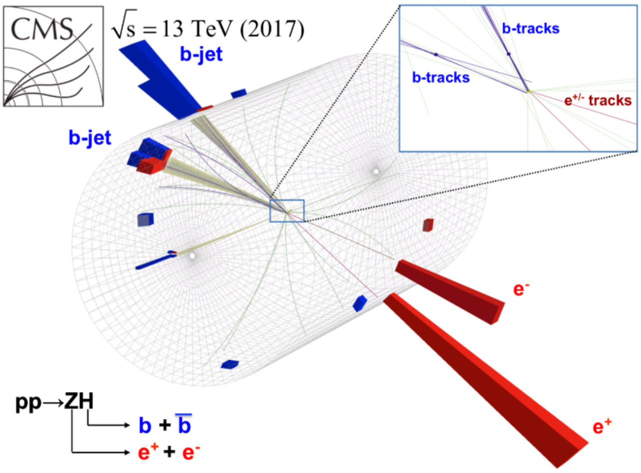


Figure 1.1: A CMS candidate event for the Higgs boson (H) decaying to two bottom quarks (b), in association with a Z boson decaying to an electron (e-) and an anti-electron (e+) [1].

second upgrade of the experiments is ongoing. After the LS3, in 2025, the machine will be upgraded for the High Luminosity era (HL-LHC) [3].

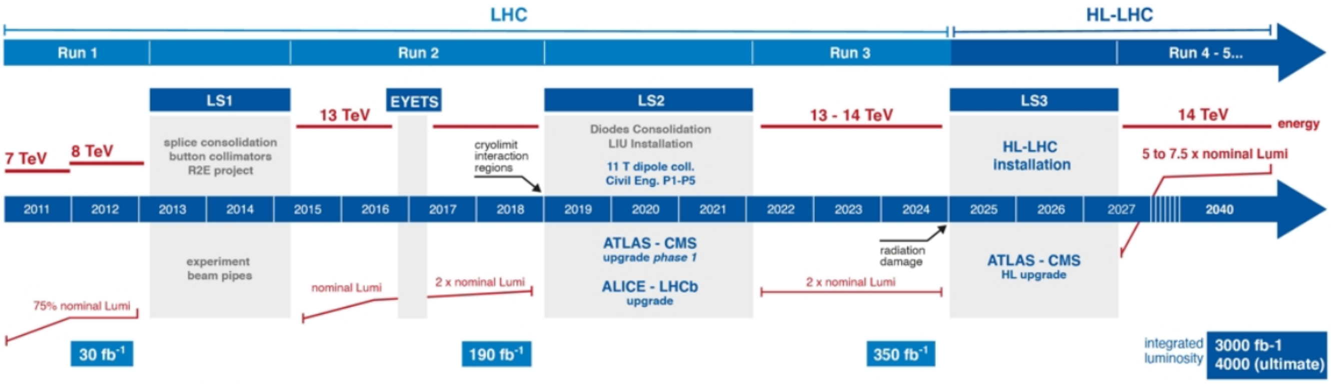


Figure 1.2: LHC operational phases with the targeted collisions energy and luminosity [3].

1.1.1 The LHC experiment design

The LHC tunnel is situated at a depth of about 100m under the border between France and Switzerland, near the city of Geneva. The beam accelerating concept is shown on Figure 1.3. It consists of an arranged set of machines, where each of them serves as a booster of particles accelerating the beam to a given energy before injecting it into the next machine in the chain [4].

At the very beginning of the accelerator chain is a hydrogen gas used as a proton source. Protons are extracted from the hydrogen atom by the application of an electric field. A linear accelerator (LINAC) is the first step of acceleration that boosts protons to the energy of 50MeV after which the beam is sent to the Proton Synchrotron Booster (PSB) for an additional energy boost up to 1.4GeV. Next, the Proton Synchrotron (PS) forces the injected

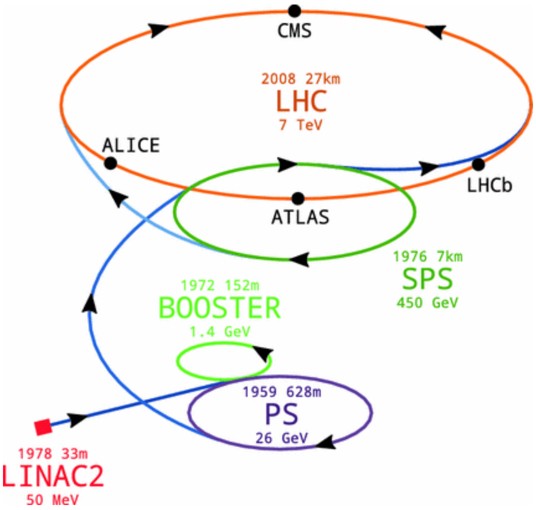


Figure 1.3: Simplified accelerator concept at CERN [5].

beam of protons to 25GeV, after which the Super Proton Synchrotron (SPS) increases its energy to 450GeV [6].

The LHC is the final part of the chain, in which the beams reach their highest energies. The beam is split to two beams that are sent in the two opposite directions. It takes about 20 minutes for protons to reach their maximum (targeted) energy while circulating inside the ring.

1.1.2 The LHC performance parameters

The LHC uses over 1600 superconducting magnets with a magnetic field strength of up to 8.3T, which is more than 100,000 times the Earth's magnetic field [7]. Their task is to keep the stability of the beams so that they are focused and precisely aligned towards each other, and that proton bunches are tightly squeezed to maximize the chances of the interaction. The particles inside the collided bunches interact with each other, but the "interesting" collisions are rare, and the essential accelerator task is to increase the interaction probability. Therefore, there is another important performance parameter besides the center of mass energy, which is the instantaneous luminosity L. The higher the luminosity of the collider, the more collisions occur in the detector, such that more data can be gathered for subsequent data analysis. By definition, the number of events per second N_{event} generated in the LHC collisions (pp) is given by [2]:

$$N_{event} = L * \sigma_{pp} \tag{1.1}$$

where σ_{pp} is the cross section of the pp interaction, which characterizes the probability that an interact will take place in a collision. For a given cross section, the larger the luminosity, the larger the number of events per second from the considered process. The cross section depends on the type of particles and the type of interaction (σ_{pp} ~80mb at 13TeV).

The cross section σ_{pp} has the dimension of an area, where larger transverse area means larger probability for the process to occur. Hence, the unit is cm^2, but usually smaller units are used such as barn (b), where $1b = 10^{-24}cm^2$.

The instantaneous luminosity L is calculated from the beam parameters and it is given in units of $cm^{-2}s^{-1}$. If we assume a Gaussian distribution of particles inside the two colliding beams with the same circular transverse sections, L is calculated with the following formula [8]:

$$L = \frac{N_1 * N_2 * f_{rev} * N_b}{4 * \pi * \sigma_x * \sigma_y} \tag{1.2}$$

where N_1 and N_2 are the number of particles per bunch for each of the two colliding proton beams, f_{rev} is the revolution frequency in the LHC ring, N_b is the number of bunches per beam, and σ_x and σ_y are the beam sizes at collision point in the horizontal and vertical directions. The Formula 1.2 shows how L depends on the particle beam parameters in practice, i.e. the number of particles per bunch and the beam sizes.

While the above instantaneous luminosity L is a measure of the number of collisions that can be produced in a detector per cm^2 and per second, the integrated luminosity L_{int} is accumulated over the time t of activity of the experiment [9]. L_{int} is expressed in inverse femtobarn ($fb^{-1} = \frac{1}{fb} = \frac{1}{10^{-15}b} = \frac{1}{10^{-15}*10^{-24}cm^2} = 10^{39}cm^{-2}$), which roughly corresponds to ~100 billion collisions [4]. It can be written as:

$$L_{int} = \int_0^t L\,dt \tag{1.3}$$

To compensate for the low cross section of interesting events, the LHC must have a high luminosity, reached through a high number of bunches per beam and a high number of particles per bunch. The resulting time between collisions of bunches is very short (25ns), which leads to the BX rate of 40MHz. The nominal LHC parameters for pp collisions are summarized in Table 1.1.

Table 1.1: The nominal parameters of pp collisions at LHC [10].

Parameter	Symbol	Nominal value
Design center-of-mass energy	$\sqrt{s}$	14 TeV
Design luminosity	L	$10^{34}\ cm^{-2}s^{-1}$
Time distance between bunches	Δt	25 ns
Number of bunches per beam	N_b	2808
Number of protons per bunch	N_p	$1.15 * 10^{11}$
Revolution frequency	f_{rev}	11245 Hz
Bunch crossing rate	f_{LHC}	40 MHz

1.1.3 Detectors

The two LHC beams cross in four different points where the main experimental detectors are placed, as shown in Figure 1.3, which are used to collect and analyze the collision data.

The biggest among them are ATLAS (A Toroidal LHC ApparatuS) and CMS, which are general-purpose detectors intended for high luminosity. Their concepts are rather different, using different subdetector technology choices for data measurements [11, 12]. The two detectors were built following the very similar physics goals. They initially targeted pp collisions, but have expanded to the study of heavy ions collisions.

The heavy ions collisions are also investigated with ALICE (A Large Ion Collider Experiment), but with low luminosity conditions [13]. Another detector devoted to a specific phenomena is LHCb [14], which studies interactions of b quarks.

1.1.4 The LHC operations

The LHC operation started with the first pp collisions in 2009 and reached the center-of-mass energy of 7TeV in 2010 (Figure 1.4). Around 30 fb^{-1} was delivered until LS1. About twice the nominal luminosity value was obtained before the LS2 (at the end of 2018) concluding the Run 1 and Run 2 operational phases (Figure 1.4). Expressed in inverse femtobarns, the LHC has delivered 150 fb^{-1} data before LS2. The history of the cumulative integrated luminosity recorded by CMS is represented in Figure 1.4.

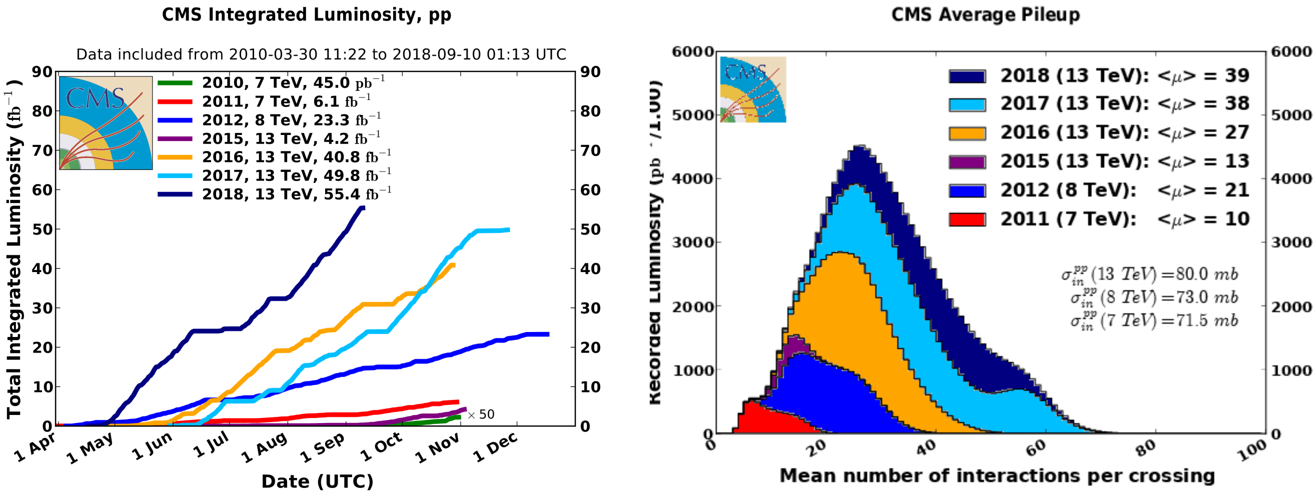


Figure 1.4: Total integrated luminosity (left) and the mean number of interactions per bunch crossing (right) for the pp collisions recorded by the CMS experiment during Run 1 and Run 2 [15].

During Run 1, 6.1 fb^{-1} was recorded by CMS from the LHC. The energy increased to 8 TeV and the recorded data of about 23.3 fb^{-1}. The Higgs discovery was announced in 2012, and the discovery was based on 10 fb^{-1} of data (5 fb^{-1} 2011 and 5 fb^{-1} 2012). During the Run 2, 40.8 fb^{-1}, 49.8 fb^{-1} and 55.4 fb^{-1} of data were recorded by CMS in 2016, 2017 and 2018, respectively. At the moment, the experiment is being prepared for the Run 3 phase of the LHC era, where the integrated luminosity is planned to reach a total of around 350 fb^{-1}. This will be the end of LHC Phase 1 after which the experiment will enter the HL-LHC phase, where the accumulated targeted luminosity foreseen by the end of the year 2038 is 3000 fb^{-1}. The technical preparation will occur during LS3 such that the experiment can cope with the luminosity conditions of HL-LHC. The goal of the upgraded machine will be to achieve a luminosity which is 5 to 7.5 times the nominal ($L \approx 5 * 10^{34}\ cm^{-2}s^{-1}$).

One of the major challenges that arises with the higher luminosity is the large increase of pile-up (PU) interactions. Namely, as the LHC collides dense bunches of protons and multiple protons interact when the bunches collide in CMS, many additional interactions occur along with the "interesting" interaction inside the same event. In these physical interactions, many other low-energy particles are produced besides the particles from the collision that is worthwhile studying (particles of interest). It is shown on Figure 1.4 how the mean number of simultaneous interactions per BX grows along with the luminosity growth over the years. The first data taking in 2011 had on average 10 PU interactions per BX. By the end of Run 2, this number exceeded by far the nominal conditions and almost 40 unwanted extra interactions (on average) are present in each event.

The former is one type of PU called the in-time, since additional pp collisions are inside the same BX as the collision of interest. Additionally, out-of-time PU results if signals spread over more than 25ns period, such that additional pp collisions are included from BX just before or just after the BX with the collision of interest [16]. In order to better understand the effect of PU inside the detector, a visualization is provided on Figure 1.5. Similar conditions are expected in the HL-LHC, where there will be on average 140 or 200 additional interactions per BX. The challenge of the upgraded operational phase of LHC is to maintain or even improve physics performance expected during Phase 1, where the environment is hard but it was by far easier (four times less PU in Run 2).

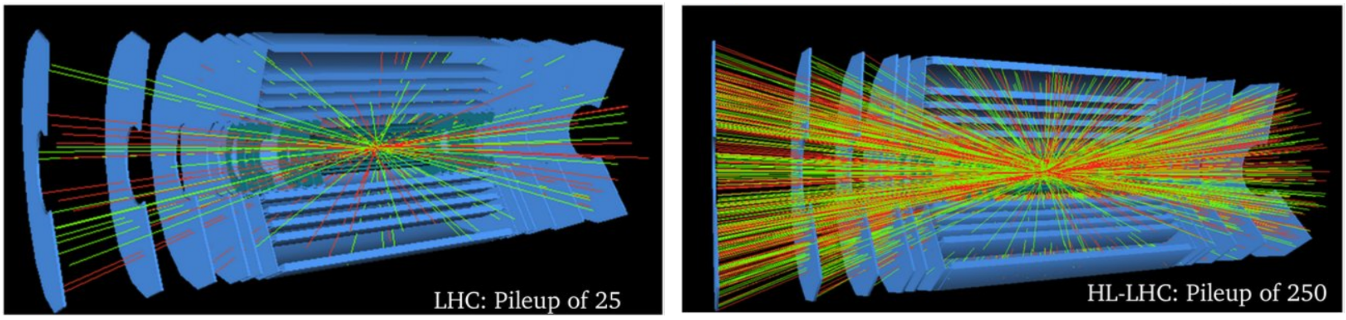


Figure 1.5: Simulated PU interactions inside the ATLAS detector [17].

1.2 The CMS experiment

The CMS detector has three main characteristics revealed from its name. It is small in dimensions compared to its mass ("compact"), being 15 meters tall and 21 meters long, with a weight of about 14000 tonnes. It is organized around a compact muon detection system ("muon") and a solenoid magnet ("solenoid") providing a strong magnetic field of 3.8T. The CMS consists of several subdetectors that are placed around the interaction point, forming a cylindrical structure, such that most of the solid angle around the interaction point is covered. Each subdetector is dedicated to the detection of different kinds of particles, as is described in Section 1.2.1. The coordinate system adopted by CMS is illustrated in Section 1.2.2, and the details of the CMS two-levels trigger system are provided in Section 1.2.3.

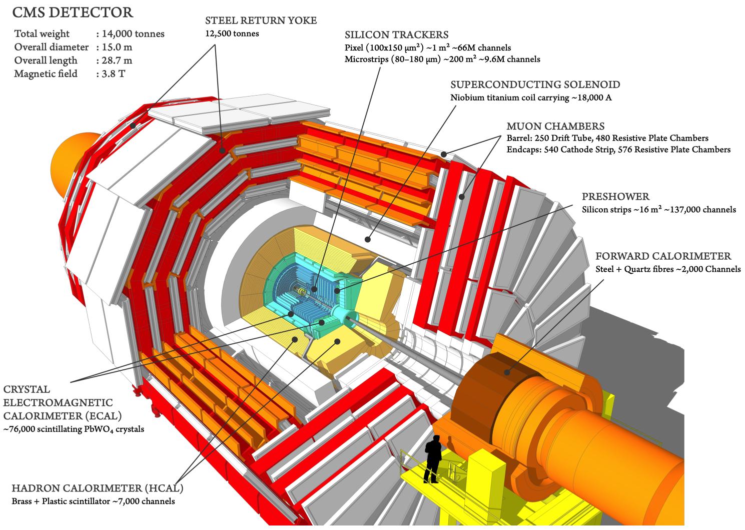

CMS DETECTOR
Total weight : 14,000 tonnes
Overall diameter : 15.0 m
Overall length : 28.7 m
Magnetic field : 3.8 T
STEEL RETURN YOKE
12,500 tonnes
SILICON TRACKERS
Pixel (100x150 μm²) ~1 m² ~66M channels
Microstrips (80–180 μm) ~200 m² ~9.6M channels
SUPERCONDUCTING SOLENOID
Niobium titanium coil carrying ~18,000 A
MUON CHAMBERS
Barrel: 250 Drift Tube, 480 Resistive Plate Chambers
Endcaps: 540 Cathode Strip, 576 Resistive Plate Chambers
PRESHOWER
Silicon strips ~16 m² ~137,000 channels
FORWARD CALORIMETER
Steel + Quartz fibres ~2,000 Channels
CRYSTAL ELECTROMAGNETIC CALORIMETER (ECAL)
~76,000 scintillating PbWO4 crystals
HADRON CALORIMETER (HCAL)
Brass + Plastic scintillator ~7,000 channels

Figure 1.6: CMS detector structure [18]. The cylindrical central part is "barrel", and the two facing "endcap" sections are covering the forward regions.

1.2.1 Detector design

The overall layout of CMS is shown in Figure 1.6. All cylindrical subparts are divided into a barrel covering the central region, and two endcaps at the detector ends covering the forward region. The former is closer, while the latter is further away from the interaction point. The goal of such a structure is to provide a good coverage all around the collision, such that almost everything arising from it is detected, enabling the reconstruction of the events. The heart of CMS is a 6-m-inner-diameter superconducting solenoid providing a large bending power before the muon bending angle is identified by the muon system [11]. The bore of the magnet coil is large enough to hold the inner tracker and the calorimetry inside.

Considering the collision happening at the detector center, the first detector subpart is the tracking system whose volume is a cylinder of 2.6-m diameter. This tracking detection is fully silicon-based and, in order to deal with high track multiplicities, CMS employs 10 layers of silicon microstrip detectors in the outer region, together with 3 (now 4) layers of silicon pixel detectors placed close to the interaction region to improve the measurement of the charged-particle tracks and the position of track vertices. The signals measured in the tracker layers are called "hits", and these are connected to reconstruct the particle tracks and their origin called "vertices". Surrounding the tracker is a scintillating crystal electromagnetic calorimeter (ECAL), where electromagnetic (EM) showers develop. They are initiated by the EM particles, electrons and photons, whose energy and position are measured with the ECAL, by absorbing the EM shower in the calorimeter, and the particles are identified as clusters of energy in neighbouring ECAL cells.

There are two preshower detectors (one in each endcap), which are of high-granularity. Placed in front of the ECAL, they help to distinguish single-photon energy deposits from double-photon ones, to reduce some background events. The ECAL is surrounded by a hadronic calorimeter (HCAL). Hadronic (HAD) showers are absorbed with the HCAL, while they may have started in the ECAL, leaving some of their energy. The muon detectors are placed in the most external part of the CMS structure, since muons pass all the subdetectors with very little interactions. They are precisely measured by the tracker and identified by the muon system.

The tracking system

The CMS tracker aims at reconstructing the trajectories of charged particles coming from the collision in its acceptance. This requires fast response and high efficiency with the 25ns spacings between BXs, and on average about 1000 particles from more than 20 overlapping pp interactions traversing the tracker for each BX (calculated for the design luminosity). The "interesting" vertices must be separated from the ones originated by the large number of PU interactions.

The tracking system is composed of the two parts: a pixel detector in the innermost region, and a silicon strip tracker, as shown in the tracker scheme on Figure 1.7. The pixel detector consists of three barrel layers at a radius $4.4cm < r < 10.2cm$, in the central part and cover an area of about $1m^2$. The silicon strip tracker barrel has a total of 10 layers and extends to a radius of 1.1 m [11].

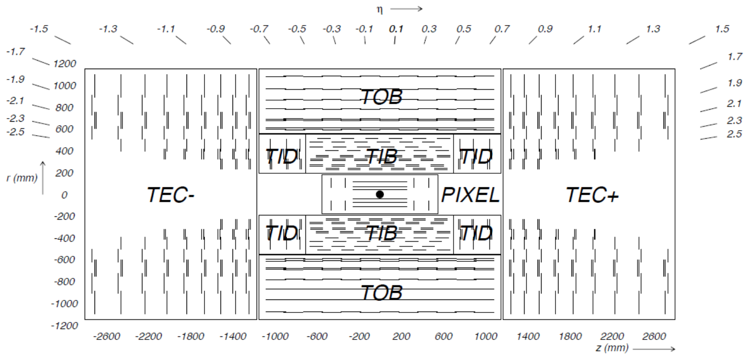


Figure 1.7: CMS tracker schematic. The central pixel detector and the silicon strip detector, which consists of: the tracker inner barrel (TIB), the tracker outer barrel (TOB), and the two mirrored endcaps, the tracker inner disc (TID) and the tracker endcap (TEC) [11].

Each part of the tracking system is complemented by the endcaps on each side, i.e. two tracking discs in the pixel detector and three followed by nine tracking disks in the strip tracker. It is important to note that the pixel detector of the CMS tracker achieves a spatial resolution of 10μm in (x, y) and 20μm along z, offering a three-dimensional (3D) vertex reconstruction of particle trajectories. During 2017, the pixel detector tracker was upgraded with one more layer in the barrel and in the endcap parts [19].

The electromagnetic calorimeter

The CMS ECAL is a hermetic, high-resolution, high-granularity homogeneous calorimeter made of lead tungstate ($PbWO_4$) crystals. The position of the crystal arrays in both the ECAL barrel and endcaps, is illustrated on Figure 1.8. The ECAL provides a coverage in pseudorapidity of up to $|\eta| < 3.0$, whereas the barrel covers the region of $|\eta| < 1.479$. The crystals are approximately pointing towards the collision point, such that there are no gaps in between.

The ECAL is designed to provide the energy measurement for electrons and photons. The EM shower development inside the absorber is illustrated on Figure 1.8. It is a set of EM interactions with the detector material (the lead tungsten crystals), where the two dominated processes are the pair production (or photon conversion into an electron-positron pair) and the bremsstrahlung (emission of a photon from an electron or positron) [20]. The step of the avalanche of particles is measured with radiation length (X_0), which depends on the material. Each time a gamma particle is produced, the initial energy of the particle in the former step is further decreased. The particle multiplication process stops when the critical energy E_c is reached. At that point, the multiplications do not continue because the energy loss by other processes starts to dominate.

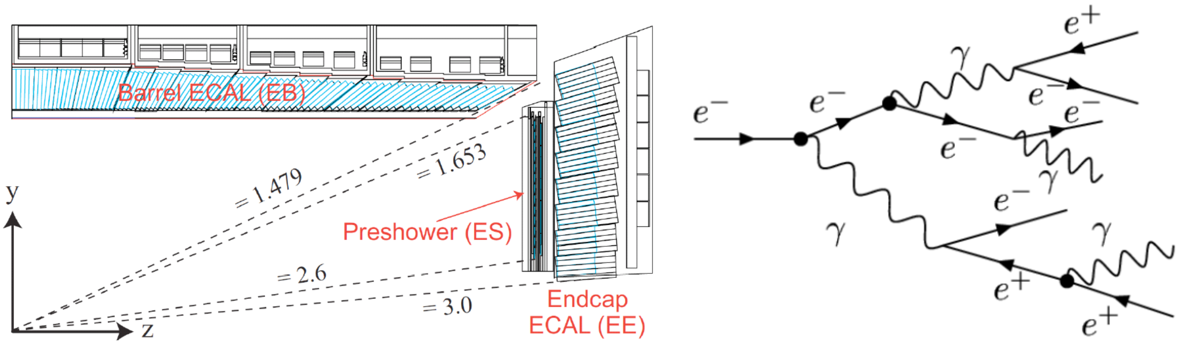


Figure 1.8: CMS ECAL slice in the first quadrant [21] (left) and the EM shower development (right) [22].

The maximum of the shower t_{max} depends on the absorber material, and it is the length until the shower develops, after which particles just travel inside the medium and gradually lose the rest of their energy. In the simplified model such as the one on the Figure 1.8, the number of particles produced at the step t is $N(t) = 2^t$, and the average energy of all the particles produced is $E(t) = \frac{E}{N(t)}$. It follows that: $t_{max} = \frac{ln\frac{E_0}{E_c}}{ln2}$. This suggests that the maximal shower depth t_{max} behaves as the logarithm of the initial energy. Also, it predicts that the shower longitudinal energy profile will rise rapidly up to a peak value of t_{max} and then fall to zero [20].

Besides the radiation length X_0 which characterizes the longitudinal EM profile, where X_0 is defined as the mean distance over which a high-energy electron loses all but $\frac{1}{e}$ of its energy by radiation, the EM shower is characterized in the transverse plane with another metric called the Moliere radius R_M. Namely, about 90% of the total EM energy is contained inside a cylinder of radius $r = R_M$.

The choice of $PbWO_4$ material in ECAL is motivated by the very high density and good light yield, and the former leads to a small radiation length ($X_0 = 0.89$cm) and a small Moliere radius ($R_M = 2.19$cm). It allows for a compact calorimeter, where $25X_0$ of the absorber (the average lengths of crystals in the barrel and the endcaps is $25.8X_0$ and $24.7X_0$ respectively, i.e. about 22cm) is used to contain the shower and capture the total energy. The power of ECAL crystals is in the fine transverse granularity offered. Furthermore, $PbWO_4$ crystals enable a fast response, where 99% of the light is collected in 100ns, which is permits to work with the 40MHz interaction rate of the LHC [21]. The light results from the photon emission from the scintillator material and it is measured with the electronics positioned at the crystal rear. In the barrel and the endcaps, avalanche photodiodes (APD) and vacuum phototriodes are used, respectively. The intensity of the light emitted in the reaction inside the crystal is proportional to the energy absorbed by the crystal so to provide a measure of the energy of electrons or photons that initiated the shower.

The hadronic calorimeter

The HCAL measures the energy of hadrons that traverse the tracker and may leave already about 30% of their energy in ECAL. Unlike ECAL, the HCAL is a sampling calorimeter (see Section 2.1), whose structural design is illustrated in Figure 1.9.

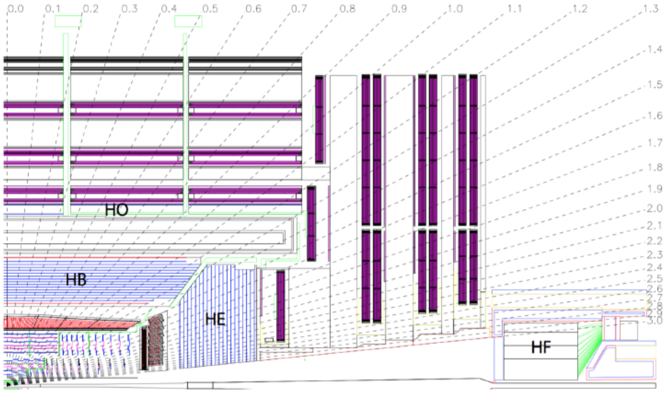


Figure 1.9: CMS HCAL slice in the first quadrant [11].

The HCAL consists of heavy absorbers and scintillator (detector) layers. When a hadronic particle interacts with the absorber (brass or steel), an interaction occurs producing numerous secondary particles. Again, they flow through successive absorber layers and further interact and produce a shower of particles. As the hadronic shower develops, the particles pass through alternating layers of active scintillation material, where the scintillation light is produced by ionization and excitation of the medium and measured as a signal.

The Barrel Hadronic Calorimeter (HB) covers up to $|\eta| < 1.4$, and the Endcap Hadronic Calorimeter (HE) covers the $1.3 < |\eta| < 3.0$ region. The design of both, HB and HE, is similar, where the HB is made out of 2304 towers of $\Delta\eta$x$\Delta\phi = 0.087$x0.087. Since this is not enough to capture the long hadronic shower with a large spread in longitudinal direction, an Outer Hadronic Calorimeter (HO) is placed outside the solenoid volume, covering the $|\eta| < 1.4$ region. Finally, the $3 < |\eta| < 5.2$ region is covered by the Forward Hadronic Calorimeter (HF) that has to be more resistant to the intense radiation striking on the forward detector regions [11].

1.2.2 Coordinate system

The convention for the definition of the right-hand CMS coordinate system is given on Figure 1.10. The center of the coordinate system is the interaction point and the beam direction is parallel to the z axis. The y axis is positioned perpendicular to the beam, where the x axis points towards the center of the LHC ring. The former are Cartesian coordinates, but the cylindrical or polar dimensions r and ϕ are also used, which are calculated based on x and y [22]. Hence, the azimuthal angle ϕ is measured from the x axis in the transversal x-y plane and it takes values $\phi \in [-\pi, \pi]$. The polar angle θ is measured from the z axis and takes values $\theta \in [0, \pi]$.

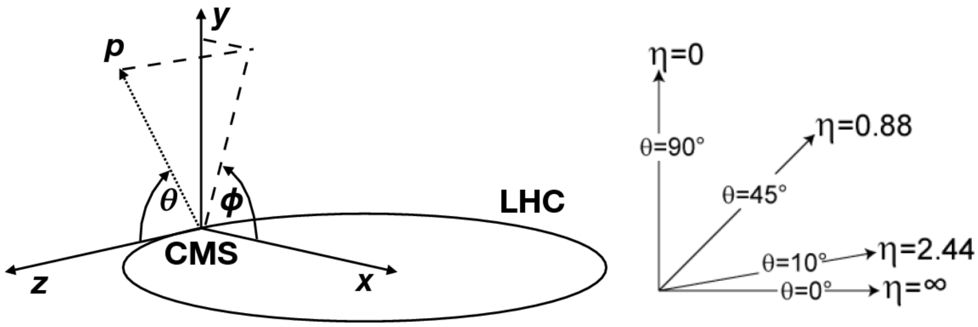


Figure 1.10: CMS coordinate system showing a particle with momentum p, produced at the origin of CMS (left) [23] and the pseudorapidity η growth with lower polar angle θ (right) [22].

The polar angle coordinate is usually expressed as the pseudorapidity η (Figure 1.10) that is calculated with the following formula:

$$\eta = -ln(tan\frac{\theta}{2}) \tag{1.4}$$

The particle momentum p is shown on Figure 1.10, which corresponds to $p = E$ at high energies. Even though CMS measures energy E in the calorimeter cells, in this thesis we use the transversal energy or transversal particle momentum, where $E_T = p_T = \sqrt{p_x^2 + p_y^2}$. Naturally, p_T corresponds to the x-y plane component of the momentum and it is calculated as $p_T = p * sin\theta$.

1.2.3 Trigger system

It has already been emphasized that the pp beams crossing interval is 25ns, which leads to a BX working frequency of 40MHz. Depending on the targeted luminosity, many collisions can occur at each proton bunch crossing. For example, around 20 simultaneous pp collisions were present at the nominal, and around 40 collisions were really present in the 2018 LHC luminosity, producing terabytes of data per second. As it is impossible to store and process all this data, a data reduction mechanism has to be used [11]. This is accomplished with the CMS trigger system, which quickly goes through the event data (the data from the selected BX), and performs a selection to decide whether or not to keep the event. The total output data rate is reduced down to $\sim$1kHz in the two-step process divided between the Level-1 (L1) and the High-Level Trigger (HLT), described in the following.

The Level-1 Trigger

The L1 trigger is completely hardware-based, and reduces the data rate from 40MHz to 100kHz, which means that we keep roughly 100 000 events out of 40 million in total (per second). The L1 electronics is based on Application-Specific Integrated Circuit (ASIC) and Field Programmable Gate Array (FPGA). The L1 trigger system is composed of local, regional and global views of the detectors, which means that several application-specific trigger subsystems are used. An example of L1 subsystem that exhibits a local view of the detector is the calorimeter trigger which analyzes the data from the CMS calorimeters. Such local trigger is examined in the context of this thesis and it is called Trigger Primitive Generator (TPG), being a preliminary step for the L1 trigger based on energy deposits in the calorimeter cells.

The energy measured in the calorimeters is delivered to the central L1 trigger in the form of the "trigger primitives" (TPs). These are combined to reconstruct calorimeter trigger objects such as electrons, photons, hadrons, jets or total energy sums. The L1 trigger has a very limited time to decide about which events to keep. There is no possibility to re-visit the decision once the data is thrown away. The maximal allowed L1 latency must not exceed 4 microseconds, after which data is taken over by the HLT.

The High Level Trigger

The HLT further reduces the data rate from 100kHz to 1kHz, which means that only 1000 events per second are kept from a BX for the storage and the full reconstruction. Unlike the L1 trigger, which consists of custom-designed and programmable electronics, the HLT is a software system implemented as a farm of about one thousand parallel data processors. Also, the HLT may use the information from the full detector. By having the complete read-out data, it can perform higher-level calculations [11]. The HLT makes a software-based decision in about 200ms.

1.3 Physics event reconstruction

The physics event reconstruction is based on the experimental signatures from the full detector, from which one can identify various types of particles. To illustrate this, the longitudinal slice of CMS is presented on Figure 1.11. First, all charged particles such as electrons, muons, or charged hadrons leave tracks in the tracking sub-detector, while unconverted photons or neutral hadrons pass the tracker without producing a signal. Besides the track itself, the particle signature consists of the energy deposits measured in the specific sub-calorimeter parts.

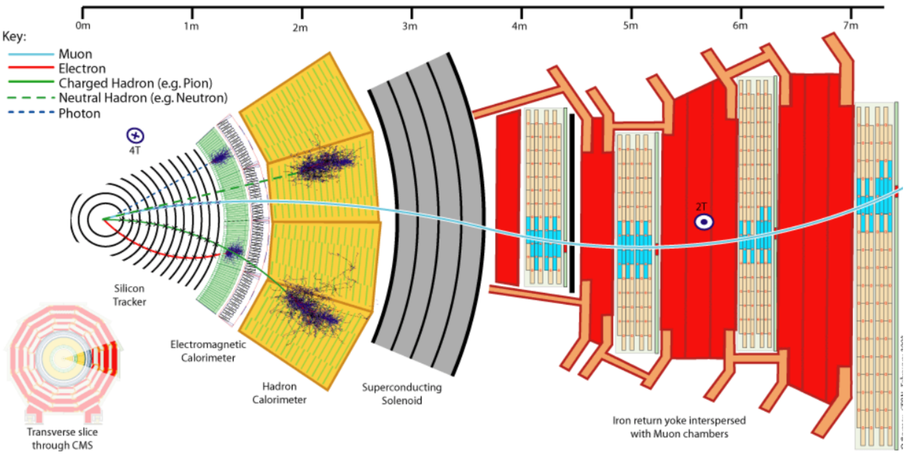

0m
1m
2m
3m
4m
5m
6m
7m
Key:
Muon
Electron
Charged Hadron (e.g. Pion)
Neutral Hadron (e.g. Neutron)
Photon
4T
2T
Silicon Tracker
Electromagnetic Calorimeter
Hadron Calorimeter
Superconducting Solenoid
Iron return yoke interspersed with Muon chambers
Transverse slice through CMS

Figure 1.11: Longitudinal slice through CMS [24].

Hence, the energy deposited by photons and electrons is detected in ECAL, where also energy deposits can originate from hadrons interacting in the ECAL. The largest part of hadron energy is detected in HCAL calorimeter, while the combination with the track information from the tracker enables the discrimination between the charged or neutral hadron particles. Finally, muons pass both ECAL and HCAL essentially undetected, and their measurement is performed by the tracker and the dedicated muon detectors at the CMS external layers.

1.4 The CMS upgrade in HL-LHC

To summarize, CMS experiment is one of the most famous big data sources, with a 40MHz bunch crossing rate producing terabytes of data per second. It is not possible to transfer and store this data volume directly, so a trigger system is designed to reduce the number of events. These big data conditions will be even more demanding after the major luminosity upgrade of LHC planned around year 2026 with the HL-LHC phase. Since the LHC experiments are constantly being improved to meet the new technology trends and replace the parts damaged by radiation, the upgrade of both mechanical parts and trigger system, is foreseen.

In particular, the ECAL and HCAL endcaps will be replaced by new versions with unprecedented transverse and longitudinal segmentation. It is called the High Granularity Calorimeter (HGCAL), where the fine structure of showers can be reconstructed and used to enhance particle identification. The new calorimeter consists of both ECAL and HCAL sampling layers, where a hexagonal sensor geometry will be used, with small hexagonal silicon sensors as active material.

At HL-LHC, a better chance of observing rare events is expected, but the conditions are very challenging. The first goal is to maintain or even improve the reconstruction performance from the previous LHC operational phase in this highly demanding HL-LHC environment. The luminosity will increase by a factor $\sim$3-4 (from 2*10^{34} $cm^{-2}s^{-1}$ in 2018 to $\sim$5-7.5*$10^{34}cm^{-2}s^{-1}$ in 2026). The integrated luminosity after $\sim$10 years of operation is expected to reach $\sim$3000fb^{-1}, and 10 times the one of the first phase of LHC. The new upgraded detector parts must be very radiation hardened to be able to cope with the increased radiation dose. Moreover, a higher number of concurrent interactions per BX called PU is expected, since the average PU will be 140-200 at HL-LHC, a factor of four larger than the Run 2 LHC values. This PU challenge requires a finer sensor granularity as will be accomplished with the new HGCAL technology.

The larger detector granularity will affect the trigger, because larger data volume must be readout compared to the LHC era. Also, both fast and high-quality decisions must be made on this data, which requires a very high hardware processing power. An upgraded HGCAL trigger chain for L1 is proposed by the CMS collaboration and needs to enable that the enhanced decision-making is possible. A compromise is needed between the available data links for the communication between the various trigger stages and the processing power required for the algorithms to fulfill the trigger functions within the allowed time and with the available hardware.

Chapter 2

The new Phase-2 CMS endcap HGCAL calorimeter

The future HGCAL detector will provide a new paradigm to calorimetry, and it is one of the most demanding projects, with an extremely high granularity and therefore the high number of channels. Unlike the current homogeneous calorimeter, the sampling calorimeter concept will be used, providing a three-dimensional (3D) shower image. The advantages of the new detector will be described in Section 2.1. Next, the engineering design will be summarized in Section 2.2, as well as the hexagonal geometry used in the detector sensor module design. Finally, Section 2.3 will introduce the trigger readout architecture. The on-detector electronics adjusted to the the harsh radiation environment during HL-LHC operation will be summarized, as well as the TPG algorithm concept and the baseline architecture. Section 2.4 describes the simulation software and the data samples used in the trigger studies.

2.1 Detector longitudinal structure and granularity

Unlike the homogeneous ECAL crystals in CMS, where the medium is both the absorber and the active material, the HGCAL is a sampling calorimeter in which these two functions are separated. Hence, the dense absorber material is interleaved with the active silicon layers. The absorber is where particles interact, initiating the EM shower development and forming the cascade of particles. The initial particle energy E_0 is absorbed, where the number of secondary particles is proportional to E_0. The goal of the calorimeter is to measure the energy loss ΔE or energy absorbed, and thus enable to count the number of particles produced in the shower.

In a perfect calorimeter, if all particles were counted, the energy resolution R of the calorimeter is $\Delta E \approx \sqrt{N} \approx \sqrt{E}$, where N is the number of particles [25]. Since the R is expressed as a ratio $R = \frac{\Delta E}{E}$, the performance of the ideal measurement case is:

$$R = \frac{\Delta E}{E} = \frac{1}{\sqrt{E}} \tag{2.1}$$

where E is the incident energy. For a good detector resolution, the factor R should be minimized, and this is accomplished when the measured energy is very high. In reality, when effects such as leakage, non-uniformities, noise etc. are included, the Formula 2.1 is transformed to:

$$R = \frac{\Delta E}{E} = \frac{a}{\sqrt{E}} + \frac{b}{E} + c \tag{2.2}$$

where a is the stochastic term (caused by shower fluctuations and sampling fluctuations), b is the noise term (caused by the readout-electronics), and c is a constant term (independent of energy and caused by imperfections in the calorimeter construction). For homogeneous calorimeter such as ECAL, the stochastic term is dominant (due to lower energies). It is satisfactory because the active material is put everywhere, such that the range $a = 2\%$ to $a = 3\%$ offers the minimal stochastic factor and thus a good statistical precision. The other factors are the electronic noise and the rear leakage of the measured signal on the crystals. Their values are $b = 12\%$ and $c = 0.3\%$ (coming only from the geometry) [26]. The additional source of fluctuations in ECAL is the variation of properties from crystal to crystal (in the full calorimeter) or in a single crystal. A benefit is the homogeneity itself, which enables the same detector response from everywhere.

On the contrary, in a sampling calorimeter such as HGCAL, there is a larger stochastic term with typically between $a = 10\%$ to $a = 30\%$. It is caused by the sampling fluctuations, because the active material is positioned at specific places (interleaved with the absorber). Hence, there will be variations of the measured energy from shower to shower, when particles traverse the active layers. Even though the HGCAL detector resolution is degraded compared to a homogeneous calorimeter, it is intended for a better particle separation at HL-LHC that will be necessary because of the amount of PU. What matters is the very high segmentation for PU rejection. At high energy, the constant term is the most important, so a design goal of the future HGCAL is to keep it at the level of $\approx 1\%$ or below.

A general advantage of the HGCAL is that it is more radiation hard than the ECAL lead tungstate, where radiations create defects in the crystal structure. Also, unlike for a the homogeneous calorimeter, HGCAL enables both transverse and longitudinal fine segmentation, which will help to significantly suppress PU contributions. Transverse segmentation is accomplished with the use of very small silicon readout cells and helps to separate nearby showers. Therefore, the detector transverse granularity is very important for the event reconstruction and the cell-size should be less than the Moliere radius. This is needed to detect the shower spread in the lateral direction. If cells size was larger or equal to Moliere radius, it would contain all the shower in a single cell and it would not be possible to detect the shower energy pattern.

The chosen cell-size varies from ≈1.2cm^2 to ≈0.5cm^2 depending on the pseudorapidity. The choice of the cell size is also constrained by considerations of the trigger cell (TC) size that is adjusted to obtain an integer number of TCs within a sensor module (SM). This is important because sharing TCs between modules is not feasible in terms of the communication limits and it should be avoided. TCs are groups of either four of the larger-sized basic cells or nine of the smaller-sized basic cells [27]. The basic cell area (and therefore the TC area) is related to the minimal ionizing particle (MIP), which is a very small signal without shower initiation (but the ionization only). Each cell size corresponds to a specific signal over background ratio concerning MIP. This one is decreased for a cell larger than 1.2cm^2, because larger area means larger noise, so the MIP signal would be lost.

The HGCAL consists of an electromagnetic compartment (CE-E) followed by a hadronic compartment (CE-H), as shown schematically on Figure 2.1. The silicon part of the calorimeter will be followed by a scintillator part, for which the active medium changes from silicon sensors to plastic scintillator tiles. This one is used in the regions with the lowest radiation, where the large radiation hardness of silicon is not mandatory, so that the cost of the mechanical construction is reduced. The plastic scintillator tiles are of size 2x2 to 5.5x5.5cm^2. They are structured to match the geometry with the silicon cells, so that the cells in the inner endcap edge will be smaller in area (≈4cm^2) than those at the outer edge (≈32cm^2). The scintillation light is read out by silicon photomultipliers that measure the collected light and transform it to an electrical signal.

2.2 Engineering design

The HGCAL mechanical design is described in the Technical Design Report (TDR) [27]. The future detector will be realized as a 52-layers structure, where the CE-E will have 28 layers in a depth of ≈26X_0 and ≈1.7λ. The absorber material in CE-E is mostly lead (Pb) mixed with copper (Cu) sandwiched between the two layers of the copper tungstat (WCu). Hexagonal silicon sensors are used as active detector elements. The total thickness of CE-E sampling layers is 34cm. Also, sensors with three different sensitive thickness are deployed in the CE-E part: 300, 200, and 120 micrometers (lower thickness in the highest radiation environment). Modules are tiled together to cover the detector layers. Each plane of the layer is subdivided into $60°$ wedges called cassettes.

The CE-H will consist of 24 layers arranged in depth for a total thickness of ≈8.5λ. Silicon sensors will be used in the innermost region of this section such that the same technology is used as in CE-E, due to the higher expected level of radiation. For the outermost region, plastic scintillator tiles will be used together with steel as the absorber. The absorber in the CE-H consists of 12 planes of 35mm thick steel plates followed by another 12 steel planes with a thickness of 68mm. The silicon modules are placed between these absorbers, together with the scintillator tiles mounted on 6mm thick plates and forming cassettes with boundaries at $30°$.

The total calorimeter thickness, perpendicular to the layers, will be 10.7λ. The whole calorimeter will be positioned in a volume cooled by a two-phase CO_2 system and maintained at the temperature of $-30°$. All layers will be

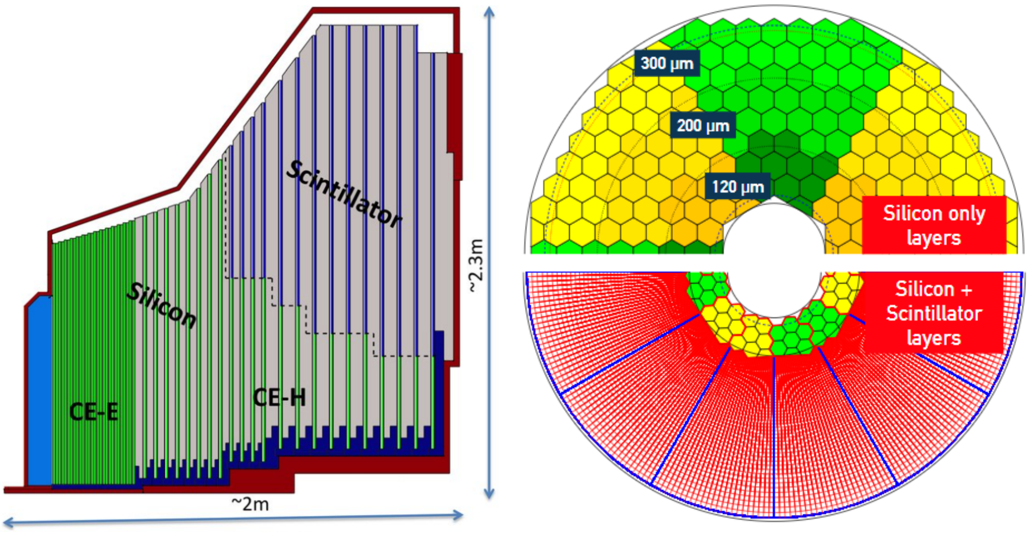


Figure 2.1: Schematic of transverse and longitudinal view of the HGCAL. The electromagnetic calorimeter is 28 layers long and comprised by silicon, and the hadronic calorimeter has 22 layers with a combination of silicon and scintillator [28, 29].

readout for the energy measurements. However, only half of the layers in CE-E (alternating layers) together with all the CE-H layers will be used for the trigger [27].

Table 2.1: Parameters for the HGCAL design [30].

Both endcaps	Silicon	Scintillators
Area	600m^2	500m^2
Number of modules	27000	4000
Channel size	0.5-1cm^2	4-30cm^2
Number of channels	6M	400k
Operating temperature	-30°	-30°

The parameter values of the new HGCAL design are summarized in Table 2.1. Further details of the silicon sensor cells (SCs), the construction of SMs and the overview of the cassettes forming structural design are given in what follows.

2.2.1 Silicon sensors

Silicon sensors active material is present in the CE-E and the inner parts of the CE-H sections. Concerning the CE-E, the sensors thickness depends on the radiation levels, such that 120μm thickness is used in the region where the highest radiation is expected (highest pseudorapidity, close to the z axis), 200μm in the medium radiation region, and 300μm in the region where the lowest radiation is expected (Figure 2.1). Silicon thickness is related to radiation, and we use lower thickness sensors in the most energetic detector part because they are more resilient to irradiation. Thicker sensors (larger thickness) have the improved charge collection before irradiation, but they are more affected by radiation (the charge collection efficiency is more reduced when irradiated).

The choice of the hexagonal sensor is driven by cost, since a reduction of the silicon material waste is obtained when produced from a circular wafer compared to rectangles or squares. Also, deploying hexagonal modules also allows for a better coverage of the detector layer, whereas a truncation of the vertices of the modules must be foreseen to allow clearance for the mounting and the module fixation system [27]. The inner radius is 32.8cm and the outer radius is 160cm. The circular wedges or cassettes are formed as layer sub-regions separated by $60°$.

In CE-H, the silicon coverage is for approximately $|\eta| > 2.4$ (depending on the layer), i.e. in the eta region where more important radiation levels are expected. The size of the scintillator cells located at the boundary between silicon and scintilator is kept similar to the silicon TCs, i.e. ≈4cm^2. The goal is to ease the transition between the two, such that this does not affect the efficiency of the reconstruction algorithm.

Also, as illustrated on Figure 2.1, the cells closer to the beam line will be smaller than the cells at the outer edge (32cm^2). The former will provide better signal at regions where both the radiation and noise will be the largest. On the other hand, at a larger radius, where the radiation level and the noise are smaller, the scintillator cells are larger to reduce the number of channels and the total calorimeter cost [27].

2.2.2 Sensor modules and layout

The hexagonal silicon sensors are fabricated from 8inch (200mm in diameter) circular wafers, which limits the SM size. For simplicity, a hexagonal sensor wafer (the cut out hexagonal sensor) is a module used for readout [27]. It is foreseen by the TDR that 192 large-size SCs and 432 small-size SCs are fit inside the SM. The total number of channels needed for the readout depends on the module architecture, which results from the cell size and the module size. There were several possible choices for the module design, and the studies on how to efficiently assort the SCs inside and calculate their total number is presented in Section 3.

Figure 2.2 presents a drawing of the final choice on the selected SMs for HGCAL with large and small SCs. The vertex module cuts are shown (called "mouse bites"), that are needed for the fixation of the module onto the detector layer. A similar kind of geometry studies resulted in the choice of how to group inner SCs to form TCs. Different geometries and tiling options have been examined, and their advantages and disadvantages have been considered (Section 3). The symmetric diamond TC of 2x2 and 3x3 hexagonal SCs is chosen, forming a SM with three symmetrical sub-parts to be readout separately with the electronics (Figure 2.3). The TC area is the same for large and small cells, 4*1.18=4.7cm^2 and 9*0.52=4.7cm^2 respectively. There are in total 48 TCs inside the module in both cases.

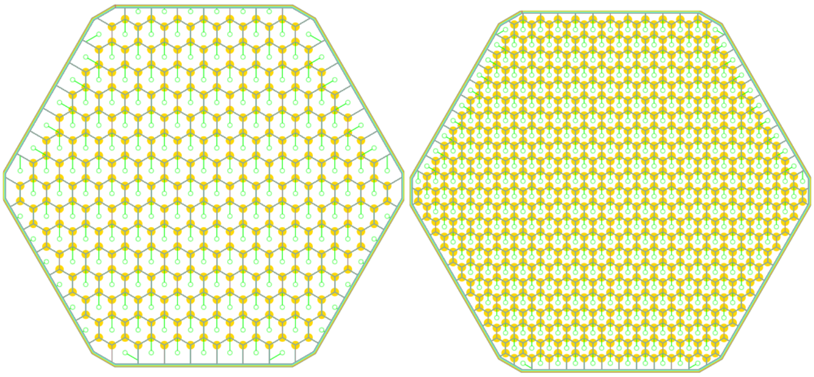


Figure 2.2: Layout of an 8inch SM for the HGCAL, which is divided into large-cells of 1.18cm^2 (left) and the small-cells of 0.52cm^2 (right) [27].

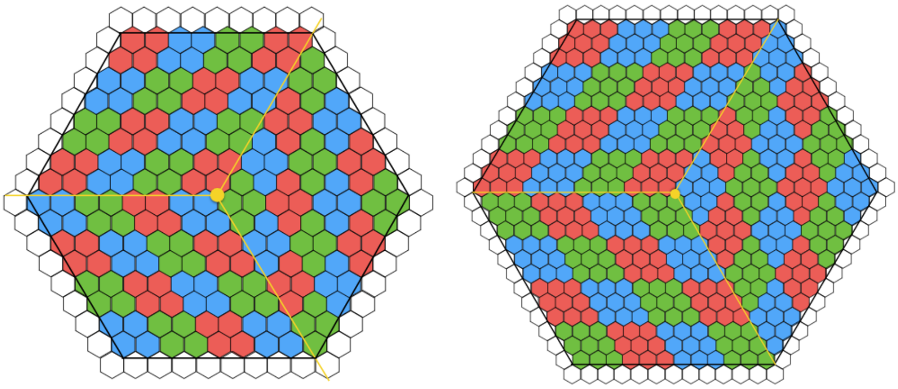


Figure 2.3: Formation of TCs by a clustering procedure with four large-cells of 1.18cm^2 (left) and nine small-cells of 0.52cm^2 (right) [27].

2.3 The HGCAL trigger readout architecture

The general trigger architecture is shown on Figure 2.4, which consists of the three main parts [31]:

- Very Front End (VFE) - It consists of the HGCAL read-out chip (HGCROC), which is an ASIC in the high radiation "on-detector" zone. It measures and digitizes the charge deposited in the silicon SCs and forms TCs.
- Front End (FE) - It consists of the concentrator ASICs called endcap concentrators (ECONs). The ECON-T is used for the trigger, which selects the fraction of trigger data before transmitting to the next stage. The ECON-D sends zero suppressed fine granularity data to the data acquisition system.
- Back End (BE) - It consists of two processing layers outside the high radiation zone that perform the TPG on FPGAs and finally deliver the TPs.

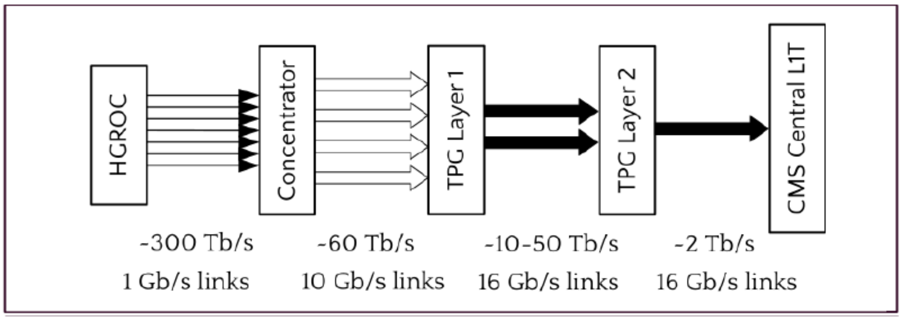


Figure 2.4: General scheme of the trigger, describing the data-flow between the several processing steps [31].

2.3.1 On-detector electronics

The HGCROC is the first interface between the detector signals and the VFE electronics starting to form the detector event data. The signals delivered from the detector are collected, shaped and digitized on the VFE. Namely, the silicon sensors (from the CE-E and the CE-H) or the silicon photomultipliers (from the CE-H) are connected to the FE ASIC HGCROC that measures and digitizes the charge. For the silicon sensors, the signal is first pre-amplified, after which the charge measurement is performed. For charges up to 100–150fC, a low-power 10-bit analog-to-digital-converters (ADC) in 130nm technology is used, while for charges above 50fC the time-over-threshold (ToT) technique is applied with a 12-bit time-to-digital-converters (TDC). ADC and ToT data are recombined, linearised, and calibrated to provide a single charge measurement which is used to build trigger sums. Sums of 4 large-cell charges or 9 small-cells charges of adjacent channels are formed and the result is truncated and compressed with an 8-bit format.

The VFE transmits the digitized data to the BE electronics. The TPG actually starts from the early VFE stage that is subject to critical requirements driving the performance of the HGCAL trigger [29]. There are three HGCROCs on a large-cell module and six HGCROCs on a small-cell module (Figure 2.5), where each of them provides 16 and 8 TC energies respectively. The HGCROC deals with the signal analog charge and performs the time measurements. Since each SCs represents a single HGCROC input, there are in total 4*16=64 and 9*8=72 inputs for large-cell and small-cell module. The HGCROC does two main functions: it sends the zero suppressed event data (via two 1.28 Gbps links) to the ECON-D, and performs the first trigger processing forming digital sums of TCs. Two or four links are used for the trigger data, depending on the number of TCs to be sent.

The characteristics of HGCROC are the following. First, it supports a large dynamic range of signal charge, such that high-energy deposits can be measured with low noise. Next, it is radiation hard, and a low power is needed for the data readout from each sensor channel ($<20mW$) and for the power for the whole circuit. Also, it can accommodate the 12.5 microseconds latency of the L1 trigger, by buffering the uncompressed event data until a L1 decision is made [27].

There were several test beam periods in 2016 and 2017 at CERN, where the first CMS sensor module prototypes

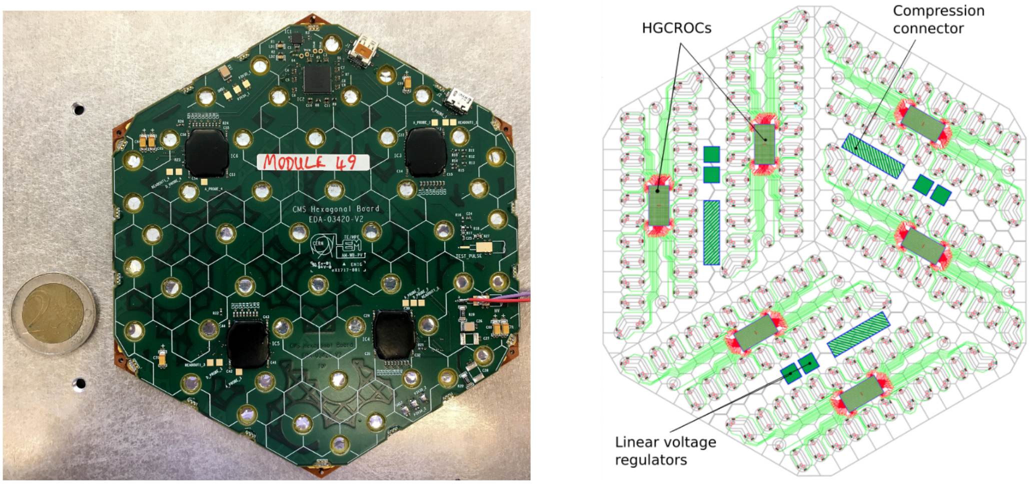


Figure 2.5: A hexagonal SM prototype used in the 2017 test beam (left) and the HGCROC schematic (right) [29, 27].

were tested, such as the one presented on Figure 2.5. Also, a prototype of the final HGCROC is built in 2020. The main goal is to mimic the CMS-like conditions and to identify the source of the noise and other factors affecting the physics performance [31].

The HGCROCs are connected to the motherboard via the compression connectors, and each motherboard serves (distributes power) to up to six modules (Figure 2.6).

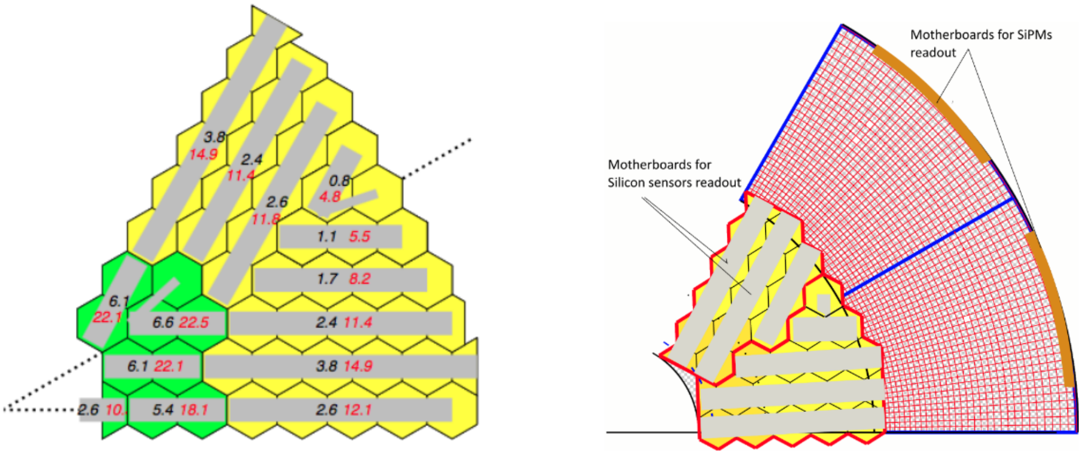


Figure 2.6: Layout of the cassette motherboards on the 8th layer of CE-E (left) and the CE-H (right). The numbers in black (red) give the average output bandwidth in GBps of the motherboard for data (trigger) [27].

The HGCROC sums are transmitted for every BX to the trigger ECON-T, which has 12 input electrical links of 1.28 Gbps (3*4=12 in the case of large-cells or 6*2=12 small-cells). The ECON received sums are used for the formation of HGCAL TPs in the BE electronics prior to which a reduction is made by selecting only a fraction of TCs. Alternatively, it is possible to select the TCs to be read out in the HGCROC itself. This would reduce the data rate out of the ASIC, but increase the complexity of the HGCROC design [27]. Finally the HGCROC has a partial module view, while ECON has the full single-module view (with 12 input links as mentioned above) and thus a more sophisticated selection is possible.

The motherboard layout (example in Figure 2.6) results from an optimization of the number of different motherboard shapes such that the whole cassette is covered, as well as the number of optical 10Gbps links required by data rates. Hence, in regions with high occupancy (close to the beam axis so lower z and lower radius), there is a larger number of links needed. We can estimate the number of ECON links needed to transfer the data. The average number of optical links per motherboard is 2-4, and the total number of on-detector links is above 5000.

2.3.2 Trigger primitive generator

The TPG algorithm is performed in the BE, and its input data are the selected TCs or module sums from the trigger ECON-T. Several working conditions are assumed [27]. First, the TPG receives data from the FE every BX (at a frequency of 40MHz), whereas each BX data will arrive up to 1.5 microseconds after the BX occurred. The main goal is to produce TPs in the form of 3D clusters reconstructed from the TC data per each endcap, and an energy map consisting of the total HGCROC energy sums. The output data needs to be delivered to the central L1 trigger within a time frame of 5 microseconds from the BX beginning, meaning that the TPG has to both process and transmit the data in 3.5 microseconds.

The baseline 3D algorithm architecture

The baseline TPG concept from [27] was based on the following. The BE is organized in two stages. The first stage FPGAs read the TCs on each endcap layer (or half of the layer), such that two-dimensional (2D) clusters are formed. The energy maps are derived based on the HGCROC sums delivered from each layer. Afterwards, the 2D clusters are linked in depth to form 3D clusters. Also, all the single-layer energy maps are combined into a total energy map. The time multiplexing concept is used in the design, and this is a mechanism used to enable that the FPGAs in the second stage of the architecture receive the 2D clusters from the full endcap, which correspond to a specific BX. For example, with the time multiplexing period $T_{mux} = 24$, there are 24 stage 2 FPGAs such that each receives and processes the data from a specific BX and 24 BXs are processed at the same time (in parallel).

A sketch of the architecture is shown on Figure 2.7. The FE electronics input assumes 3672 output links, that are divided into the readout of the electromagnetic and hadronic detector parts separately. Concerning the CE-E, there are more than 96 links available per layer, so the data from the layer will be shared between boards. For example, assuming that there is a single FPGA per board, it means that each of them reads half of a single ECAL layer. Since there are 28 FPGAs here (VU9P chips from the Xilinx Virtex UltraScale family), that is in total 28*96=2688 links that can fit to the FPGA inputs. The first 16 CE-H layers fit into one board (a single FPGA is used per layer), while the last 8 layers have only 48 links each (so two layers fit to one board). The second-stage FPGAs receive the data via 24*96=2304 links in total, where again the same FPGA family is assumed.

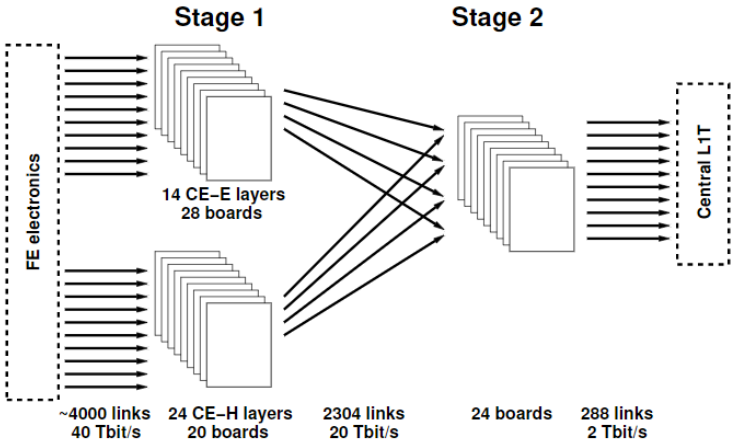


Figure 2.7: Baseline TPG hardware architecture for a single endcap [27].

The TPG scheme for the baseline requires a total of 48 FPGA boards per endcap in the first stage, and 24 boards in the second stage. Concerning the data flow between the stages, the total average rate of TC data from the FE is roughly 1MB per event, which means around 1*10^6B * 40*10^6Hz=40*10^{12} bytes per second, or 40TBps. The data rate between the stages, as well as at the interface with the central L1 trigger, results in the usage of 16Gbps links. These links are driven directly by the transceivers of the used FPGAs. It is needed to have faster links in the BE design than in the interface from the FE to the BE (where 10Gbps links are used), to have as much time as possible for the TPG algorithm and less time is used for data transfer between BE stages. Also, we could use even faster links of 25Gbps, but this would require more expensive FPGAs, so we use the less expensive solution.

After the description of the new Phase 2 CMS upgrade, the following Section 2.4 will briefly describe the simulation software and the data samples used in the trigger studies.

2.4 Simulation software and samples used in the trigger studies

The CMS Software (CMSSW) is a simulation and reconstruction framework for the analysis of data taken from the CMS detector. The main CMSSW tasks are illustrated in the Figure 2.8. An event is the result of a single readout of the detector electronics that contains signals generated by particles present in a number of BXs [32]. As shown on the Figure 2.8, the simulation starts from the collection of data from the detectors (raw data), which is processed and reconstructed to produce final data objects (reco data). CMSSW includes many packages, with the HGCAL TPG one located inside the L1TriggerL1HGCAL package. It consists of three main parts [33]:

- A geometry part defining the TCs and modules,
- A simulation part of the FE that creates trigger cells and selects a fraction of them,
- The BE clustering and reconstruction algorithms part.

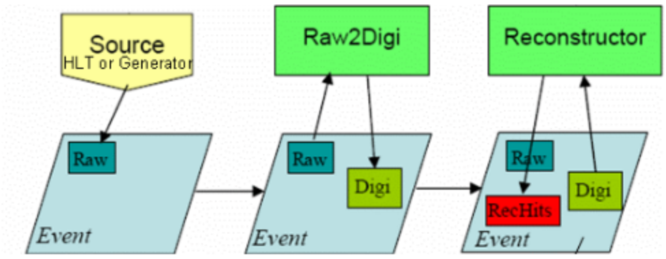


Figure 2.8: Simplified CMSSW architecture [32].

The input of the simulation is a list of particles (originating from an event generator or a simple particle gun) characterized by their momentum and origin vertex, with mother and daughter particle relationships to follow the various decay chains in the event [34]. The input data used in our studies is the output of the CMSSW L1Trigger simulation (from the HGCAL TPG only), where the (stable) event particles are propagated trough the HGCAL L1 trigger chain. They are not propagated directly to the HGCAL L1, and prior steps are the simulation of the interactions with the detector (done with Geant4), a simulation of the analogue electronics (noise, shaping, etc.) and a simulation of the digitization. The simulated data samples used in the studies are:

- photons (pT=25GeV, 35GeV, 50GeV) without PU
- photons (pT=25GeV, 35GeV, 50GeV) with PU=140 and PU=200
- electrons (pT=15GeV) without PU
- electrons (pT=15GeV) with PU=140
- electrons (pT=5GeV-100GeV) without PU
- electrons (pT=5GeV-100GeV) with PU=140
- neutrino with PU=200

We have been using events from particle guns. It is an unrealistic generation of a single particle or two back-to-back particles. The actual physics events are much more complex (even without the PU added). Such particle guns are used for the studies, because it is the best way to understand the response to single particle inside the calorimeter. When there are two back-to-back particles, this is just to simulate two events inside a single event, as they are sent in opposite directions so not to influence one another. For instance, Figure 2.9 shows random events with two photons of 25GeV without PU. This is the distribution of hits or energy deposits in HGCAL in the (x, y) plane, where both endcaps and all layers are summed. We can clearly identify the two photon-induced showers.

As said above, the PU=0 sample is unrealistic and artificial, and the same is for the sample with the PU=140 or PU=200. The latter can be considered "more realistic", to study the effects with the PU included. For instance, the PU200 sample includes on average 200 additional low-energy interactions on top of the high energy one.

The signal sample used in the studies consists of events containing EM showers initiated by electrons or photons of fixed transverse momentum pT=25, 35, 50 GeV. In most samples a constant pT is used (it is the energy projection on the transverse plane, refer to Section 1.2.2), but we have also used a flat pT sample, with several particle showers contained in each event. The flat pT sample is also an unrealistic event simulation, and we use it to produce a distribution of pT values. This is important for the physics analysis, as it can be re-weighted and one can simulate the pT distribution that is specific to a physics process. The flat pT was used for the machine learning (ML) study in Chapter 6. Namely, in order to provide more general database of the training images, we have trained on the flat pT electrons from the GeV in range [5, 100].

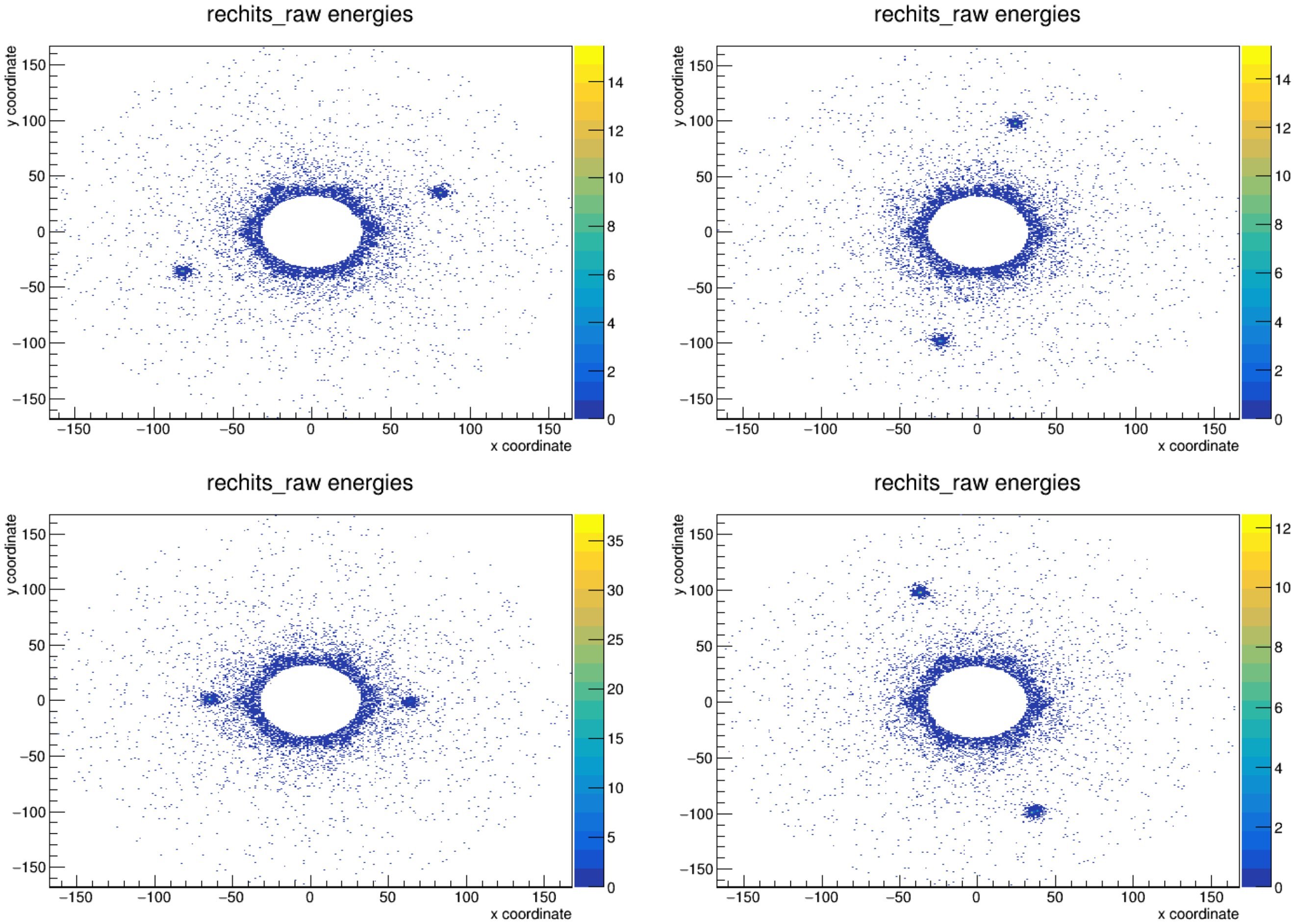


Figure 2.9: CMSSW simulated event data (photons 25GeV, PU=0) projected in (x, y) coordinate system. Hence, these events show the simulation of the two back-to-back photons of 25 GeV.

Finally, we have also used a neutrino sample to simulate the PU only background, since the neutrino particle does not initiate any showers in HGCAL, so that the deposits coming from the particles in the sample can be used as "additional" minimum bias collisions to be superimposed to the primary event. This sample is again used in the ML study, for the generation of the background images in the training data set.

Chapter 3

HGCAL detector geometry studies

The CMS HGCAL detector design upgrade in the new phase of the LHC is already given in Chapter 2 and the application of a hexagonal geometry in the new detector end-caps is described. As it is already well-known, hexagons are applied in many different fields due to the many advantages they offer compared to other geometrical shapes. One of the most important things is the possibility to be closely packed and to form a hexagonal grid that fully covers the Region of Interest (ROI) without overlaps or gaps. This concept is called mathematical tessellation and the purpose of our research presented in this chapter is to explore how it can be efficiently applied in HGCAL. The main goal is to obtain the possible models used to efficiently pack sensors in the SM approximated by a hexagon. There is another ROI type examined in this context, and that is how to efficiently cover the circular detector sensing layer. To summarize, hexagonal SC must be efficiently embedded into the hexagonal SM as a first step, after which SMs with inner packed items must be efficiently packed into the circular ROI.

In Section 3.1, we provide a short literature review summarizing the existing state-of-the-art on this topic and offering some application-specific advantages. We are interested in the packing efficiency (PE) of embedding as much sensors as possible in the container ROI area. We report on existing researches deriving formulas to calculate the number of embedded inner hexagonal cells or their vertices and/or edges. In case only the number of edges or vertices is provided for the targeted application in the literature, we derive formulas for calculating the number of inner hexagons. Next, in Section 3.2, studies are provided on the SM design, approximating the SM with both regular and irregular hexagons. General architectures are derived as well as visualization of the detector sensor plane. Also, the production cost is evaluated in terms of silicon efficiency (SE) when the aforesaid SM types are produced. Section 3.3 describes data reduction based on the hexagonal geometry, but in the context of the FE detector design from Chapter 2. The concept of TC is explained, and the analysis is done on how to efficiently pack them in the hexagonal SM.

3.1 Survey on embedding hexagons in circular and hexagonal ROI

This section is formulated based on the published paper [35], where we address a specific problem of embedding hexagonal cells in a selected ROI. The goal of this section is to provide a short review on the state-of-the-art studies where the ROI is approximated with a circular or a hexagonal shape. The application-driven guidelines are expressed in the context of the CMS HGCAL upgraded end-cap design, modelled such that hexagonal SCs are packed in the hexagonal SMs, after which these are packed in the circular ROI in an optimal way (Figure 3.1). The research questions (RQs) answered in this section are:

- RQ1: Which applications can be found in the literature on embedding the hexagonal cells in a circular or hexagonal ROI?
- RQ2: Are there formulas in the literature that calculate the number of ROI-embedded hexagonal cells? If not, can they be derived based on the number of inner hexagon's vertices and edges?
- RQ3: Which of the existing models can be applied for the HGCAL design?

First, we classify the papers from the literature in two main classes depending on the hexagonal or circular ROI used. Next, sub-classes are derived based on the criterion that formulas are provided in the paper, whether for the calculation of the total number of hexagonal cells embedded in the ROI, or the number of the corresponding inner hexagon vertices and edges. In case that the number of inner hexagonal cells is not provided in the referent papers, we derive formulas for the total number of hexagons embedded in the ROI. Our main intention with this overview section was to conclude whether it is possible to adjust or adapt some of the existing solutions and use it as a potential HGCAL detector model. Based on this analysis of prior work, the motivation for our geometry studies was further enforced, and it encouraged us to contribute by developing a general framework of HGCAL architectures.

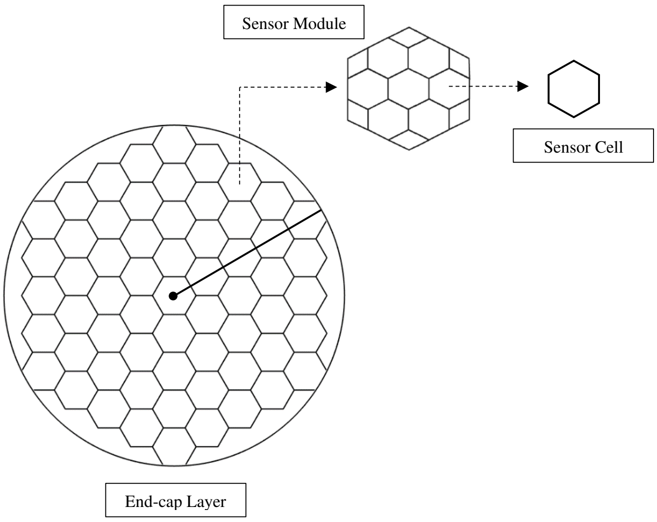


Figure 3.1: Simplified structure of the CMS HGCAL upgrade [35]. The circular region is a single HGCAL layer covered with the hexagonal modules, where each module consists of the hexagonal cells.

Table 3.1: Classification of papers.

	Provided formulas #hexagonal cells	 #vertices	 #edges	Total
Circular ROI	[36, 37]	-	-	2
Hexagonal ROI	[38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48]	[40, 49, 41, 42, 43, 50, 51, 52, 53, 54]	[40, 49, 42, 43, 50, 53, 54]	17
Total # papers				19

In [36, 37], hexagonal sensors are packed inside a circular ROI, and the efficiency is expressed as the total number of inner packed items in a fixed radius R. Davis and Sinha [36] apply regular hexagon tessellation to arrange rings of polygons around the central cell. Authors show that a hexagonal grid is better than a square, as it enables more sensors to be placed on the ROI of constant radius. Kim et al [37] also deploy hexagons on a targeted region and show that for a circular ROI of size R and hexagon cells of side a, one can calculate the number of hexagon rings k needed to fully cover the ROI. The formula to calculate the number of embedded hexagonal cells is the following:

$$R = \begin{cases} \frac{(3k+1)a}{2}, & \text{if k is odd} \\ \frac{\sqrt{(3k+1)^2+3}}{2}, & \text{if k is even} \end{cases} \tag{3.1}$$

This model can be applied for calculating how many rings of SMs of a fixed size fit to the detector ROI layer, or how many rings of SMs are needed to cover a predefined ROI. For example, having a ROI of radius $R = 20cm$ and given the cell size $a = 10cm$, the number of hexagon rings is $k = 1$. Also, if the number of hexagons is known in advance, for example $k = 2$ and $a = 10cm$, a radius of $R \approx 3.6a$ can be covered. A small correction of the result with respect to integer values can enable using a larger number of full hexagons (Figure 3.2).

The basis for mathematical tessellation of hexagons is the work of Stojmenovic [40], and it can be applied in cases where the ROI is approximated by a hexagon. The example is given on Figure 3.2, where the ROI radius t is a tessellation factor defined as the number of hexagon rings between the ROI center and the ROI border. The total number of vertices and edges inside the ROI is calculated as $6t^2$ and $9t^2 - 3t$ respectively. The total number of

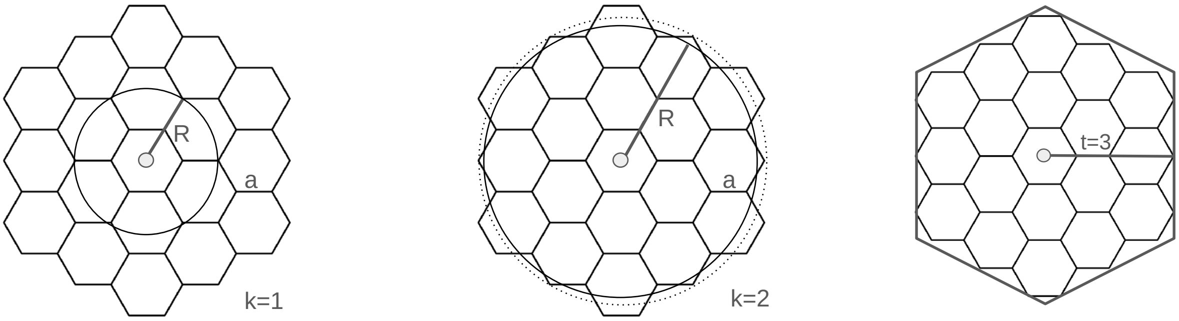


Figure 3.2: Calculating the number of hexagons in a circular and hexagonal regions.

hexagons inside a ROI of size t is:

$$N_{hexagons} = 3t^2 - 3t + 1 \tag{3.2}$$

A similar concept is applied in [41, 42, 50]. On the other hand, while the Formula 3.2 provides the number of hexagons in a certain number of hexagonal rings, the model proposed by Chen et al. [49] (Figure 3.3) counts the total number of the inner hexagons nodes and edges as $3t^2 - 3t + 1$ and $9t^2 - 15t + 6$ respectively. The authors did not provide formulas for calculating the total number of embedded hexagonal cells. Therefore, we derive the formula based on the total edges/vertices count:

$$N_{hexagons} = 3t^2 - 9t + 7 \tag{3.3}$$

The concept of overlapping inner hexagons is introduced with this model, and it is valid in Equation 3.3. However, since the practical application of packing SCs in SMs does not allow any overlapping, the formula is adjusted as follows:

$$N_{hexagons} = \begin{cases} 3k^2 - 3k + 1, & \text{if } r = 0 \\ 3k^2 - k, & \text{if } r = 1 \end{cases} \tag{3.4}$$

where $t = 2k + r, k \in N_0, r \in 0, 1.$

A mathematical model for embedding hexagons into hexagon is derived in [51], where the authors denote the

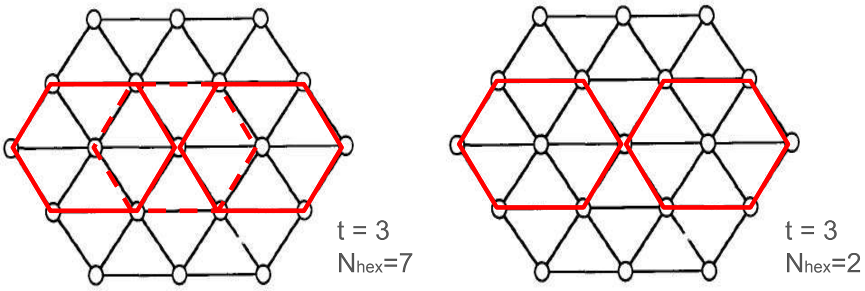


Figure 3.3: Hexagons inside a hexagonal ROI with (left) and without overlapping (right). Adjusted from [49].

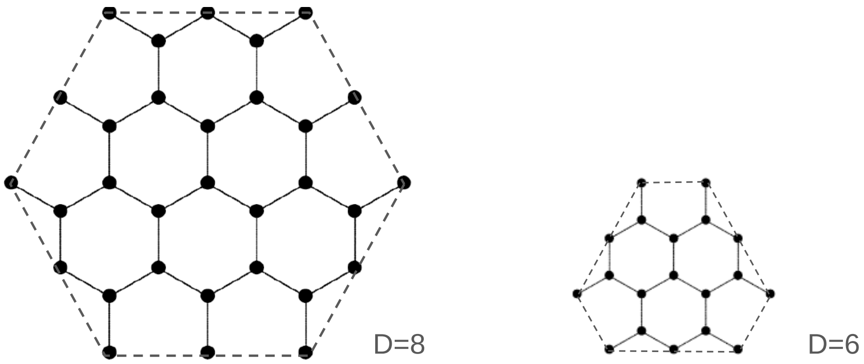


Figure 3.4: Hexagonal grid embedded inside a hexagonal ROI. Adjusted from [51].

ROI size as a diameter D, expressed as the total number of edges on the circumscribed circle diameter of the ROI hexagon. We added the border around the resulting grid on Figure 3.4 to visualize the ROI shape for $D = 8$ and $D = 6$. The ROI size D is defined as $D = 4k + r, k \in N_0, r \in 0, 2$, where the ROI is a regular hexagon for $D = 4k$, and it is an irregular hexagon for $D = 4k + 2$. Authors in [51] do not provide the total number of hexagons embedded inside the ROI, so we derive them with the following formulas:

$$N_{hexagons} = \begin{cases} 3k^2, & \text{if } r = 0 \\ 3k^2 + 3k, & \text{if } r = 2 \end{cases} \tag{3.5}$$

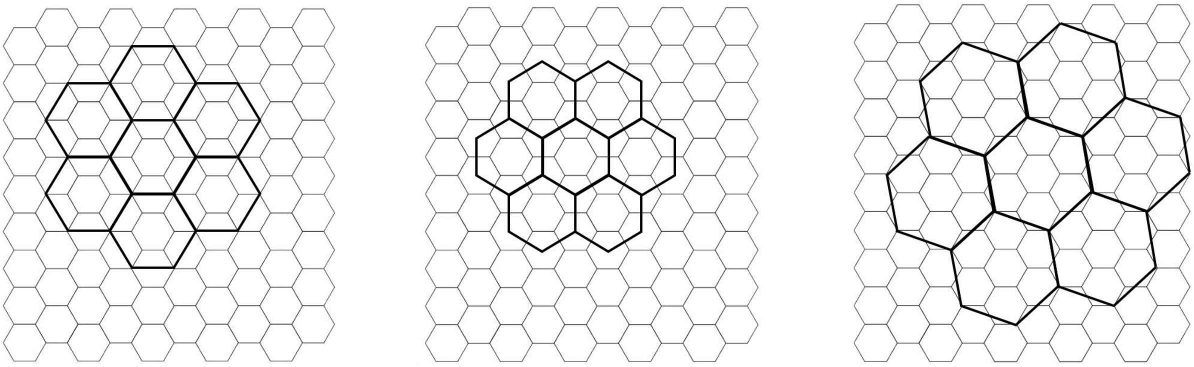


Figure 3.5: Subdivision of the center-sharing ROIs. Fraction 1/4 (left, middle) and 1/7 (right). Adjusted from [39, 48].

Previous models are meaningful because of their potential to be used as guidelines for the new HGCAL module design. However, even though the number of inner packed sensors is calculated, there are no partial cells identified at the ROI border, which is inevitable for the architectures to be efficient in packing. Namely, it is shown by Sahr et al. [47] that a large hexagon cannot be composed only of full hexagons, as there is always a combination of hexagonal cells in the middle of the ROI and the non-hexagonal cells at the ROI border. Authors in [39] show that each inner hexagon cell has a fraction $1/N$ of the ROI area. Examples of 1/4 and 1/7 subdivisions are given in Figure 3.5, and they are referred to as center-sharing, when the central inner cell shares its center with the ROI hexagon in which the cells are packed. The vertex-sharing variant is given by [48] and it is obtained when the central cell vertex is at the ROI center. Also, as visualized on the example from Figure 3.6, a single hexagonal ROI can be subdivided by using 9 hexagons (7 full and 6 border hexagon thirds), or by using 16 hexagons (13 full hexagons and 6 border hexagon halves).

All these visualizations can be used for modelling the HGCAL detector sensing layer, when the SMs with the inner packed SCs are tessellated on the higher-level ROI. The idea of hexagonal ROI SM subdivision is extended in [46]. Authors calculate the total number of inner hexagons as a function of the ROI size and prove that, for subdivision R/n, the total number of equal hexagonal cells is n^2.

3.1.1 The differences between the state-of-the-art models

The basic difference between the presented models is whether they use a circular or a hexagonal ROI to describe the container in which the small hexagons are packed. Hence, we have presented the usage of both the circular and hexagonal ROIs. In the first case, it is shown how to calculate the number of the inner packed hexagons of the fixed size in a fixed-size circular container. Also, we can use the provided formulas from the literature to estimate the circular region that can be covered with the hexagons of a fixed size.

Concerning the presented models using the hexagonal ROI, they differ depending on the way they calculate the number of the inner packed hexagons. For example, the model on Figure 3.2 supposes that the ROI always contains the certain number of the hexagonal rings, and do not consider that the ROI border can "cross" the cells in the ring. This way, they have just allowed the gaps at the very border of the ROI, without identifying the broken (triangular) parts. On the contrary, the model from Figure 3.3 does not require that the inner hexagons are aligned in rings, but they allow us to calculate the inner hexagons number in both cases, with and without using the hexagonal ring scheme (depending on the desired ROI size).

The models on Figure 3.4 show that a grid of the tessellated hexagons inside the ROI does not need to be a regular ring of hexagons like in the former cases. The grid definition can be such to provide a larger number of the inner packed items, and without the central small hexagon positioned in the ROI center. What is emphasized in this case is that, unlike before, the ROI border "cuts through" ("crosses") the full hexagons. However, authors do not consider the border compromises.

3.1.2 Discussion and evaluation

We define several criteria to examine the possible implementation of the state-of-the-art findings in our targeted application. First, all works provide the potential models on how to design a specific structural scheme that we are interested in for the HGCAL SM geometry. However, these models are not classified in the literature and the models and formulas are provided in an ad-hoc manner, depending on the targeted application. Some of them calculate the number of inner hexagons inside a larger hexagonal region, while others concentrate on the inner hexagons

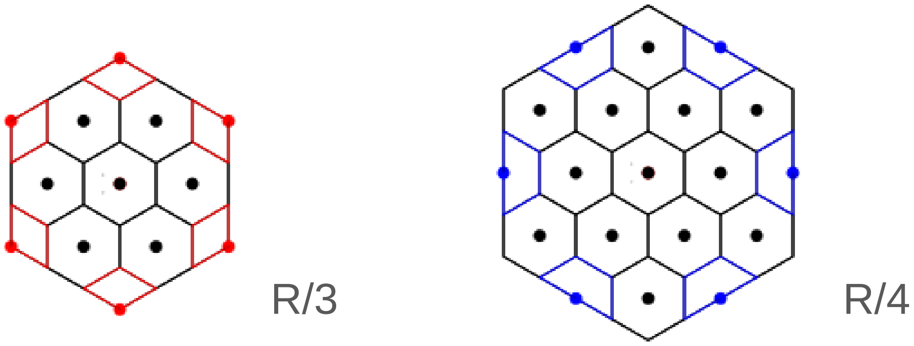


Figure 3.6: Subdivision of the hexagonal ROI based on its size. Adjusted from [46].

vertices or edges [40, 41, 42, 50, 53]. We have shown that even when only these formulas are provided, one can easily derive the number of inner the inner packed full cells.

In this context, without the direct border cells identification, the models from Figure 3.4 are important, and we follow them in our study on the module design (Section 3.2). These models indirectly provide insight to the broken cells at the ROI (module) border, and also give the possibility for the module to be an irregular symmetric hexagon.

The state-of-the-art models that are also important for our study on the module design are the ones in [43, 44, 45, 54], since they provide a possible theoretical model for the hexagonal structure, indirectly considering that there should be partial cells. These are clearly visible at the ROI border, but they are not identified or quantified.

There are only few papers in the literature that directly tackle the issue of the broken cells and indicate that there are hexagon parts which appear on the hexagonal ROI border during the packing procedure. For example, authors in [38, 48, 47, 39, 46] decompose a large hexagon into smaller ones with the scaling schemes that completely rely on partial hexagons identification.

In many cases, ROIs containing the packed objects in the literature are tessellated to form several variants of multi-resolution hexagonal grids [54, 55, 56]. This is important, because it enables us to follow the similar principle when covering the circular HGCAL area with the designed hexagonal modules. Considering the application in the potential HGCAL module design, authors calculate only the subdivision fraction when the module is decomposed into smaller hexagons (this can be defined as a total equivalent cell number, or the number of cells that have the equal area as a full hexagon). However, authors do not give a generalized approach to the cells quantification which would identify and quantify the hexagonal parts such as halves of thirds at the module borders.

Certainly, there is a need for further research on detector sensing layer geometry in addition to the summarized review findings. Namely, the total number of SCs inside a SM should be calculated with the objective to estimate the overall SM production cost. Therefore, in Section 3.2, we derive a framework of architectures that can be used for the hexagonal SM design. Also, the total sensor production cost is evaluated for each architecture. Hence, Section 3.2 provides solutions for solving the design problems and identifies the various sensor shapes that need to be produced for placing at the SM border. At the very end of the Chapter 3, we describe how our HGCAL geometry studies inspired other people and served as a good starting point for finding a final solution for the new HGCAL sensor module design.

3.2 HGCAL detector sensor module design and sensing layer model

In Section 3.2.1, we explain the mechanical constraints for the HGCAL design layout, originated from the CMS technical proposal [57] in 2015. Next, in Section 3.2.2 a classification framework of hexagonal architectures is derived for the HGCAL module, together with the visualization of the detector sensing layer model. In Sections 3.2.3 and 3.2.4, we present the design of the SM as a regular hexagon. Next, in Section 3.2.5, we examine the possibility to form a SM from an irregular hexagon. The Section 3.2.6 provides the comparison between the presented module architectures. Finally, in Section 3.2.7, the production cost of an irregular hexagonal SM is evaluated.

3.2.1 Problem formulation and mechanical constraints

The research problem in this section arises from the mechanical constraints given by the CMS technical proposal for the HGCAL design upgrade [57]. Basically, detector end-caps are circular structures using silicon material as the active medium and having several sampling layers. The simplified structure of the active layer is presented on Figure 3.1.

There are several technical details foreseen by the proposal. First, a hexagonal SM should be designed. The number of full SCs inside the module should be maximized, with the reduced number of different SC types. Also, SCs should possibly be of equal size and area and their size must correspond to the physics and electronics needs in terms of granularity and cell capacitance. The technical proposal foresees using cells of 1.05 cm^2 area in the low η and 0.53 cm^2 cells in the high η region, fabricated on six-inch or eight-inch sensor wafer production lines. The initial proposal for the number of cells packed in the module is 128 and 256 for large and small cells respectively.

Possibly, partial and odd-shaped border cells should be avoided, because they increase the number of module channels and complicate data readout at the module border. However, if they are inevitable for a certain architecture solution, the number of each border cell types should be calculated because these sensors must be produced separately. Also, the SC plane must remain uniform and non-distorted, with SCs keeping the initial positions defined by the hexagonal grid. All SC types for a specific SM architecture should be quantified in order to evaluate the total sensor production cost. The proposal declares that there are approximately 22 000 sensor wafers to be produced.

There is another requirement for the module design, i.e. the grouping of the inner SCs into TCs. The module architecture should allow the readout cells to be grouped together in clusters of four, forming in total 32 or 64 groups or TCs inside the SM with 128-channels or 256-channels respectively. Also, the TC plane should remain uniform and non-distorted in order to simplify the nearest-neighbor (NN) finder algorithm when TC clustering is performed in the TPG algorithm. Finally, a circular detector end-cap layer is built by covering the sensing region with the produced hexagonal SMs. There should be triangular cuts on the wafer vertices providing mechanical apertures used for fixating the SMs. In addition, $30°$ or $60°$ cuts are predicted in the technical proposal, to construct a mechanical cassette from a cut wedge sector. The engineering details of the new HGCAL design concept can be revisited in Chapter 2.

3.2.2 Study on general hexagonal architectures

We classify papers from the state-of-the-art in Section 3.1 and use the mechanical constraints in HGCAL design as guidelines to evaluate the architectures. The technical proposal foresees using a hexagonal module, but it does not specify if it should be a regular or an irregular hexagon so we explore both directions. The classification of papers is given in Table 3.2. Almost every research approximates the ROI container by a regular hexagon and an exception is the work in [51] where an irregular hexagon is considered. Hence, we separate hexagonal architectures in two basic classes, depending on whether a designed SM is regular or irregular hexagon. Adopted from the terminology in [43], regular architectures are considered as aligned (with a vertex at the top) and rotated (with an edge at the top). Aligned architectures have the property that inner SC and SM orientations coincide. Rotated architectures have the property that the SM hexagon is rotated by $30°$ compared to the SC orientation.

Table 3.2: Classification of architectures based on state-of-the-art.

Centred		Non-centred	
Aligned	Rotated	Vertex-sharing	Edge-based
[38, 47, 39, 46, 43, 54, 55, 56]	[50, 42, 46, 39, 43, 44, 54, 56]	[45, 48, 51, 55]	[54]

As a result of the classification presented in Table 3.2, a general framework of hexagonal architectures is developed (Figure 3.7). We derive formulas for calculating the total number of different SC types which arise from the compromise on the SM border during the packing procedure. We tessellate the SMs on to a circular ROI and visualize the detector sensing layer with each architecture. Also, we analyze the cuts provided by forming the $30°$ sectors and evaluate the total number of SM types that need to be produced.

Architectures are referred to as centered if the central inner SC overlaps with the centre of the SM container [48]. Otherwise, the container is moved in the hexagonal grid so that the SM center is at a SC vertex (vertex-sharing architecture). We name these non-centered or moved architecture, and only movement in the up direction in the SC plane is considered because architecture moved down is the same as moved up but flipped vertically. Hence,

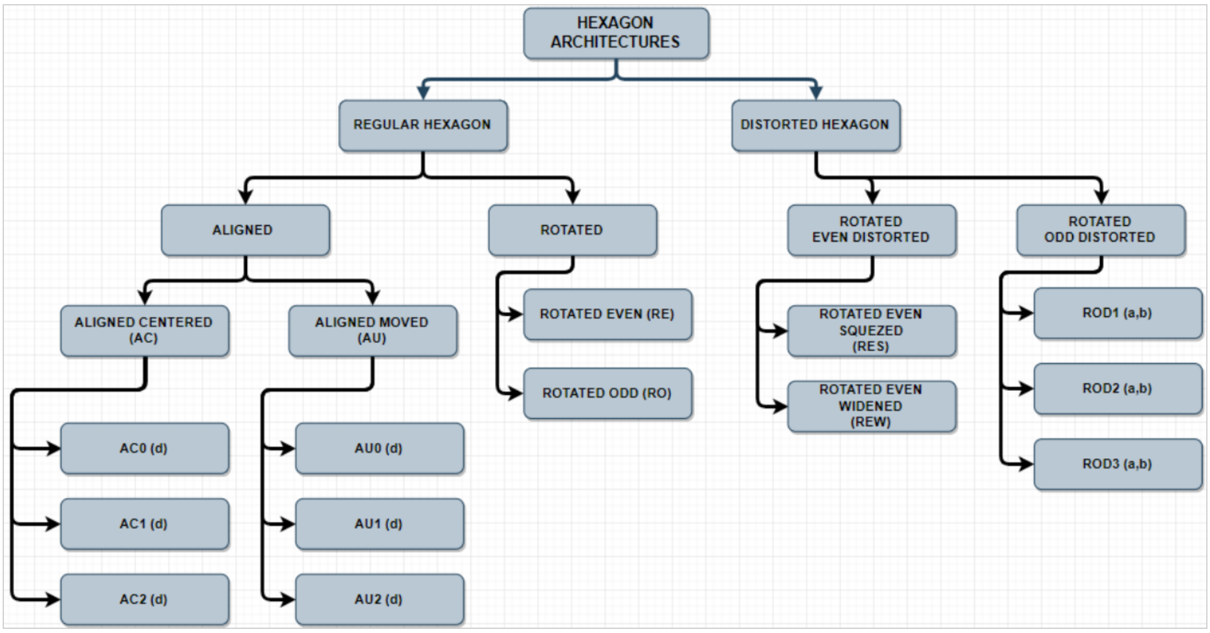


Figure 3.7: Derived classification of architectures.

we refer to aligned architectures as Aligned Centered (AC) and Aligned Up (AU). Each of them is further divided in three sub-classes based on the size of the inscribed circle diameter or apothem of the hexagonal container d which is expressed as the number of full hexagonal SCs.

The rotated class of the architectures is divided in two sub-classes: architectures with and without the central SC, depending on whether the SM center is at a SC center (centered architecture) or edge (edge-based architecture). We name these Rotated Odd (RO) and Rotated Even (RE) architectures respectively. They are identified based on the property of the SM size expressed as the number of full SCs on the circumscribed circle diameter. Besides, RO-small has partial SCs at the module borders, while in RO-large these are omitted completely.

The common property of the aligned and RE architectures is that, with them, border partial cells cannot be avoided inside the SM. On the other hand, the RO-large architecture has the nice property of having no partial cells at all, while cells at the border have the same area as a full hexagon. The motivation for architectures where the SM is an irregular hexagon comes from the property of having no partial cells. Hence, two new sub-classes are derived from the RE architecture, called Rotated Even Squeezed (RES) and Rotated Even Widened (REW). Also, three new sub-classes are defined from the RO architecture and divided by the difference in the hexagon edges $|a - b|$. We named them as Rotated Odd Distorted (ROD1, ROD2 and ROD3).

3.2.3 Using aligned architectures to design SM from a regular hexagon

The class of aligned architectures is described with the size of the inscribed circle diameter $d = 3k + r, k \in N, r \in 0, 1, 2$ expressed as the number of full hexagons where k is a parameter to generate d. In this layout there are in total 3 types of SCs: full hexagons (FH) in the inner part and two types of partial sensor cells at the module border, which we identify as rhomboid hexagonal thirds (RHT) and vertical hexagon halves (VHH), generated by cutting a SC hexagon from vertex to vertex. Border cell definition is visualized on Figure 3.8.

This class of architectures is subdivided based on the property of having a central SC. Hence, we define aligned centered (AC) and aligned moved up (AU) which are non-centered architectures. Each sub-class is further divided based on the property of the inscribed circle diameter being a multiple of the number 3, as shown on Figure 3.9.

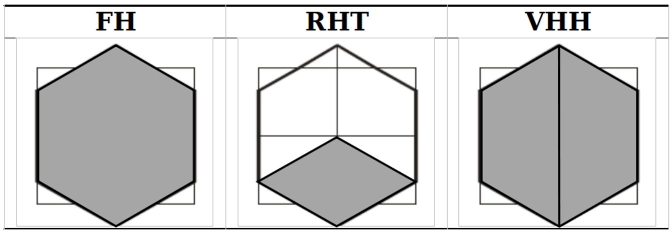


Figure 3.8: Definition of SCs in the aligned hexagon module.

Aligned-centered architecture

Three subclasses of AC architectures ($AC_0(d)$, $AC_1(d)$ and $AC_2(d)$) have partial SCs at the module borders and a specific number of inner partial and full hexagon cells. Hence, architecture type $AC_0(d)$ has six RHT cells at the module vertices and VHH cells at the borders, while architectures $AC_1(d)$ and $AC_2(d)$ have VHH cells only. This is illustrated on Figure 3.10.

We derive the number of full hexagons packed inside the aligned architecture $AC(d)$ of size d, where $d =$

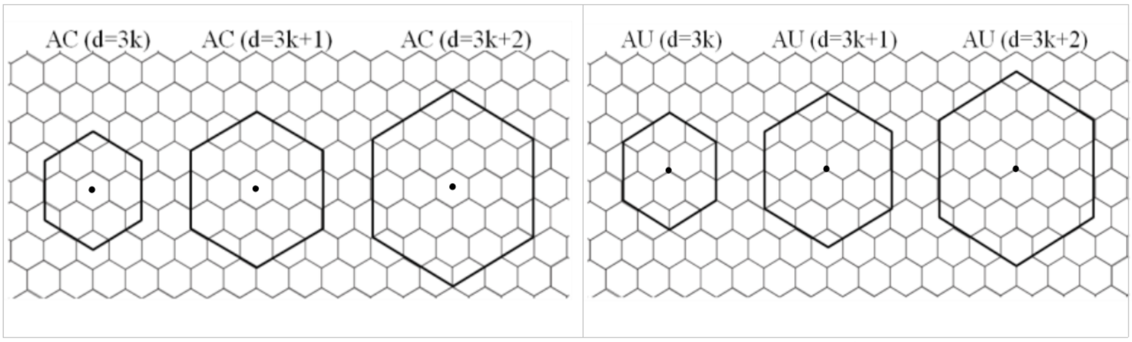


Figure 3.9: AC and AU architectures subdivision.

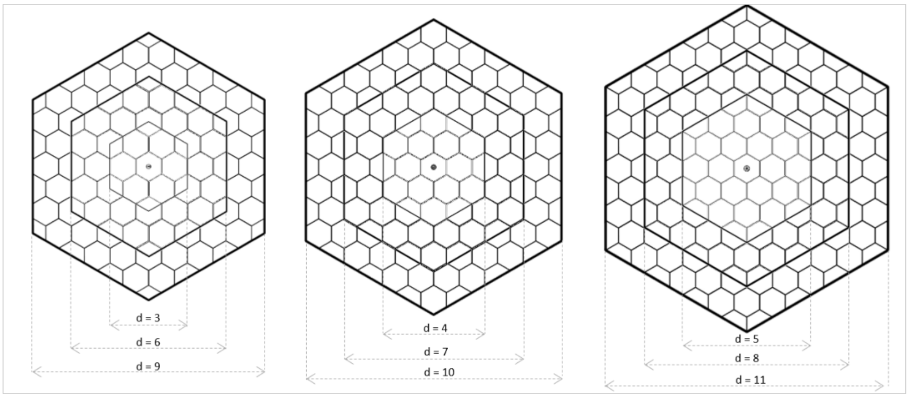


Figure 3.10: AC architectures for k = 1, 2, 3.

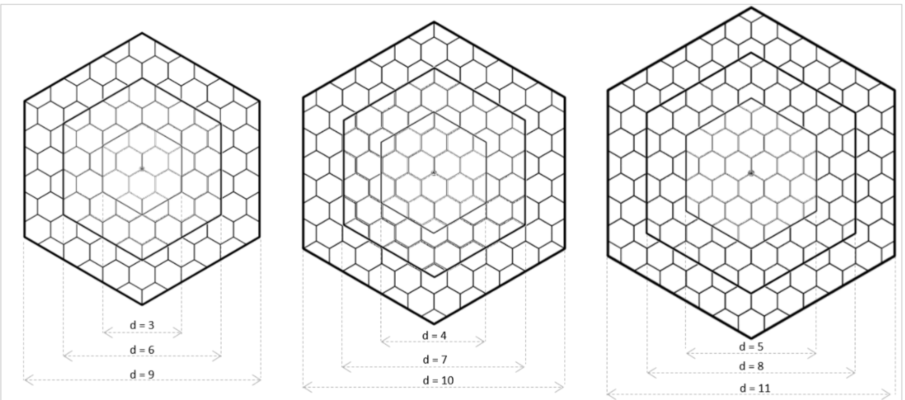


Figure 3.11: AU architectures for k = 1, 2, 3.

$3k + r, k \in N, r \in 0, 1, 2$:

$$N_{FH} = \begin{cases} d^2 - d + 1, & \text{if } r = 0 \text{ and } r = 1 \\ d^2 - d - 1, & \text{if } r = 2 \end{cases} \tag{3.6}$$

The number of hexagon halves is defined as:

$$N_{VHH} = \begin{cases} 2(d - 3), & \text{if } r = 0 \\ 2(d - 1), & \text{if } r = 1 \\ 2(d + 1), & \text{if } r = 2 \end{cases} \tag{3.7}$$

The number of hexagon thirds is defined as:

$$N_{HT} = \begin{cases} 6, & \text{if } r = 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.8}$$

Vertex-sharing architecture: Aligned non-centered

Moved architectures arise by moving the AC module up in the sensor plane such that one SC vertex overlaps with the SM center. Three sub-classes of aligned non-centered or aligned up architectures ($AU_0(d)$, $AU_1(d)$ and $AU_2(d)$) have partial SCs at the module vertices and a specific number of partial and full hexagon cells (Figure 3.11). Hence, $AU_0(d)$ has only VHH cells at the borders, while architectures $AU_1(d)$ and $AU_2(d)$ have VHH cells as well as three RHT cells at the opposite vertices.

We derive the number of full hexagons packed inside the aligned architecture $AU(d)$ of size d, where $d = 3k + r, k \in N, r \in 0, 1, 2$:

$$N_{FH} = \begin{cases} d^2 - d, & \text{if } r = 0 \text{ and } r = 1 \\ d^2 - d + 1, & \text{if } r = 2 \end{cases} \tag{3.9}$$

The number of hexagon halves is defined as:

$$N_{VHH} = \begin{cases} 2d, & \text{if } r = 0 \\ 2d - 2, & \text{if } r = 1 \\ 2d - 4, & \text{if } r = 2 \end{cases} \tag{3.10}$$

The number of hexagon thirds is defined as:

$$N_{HT} = \begin{cases} 0, & \text{if } r = 0 \\ 3, & \text{otherwise} \end{cases} \tag{3.11}$$

The number of equivalent cells for both AC and AU architectures is verified to be $N_{eq} = n^2$ [46].

3.2.4 Using rotated architectures to design a SM from a regular hexagon

Rotated architectures arise from rotating the aligned module by $30°$. The basic idea is to avoid partial cells at the hexagonal module border. Hence, two sub-classes are defined based on an even or odd number of full hexagonal cells at the module's circumscribed circle diameter D, as illustrated on Figure 3.12. Partial SCs exist at the SM border of the RE and RO-small models, and they are completely omitted in the RO-large.

Edge-based architecture: Rotated Even

This architecture has a symmetric topology in which partial cells are present only at the module top and bottom edges. To obtain a straight SM cut, edge hexagons are somewhat distorted. Bottom and top edges contain a new type of sensor halves which we call horizontal hexagon half cells (HHH) generated by cutting the SC from edge to edge. Also, there are always four extended half cells at each of the top and bottom vertices. We refer to them as left and right horizontal hexagon halves (LHHH and RHHH) respectively. There are no half cells at the remaining

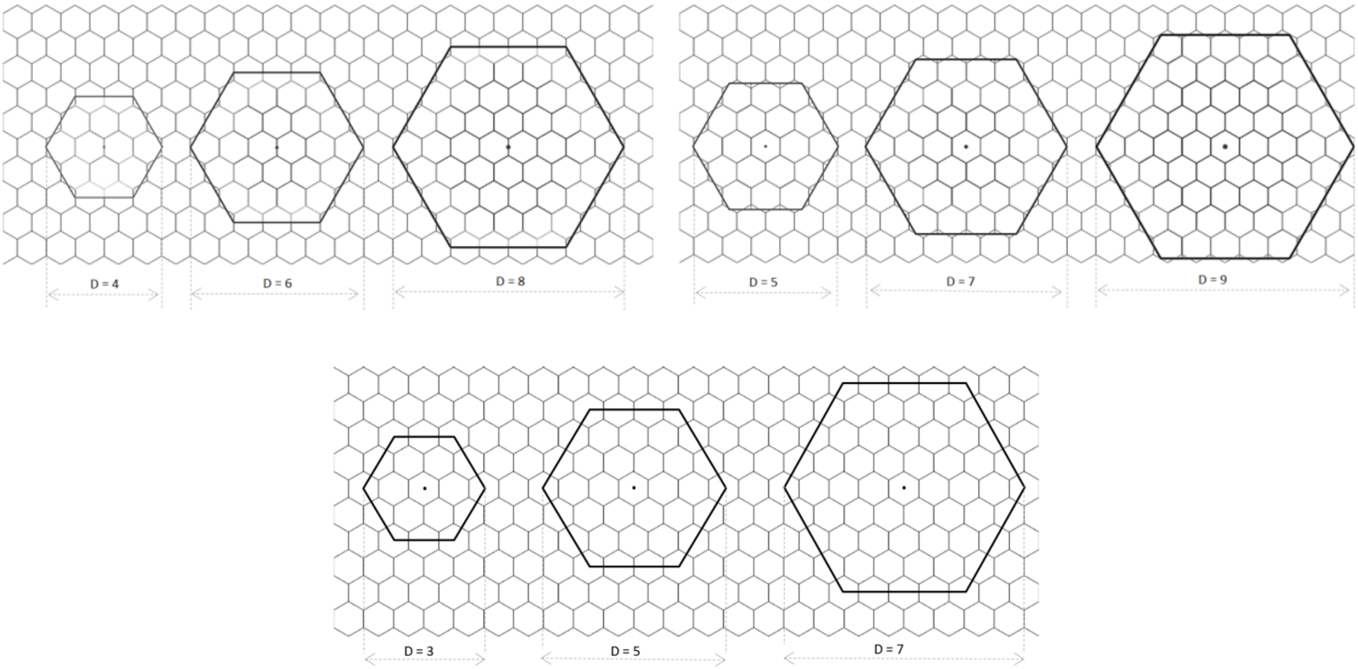


Figure 3.12: Rotated architectures. RE (D=2k), RO-large (D=2k+1) (up) and RO-small (D=2k+1) (down), k=1,2,3.

module borders, but there are distorted Full Hexagon (FH), Edge Pentagons (EP) and Corner Pentagons (CP), as listed in Table 3.13. They are the same in area as FH cells, so they can be included in the total FH or equal or equivalent cells (EQ) count. The number of equal cells is the total number of cells that have the same area as the regular hexagon (FH cell area). We derive a formula for the total number of equal cells in architecture $R(D)$, where $D = 2k, k \in N$:

$$N_{EQ} = \frac{3D^2 - 2D}{4} \tag{3.12}$$

The numbers of individual cells (CP, EP, FH) are:

$$N_{CP} = 2; N_{EP} = 2D - 4; N_{FH} = \frac{3D^2 - 10D + 8}{4} \tag{3.13}$$

The numbers of different types of horizontal hexagon halves are:

$$N_{HHH} = D - 4; N_{LHHH} = N_{RHHH} = 2 \tag{3.14}$$

Centered architecture: Rotated Odd

Unlike the aligned and rotated even architectures, in the rotated odd layout, partial cells are present only in RO-small, and we refer to them as vertical hexagon halves (VHH) and pentagonal hexagon thirds (PHT). On the other hand, in RO-large, they are completely omitted at the module border. All inner SCs are FHs and non-hexagonal EP and CP, as shown in Table 3.14. This type of architecture has a good property of all cells being the same area. We derive formula for FH, CP and EP calculation in the architecture $R(D)$, where $D = 2k + 1, k \in N$:

$$N_{EQ} = \frac{3D^2 + 1}{4} \tag{3.15}$$

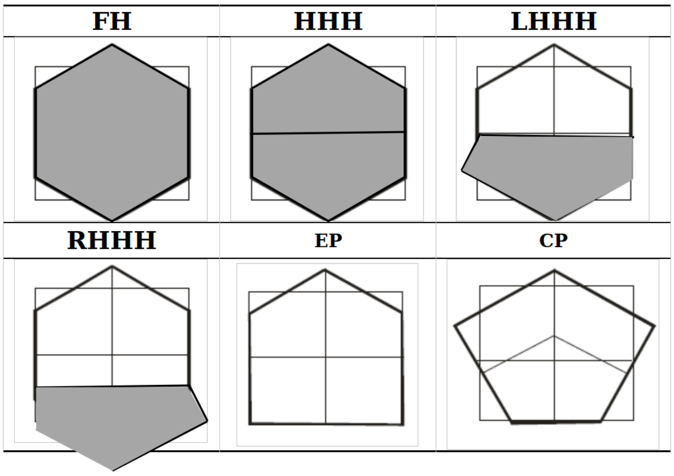


Figure 3.13: Definition of SCs in the rotated even module.

The numbers of individual cells (CP, EP, FH) are:

$$N_{CP} = 6; N_{EP} = 3D - 9; N_{FH} = \frac{3D^2 - 12D + 13}{4} \tag{3.16}$$

Since in the RO-small architecture there are still partial SCs left at the module border, we calculate the total number of each cell type as follows. The number of full cells (FH) and partial cells (PHT) in RO-small is equal to the number of N_{EQ} and N_{CP} in RO-large respectively. Also, the total number of VHH cells is:

$$N_{VHH} = 3D - 3 \tag{3.17}$$

3.2.5 SM designed from an irregular hexagon

This class of architectures is derived to maximize the total number of FHs inside a module and to avoid having different types of partial cells at the module boundaries. Thus, the upper and bottom edges of the RE architecture are extended upwards and downwards, creating a distorted hexagonal module of different edge sizes. The new architecture has modified edges with respect to the original one. Following a similar logic, edges of the RO architecture can be modified as well, providing a larger number of cells in a module. We refer to these new architectures as rotated distorted and they are defined with referent edges a and b and step s. The longer edge equals $b = a + s$ and a non-distorted or regular hexagon would be provided if the step $s = 0$.

There are two types of these architectures and we refer to them as irregular or distorted. Distorted even ones have top and bottom edges adjusted to contain only full cells, basically EP and CP at the edges and preserve orthogonal symmetry. If the parameter s is set to $s = 1$ the RES and REW architectures are obtained. As given on Figure 3.15, top and bottom edges are adjusted to contain full SCs and the orthogonal module symmetry is

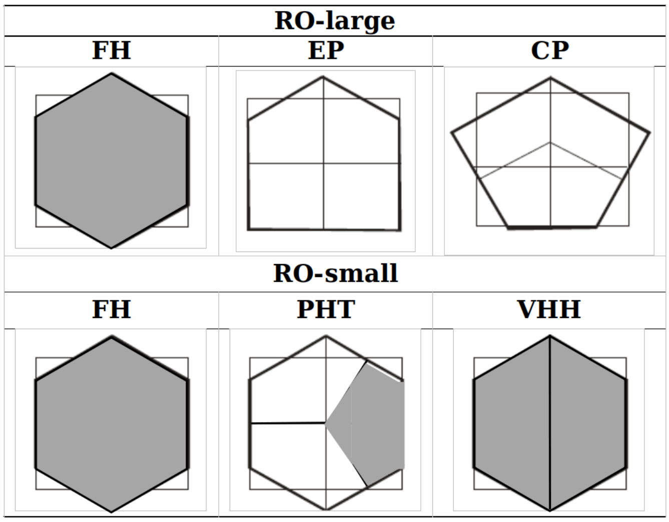


Figure 3.14: Definition of SCs in the rotated odd module.

preserved. On the other hand, rotated odd distorted architectures, have every other edge modified thus preserving radial symmetry. Selected sizes of parameter $s = 1, 2, 3$ are analyzed, which resulted into three sub-classes of architectures; ROD1, ROD2 and ROD3. Examples are given on Figure 3.16. Since all these architectures are defined based on their size or step s, we refer to them as RES(a,b) and ROD(a,b), where a is the size of the shorter and b is the size of the longer edge.

Rotated even distorted architectures RES and REW have the same topological properties, but they differ in the total number of full hexagonal cells inside the module. The definition of sensor primitives inside the distorted hexagon architectures is the same as in the rotated odd architectures RO-large (Table 3.14). The definition of the module width D for architectures RES(a,b) and REW(a,b), where $|a - b| = 1, a, b \in N$, expressed as the number of full hexagon cells is the following:

$$D = 2a \tag{3.18}$$

The total number of equal cells (CP, EP, FH) for the RES(a,b) architecture is:

$$N_{EQ} = 3a^2 + a \tag{3.19}$$

The total number of equal cells (CP, EP, FH) for the REW(a,b) architecture is:

$$N_{EQ} = 3a^2 - a \tag{3.20}$$

The definition of the module width D for architectures ROD(a,b), where $s = |a - b| \in 1, 2, 3, a, b \in N$, expressed as the number of full hexagon cells is the following:

$$D = a + b - 1 \tag{3.21}$$

The total number of equal cells (CP, EP, FH) for ROD(a,b) architecture is:

$$N_{EQ} = \begin{cases} 3a^2, & \text{if } s = 1 \\ 3a^2 + 3a, & \text{if } s = 2 \\ 3a^2 + 6a + 1, & \text{if } s = 3 \end{cases} \tag{3.22}$$

3.2.6 Evaluation of architectures

The important criteria having an impact on the choice of the hexagonal module architecture for the HGCAL SM design are:

1. Uniformity of the SC plane - SCs should remain in their initial positions defined with the hexagonal grid.
2. Full SCs maximization inside SM - The number of partial cells generated at the border of the hexagonal container during the packing procedure should be minimized and all SCs should be the same in area.
3. Minimal number of different SM types - Circle sector cuts are examined when the detector sensing layer is constructed with multiple SMs. The number of different module types should be minimized.
4. The silicon efficiency - The silicon material waste should be minimal when an irregular hexagonal SM is produced from a circular wafer.

We refer to a distorted SC plane as architectures having a “general tessellation problem”, which results in initial SC positions not overlapping with the positions of SCs inside the SM. Figure 3.17 is a brief preview of the procedure,

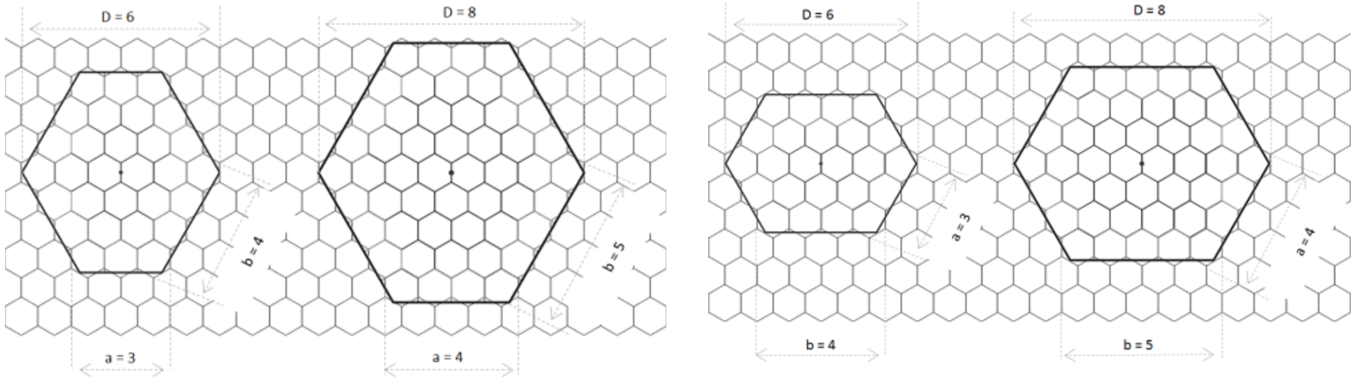


Figure 3.15: RE distorted architectures RES(a, b) and REW(a, b).

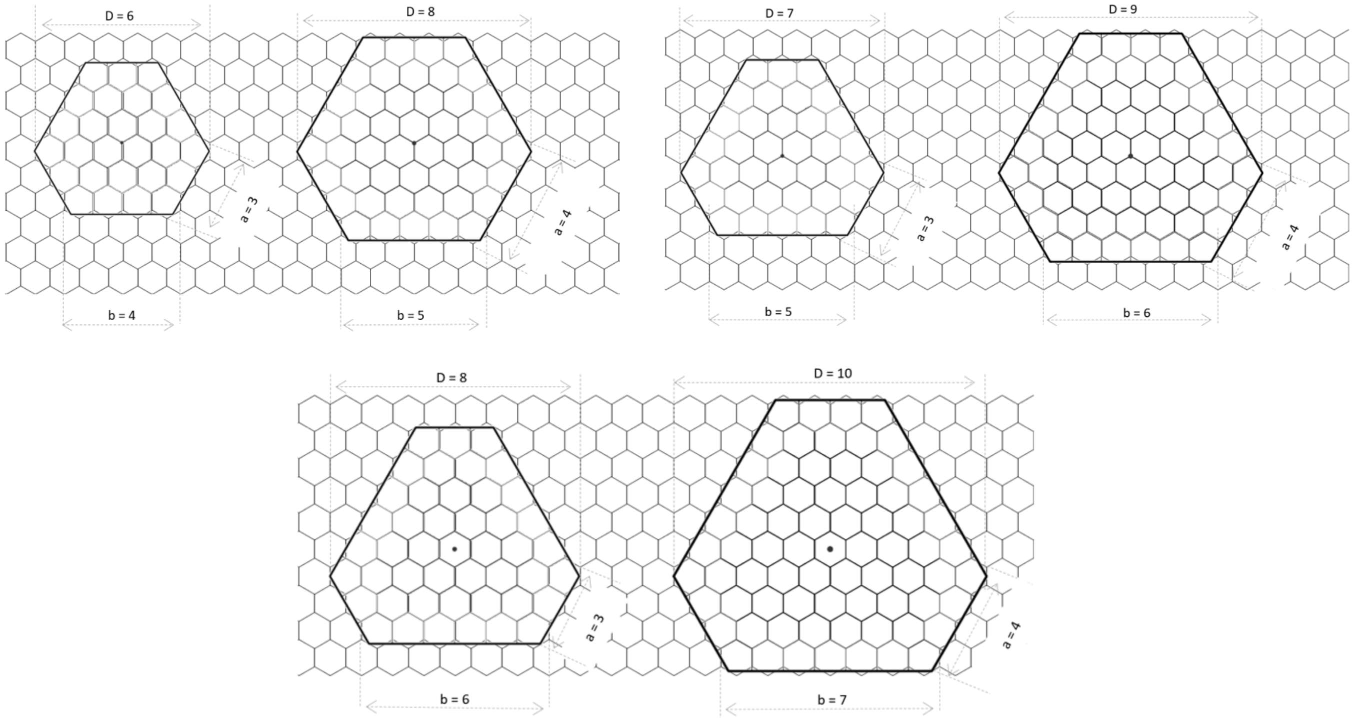


Figure 3.16: RO distorted architectures ROD1(a, b), ROD2(a, b) (up) and ROD3(a, b) (down).

visualizing problems that emerge when using different module architectures. Only few architectures (RO-large, RES, REW and ROD2) suffer from the general tessellation problem.

Table 3.3: Summarized differences between SM architectures.

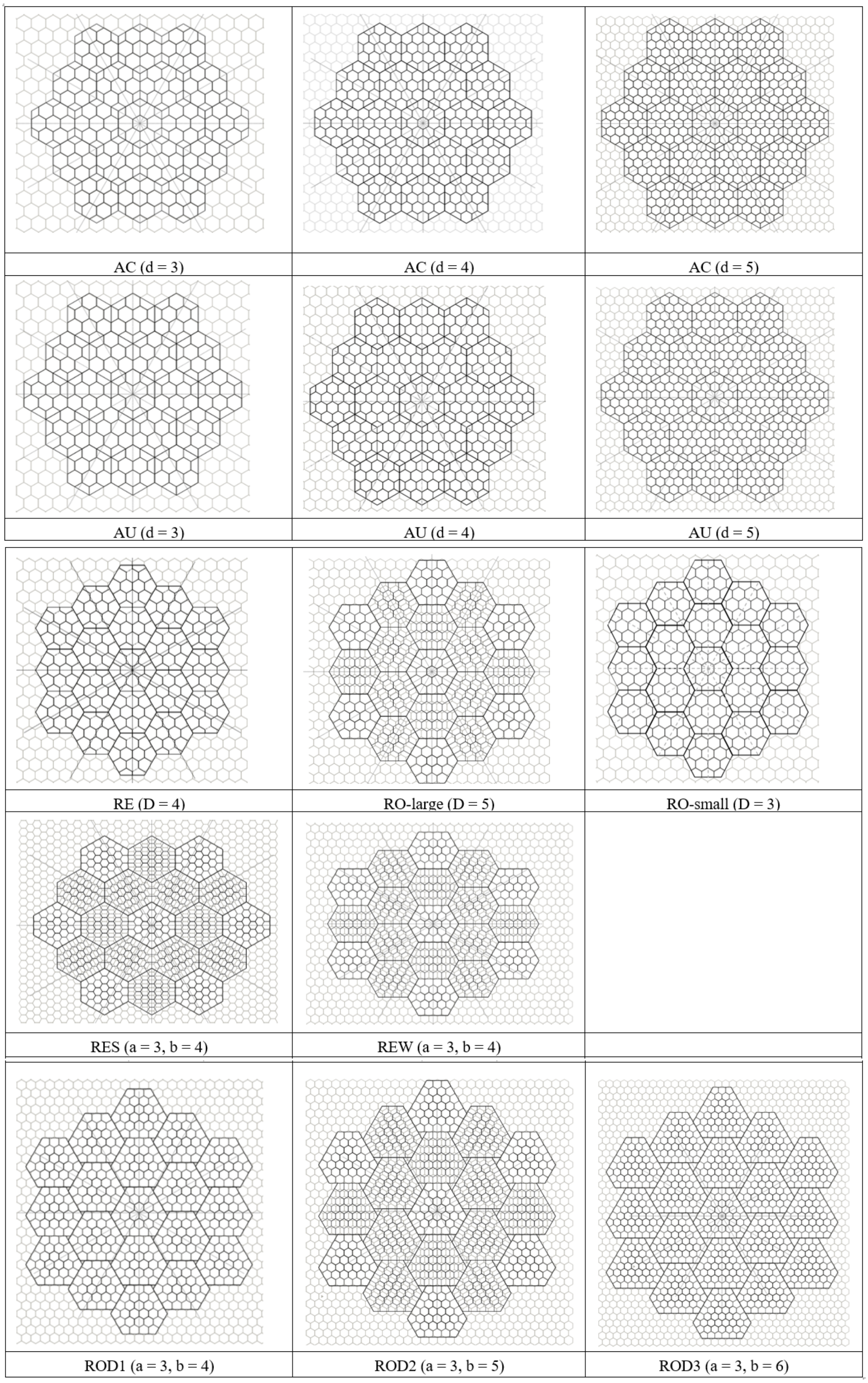
	#SC types	SC types	Gen. tessellation problem (criteria 1)	All SCs of same area (criteria 2)	#SM types (criteria 3)	
					sector=30°	sector=60°
AC0	3	FH, RHT, VHH	no	no	2	1
AC1	2	FH, VHH	no	no	2	1
AC2	2	FH, VHH	no	no	2	1
AU0	2	FH, VHH	no	no	3	2
AU1	3	FH, RHT, VHH	no	no	3	2
AU2	3	FH, RHT, VHH	no	no	3	2
RE	4	FH, HHH, EP, CP	no	no	4	2
RO-small	3	FH, VHH, PHT	no	no	2	1
RO-large	3	FH, EP, CP	yes	yes	2	1
RES	3	FH, EP, CP	yes	yes	many	many
REW	3	FH, EP, CP	yes	yes	many	many
ROD1	3	FH, EP, CP	no	yes	3	2
ROD2	3	FH, EP, CP	yes	yes	3	2
ROD3	3	FH, EP, CP	no	yes	3	2

While there is no tessellation problem for AC and AU, there are more odd-shaped cells at the SM boundaries and SCs are not the same in size (Table 3.3). We can rotate the module to provide RO-large architecture so that all cells are the same in area, but it tessellates with problems if we cover the whole plane with these modules. On the other hand, RE architecture tessellates well, but it has again a larger number of odd-shaped cells at the boundaries. If all SCs are the same in area in Table 3.3, it means the architecture does not have partial cells, and the number of equivalent cells N_{EQ} is equal to the number of full cells N_{FH}. In other words, the ratio between the number of FH cells and the number of equivalent cells is 100%, so the number of full SCs is maximized.

It is shown on Figure 3.18 that a larger number of full SCs can be obtained in aligned than in the RE architectures for the same module area. On the other hand, RO-small provides the smallest number of full SCs among all regular SM architectures, while in RO-large all inner SCs are the same in area ($N_{FH}/N_{EQ} = 100\%$). When including the distorted architectures in the comparison (Figure 3.18), we can see that ROD2 and ROD3 provide the largest number of equivalent or full SCs, while REW has a rather low total SC number.

When we calculate the total number of different SM types in Table 3.3 (to evaluate the criterion 3 for architectures comparison), the full SMs are not counted, but only partial or odd-shaped SMs that originate from the 30° or 60° sector cuts. Architectures RES and REW require that many odd-shaped modules are produced, as indicated in the table. For details on the SM shapes with 30° sectors one can refer to Figure 3.19, where we visualize the partial SM types that need to be produced when covering the circular detector layer.

It is shown that there is no perfect solution for SM geometry without any compromises along the way. Namely, each architecture has advantages and disadvantages regarding the defined requirements. Regular SM architectures that satisfy most of the criteria are AC, AU and RE, but the compromise which should be taken into account with



AC (d = 3)

AC (d = 4)

AC (d = 5)

AU (d = 3)

AU (d = 4)

AU (d = 5)

RE (D = 4)

RO-large (D = 5)

RO-small (D = 3)

RES (a = 3, b = 4)

REW (a = 3, b = 4)

ROD1 (a = 3, b = 4)

ROD2 (a = 3, b = 5)

ROD3 (a = 3, b = 6)

Figure 3.17: Layouts of the tessellated detector layer.

these architectures is a slightly larger number of different SC types at the module border. Additionally, as the SM geometry study evolved, we have seen that it is possible to obtain a larger number of SCs inside the module in the packing procedure while keeping all packed cells the same in area even at the boundaries. To obtain this, a hexagonal module had to become distorted, but with the preserved symmetry. We succeeded in keeping all SCs the same in area with this distorted SM, i.e. we accomplished the minimum number of different SC types (the maximized number of full SCs). Therefore, ROD2 architecture satisfied most of the criteria, but without the main requirement of avoiding the general tessellation problem. It is very important that the SC plane remains homogeneous, as it enables a simpler navigation between tessellated items on a higher level.

On the other hand, architectures ROD1 and ROD3 satisfy all criteria, and they have been emphasized as a promising architectures to use for SM design. One of the very important properties of these distorted designs is that there are no cuts needed on the SM vertices to provide mechanical apertures for fixating the SM. These spacing areas are provided naturally in the process of tessellating modules on detector layer (Figure 3.17). However, the silicon efficiency (SE) of the distorted architectures is examined in order to see if using distorted SMs is cost effective. It is the analysis of the silicon waste that needs to be done, as the SE in this case differs from the SE preserved when producing a SM which is a regular hexagon (SE=83%). We defined this as a criteria 4 when selecting the SM geometry, and based on the importance of the issue, we devote the whole Section 3.2.7 to this topic.

3.2.7 Study on sensor module production cost

The basic requirement when producing sensors is cost reduction by improving wafer productivity. This is defined as the fraction of the used wafer area to the total wafer area [58]. Szabo et al [59] defined the density of a circle packing, where r is a radius of the inner circle and S is size of the square container. The efficiency of packing n

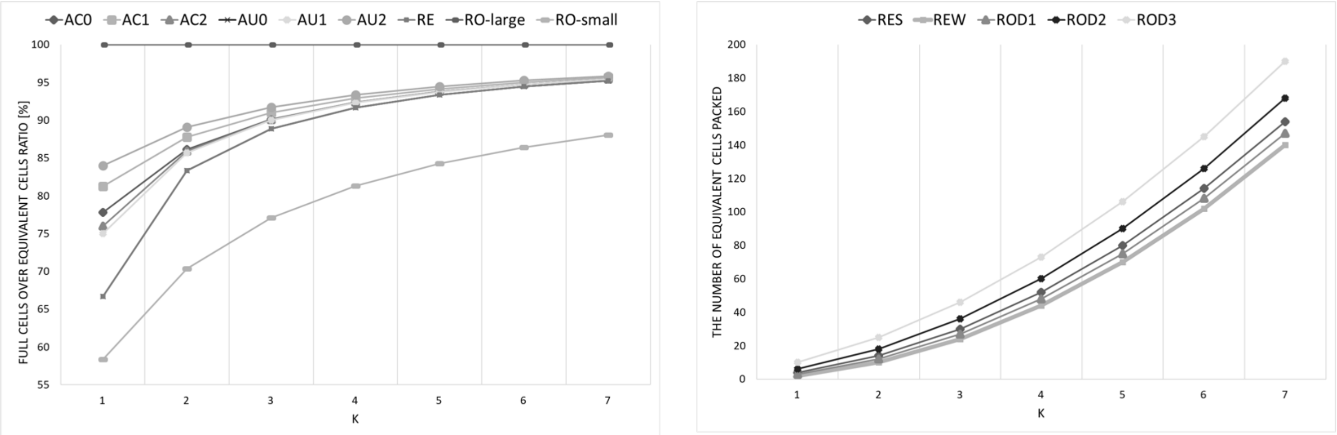


Figure 3.18: Comparison between different SM architectures. Module approximated by a regular (left) and an irregular or distorted hexagon (right).

circles is given by the formula:

$$d_n(r, S) = \frac{nr^2\pi}{S^2} \tag{3.23}$$

This means that packing density can be expressed as the ratio between total area covered by the packed items and the area of the usable container [59]. We adjust the former formula 3.23 to our case of using the hexagonal container area as well as the number of inner packed hexagonal items, to approximate the SE and to evaluate the SM production cost.

The study in this section is based on the published paper in [60], where we mathematically formulate the engi-

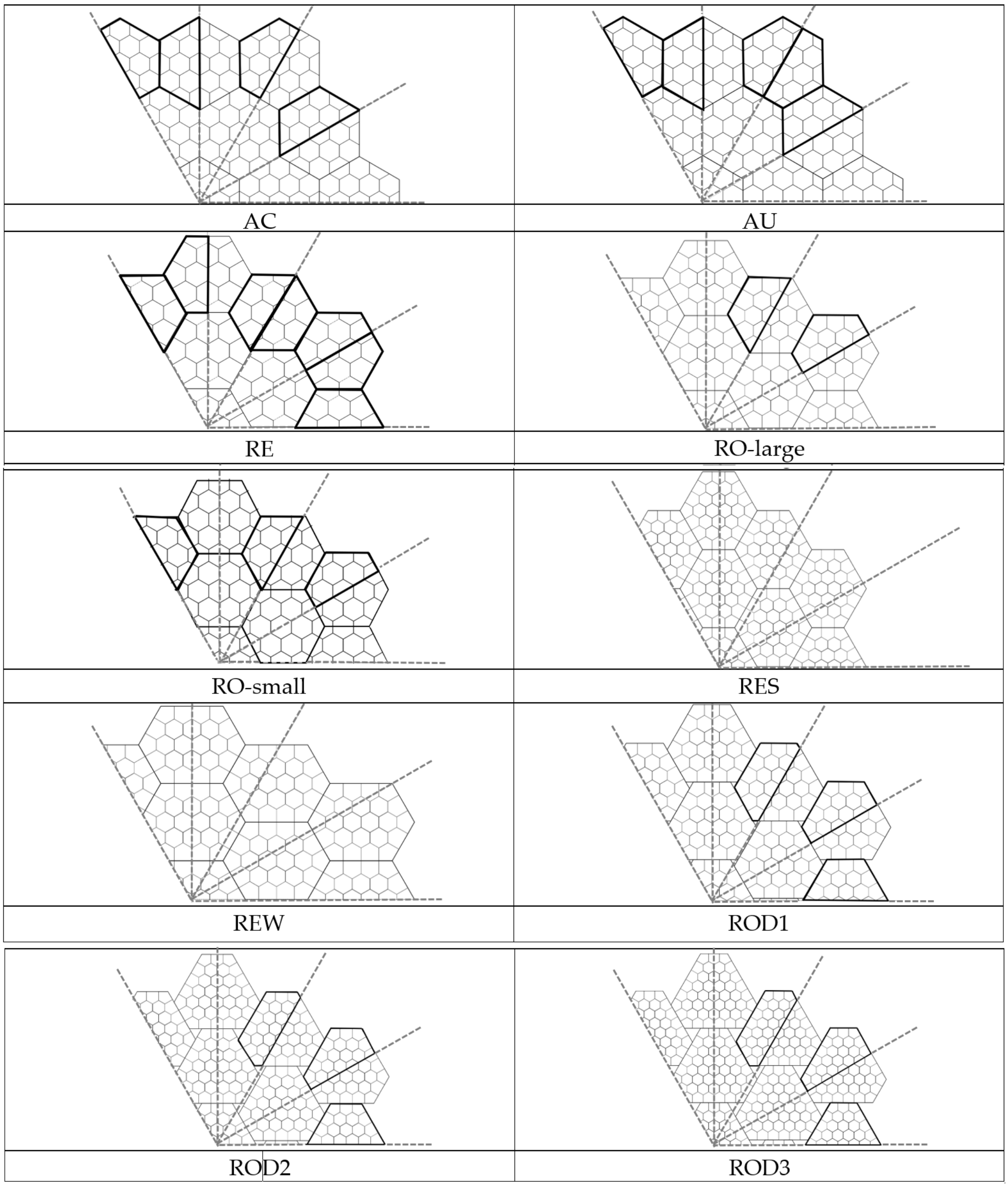

AC
AU
RE
RO-large
RO-small
RES
REW
ROD1
ROD2
ROD3

Figure 3.19: SM shapes (black) that need to be produced when covering the detector sensing layer (30° sector cuts are marked with a dashed line).

neering problem of producing ROD SM types from a circular wafer with maximal SE. As it is already stated before, the reason why we have chosen this distorted module shape instead of a regular hexagon is to provide spacings at the sensor vertices which will be used for placing mechanical apertures in the design of the new HGCAL detector [61]. For this, we have replaced the "perfect" tessellation which is naturally provided by regular hexagons, with semi-regular tessellation by the constructed irregular hexagonal SM. Otherwise, according to Figure 3.20, the corners of the modules would need to be cut, providing a triangular gap with a central point where three neighboring hexagon vertices meet. These cut out triangles or "vertex cuts" are clearly identified on a sketch when covering the detector surface in the tessellated manner. The tessellation of the plane is well-known in mathematics, which is covering a flat surface using one or more tiles without any overlaps or gaps. When tiles are regular polygons, we say that the tessellation is regular.

We construct an irregular hexagonal module that is semi-tessellating the targeted area and it naturally provides the vertex cuts during the construction of the detector layer. With this design, the SM remains symmetric and hexagonal in shape, even though irregular, and its efficiency remains satisfactory. Namely, we show that by producing the proposed irregular hexagon sensors from the same wafer as a regular hexagon, we can obtain almost the same SE.

Deriving formulas for SE with the irregular hexagon ROD(a,b)

We define regular and irregular SM shapes produced from a circular silicon wafer with six straight cuts, as presented on Figure 3.21. The irregular hexagon with three symmetry axes is adjusted from [52]. We extend the possible irregular hexagon types with respect to their size defined by the edges a and b, as it is already described in ROD definition. We prove that silicon waste would be minimized when the ratio $\frac{a}{b}$, $a \geq b$ is minimal. Also, increasing the ratio is proportional to the silicon waste. However, we show that the silicon waste is negligible compared to using a

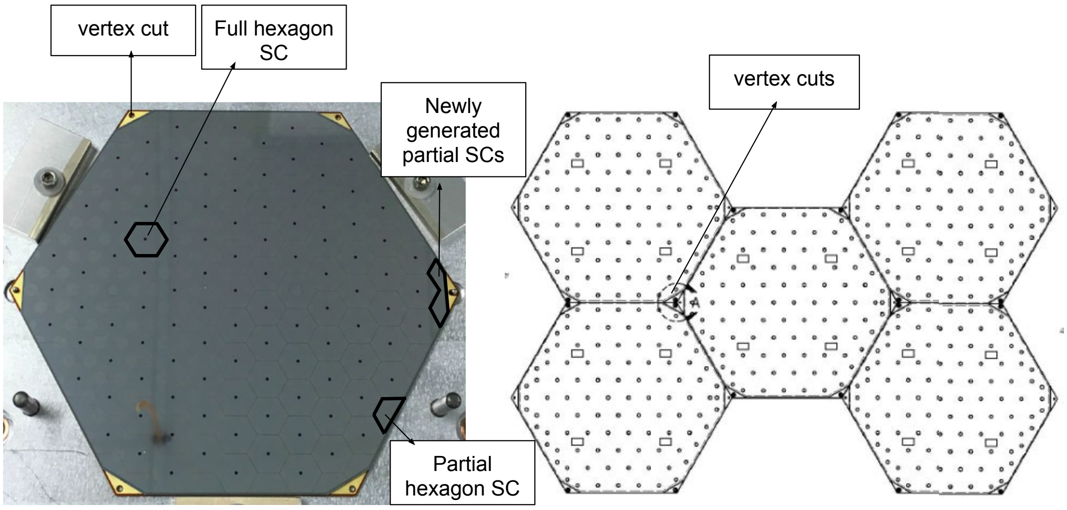


Figure 3.20: Hexagonal silicon SM of type AU(11) and the tessellation scheme with vertex cuts. Adjusted from [61, 62].

regular hexagon.

Let us consider an irregular hexagon with three symmetry axes in the triangular coordinate system (Figure 3.22). We define the hexagon size by two parameters; sides a and b, and the ratio $\frac{a}{b}$ is constant. We extend the class of hexagons by increasing the ratio $\frac{a}{b}$, which is shown on Figure 3.22 and Figure 3.23. As we have already stated before, we construct irregular hexagon which can be varied in size to enable the flexibility of the triangular spacings. We define irregular hexagon types denoted with H (Δ), where $\Delta = |a-b|$. Following the notation on the α and β angles from Figure 3.21 and from the relation $\alpha + \beta = 120°$ it follows [60]:

$$\tan\frac{\alpha}{2} = \frac{\sqrt{3}}{1+2\frac{b}{a}} \tag{3.24}$$

Therefore, if $\frac{a}{b} = \frac{7}{5}$, then it follows that $\alpha \approx 71°$ and $\beta \approx 49°$. The triangular spacing is made of equilateral triangles with side x that is defined as $x = \frac{\Delta}{2}$. Based on the size of the triangular spacing, higher Δ provides larger triangle area that causes larger waste in the sensor production, but it provides larger area for mechanical apertures when N sensors are covering a circular area of interest. For $\alpha > \beta$, the area of this irregular hexagon can be calculated by using the formula [60]:

$$A_H = \frac{3\sqrt{3}}{2} r^2 \sin\left(150° - \alpha\right) \tag{3.25}$$

where the circumscribed circle radius r is: $r = \frac{a}{2\sin\frac{\alpha}{2}}$. The area comparison can be expressed as the ratio between the regular and the irregular hexagon with areas A and A_H respectively, having the same circumscribed circle radius [60]:

$$\frac{A}{A_H} = \frac{1}{\sin\left(150° - \alpha\right)} \tag{3.26}$$

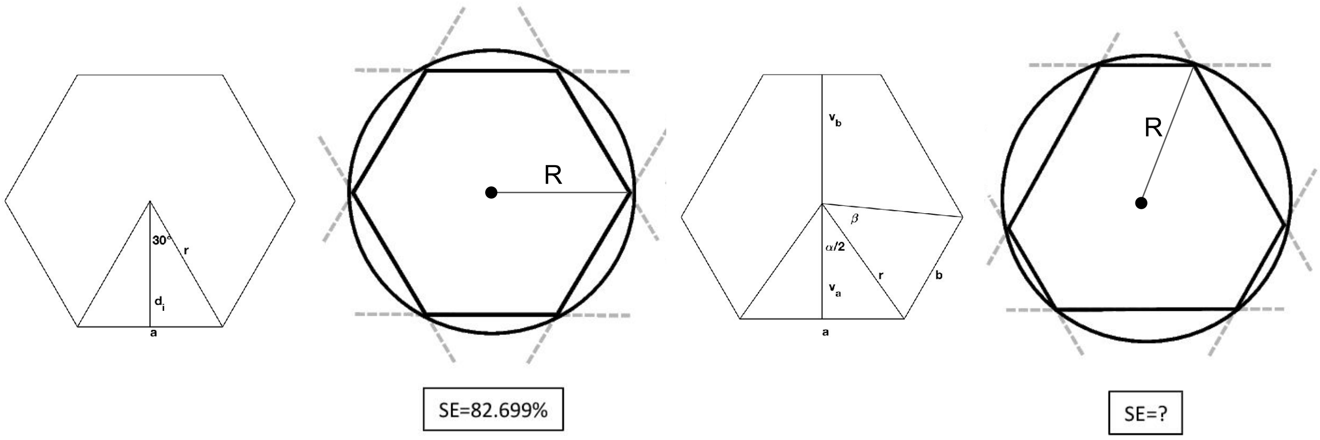


Figure 3.21: SM geometrical shapes and production. Regular hexagon and irregular hexagon [60].

Consequently, for the case where the ratio $\frac{a}{b} = \frac{7}{5}$, it follows that $\frac{A}{A_H} \approx \frac{1}{0.98}$. We can conclude that the SE would be almost the same when a single regular or irregular hexagon of type H(1) is produced from a circular wafer of a fixed radius. We derive the formula to calculate the SE when H(Δ) is produced from a circular wafer [60]. The area of an irregular hexagon is given by Formula 3.25, and when we compare it to the area of a circle containing the hexagon similarly to 3.23, the SE can be calculated [60]:

$$SE = \frac{3\sqrt{3}\sin(30^\circ + \beta)}{2\pi(1 + (\cot\frac{\beta}{2})^2)(\sin\frac{\beta}{2})^2} \tag{3.27}$$

Results and evaluation

First, we discuss the dependency of the angle α to the size of an irregular hexagon H(Δ). Then, we analyze how close can we get to the SE of a regular hexagon with various H(Δ). The results are presented on Figure 3.24. Naturally, based on the relation 3.24, when $\frac{b}{a}$ approaches 1, H(Δ) approaches the SE of a regular hexagon as α approaches 60° angle. Naturally, H(1) is the closest to the regular hexagon (the closest to SE=83%), so it obtains the highest efficiency (above $\approx 80\%$). Hence, the silicon waste is negligible when using this type of irregular hexagonal shape with respect to a regular hexagon. However, it is shown that using other irregular hexagons is also cost-

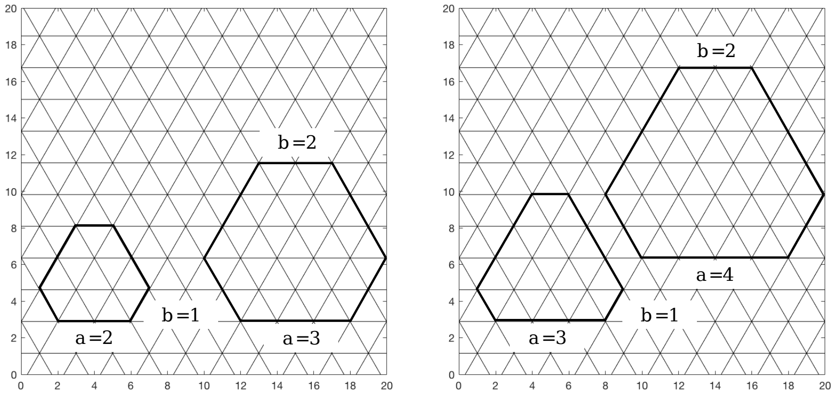


Figure 3.22: Irregular hexagon H(1) (left) and H(2) (right).

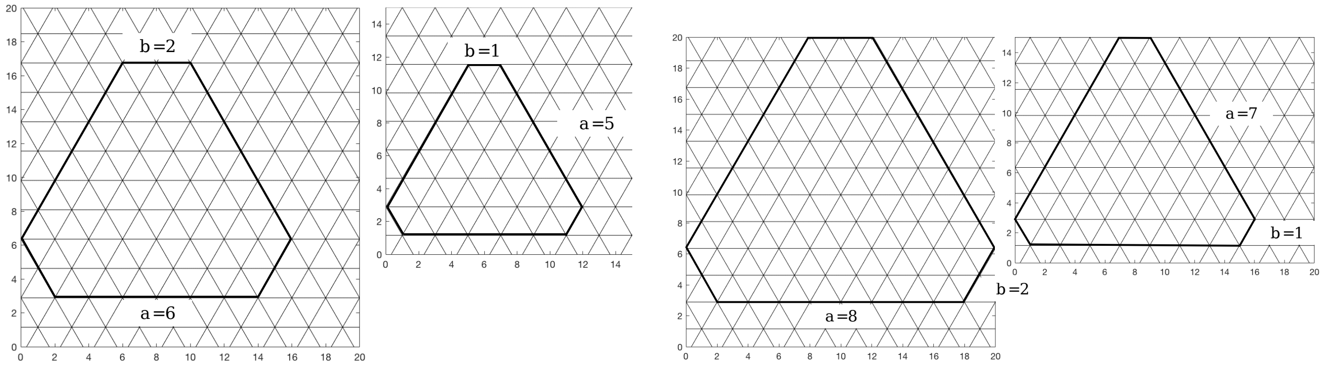


Figure 3.23: Irregular hexagon H(4) (left) and H(6) (right).

effective. We can see on Figure 3.24 that $SE > 55\%$ for $\Delta \leq 5$, which means that more than half of the wafer is utilized. As we saw in Section 3.2, larger ROD architectures H(Δ) where $\Delta > 1$ are important because of the larger number of total SCs that can fit inside, and the larger triangular spacings are provided in the construction of the detector layer, avoiding the vertex cuts of the existing SCs. Also, we saw that only architectures with odd Δ value are interesting, since they do not suffer from the general tessellation problem.

3.3 Front-end data reduction based on geometry

The data reduction mechanism on the FE is already described in Chapter 2, where we saw the full HGCAL multi-staged trigger system which generates the trigger primitives based on energy deposits. As a reminder, the HL-LHC will deliver busy high-energy events at a 40MHz rate and the CMS Level 1 trigger will have to reduce the data rate to 750kHz while preserving interesting physics events. Namely, the future HGCAL of CMS will consist of about 6 million channels and not all sensor data could be read and stored for further processing. There are 40 million events per second, while only a few hundreds of events per second can currently be recorded offline. CMS uses a trigger

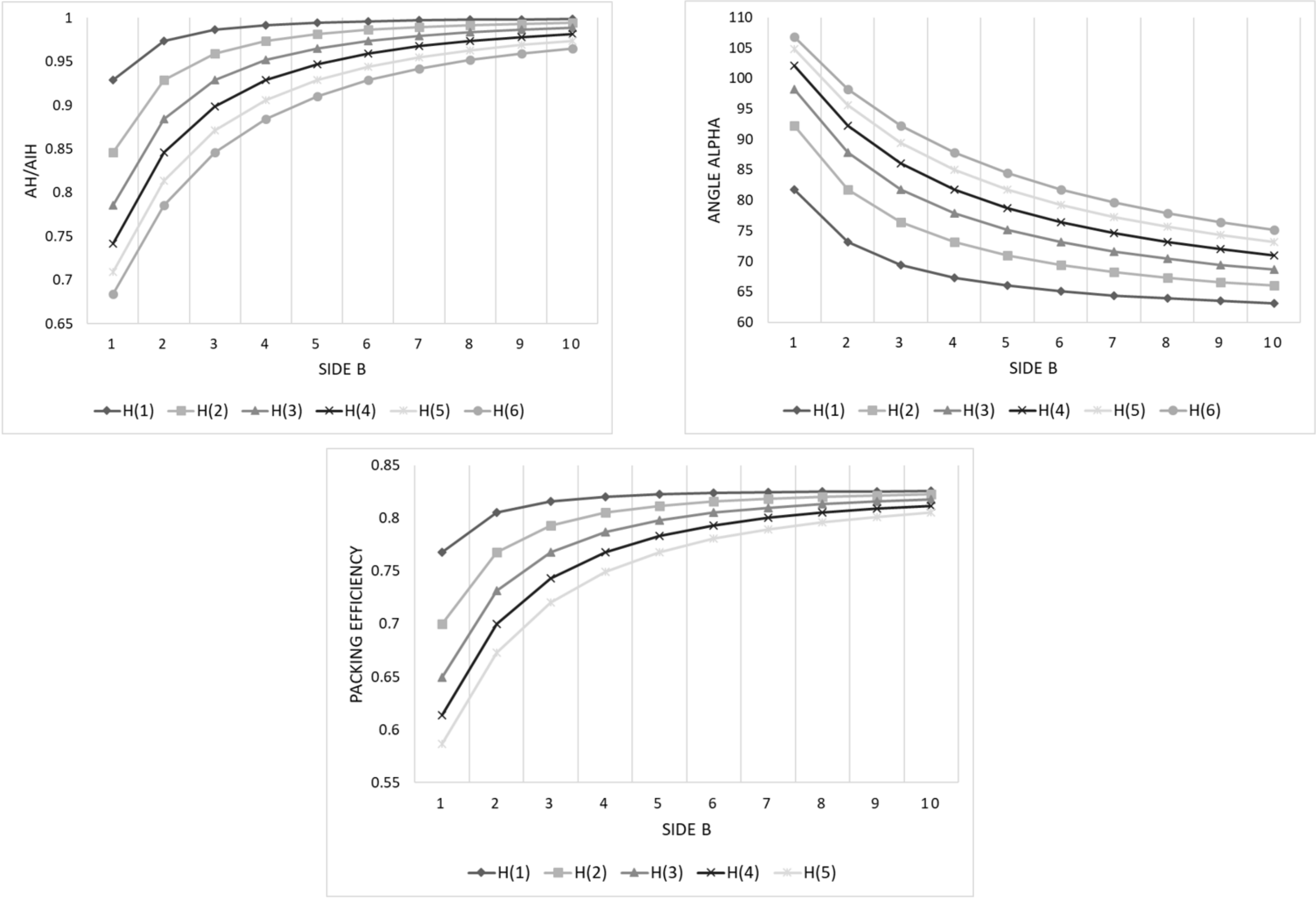


Figure 3.24: Irregular hexagon efficiency. Comparison to regular hexagon area (up left) and H(Δ) angle α (up right); SE for H(Δ), Δ=1, 2, 3, 4, 5 (down).

system to select the events of interest whose data is stored and analyzed [63]. Also, the bandwidth requirements should be met in order to send the data, so a reduction is applied in several forms. One of the approaches is based on the detector geometry design, where the data reduction is performed by using the clustering procedure. Hence, detector hexagonal SCs in the SM are grouped into larger polyhex clusters or TCs [61, 27], so there is another aspect that must be considered when designing a SM, and that is how to efficiently form TCs. This issue has already been tackled in Section 3.2.1 when we described the SM design requirements from the technical proposal [57].

In this section we present a study on TC formation. First, there are many different ways on how hexagonal SCs can be clustered by merging hexagonal SCs together in a polyhex structure [64]. The symmetric TC candidates that are the most closely packed are the most promising, such as triangular trihex, diamond tetrahex and hexagonal heptahex (Figure 3.25). We concentrate on these TC shapes in Section 3.3.1 and examine how they can be efficiently packed inside a SM. Namely, since TCs are formed by grouping hexagonal SCs inside the SM, intuitively, clusters should also be packed in the module. Each SM should possibly contain its own clusters or, at least, the number of shared clusters at the SM border should be minimized. This is to reduce or possibly to avoid communication between boards that are processing data from each of the neighboring SMs. Ideally, the cluster plane should remain uniform, to keep the simplicity of the nearest-neighbor finder algorithm [63].

We use only AC, AU and ROD1 architectures to model the SM in our TC studies. Namely, the potential of ROD1 has already been shown, while the AC(d) where $d = 11$ and AU(d) where $d = 15$ have been used for a long time as default architectures in the CMSSW simulation [65]. The Section 3.4 is based on the published paper in [66], where a new vertex-sharing or vertex-aligned model is described, which is accepted as a new HGCAL SM design.

3.3.1 Trigger cell definition and study on TC regularity

There was a clear intention to avoid partial SCs from SM design in Section 3.2.2. The reason is that when SCs are situated at the module border or module vertex, they are actually "shared" between two (or three) neighboring modules that are tessellated in the detector sensing plane. It implies that TCs whose SCs belong to are also shared, so that HGCROCs which process data from these modules need to communicate. This will complicate

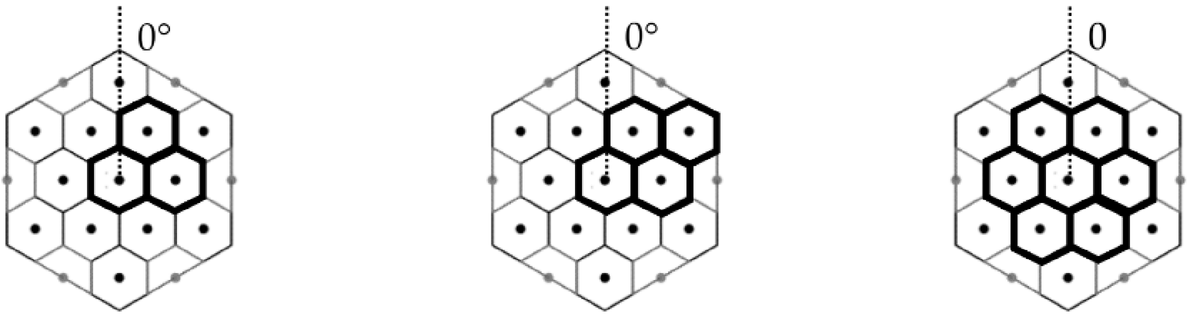


Figure 3.25: TC polyhex types; triangular trihex, diamond tetrahex and hexagonal heptahex.

the architecture design, as well as cause data reduction that will occupy the available data bandwidth. Hence, we introduce a new term which is a number of shared cells (N_{shared}) or a communication factor. For example, when AC modules with inner packed items are tessellated, we can calculate the total number of shared cells by summing the Formulas 3.10 and 3.11. Also, the number of equivalent shared cells for AC(d) can be calculated as follows:

$$N_{shared} = \begin{cases} d-1, & \text{if } k=0 \text{ and } k=1 \\ d+1, & \text{if } k=2 \end{cases} \tag{3.28}$$

This can be checked on Figure 3.17, where for AC(d), $d = 3$ and $d = 4$, the number of shared SCs for each of the tessellated modules is $N_{shared} = 2$ and $N_{shared} = 3$ respectively. We follow the same logic when analyzing the number of inner and shared TCs, when each of the TC types is packed in the SM of type AC. Also, we analyze the number of different SM types with the TC clustering.

Packing trigger cells TC3 in AC(d), $d \neq 3k$, $k \in N$

The packing trihex TC3 results when the SM size is not a multiple of 3 are given on Figure 3.26. We can see that the packing structure is not equal for every SM in this case, and we have several SM types with the TC clustering.

Packing trigger cells TC3 in AC(d), $d = 3k$, $k \in N$

Also, we visualize on Figure 3.26 the packing result for trihex TC3 when the SM size is a multiple of 3 ($d = 6$). The packing structure is constant for every SM as shown, so we have a single SM type in the TC plane. The quantized packing results are given in Table 3.4. We derive a formula for calculating the number of full polyhexes of size TC3, and the number of shared equivalent polyhexes at the SM border:

$$N_{full} = \frac{d^2 - 3d + 3}{3}; \quad N_{shared} = d - 1 \tag{3.29}$$

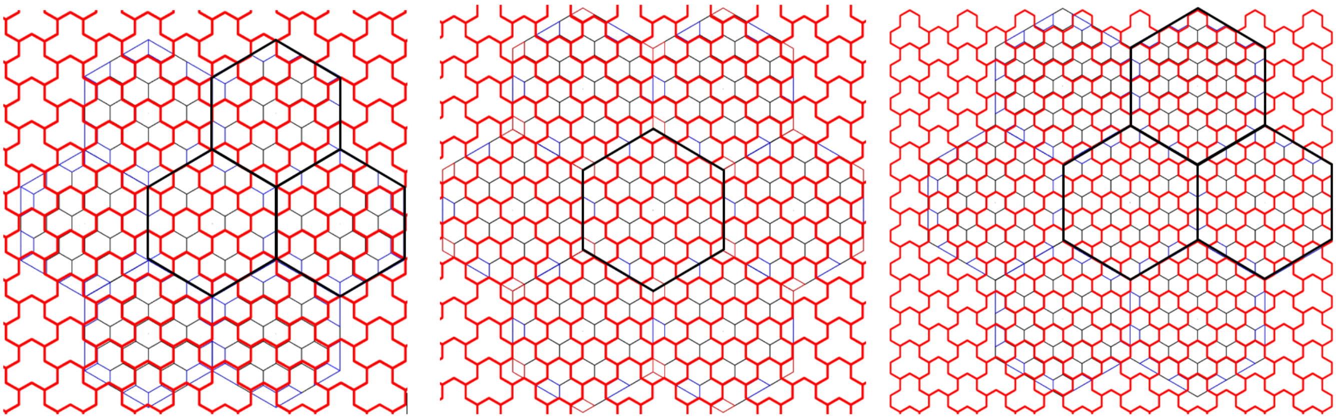


Figure 3.26: TC3 packing result; $d = 5, 6, 7$. The SM types are marked in black.

Table 3.4: Packing results for polyhex TC3.

SM size (d)	N_{full}	N_{shared}
3	1	2
6	7	5
9	19	8

Table 3.5: Packing results for polyhex TC4.

SM size (d)	N_{full}	N_{shared}
4	1	3
6	5	4
8	10	6

Packing trigger cells TC4 in AC(d), $d \neq 2k$**,** $k \in N$

The result of packing tetrahex TC4 when the SM size is not a multiple of 2 are given on Figure 3.27. We can see that the packing structure is not equal for every SM in this case.

Packing trigger cells TC4 in AC(d), $d = 2k$**,** $k \in N$

On the other hand, when the SM size is a multiple of 2 ($d = 6$), the packing structure is constant for every SM as shown on Figure 3.27. The quantized packing results are given in Table 3.5. We derive a formula for calculating the number of full polyhexes TC4 inside the SM:

$$N_{full} = \begin{cases} \frac{3d^2-10d+12}{12}, & \text{if } k = 0 \\ \frac{3d^2-10d+4}{12}, & \text{if } k = 1 \\ \frac{3d^2-10d+8}{12}, & \text{if } k = 2 \end{cases} \tag{3.30}$$

We derive a formula for calculating the number of shared equivalent TC4 at the SM border and the expression

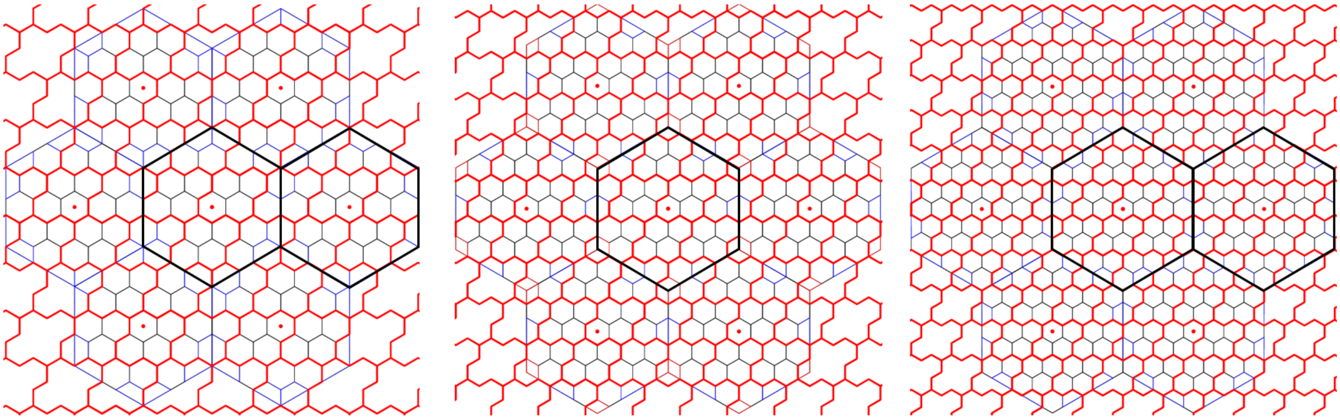


Figure 3.27: TC4 packing result; $d = 5, 6, 7$. The SM types are marked in black.

is as follows:

$$N_{shared} = \begin{cases} \frac{5d-6}{6}, & \text{if } k = 0 \\ \frac{5d-2}{6}, & \text{if } k = 1 \\ \frac{5d-4}{6}, & \text{if } k = 2 \end{cases} \tag{3.31}$$

Packing trigger cells TC7 in AC(d), $d \neq 7k$, $k \in N$

The packing heptahex TC7 results when the SM size is not a multiple of 7 are given on Figure 3.28. We can see that the packing structure is not equal for every SM in this case.

Packing trigger cells TC7 in AC(d), $d = 7k$, $k \in N$

We visualize on Figure 3.28 the packing result for heptahex when the SM size is a multiple of 7. The packing structure is constant for every SM as shown. We derive a formula for calculating the number of full polyhexes TC7 inside the SM:

$$N_{full} = \frac{d^2 - 5d + 7}{7} \tag{3.32}$$

We derive a formula for calculating the number of shared equivalent TC4 at the SM border and the expression is as follows:

$$N_{shared} = \frac{d^2 + 4d - 7}{7} \tag{3.33}$$

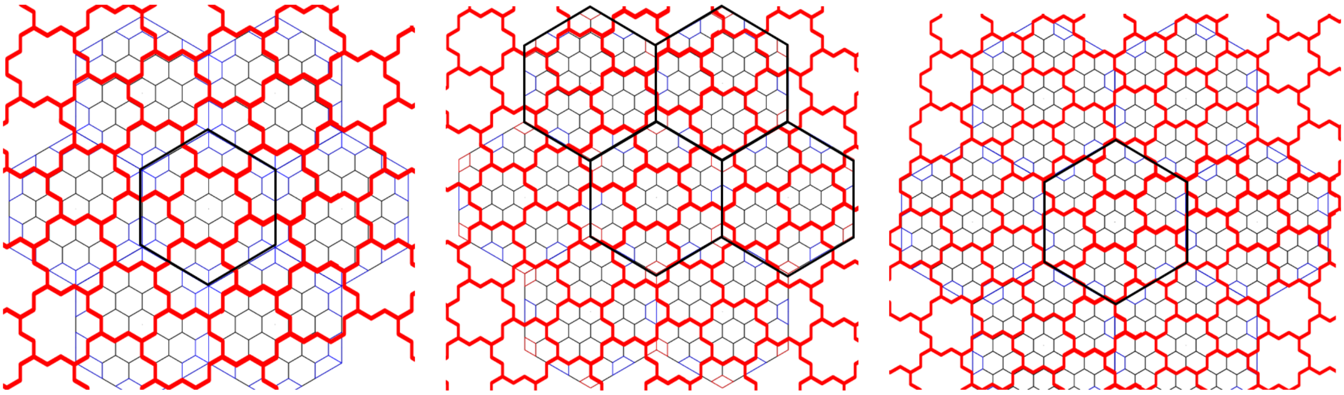


Figure 3.28: TC7 packing result; $d = 5, 6, 7$. The SM types are marked in black.

3.3.2 Discussion and evaluation

Based on the analysis, we conclude that the general packing structure for packing TCn polyhexes in the hexagonal container when $n = 3, 4, 7$ will be constant for each of the tessellated SMs if the size d of the SM is a multiple of the smallest prime divisor of n, i.e. $d = 3, 2, 7$. Hence, these SM sizes are optimal for the TC clustering procedure, as they keep the single SM type in the whole detector sensing layer.

To evaluate the efficiency of different polyhex packing approaches, we calculate the fraction of N_{full} and N_{shared} towards the total number of TCs. However, packings are comparable only if the polyhex size is the same, which puts restriction since the size of the small hexagonal cell inside the polyhex will be scaled based on the chosen SM type. Hence, to get a fair comparison, we chose the SM size d to be the common multiples of $3, 4, 7$, i.e. $84, 168, 252, 336, 420$. In this case, the small hexagon side is constant for each packing type. Naturally, the number of full polyhexes packed inside the SM will be lower when we increase the size of the hexagonal cells in a single TC, which means that a larger number of full TCs can be fit in the same module if we use TC3, then TC4 and the smallest number is for TC7. However, the arrangement will be the other way round for the number of shared polyhexes at the SM border. We want to minimize this, so that there are as few shared TCs as possible, and in this case TC7 is the best (Figure 3.29).

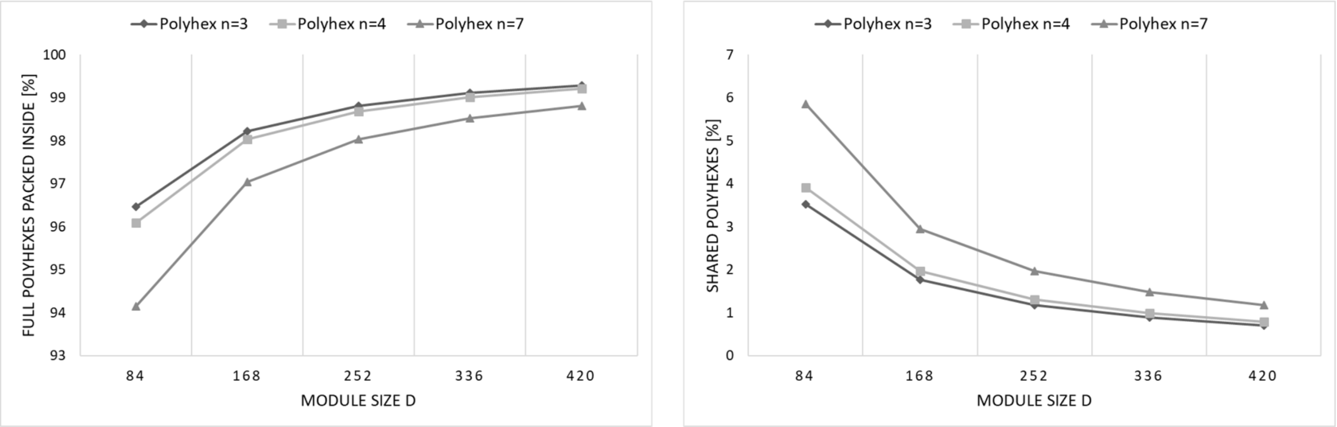


Figure 3.29: Comparison of packing TC3, TC4 and TC7. The fraction of the full TCs packed inside (left) and the fraction of the shared TCs at the module border (right).

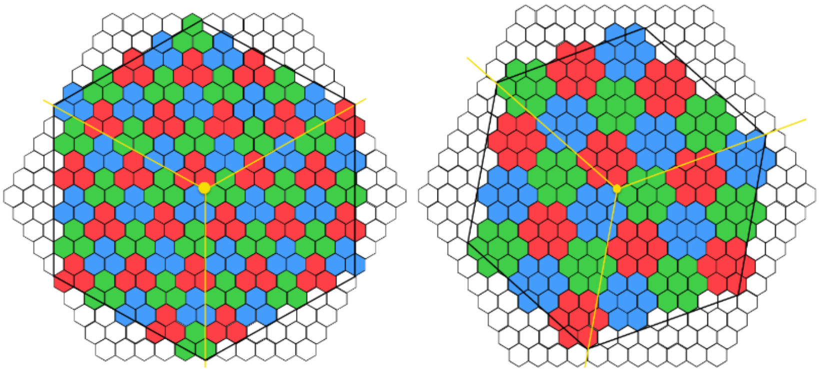


Figure 3.30: Packing TC3 and TC7 in vertex-sharing architectures [67].

Other architectures are analyzed besides AC like vertex-aligned or vertex-sharing SM models, where TCs can be formed with packed TC3 and TC7 polyhexes (Figure 3.30). Again, the intention is that all TCs have the same size and orientation, so that the TC plane remains uniform [67]. In our published paper [66], we have examined all the possibilities of cluster forming with AC and AU architectures and TC4, where we rotate the TCs by $60°$, so that $\alpha = 30°, 90°, 150°, 210°, 270°, 330°$. The example of TC4 on Figure 3.25 represents the basic $30°$ angle of the diamond tetrahex cluster. We concluded that TC sharing at the SM border is inevitable with these architectures.

As a main conclusion of the study with different TC definitions in Section 3.3.2, we saw that there is not much difference between TC3 and TC7 when the number of shared TCs is considered. Also, the number of full cells makes TC4 a good compromise, so we keep TC4 polyhex as our main target for the trigger. Also, we show the possibility of packing TC4 in ROD1, where $a = 8$, $b = 9$ on Figure 3.31. The main advantages of ROD1 were already explained in Section 3.2.2, while at the same time the number of SCs in the module is 192 (based on Formula 3.22) with 44 full and 4 shared trigger cells TC4. Hence, it is good compromise to replace 128 and 256 cell architectures (6 inch wafers) that are foreseen in the technical proposal with this single architecture [68]. Also, there is a single type of SM in the clustered TC plane, because the condition is valid that $a = 4k, k \in N$.

We can notice one more advantage of the ROD1 architecture when TC4 clusters are formed. Namely, there is a reduced number of shared TCs at the SM border, which is now present only on two edges, unlike with other architectures where TC sharing is present on all SM edges. However, a disadvantage is that the SM remains distorted or irregular hexagon, so in Section 3.4 a new vertex-aligned solution for a SM design is described. With this solution, the same total number of SCs is kept, but SM is a regular hexagon. There are two possibilities with this new model. First, the same effect with reduced communication factor like in ROD1 can be accomplished if the TC plane remains uniform. Next, in the non-uniform TC cluster packing scheme of $120°$, where the TCs are packed on each third of the module but rotated by $120°$, it is accomplished that all clusters are contained inside the module region, so that the communication factor between modules is reduced to zero. Also, since cluster-sharing is completely avoided at the SM border, a maximal packing efficiency is obtained compared to the other models [66].

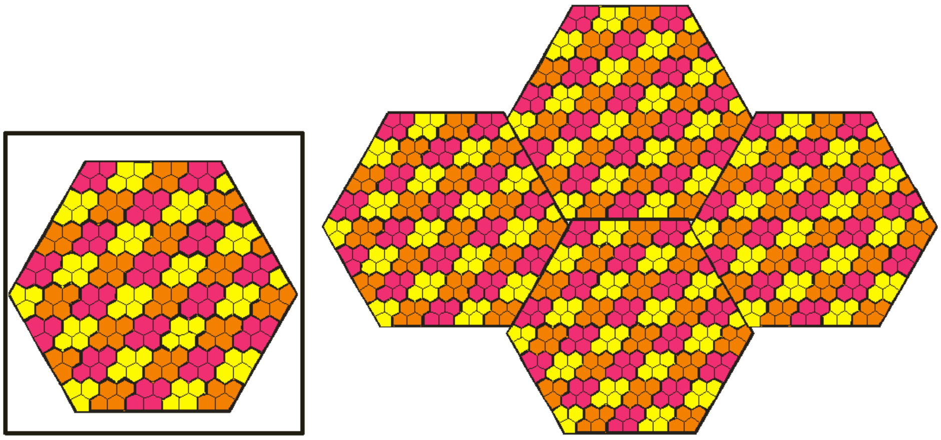


Figure 3.31: Packing TC4 polyhex in ROD1, $a = 8, b = 9$ [68].

3.4 Proposed SM geometry design for HGCAL

Following our geometry studies, another module schema was proposed for HGCAL in [67]. Its general structure is similar to the work of Holub et al [51], from where we extend the notation in this section to formally describe the geometry. The Holub model can be referred to as H(D), where $D = 8k, k \in N$ is the total number of inner small hexagonal edges, situated on the circumscribed circle diameter of the SM. Example for $D = 8$ is given on Figure 3.4.

3.4.1 Uniform TC4 formation with H(D)

We examine a uniform clustering with the model proposed in [67]. As shown on Figure 3.32, we have 3 possibilities for the inner central TC: central cluster position down, central cluster position left (symmetric to position right), and

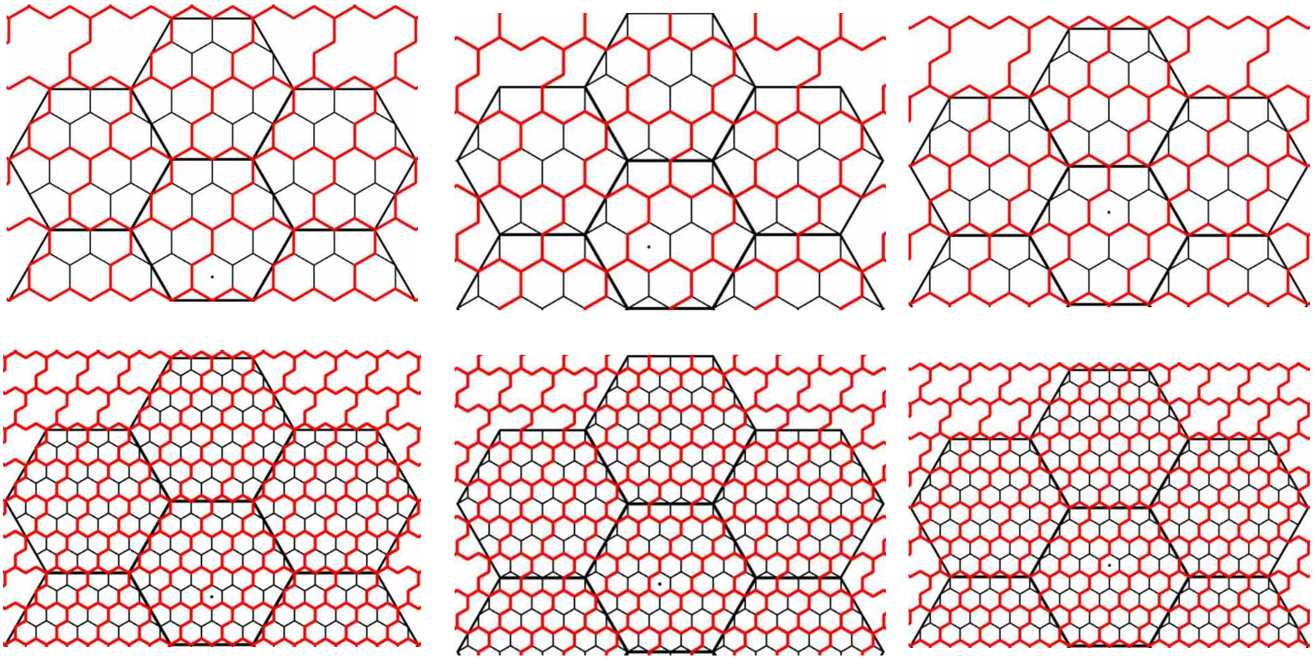


Figure 3.32: Options for clusters centers in the H(D) architecture. Position down (left), left (middle) and up (right).

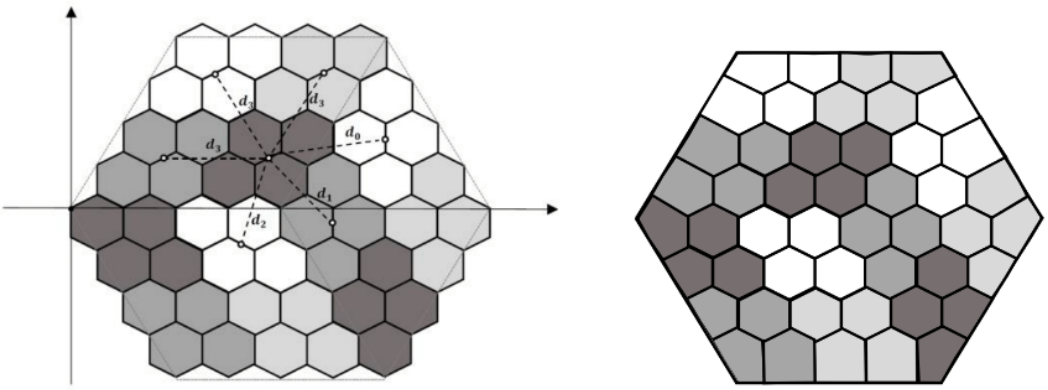


Figure 3.33: Non-uniform clustering with H(D). Cluster NN distances d_i (left) and model without voids (right).

central cluster position up. It can be seen that if the SMs are arranged in a tessellated manner, they all have a single TC4 clustering structure. The results from Figure 3.32 are summarized in Table 3.7, providing the number of inner full clusters packed inside the SM and the number of shared clusters at the border. Algebraic expressions for calculating the number of inner packed cluster items as well as the number of shared clusters are as follows:

- Central TC position down:

$$N_{full} = \frac{3D^2 - 8D}{64}; \quad N_{shared} = \frac{D}{4} \tag{3.34}$$

- Central TC position left:

$$N_{full} = \frac{3D^2 - 24D + 64}{64}; \quad N_{shared} = \frac{3D}{4} \tag{3.35}$$

- Central TC position up:

$$N_{full} = \frac{3D^2 - 16D}{64}; \quad N_{shared} = \frac{D}{2} \tag{3.36}$$

Table 3.6: Calculation of full and shared clusters for H(D) with TC4.

Table 3.7: Uniform clustering.

SM size	TC up		TC left		TC down	
(D)	N_{full}	N_{shared}	N_{full}	N_{shared}	N_{full}	N_{shared}
8	1	4	1	6	2	2
16	8	8	7	12	10	4
24	21	12	19	18	24	6
32	40	16	37	24	44	8
40	65	20	61	30	70	10

Table 3.8: Non-uniform clustering.

SM size	TC $120°$	
(D)	N_{full}	N_{shared}
8	3	0
16	12	0
24	27	0
32	48	0
40	75	0

3.4.2 Non-uniform TC4 clustering with H(D)

Based on our geometry studies, another clustering model is proposed that is non-uniform [67]. The orientation of tetrahex clusters is rotated by $120°$ on each SM third, similar to [69]. Hence, the non-uniformity is present in the cluster plane, as the distance to all NN clusters is not constant. In this architecture, all clusters are entirely contained inside the SM with no shared clusters at the border, where border TCs are not completely the same in area (Figure 3.33).

Clustering results for the non-uniform TC4 clustering with H(D) (the example on Figure 3.33) are quantized in Table 3.8. We derive the corresponding algebraic expressions. In this clustering approach, there are no shared clusters at the SM border, since they are all contained inside SM. Thus, the number of full clusters is calculated with the following formula:

$$N_{full} = \frac{3D^2}{64} \tag{3.37}$$

3.4.3 Discussion and evaluation

To compare the models for uniform and non-uniform clustering, we use Formula 3.34, Formula 3.35, Formula 3.36 and Formula 3.37, and the efficiency result is shown on Figure 3.34. The maximal number of full clusters or packed items is obtained for uniform with cluster plane moved down. Also, it has the lowest number of shared clusters among uniform architectures. On the other hand, moved left is the least efficient, since sharing of clusters is present at every edge of the SM. For most of the other uniform H(D) models, sharing is present only at two edges, which is the minimal solution that can be obtained. The non-uniform model is the most efficient of all, since the number of packed items in the SM is maximized and it requires no cluster sharing. Based on its great advantages, it is this architecture which will be used in the future HGCAL (Figure 3.35).

Let us discuss how the H(D) architecture compares to the other module architectures presented before. The criteria from Section 3.2.6 is considered:

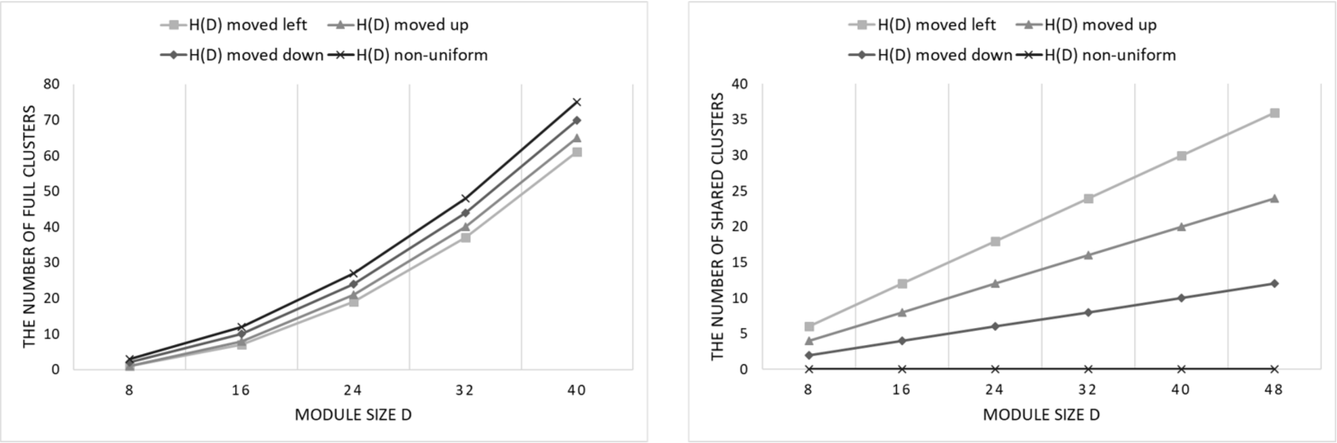


Figure 3.34: Results on packing TC4 in H(D) architectures. The number of full TCs packed inside the module (left) and the number of shared TCs at the module border (right).

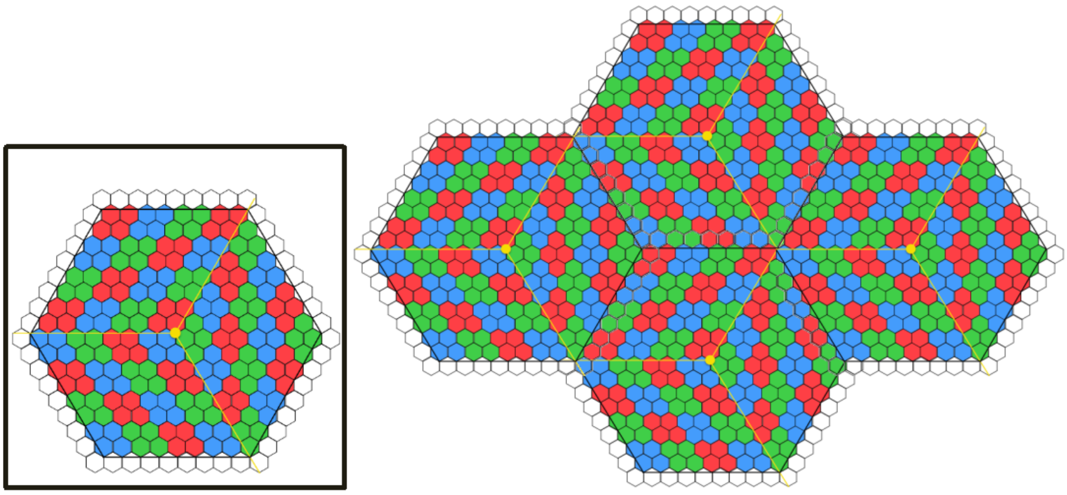


Figure 3.35: Packing TC4 in non-uniform H(D), $D = 32$. Adjusted from [27].

- The uniformity of the SC plane is preserved, and there is no "general tessellation problem".
- The full SC maximization inside the SM is not accomplished, since the SCs are not all the same in area. Also, the number of SC types is not minimal, as there are 4 inner-packed SC types, which is larger than for other architectures (Figure 3.33).
- The number of SM types when covering the detector sensing plane with H(D) is minimized, since there is only a single module type (for both 30° and 60° sector cuts).
- Since H(D) is a regular hexagon, the SE is maximized (SE=83%).

Although Gecse was the actual author of the architecture that will be used in the future HGCAL [67], we performed the extensive set of the geometry studies that were presented to others [70], inspiring them to contribute to the field. The H(D) was accepted, but that was not the point of the thesis. The presented set of geometry studies is a significant step towards the final solution, and what is accepted at the end is decided by the CMS collaboration.

Our main intention in the geometry research was to keep the TC plane uniform, and even with the H(D) model, we still explored the TC plane uniformity (Section 3.4.1). However, in all our strategies and approaches, there was always an inevitable communication between neighboring modules. As the main focus from the group became how to omit this, the significant contribution from Gecse was to find the way to avoid the inter-module communication

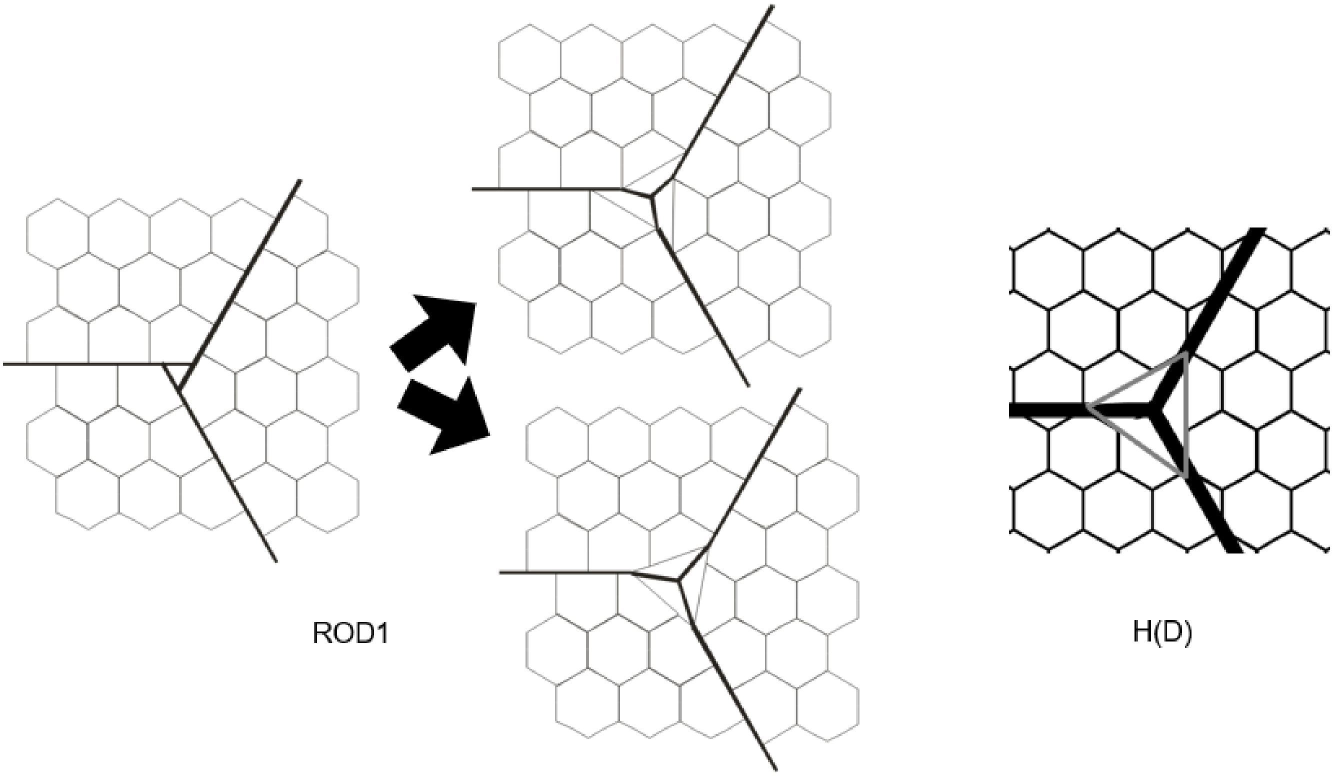


Figure 3.36: Comparison of the ROD1 (left) and H(D) (right) cuts at the module vertices used to provide spacings for mechanical construction purposes.

with the non-uniform H(D). Our contribution is the analysis of various geometries based on which the final solution was derived. We have proposed our own ROD1(8,9) architecture (Figure 3.31), and it differs from the final H(D) solution in the following: the rotation of the TCs in the non-uniform scheme (120°) and the cuts provided at the SM vertices.

Considering the vertex cuts used for mechanical apertures, ROD1 provides it naturally in the module tessellation procedure when covering the detector sensing region. In order to increase the spacings, a module displacement is needed for ROD1, as shown on Figure 3.36, to avoid a large number of partial SCs generated at the module vertex. There are two possible vertex cuts shown, where the first one is more regular spacing (equilateral triangle), while the other is cut does not follow a regular shape, but less silicon is wasted because hexagon halves can be produced with the maximal silicon efficiency of a regular hexagon.

Shortly, it can be seen that ROD1 cut is the position of the module edges referenced to SCs, where module verices are positioned at the middle of the SC edges, resulting that module displacement is needed to keep SC plane smooth. It provides SC areas with less variations and provides some space for mechanical fixation of the modules. On the contrary, the H(D) cut is the position of the module edges referenced to SCs, where module vertices are positioned at SC vertices, resulting that natural tiling of SMs is possible. The modules tile well, and all good properties of ROD1 are preserved. However, all this is at the cost of more different SCs (especially pentagons) generated at the position of the vertex cut. In this case using ROD1 architecture may be better, having a minimal impact on the cells generated by applying a cut at the vertex.

The advantage of H(D) is that a SM remains a regular hexagon, and it was finally acquired as module architecture applied for HGCAL. The communication between boards that process the data of neighboring modules is completely avoided, meaning that the FE HGCROCs do not need to communicate when processing the TC data. This is makes H(D) a better option than ROD1. The avoided inter-module communication as well as the fact that the partial TC sums do not need to be calculated any more, is very significant for the trigger and it simplifies the FE design.

Chapter 4

Front-end data selection and the TPG architecture design

As described in Chapter 2, the first step of the data processing is performed with the on-detector ASICs in the FE electronics. The main goal is to reduce the amount of off-detector data that is sent to the next stage for further processing. There are many selection algorithms that can be used at the FE to select the high-energy TCs. In Section 4.1, we examine prior work on maximum-finder circuit implementations in hardware. We classify existing solutions depending on whether a single maximum is found or N highest energies are extracted. We design a selection algorithm, called Best-Choice Topology (BCT), and we show its performance when implemented in ASIC. Finally, we discuss the advantages and disadvantages when compared to the other solutions used in the trigger.

The second processing step in the trigger algorithm is related to the reconstruction in the BE, where the main process is the energy clustering performed by the off-detector FPGAs. There are two BE processing sub-steps, which are connected with optical links, and finally produce the HGCAL trigger primitives. We refer to the algorithm as the TPG, whose output is used together with primitives from the other sub-detectors in order to select higher-level objects such as electrons and photons. The formation of the clusters (TPs) can be done in two processing sub-steps, starting with a 2D clustering layer-by-layer, followed by a procedure that links 2D clusters into 3D clusters. The clustering can also be done directly in a single step, by using a full 3D information from the detector geometry. While providing a clear benefit on the reconstruction efficiency, it is expected to be more resource consuming in terms of hardware. This scenario is explored in Section 4.3, where we study possible TPG hardware architectures that can be used for a 3D clustering implementation at the L1 trigger.

4.1 Data reduction mechanisms in HGCAL

There are limitations in the available bandwidth and number of available links (10Gbps) when sending the data off-detector to FE stage. First, the number of bits that can be sent from one module is limited. For example, let us assume 160 bits processing element on the ASIC that reads the module data [71]. It means that the maximal data code word size that can be sent from a single module is 160 bits, which is not sufficient if there are 256 sensor cells on the module. Ideally, we would like to send as much data per module as possible, but obviously there should be some compromises with the hardware constraints, so that some kind of data reduction mechanism must be applied on the data transmitted from the FE. The same happens for the second stage, when the data is sent to the BE.

The first kind of reduction based on the geometry has already been explained in Chapter 3. It is the reduction of the off-detector data received in the FE, and it is based on the grouping procedure, such that groups of sensors or TCs are formed. Hence, instead of sending the data from each sensor, we send the data from a group of hexagonal sensors forming a tetrahex structure (a cluster of 4 sensors). Also, TCs are made from 9 sensor cells, depending on the detector region. Let us assume, for instance, that there are 256 sensors in the module, and that 64 groups are created by the procedure of forming tetrahex TCs, reducing the amount of data to be sent by a factor of 4. Moreover, a simple calculation can show that the transfer of 64 TC energies means about 2 bits per group ($\frac{160}{64} = 2.5$) in this example case, which is very small to code high energies.

The second kind of data reduction performed at the FE is the selection of N TCs for the further BE stage. One of the possible coding schemes is illustrated in Figure 4.1. For the low-energy TCs we do not have to send any data, while we encode the high-energy TCs with 8 bits. Thus, it is necessary to have 64 bits (one bit for each sensor group) indicating whether the group has the data or not, followed by the encoded energy bytes from the non-zero groups. This kind of technique is similar to a zero-suppression as it omits sending the TC energies where the energy is below the threshold. Also, with this strategy, there are $160 - 64 = 96$ bits (12 bytes) available for data transfer. It means that one byte is reserved for each group that correspond to a high energy, and we can send the energy data from at most 12 such groups. However, only $\frac{12}{64} \approx 19\%$ of the information from one module can be transmitted in this way.

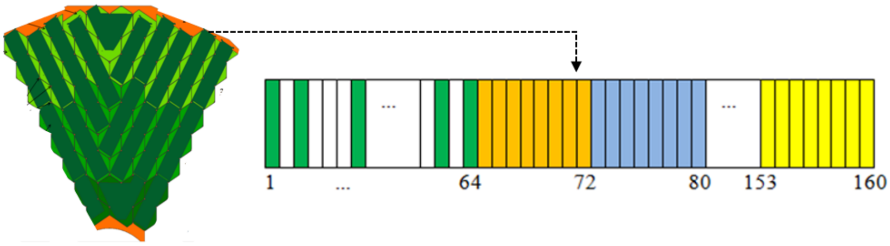


Figure 4.1: An example of the simplified coding scheme for data transfer.

Another kind of reduction is introduced, which increases the fraction of selected trigger data on the module.

It is the aggregation of data that originates from several consecutive BXs. The 40MHz working frequency means that BXs occur in the time interval of 25ns. In Phase-1, the latency of the trigger path was ≈4 microseconds, while for Phase-2, the latency of the full trigger rate is increased (12.5 microseconds). It means that the decision time limit is extended, so the total latency of the HGCAL trigger primitive generator is increased from 3.8, to 5 microseconds. This latency is the overall sum when adding the contributions from the electronics (FE, concentrator ASIC, serialization and de-serialization) and the contribution from the data processing in the trigger firmware. The fixed latency from the FE electronics and the transmission towards BE is ≈2.2 microseconds. This leads to the required ≈2.8 microseconds for the TPG algorithm to execute.

There is maximally 5 microseconds allowed to decide on which TC energies to keep or throw away in the HGCAL trigger. However, since the probability of "interesting" events with high energies is rather low, instead of sending the data for each BX, we can send the data aggregated over N BXs. In this way we actually select the N*12 highest energies from N events. Suppose, for instance, that we want to send the most significant data from 4 BXs ($N = 4$). We select 4*12=48 maximal energies from the aggregated set of BX data. Next, we fill-in the selection bits (zero or one) to indicate which TCs data we are sending the data. By means of aggregation, if something interesting happens in the first BX and nothing in the other three, then most of the data will originate from the first BX (as preferable). This is certainly the most significant BX for which most data is transferred.

Even though at the moment we do not have aggregation in the trigger, in case of the aggregated data from several BXs, the identification of the BX is kept by having TC addresses together with energies inside the aggregation. The addresses can contain both, the aggregation and the BX identification numbers. In general, we know which data correspond to which BX with synchronization patterns and BX counters in the data headers.

To summarize, it would be the best if each sensor data could be read-out such that all ≈6 million sensors data would be received at the BE part for further processing. In practice, a reduction is performed at the FE to fit within the bandwidth available to transfer the trigger data out of the detector. It is done in several ways, as visualized on Figure 4.2 [63]:

- The HGCAL sensor cells are grouped into larger trigger cells.
- The aggregation of N consecutive events is performed to get the collection of data to be transferred.
- The selection algorithm passes only the most energetic data.

It is to note that we elaborated only on a TC selection that is performed on the single module (either in one BX or in several consecutive BXs). However, there are other variants of data transfer organization. First, there is the selection of fraction of TCs from each of the modules. It is also possible to aggregate the data from several modules inside a single BX and perform the selection, or to calculate the total energy sum of each module and select the N modules with the highest energy. Then, all the TCs could be sent coming from these modules. In any case, there is a need for the design of an efficient selection algorithm in the hardware, which is given in Section 4.2.

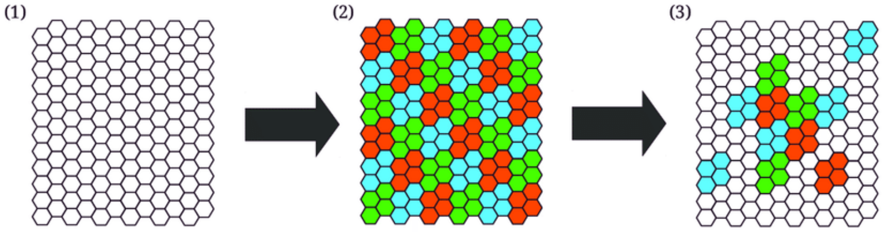


Figure 4.2: Trigger data reduction performed in the front-end ASICs in three steps; 1) read SC data, 2) form TC sums and 3) select only the most energetic TCs [63].

4.2 Trigger data selection algorithm

In this section, we describe the main functionality of the selection algorithms in the FE trigger block. We focus on the strategy selecting the highest energy TCs. First, in Section 4.2.1, we examine the existing state-of-the-art from the literature on maximum-finder circuits. We confront the solutions for the selection of only the highest value, as well as circuits selecting M largest values out of N. A possibility is to sort the data prior to extraction of the first M values, which may be more feasible for the dynamic variation of the M parameter in the selection, but also has its disadvantages. We design an efficient maximum-finder circuit that provides resource optimization towards the standard array-based topology (Section 4.2.2).

4.2.1 Background and related work

The general problem of finding a winner or a maximum value in an unsorted set can be defined as follows:

Given an unsorted set S of n elements, $S = \{D_0, D_1, \ldots, D_{n-1}\}$ where each element D_i is a k-bit unsigned binary number, the winner element D_{max} is a maximum binary element extracted from S if $\{D_{max} > D_i, \forall D_i \in S, i \neq max\}$.

The position of the winner element can be extracted in the form of a binary value or a one-hot binary address. The design of an efficient circuit solving the above-defined maximum-finding problem is a very important task. Depending on the type of application, circuits can be designed to produce only the value or only the address of the winner element. There are many applications that require both the value and address of the maximum element to be extracted [72, 73]. There are also applications that require only the fast computation of the maximum element value in a group of binary numbers [74].

Very few research papers on the maximum-finder algorithms have been published in the recent years. Yuce et al. [75] provide a detailed literature survey on this topic. They report on some sequential circuits that are synchronised with a clock, but target mostly the combinational circuits that provide the winner as soon as the input data is changed. The simplest one is the Array Topology (AT) based on a filtering concept where all candidates are examined in parallel from the most significant bit (MSB) to the least significant bit (LSB), progressively reducing the number of candidates at each bit-slice. The reduction is done by using the enable signal (enable=1), which becomes the disable signal when inverted (enable=0). It will disable the candidate that loses the chance to become a maximum when it has lost on a specific bit. As shown on Figure 4.3, the basic building block of the topology is the AT block and there are n AT blocks for one bit slice, one for each candidate in an input data array. There is also one AT block for each of the k bits.

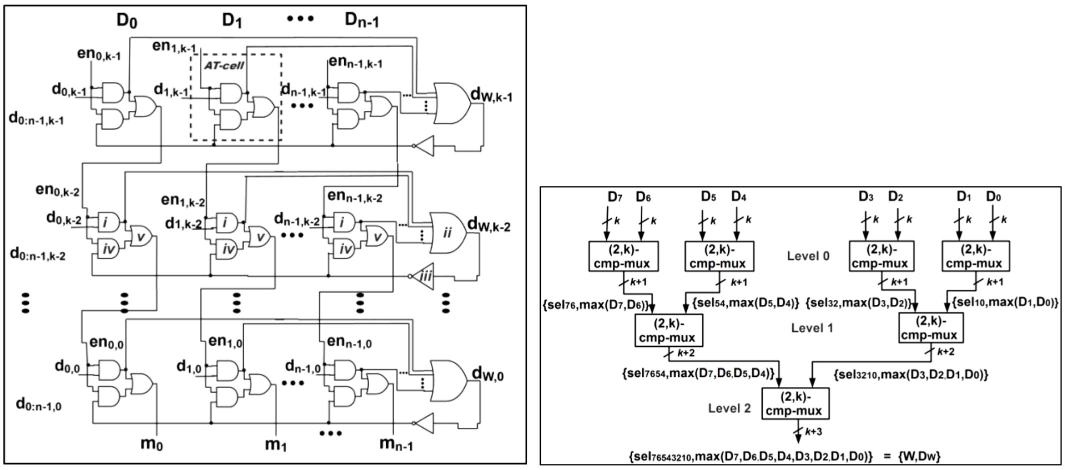


Figure 4.3: AT topology (left) and TBT concept (right) [75].

The AT works by first using the 2-input AND gate (i) to select the input data bits from n numbers that are enabled, and send the signals to the n-input OR gate (ii) providing the corresponding bit of the maximum. The inverted maximum data bit (iii) together with the 2-input AND gate (iv) ensure that even if all corresponding bits are zero, the result is still transferred to the next AT row, otherwise the enable signals would all be low. There are 2-input OR gate (v) that generate the enable signal. Depending on the input signals, low or high output is generated, enabling or disabling the current input bit. The advantage of this is that we can get the winner element just after the first level if only one candidate has a high MSB bit, as the AT is going to pass only the bits of that candidate. Besides the winner element itself, the AT also gives its corresponding address. The address is generated by using the enable signal that is propagated trough the bits of the individual candidate number. Value of the resulting address bit is high if the candidate was enabled for all bits, and it is low otherwise.

Yuce et al. also reported on the Traditional Binary Tree (TBT), a simple tree-based architecture most commonly used when solving the general maximum-finder problem. As it can be seen on Figure 4.3, simple blocks composed by a multiplexer driven by a comparator are connected in a binary tree configuration. The basic idea is to compare two by two the elements in the input array, and to propagate to the root of the tree the winner element as well as the selection bits giving its address.

The comparison of the two k-bit numbers is implemented by using a comparator circuit, where the selection bit is generated so to obtain the value of the greater element from the 2-to-1 multiplexer. Even though the TBT has a great advantage concerning the area, the authors in [75] emphasize the speed advantage of using Parallel

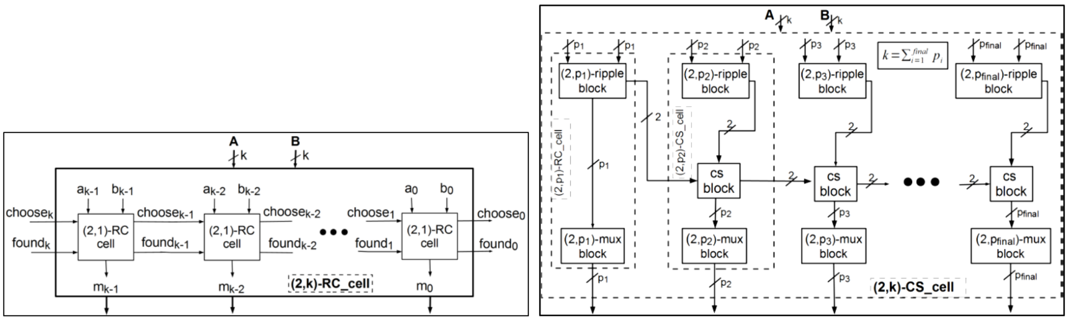


Figure 4.4: RCT (left) and CST (right) comparison scheme inside a TBT node [75].

Binary Tree (PBT). There are several variants of this topology such as the Ripple Carry (RCT) and the Carry Select (CST). The RCT is a PBT variant iteratively propagating the comparison result signals from the MSB to the LSB. The comparator-multiplexer block from the TBT for comparison between two k-bit numbers is replaced by the RCT node. It consists of k lower level elementary blocks connected serially. Each of them performs a two-bit comparison and generates the corresponding bit of the winner element. An example of RCT node comparing two bytes is given on Figure 4.4.

An elementary (2,1)-RC block generates "choose" and "found" signals that constitute the two-bit carry which is propagated to the next (2,1)-RC block. It also receives a carry from the previous (2,1)-RC block. the "choose" signal will become high if $a_i > b_i$, meaning that the first number of the two given to the RCT node is winning, and it is low otherwise. The "found" signal is high when the maximum is found. The RCT suffers from the ripple carry drawback, which means that the "choose" and "found" signals for the next RCT node cannot be generated until all the bits are compared (from the MSB to the LSB) inside the current RCT node. This is solved with the CST algorithm. As it is shown on the Figure 4.4, this topology has again k (2,1)-RC blocks to compare two k-bit numbers, but each of them is extended with a 2-to-1 multiplexer. There are k-1 CS blocks as well, and every (2,1)-RC block except the one comparing the two MSB bits is connected to its CS block. The idea is that each of the (2,1)-RC blocks can operate in parallel, producing the two-bit carry result. The CS block takes the carry signals from the current (2,1)-RC block and the previous CS block to select the corresponding bit of the winner.

The authors in [76] propose their Array Based Topology (ABT) solution, as well some even more efficient circuits that use the ABT as a basic building block. The ABT is based on the idea to produce a nxn matrix of result signals by comparing every pair of k-bit input data elements. The comparison is done bit wise in parallel, and every pair of input data bits is compared. The comparison result is given in a form of a triangular matrix of signals, where the element (i, j) is high if $D_i > D_j$ and is low otherwise. The one-hot address is generated based on that output. The maximum element value is extracted from the input set of numbers based on its address. There is a built-in priority scheme in case several maximums have the same value.

Kathirvel et al. implemented an efficient topology to find a maximum value in a set of binary numbers, which they call maximum magnitude generator (MaxMG) [77]. Their design is a combinational circuit that does not require any

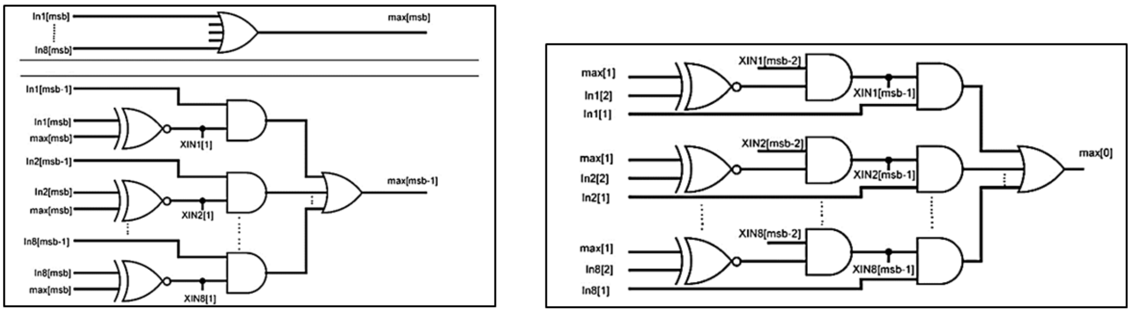


Figure 4.5: Logical design of the MaxMG topology [77].

sequential elements such as registers or flip-flops, and it is shown to be more efficient compared with the existing designs in terms of area, power and delay. The MaxMG works as a filter comparing all input data bits in parallel. It consists of three main levels of logic gates, as shown on Figure 4.5. A first level extracts the MSB of the maximum element by using OR gates. A second level uses XOR gates to compare the MSB of all input data elements with the previous level result (MSB bit). Input data elements that have the same MSB as the result one are potential maximums. The high signal generated from the second level is sent to the third level, where the XOR gate output is multiplied by each of the input data elements MSB. This level has a functionality of a filter, providing a signal if the current element is still a potential maximum or excluding it from the competition. Namely, an AND gate output is transferred to the AND gate at the next level for the calculation of the next bit of the maximum element. If the output of the AND gate is low, the number is excluded from the maximum element competition. If all input data elements have the MBS low, the next XOR gate will have a high output, so that the AND gate decide which number to choose.

The leading zero count (LZC) circuit represents an optimized circuit architecture that is easily implemented in parallel, and it is more efficient in area and delay consumption compared to a parallel comparator tree[78]. This architecture consists of two phases. In the first phase, each of the n candidates is encoded with a code word consisting of only one high bit positioned at the specific place from the left. For instance, if a candidate is 0010 (a binary number 2 expressed in 4 bits), then the corresponding code is 0010_0000_0000_0000 (16 bits) and the high bit is positioned at the second place from the left (the counting starts from zero). If a candidate is 0100 (a binary number 2 expressed in 4 bits), then the corresponding code is 0000_1000_0000_0000 (16 bits). In each code, the position of the high bit means the value of the number. When the number of input data is n, each has its own code and if the length of the input data is m bits, then the length of the coded variant is 2^m. In the second phase, a logical OR operation is performed on the bits between the generated codes and a vector is created. We count the first position of the high bit (looking from the LSB) in the vector according to the leading zero counting algorithm. The result is the value of the maximal number.

There are some other tree-like topologies found in the literature, designed to find the first two maximum or minimum values in an unsorted set of elements. One solution relies on the sorting algorithm that determines both the first and the second minimum with high efficiency [79, 80]. A more generalized architecture is designed in

[81], providing a parallel solution that relies on the sorting approach for finding the first three or more maximum or minimum values in a set. There is also an implementation of a fast sorting-based architecture that extracts the k^{th} best value in an unsorted list of elements, where the ranking position k is generic and can vary from 1 to the length of the data set [82].

4.2.2 The proposed maximum-finder algorithm design

This section is based on our published paper [83], where we propose e a new maximum-finder BCT design. It is an optimized version of the standard array topology. The usual bit-by-bit parallel comparison is applied to extract the maximum and its one-hot address having the positions of the winner element marked with the high bit.

The main BCT concept

The solution to solve the maximum-finder problem in the proposed BCT is defined as a filtering approach. A bit-by-bit comparison of all candidates in parallel from the MSB to the LSB is applied, progressively reducing the number of candidates for the winner element. The BCT functionality can be approximated with a self-organizing binary network that stabilizes when the maximum is found and settled on the result bus. This self-stabilizing concept is accomplished by using only the feedback coming from the bit-level comparisons. For example, if the input element receives a negative feedback on a bit-level comparison, it will exclude itself from further winner competition. On the other hand, if the element is high enough to become the winner, a positive feedback is generated so that the algorithm can proceed in its maximum finding goal. Therefore, the BCT logic can be summarized in two main aspects:

- The comparison is done bit-by-bit in parallel from the MSB to the LSB to extract the winner element. This concept is like the one used in the AT, but the BCT uses an improved self-exclusion technique for the bit of the input element as well as fast propagation of the winner decision towards the LSB.
- A feedback is implemented by a result (OR) bus where the winner bits are generated. It is important that this bus is unique and independent of the size of the input data set. This means that there is no additional decision logic apart from the OR bus realization.

Due to the main winner competition that is performed on the lowest bit level (Figure 4.6), if each of the input elements loses on the specific bit during the winner competition, it will be excluded from the further comparison.

BCT optimization compared to AT

The BCT comparison starts from the basic Single Bit (SB) blocks working in parallel. The SB comparator module consists of two AND gates, one OR gate and one inverter, as shown on Figure 4.7. The inputs to the SB block are

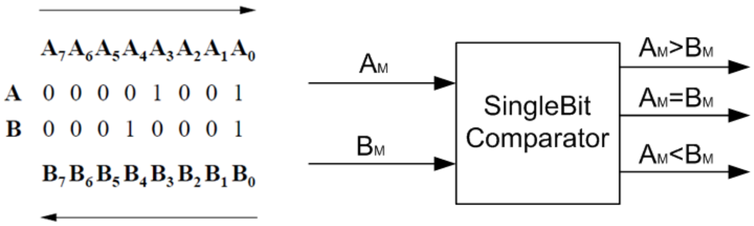


Figure 4.6: The bit wise comparison between two binary numbers $A = 9$ and $B = 11$.

denoted as signals d (data bit) and r (result bit), because r is a bit on the result bus (the bit of the current maximum) and d is a current input data bit (the bit of the current candidate).

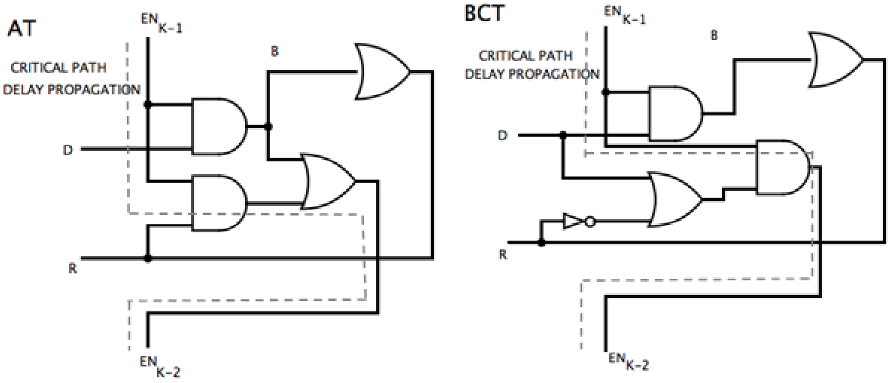
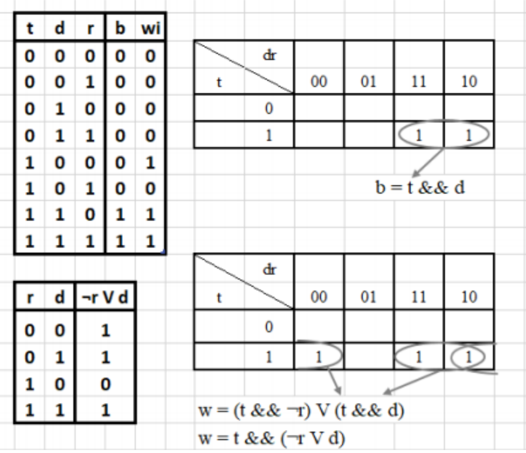

AT
CRITICAL PATH DELAY PROPAGATION
ENK-1
B
D
R
ENK-2
BCT
CRITICAL PATH DELAY PROPAGATION
ENK-1
B
D
R
ENK-2

Figure 4.7: AT and BCT SB block design for bit-wise comparison. The critical path delay is shorter for the BCT, so that an optimization is accomplished.

The comparison SB block output is marked w_i and it becomes the enable signal t for the next bit. The signal t will be high and the current input element is enabled while $d \geq r$. When $d < r$, it means that the candidate has lost on a specific bit and t becomes low (Figure 4.8). This disables the candidate from further competition since the output signal b will be zero for the current data bit and all further bits up to the LSB.



t	d	r	b	wi
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	1	1
1	1	1	1	1

t \ dr	00	01	11	10
0				
1			1	1

b = t && d

r	d	¬r V d
0	0	1
0	1	1
1	0	0
1	1	1

t \ dr	00	01	11	10
0				
1	1		1	1

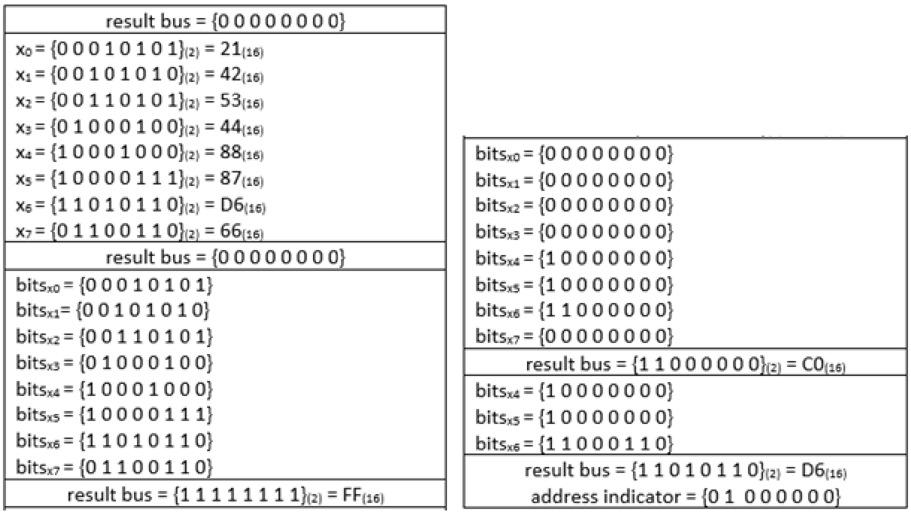
w = (t && ¬r) V (t && d)
w = t && (¬r V d)

Figure 4.8: Boolean expressions for the BCT SB block design.

The BCT filtering concept is similar but more efficient than the AT implementation. In an AT SB block, the next bit enable signal cannot be calculated before calculating previously the bit of the current maximum. This will cause one additional gate delay before calculating the winner bit. However, in the BCT, the enable signal for the next bit can be calculated immediately from the data bit, which means that the maximum calculation can start as soon as the input data becomes available. This allows to calculate faster the result element bits in the BCT with respect to the traditional AT. The BCT circuit area is foreseen to be the same as that for the AT, since there is the same number of logic gates at the lowest level of the SB comparators.

The BCT maximum finder example

The BCT topology network stabilizes when the winner is extracted on the result bus. This means that there will be several result bus configurations that will change during the winner competition. An example is given on Figure 4.9. Initially, the result bus is set to zero, so that all input elements are winner candidates. This will force the negative bit of the result to be high, enabling the enable signal generation. When the k-bit results of the SB comparisons are calculated for each of the input elements, an array of k n-input OR gates is used for the bit-wise extraction of the temporary maximum bits on the result bus. This temporary result bus configuration forces each of the elements to compete by giving its high bits to the bus.



result bus = {0 0 0 0 0 0 0 0}

$x_0 = \{0\,0\,0\,1\,0\,1\,0\,1\}_{(2)} = 21_{(16)}$
$x_1 = \{0\,0\,1\,0\,1\,0\,1\,0\}_{(2)} = 42_{(16)}$
$x_2 = \{0\,0\,1\,1\,0\,1\,0\,1\}_{(2)} = 53_{(16)}$
$x_3 = \{0\,1\,0\,0\,0\,1\,0\,0\}_{(2)} = 44_{(16)}$
$x_4 = \{1\,0\,0\,0\,1\,0\,0\,0\}_{(2)} = 88_{(16)}$
$x_5 = \{1\,0\,0\,0\,0\,1\,1\,1\}_{(2)} = 87_{(16)}$
$x_6 = \{1\,1\,0\,1\,0\,1\,1\,0\}_{(2)} = D6_{(16)}$
$x_7 = \{0\,1\,1\,0\,0\,1\,1\,0\}_{(2)} = 66_{(16)}$

result bus = {0 0 0 0 0 0 0 0}

$bits_{x0} = \{0\,0\,0\,1\,0\,1\,0\,1\}$
$bits_{x1} = \{0\,0\,1\,0\,1\,0\,1\,0\}$
$bits_{x2} = \{0\,0\,1\,1\,0\,1\,0\,1\}$
$bits_{x3} = \{0\,1\,0\,0\,0\,1\,0\,0\}$
$bits_{x4} = \{1\,0\,0\,0\,1\,0\,0\,0\}$
$bits_{x5} = \{1\,0\,0\,0\,0\,1\,1\,1\}$
$bits_{x6} = \{1\,1\,0\,1\,0\,1\,1\,0\}$
$bits_{x7} = \{0\,1\,1\,0\,0\,1\,1\,0\}$

result bus = $\{1\,1\,1\,1\,1\,1\,1\,1\}_{(2)} = FF_{(16)}$

$bits_{x0} = \{0\,0\,0\,0\,0\,0\,0\,0\}$
$bits_{x1} = \{0\,0\,0\,0\,0\,0\,0\,0\}$
$bits_{x2} = \{0\,0\,0\,0\,0\,0\,0\,0\}$
$bits_{x3} = \{0\,0\,0\,0\,0\,0\,0\,0\}$
$bits_{x4} = \{1\,0\,0\,0\,0\,0\,0\,0\}$
$bits_{x5} = \{1\,0\,0\,0\,0\,0\,0\,0\}$
$bits_{x6} = \{1\,1\,0\,0\,0\,0\,0\,0\}$
$bits_{x7} = \{0\,0\,0\,0\,0\,0\,0\,0\}$

result bus = $\{1\,1\,0\,0\,0\,0\,0\,0\}_{(2)} = C0_{(16)}$

$bits_{x4} = \{1\,0\,0\,0\,0\,0\,0\,0\}$
$bits_{x5} = \{1\,0\,0\,0\,0\,0\,0\,0\}$
$bits_{x6} = \{1\,1\,0\,0\,0\,1\,1\,0\}$

result bus = $\{1\,1\,0\,1\,0\,1\,1\,0\}_{(2)} = D6_{(16)}$
address indicator = {0 1 0 0 0 0 0 0}

Figure 4.9: Example for maximum extraction BCT.

The result is obtained when all candidates have lost the competition but the winner element that is larger or equal to the temporary result bus configuration. This configuration is generated by doing an OR operation between the k-bit result of the bits from each element. These are higher or equal to the winner, so that all maximum bits are extracted correctly after the OR operation. One-of-n binary address of the maximum is generated from the enable bits, where the address bits are set low for all candidates but the winner position.

4.2.3 Verification of the algorithm functionality for the trigger

Table 4.1 gives a brief comparison of the algorithms for the selection of the maximum element in the data set with respect to several parameters. It is analyzed whether the architecture generates the address of the selected winner together with its value, or whether the maximum value extraction depends on the generated address. For instance, the ABT and LZC architectures depend on the address because it is used to extract the value of the maximum. It is important that such architecture has a built-in priority logic, which means that for more than one maximum or minimum in the input data, a single one is selected based on a predefined priority scheme.

Table 4.1: Comparison between the selection algorithms.

Max-finder algorithm	Address extracted	Maximum value selection depends on extracted address	Priority logic added in case of few identical maximums
AT	+	-	-
TBT	+	-	+
RCT	+	-	+
CST	+	-	+
ABT	+	+	+
MaxMG	-	-	-
LZC	+	+	+
BCT	+	-	-

The array-based architectures such as AT, BCT or MaxMG do not have a built-in priority scheme, which means that a value of the maximum is extracted correctly, unlike its address. Namely, the address is a vector that has the same length as the input data set. If there is a single maximum in the input, the generated address will be a one-hot vector or one-of-N, with a single high bit located at the position of the maximum. On the contrary, in the case of several identical maximums, the address vector will have a high bit in all positions where the multiple maximum values are located. Such vector needs to be filtered so that it becomes one-hot. For example, it can keep only a single high bit located in the MSB position in the binary vector, so that only a single maximum is selected.

Also, the algorithms listed in Table 4.1 are combinational circuits, meaning that they provide the output result as soon as the input data is changed, unlike sequential or multi-cycle designs, which are synchronized with the working clock. For example, the value of the maximum of length k can be extracted bit-by-bit in k clock cycles [76].

Verilog simulation results

The L1 trigger application in the FE concentrator ASIC requires that the M largest numbers or maximums must be selected from the input data set with N energy values. The tree-based maximum-finders and the array-based maximum-finders were implemented in [84, 83], and circuits are designed using the Verilog hardware description language (HDL). Their functionality was verified by the simulation performed with the Xilinx ISE Design Suite 14.5 development environment [85].

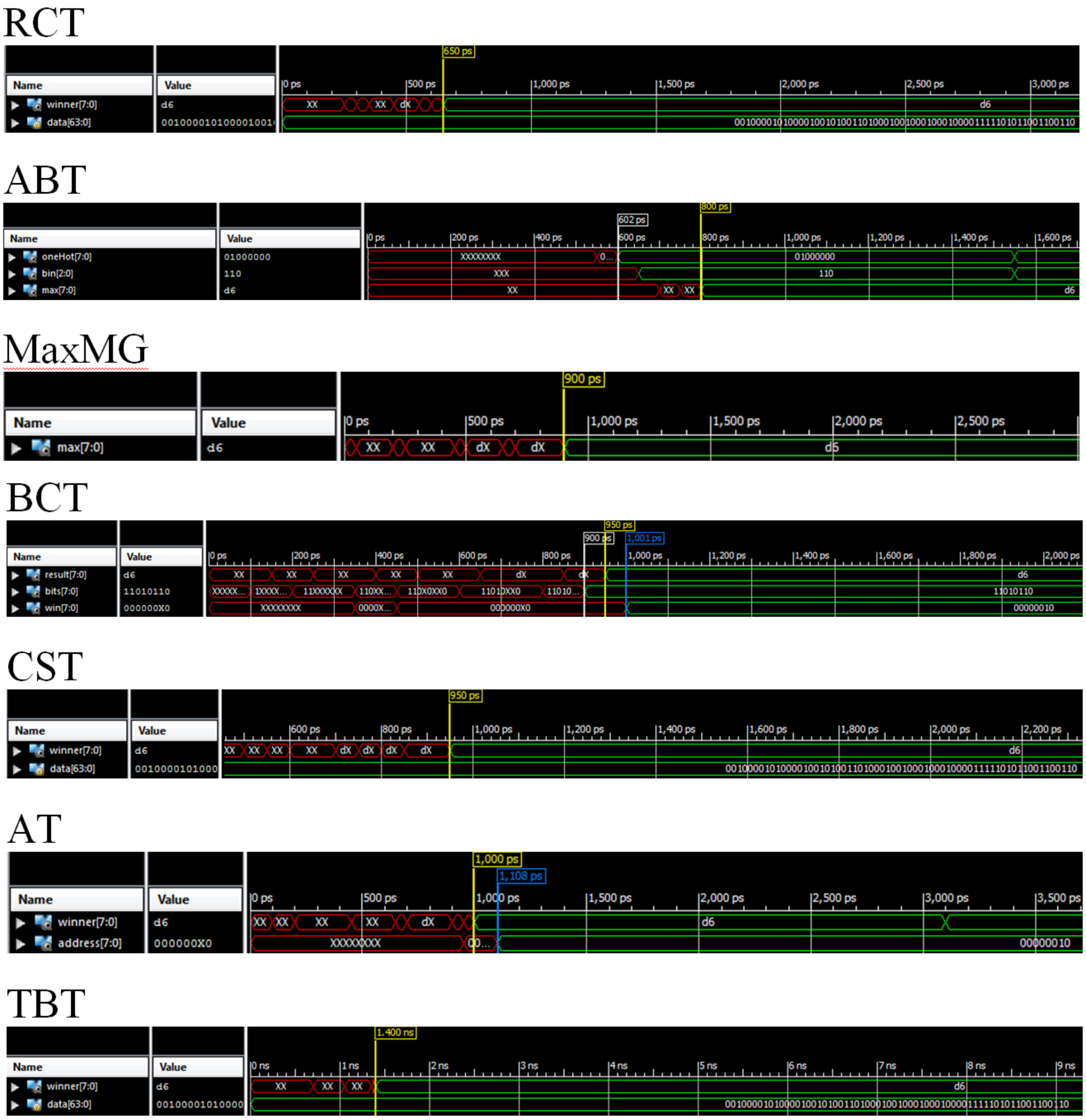


Figure 4.10: Simulation results for the $k = 8$ bits winner competition between $n = 8$ elements. The functionality of the selected maximum-finder architectures is verified in the presented simulation results, and the timing needed to extract the maximum is marked with a yellow flag (the example of the inputs is taken from Figure 4.9).

The simulation results for the winner competition between $n = 8$ input elements (each of them $k = 8$ bits wide), are presented on Figure 4.10. The input data set example is the one from Figure 4.9. As shown in the simulation, it is possible to extract a single maximum value with each of the considered circuits, and the timing performance is satisfactory ($t < 2ns$). In Section 4.1, we mentioned the simple experimental test case where $M = 48$ and $N = 256$, or 48 TCs are selected from the aggregated data of 4BXs (64 energies per BX). Based on that example, and the simulation results, we can assume that 48 winners can be successfully extracted with each of the given circuits in an upper timing limitation of $2 * 48 = 96$ ns. This is exactly needed for 4BX aggregation duration ($4 * 25 = 100$ ns). Hence, the design of a pipelined structure is enabled, which is common in high-energy physics (HEP) experiments [86, 87]. Namely, the pipeline will permit that the extraction of the maximums starts executing in parallel while the data is aggregated for the processing of the next 4BX energies.

The sequential extraction of 48 winners from 256 in 500MHz clock cycle is presented on Figure 4.11. The extraction of the maximal TC energy in each cycle is followed by the binary address generation in a form of a vector with indicator bits. We select 1 out of 256 in each clock period so that that the length of the vector is 256 bits. This vector will have multiple high bits when several identical maximums are present. We have added the priority logic that will filter out the winner position result to make it one-hot and to return only a single maximum at the leftmost MSB position. This one is disabled from the maximum-finder competition in the next clock cycle. The final indication bits are extracted with the positions of all the extracted maximums at the end of another 4BX aggregation. It should have 48 high bits indicating whether the input TC energy from the input data set is selected or not. The simulation result for selecting 48 energies out of 256 is given on Figure 4.11.

ASIC synthesis results

In the previous section, a simulation verification is shown for the maximum-finder selection functionality. Here we summarize the timing and area results when the selected maximum-finder circuits are implemented in ASICs (using the Cadence Genus synthesis tool [89]). We compare the the resources (area and timing) required to extract a single maximum by using the tree-based and the array-based selection approach [84, 83]. The main goal is to compare the proposed array-based BCT from Section 4.2.2 [83] to other competitors in [84, 83] and to discuss the possible trigger application. The results are summarized for the comparison of n k-bit input values in an unsorted set of elements where n=8,16,32,64 and k=8,16, i.e. for energies with $k = 8$ and $k = 16$ bits coding (Figure 4.12).

Based on the summarized results, we can roughly estimate the timing with each of the implemented circuits for the selection of TCs among 256 values (Table 4.2). It can be concluded that the BCT is the most efficient in timing when compared to the array topologies like MaxMG and AT, while all three architectures are less efficient in timing than the tree-based TBT, RCT and CST. However, the advantage of the array-based approach is that they are less expensive in area (lower number of logic gates used). Nevertheless, the estimated result in timing shows that in 4BX aggregation of $t = 100$ ns we can select more TCs with the tree-based architectures, especially with the parallel

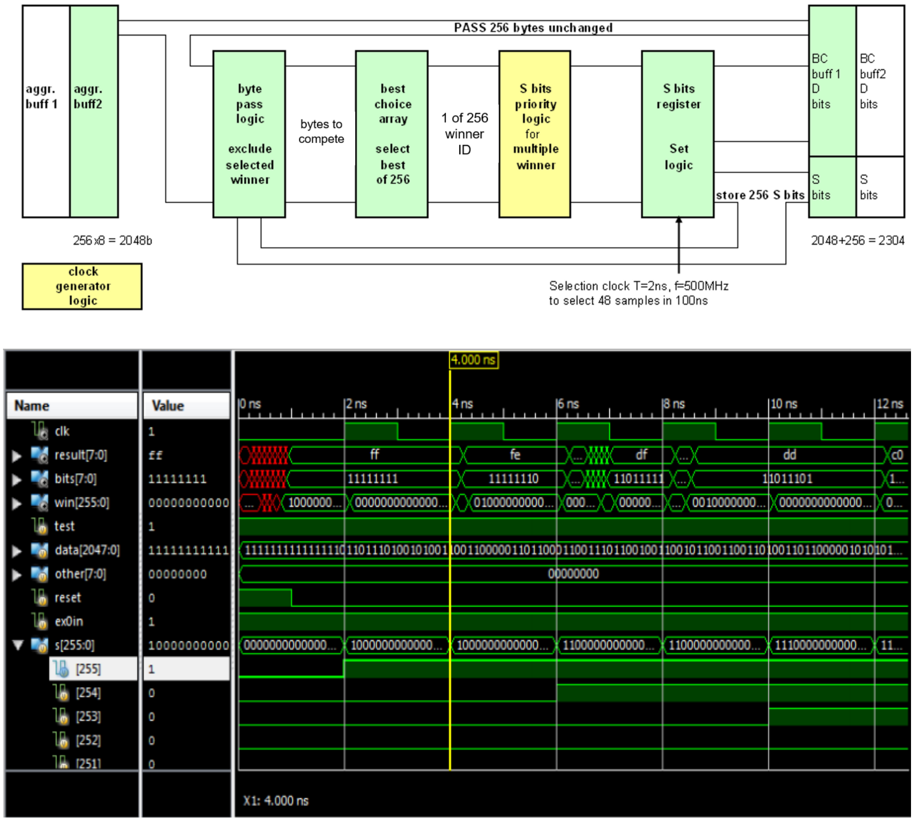


Figure 4.11: BCT logic overview (left) and the simulation result for 256 energy bytes selection with the indicator bits (right). The example input data bytes in hexadecimal format are FF_FE_DD_29_98...A6_C1_6B_DD_DF_FF [88].

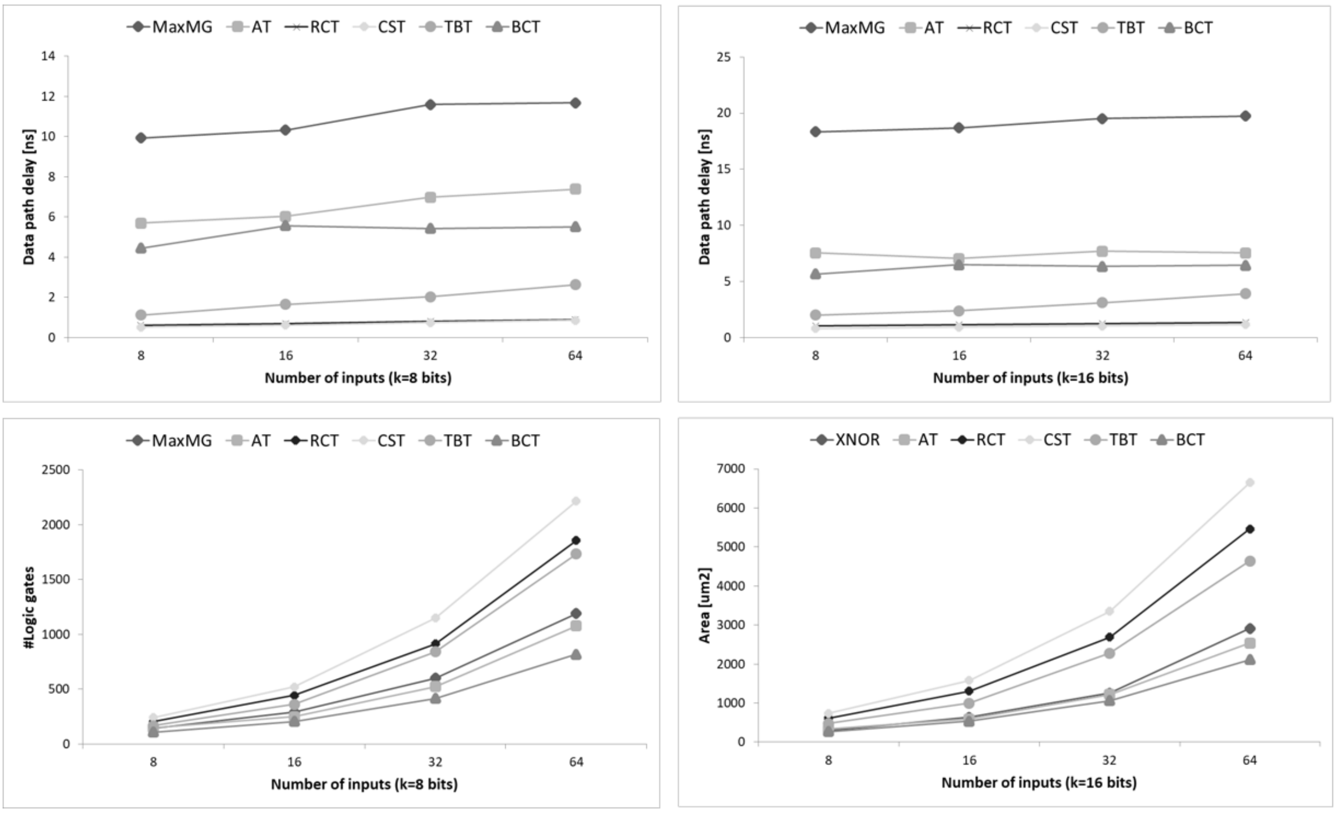


Figure 4.12: ASIC synthesis results for the implemented designs with TC energies $k = 8$ bits (left) and $k = 16$ bits (right). The x-axis shows the variation of the total number of inputs ($n = 8, 16, 32, 64$). The timing results are shown in the upper figures and the area result is given with the two figures down.

Table 4.2: Timing result for 1 of N selection. ASIC critical data path delay [ns].

	(k,n)	MaxMG	AT	RCT	CST	TBT	BCT
measured	(8,8)	9.93	5.69	0.6	0.51	1.12	4.44
	(8,16)	10.32	6.03	0.7	0.62	1.65	5.57
	(8,32)	11.59	6.98	0.80	0.75	2.02	5.42
	(8,64)	11.67	7.38	0.9	0.86	2.63	5.51
estimated	(8,128)	12.51	8.03	0.99	0.97	3.08	5.99
	(8,256)	13.16	8.63	1.1	1.09	3.57	6.30
measured	(16,8)	18.33	7.54	1.04	0.80	2.01	5.67
	(16,16)	18.70	7.03	1.14	0.93	2.38	6.49
	(16,32)	19.50	7.70	1.24	1.03	3.1	6.35
	(16,64)	19.73	7.54	1.34	1.15	3.91	6.44
estimated	(16,128)	20.32	7.62	1.43	1.26	4.46	6.78
	(16,256)	20.82	7.69	1.53	1.38	5.1	6.99

variants RCT and CST. For example, if there are 256 TCs in the input data set, we can select $M \approx 90$ TCs coded with $k = 8$ bits and $M \approx 70$ TCs coded with $k = 16$ bits in 100ns with RCT. Also, we can select on average $M \approx 25$ TCs with both TC coding schemes and the TBT algorithm. Concerning the BCT, it enables a selection of $M < 20$ TCs in 100ns, while other array-based competitors provide a smaller number of TCs ($M \approx 10$ for AT and $M < 10$ for MaxMG).

4.2.4 The sorting network implementation in hardware

The general problem of finding the M largest numbers from N inputs in an unsorted set of elements does not require the design of dedicated hardware solutions. The same functionality can be obtained by sorting the whole set and selecting the first M sorted values [86, 90]. Many efficient sorting algorithms have been designed in hardware, and various hardware solutions have been optimized for the implementation of sorting networks (SN) [91, 92].

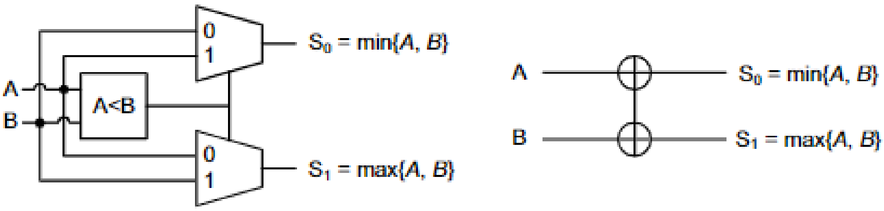


Figure 4.13: Details of the comparison element in the SN, where the CAE block has two inputs and two outputs [86].

A SN is a grid of interconnected compare-and-exchange (CAE) blocks that receive two inputs. They consist of a comparator circuit that compares the inputs and multiplexers that pass trough a set of outputs. The details on the SN CAE are illustrated on Figure 4.13. If the inputs are already in order, they are passed to the outputs, and otherwise, they are rearranged. The advantage of SNs is their simple implementation in hardware, by using parallel solutions, where a pipelined approach is used to increase the data throughput [86].

4.2.5 Verification of the SN selection used for the trigger

To the contrary of the proposed array-based BCT, the sorting-based BCT design was examined in [93, 94]. The Batcher odd-even mergesort algorithm approach [95] from Figure 4.14 is applied, adjusted for sorting 48 TCs. The choice of this algorithm is motivated by the fact that is one of the fastest sorting algorithms known, and it can be easily paralellized and pipelined for an optimal hardware implementation. The Cadence Genus synthesis results show that the latency of the sorter with 48 inputs and 48 outputs (coded with 18 bits), designed as a 1-stage pipeline, is 24.43ns or $2 * 24.43 = 48.86$ns in 2BX latency. The total number of logic gates and the total power of the synthesized circuit is 47 322 and 17.58mW, respectively.

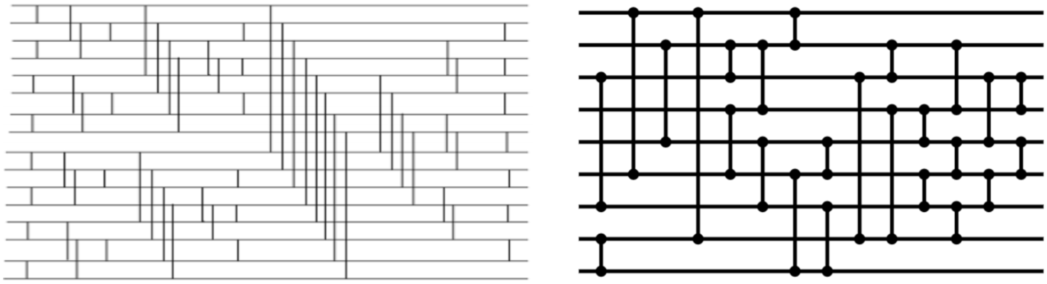


Figure 4.14: SN design for Batcher odd-even mergesort (left) with $N = 16$ inputs and the minimal SN with $N = 9$ inputs (right) [93, 96].

Intuitively, sorting the whole set for extracting a single maximum value is not feasible when a high throughput and a low latency are the most important requirements. Since in HGCAL the goal is to select the M largest numbers from N inputs, it is interesting to examine the efficiency of sorting approach applied for the selection of M highest values.

Table 4.3: Synthesis result for the SN implementation.

	M winners selected from $n = 9$ elements								
	1	2	3	4	5	6	7	8	9
#Logic gates	452	1092	1536	1568	1840	1872	1904	1936	1968
Delay [ns]	9.863	14.992	15.448	15.484	15.593	15.593	15.593	15.593	15.593

The results of a small study in [97] is shown in the following, to examine how this can be accomplished by implementing a SN in hardware, which compares $n = 9$ binary numbers of $k = 8$ bits. The minimal SN with respect to the odd-even merge sort design is used from [96]. In addition, the further optimization of the SN is possible by using less comparators in the SN design, removing the comparators that are not needed when a specific number of winners is selected. It means that the complexity would grow proportional to the parameter M because more and more elements are selected so that more values have to be sorted in the data set.

Examples of comparator reductions are shown on Figure 4.15, where it can be seen that one needs to use only 10 out of 25 comparators (40%) to extract a single element. For the selection of the first two maximums we need

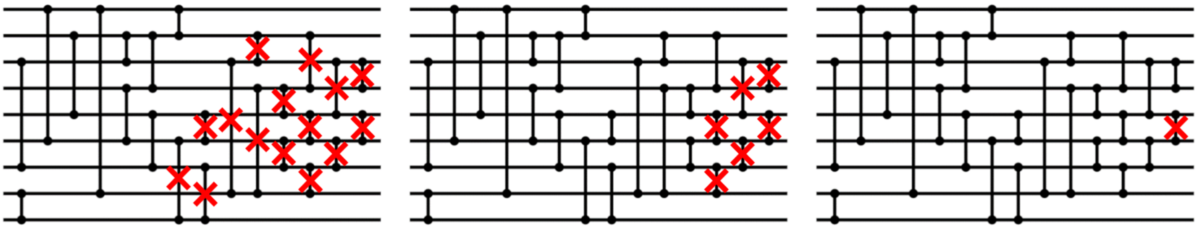


Figure 4.15: Reduced comparators in SN selecting M winners. $M = 1$ (left), $M = 2$ (middle) and $M = 3, 4$ (right).

19 comparators (76%). It is necessary to use 24 comparators for the selection of three and four winners, while all 25 comparators must be used for the remaining selections. The synthesis results are given in Table 4.3, presented for 9 cases with $1 \leq M \leq 9$. The SN was first applied to select the first element or the maximum of the data set, then for the selection of the first two maximums, first three maximums and so on until the entire data set is sorted. We have compared this SN results as a sorting-based approach to the selection, towards the array-based BCT. The result is given on Figure 4.16 and it is described in Section 4.2.6.

4.2.6 Discussion and conclusion

To summarize the study on the trigger data selection, we can derive the following conclusions. First, the existing topologies implemented from the literature have shown a good potential to be applied for the trigger. The results are especially motivating for the tree-based structures being more efficient than the array-based logical designs. It is also to note that the parallel tree-based variant such as the RCT is faster than the basic TBT, while the CST is even more timing efficient. However, the tree-based architectures are more expensive in area with a larger number of logic gates in the synthesized circuit [75].

Concerning the comparison of the proposed BCT towards the other circuits from [84, 83], it is shown to be the best among its array-based competitors (AT and MaxMG). Indeed, it is an optimized array-based maximum-finder circuit which enables the faster extraction of the winner element by using a competition concept with the parallel bit wise comparisons between binary numbers. Hence, if an array-based circuit was used for a specific application, our BCT design would be the best option to go with due to its timing efficiency, and even a better result is possible with the tree-based variant. However, for some signal processing application where the circuit area is the most important requirement, the BCT would be the best to use among all.

When comparing the array-based BCT to the SN approach from [97] with the reduced comparators (Figure 4.16), it is confirmed that a poorer timing is accomplished when selecting a single maximum value with the SN compared to the maximum-finder circuits. Also, the timing efficiency is better for BCT when the number of the selected elements M is low ($M \leq 2$). Using the sorting approach is more efficient to select M elements for $M > 2$ than BCT, since we can extract several maximums faster. Also, we can see on Figure 4.16 that the increase of the latency for the SN is not linear like in BCT. The area result of the synthesized BCT circuit is better (with less than 500 logic gates). In

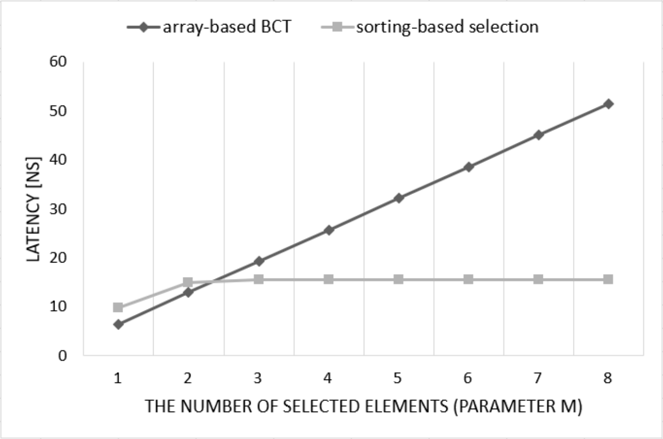


Figure 4.16: Comparison between the array-based BCT and the sorting-based selection with the reduced number of comparators.

the synthesized SN, the area increase is large, especially when we include more comparators for selecting a larger number of energies.

Concerning the array-based BCT comparison to the work from from [93, 94], let use describe the difference in the number of inputs assumed. In the array-based approach, we have shown the two experimental cases; one where we selected 48 TCs out of 256 energies aggregated from the 4BXs (64 TCs per module assumed), and one where a single TC energy is selected out of 8, and each energy value is coded with various number of bits $k = 8, 16, 32, 64$. The work on the sorting-based selection assumes a single BX without aggregation and 48 TCs coded with 18 bits and coming from the selected H(D) module described in Section 3. In order to obtain a fair comparison, we compare the synthesis results of the array-based BCT with 64 inputs (coded with 16 bits), and the sorting-based approach with 48 inputs (coded with 18 bits).

The result for the experimental case (16, 64) from Table 4.2 shows that we can select a single element with the latency 6.44ns by using the array-based BCT. In order to select M of these elements, we must go sequentially; choose one out of 64, then another one out of the remaining data set, then another one etc. This means that for 48 elements selection the latency is approximately $48 * 6.44 = 309.1$ns. On the contrary, with the proposed sorting-based approach from [93, 94], the latency of sorting 48 elements is 24.43ns (for a 1-stage pipeline), which is much faster. If we sort the whole set, we will spend more time in the hardware, but that is the most resource-consuming part, after which we can select any number of elements. We can conclude that it is more efficient in terms of latency to sort and select, than to select sequentially in several clock cycles. The sorting-based approach is finally taken for ECON-T design.

Let us discuss the advantages and the disadvantages of both approaches (BCT and SN) concerning the channel capacity. In this case, the array-based BCT is more convenient, especially for larger number of selected elements. We have shown that BCT approach includes the addressing scheme with the selection bits, where zeros and ones indicate whether the TC is sent or not. On the contrary, in the sorting approach, the fact that the input data is sorted implies that the order of data is changed, so that the relative relationship among the TC positions in the input data set is completely different. Since the BE processing of the received data from the FE includes a clustering in the reconstruction algorithms, and requires that we know exactly which TC belongs to which module, we need to transfer not only the TC energies but the TC addresses as well. These must be coded with additional number of bits, so more data is transferred.

Since the available bandwidth is limited, we can spare the bits that would be spent on data addressing (in sorting approach) and more bits can be used for the transfer of the actual energies with the BCT selection. Also, there is no sorting, so the relative relationship among TCs remains the same. The ordering of the transferred TCs is mapped to the same ordering of the corresponding high bits in the BCT selection vector transferred from the FE. This enables that the selection is correct on the BE side. The concept of the addressing with the indicator bits is used in the trigger, in case when the larger amount of data is selected. Hence, in the regions on the motherboard with the low occupancy, from which the smaller number of TC energies is selected ($N_{TC} < 8$), the 6 bits coded address is sent along the TC data. Complementary, in the high occupancy motherboard regions ($8 < N_{TC} < 48$), 48 selection bits are sent together with the selected energies [98].

There is another selection approach that is not examined in the context of this thesis, and that is the selection of all TCs above a certain threshold instead of the M highest energy TCs. Unlike BCT, the TC-over-threshold strategy cannot guarantee that exactly M TCs will be selected, since it is not known in advance how many TCs will pass the threshold. It can even happen that a single TC is selected, depending on the chosen threshold value. On the contrary, BCT guarantees that exactly M TCs are selected, with the compromise that the algorithm has to perform sequentially, extracting one TC per each clock cycle. The two approaches (threshold and BCT) are not compared in hardware, but a quick theoretical estimation can be made. For the timing requirements, the TC over threshold selection can be done in one cycle, since all candidates can be compared to the threshold in parallel. It means that the critical part delay is the timing needed for the comparison of two binary numbers. The drawback is that we do not know how many TCs to expect exactly in the BE algorithm, which can put additional complexity in the design of the trigger architecture. Also, the threshold selection needs buffers in the FE and in the BE, so the latency of the threshold algorithm is expected to be larger when used in the trigger. It is a compromise between both the resource consumption and the physics efficiency that put requirements for the selection of the final design solution.

4.3 TPG architectures for a 3D clustering at the L1 trigger

The trigger signals are generated from the detector sensor data readout chain and the selection is performed with the procedure illustrated on Figure 4.2, where only partial TC data is received on the BE. In order to decide whether the input data is "interesting" or not, it is necessary to reconstruct the event from the partial information on the TC energies. The first step towards the reconstruction is the clustering of signals, with the final goal to reconstruct a 3D shower. The baseline strategy for the clustering and reconstruction at the time of the studies was done in two steps: first a 2D clustering was performed layer-by-layer, after which 3D clusters were formed by merging the 2D clusters.

In this section, we examine architectures that can enable a direct 3D clustering. The main problem is that we need data to be projective in depth. Unlike in the baseline reconstruction strategy, where the whole end-cap layer data was fit into a single FPGA board (which was needed in order to perform the 2D clustering on the layer), it is not possible to fit all HGCAL layers into a single FPGA. Hence, we must have a sectorized view of the end-cap layer, and it should be aligned in depth to obtain projective regions. There are many implementation difficulties that complicate the design of 3D architectures and these are examined in Section 4.3.1. Also, the advantages provided by a direct 3D compared to 2D layer-by-layer approach will be studied. We propose the two-step reconstruction architecture described in Section 4.3.2. The general goals in the studies presented here are:

- To evaluate the feasibility of 3D clustering algorithms at the L1 trigger - to study the prerequisites needed and how to fit them into hardware constraints
- To study possible system architectures that would make 3D clustering possible to implement - taking into account the constraints coming from hardware and the mechanical construction of the detector
- To identify the critical points of these architectures
- To find possible architecture solutions and strategies that can be applied

4.3.1 3D clustering advantages and implementation difficulties

The advantages of a direct 3D clustering compared to a 2D layer-by-layer are as follows. First, it is easier to eliminate the noise in 3D, because we have the information from all the layers at once. For example, we can apply some data processing based on the known EM shower profile or HAD characteristics and thus we can eliminate more PU. This is something that we cannot do in 2D layer-by-layer, but we can apply a threshold to limit the 2D cluster size on each layer. However, with separate 2D clusters, we can end up with a larger number of smaller clusters that we need to deal with instead of detecting a single but larger cluster (when the z coordinate is included). To illustrate this, let us consider that the EM shower is evolving through the layers, so it is first very weak in the first couple of layers (small energy deposit) and then gets stronger as it evolves higher energy deposit). The 3D clustering will perform better than 2D, because when we do the layer-by-layer clustering, we define a threshold on the size of the cluster. It means that, when we have low-energy hits in one layer, it can be below the threshold and these hits get rejected, while if we would connect them to the hits of the next layer (or the next one), we could see that it is part of a 3D EM-like or HAD-like shower.

It is not only easier to merge clusters correctly, but also it is easier to separate the close-by or merged clusters in 3D than in 2D, because with the depth information we know how each of them evolves trough the layers. Also, it is hard to efficiently link 2D to 3D clusters, since there are large fluctuations between clusters, which makes it difficult to successfully reconstruct the depth coordinate. This has further impact on the seeding and clustering.

There are few 3D clustering implementation difficulties, which we study hereafter. First, we need projective regions if we want to do direct 3D processing, so the first problem to deal with are the non-projective regions in the detector mechanical construction. We have to take this into account when the TCs are transferred from the FE. Also, communication may be needed between FPGA boards of the same BE trigger stage.

The non-projective motherboard regions

The projectivity of the detector data coming from the FE to the BE trigger part is restricted by hardware and the detector mechanical construction. Let us assume that the selected HGCAL geometry would have around 30 thousand hexagonal modules arranged in layers, where each module contains around 64 TCs. It is not possible to fit the whole detector data into a single BE FPGA in stage 1, because we are limited by the number of input links, which is 72 in this case (KU15P FPGA from the Kintex Ultrascale family). So, instead of the layered view in depth, each FPGA may have a sectorized view of the end-cap data (through all layers), which can be useful because sectors can be projective in depth if layers are not rotated. In this case, the 3D clustering can be achieved and the tracking is enabled. However, we are limited once again by the number of links available in the selected FPGA, so sectors must be further cut and divided into smaller regions. These regions have to be precisely defined in order to retain the projectivity. As shown on Figure 4.17, one sector (ex. 60°) consists of modules, where one motherboard "covers" several modules (up to 6).

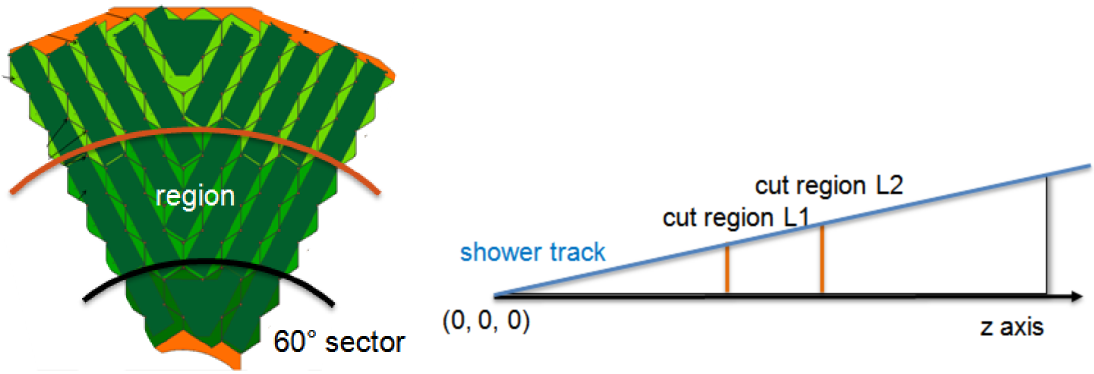


Figure 4.17: End-cap 60° sector cut into regions (left) and the shower tracking projectivity requirement (right).

Cutting the sector has to be done in a specific way. In order to apply 3D clustering we have to follow the shower track through the layers, so we need to cut the regions in such a way that a track intersects all the layers inside the same region (Figure 4.17). This requirement of cutting the region in a projective manner through the layers is needed, but it is impossible to cut over the motherboard. Also, if we want to follow track, then it cannot be separated between boards; one FPGA should "see" a track from one region through all the layers.

The communication between FPGA boards

Another problem for the implementation of a direct 3D clustering is the communication between FPGA boards, that arises from the regional view of the detector data. For 3D clustering purpose, we need a track (seed) and the neighboring cells around it. This means that we need to share the data between neighboring FPGAs, where each FPGA processes data from one region (in depth), as illustrated on Figure 4.18.

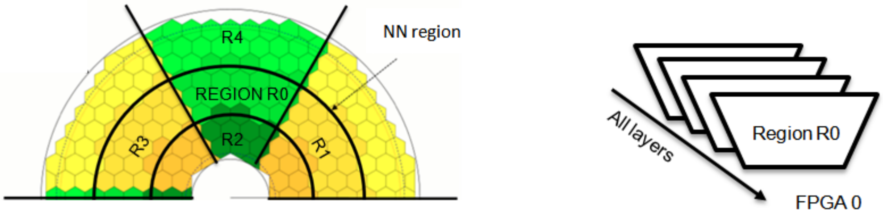


Figure 4.18: The neighboring (NN) regions of a cut 60° sector.

A forward communication is preferable, such that an FPGA in stage 1 communicates with an FPGA from stage 2, avoiding the communication between FPGAs of the same trigger stage. This is the case in the baseline reconstruction strategy of 2D followed by 3D, where each FPGA in stage 1 has the data from the full layer, and each stage 2 FPGA has the information from all the layers in one BX (Figure 4.19). However, a sectorized or regional view of the data requires the inter-communications between FPGAs, because we need to deal with borders. For example, if we have a TC seed at the border between two regions, the neighboring FPGAs that share this border should have the information from their region as well as the data from the border of the neighboring region (Figure 4.18). Adding additional data to a single FPGA increases the latency, because it receives additional input TCs, which need to be unpacked, processed or re-packed and sent. The more neighbors each FPGA has, the more data should be shared.

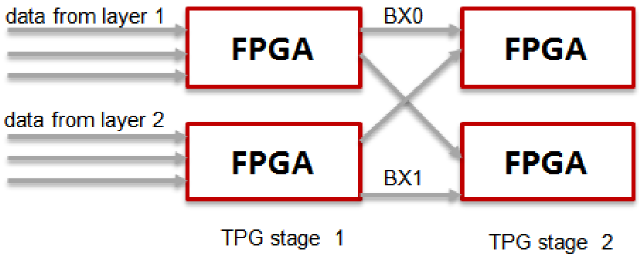


Figure 4.19: Baseline TPG architecture (2D followed by 3D).

Since the communication between boards adds latency to the system, it is better to have larger area regions but the problem is again the number of links available that needs to be considered. With larger regions, the fraction of shared data at the border is reduced (as preferable):

$$fraction = \frac{the\_number\_of\_shared\_data}{total\_data\_inside} = \frac{region\_circumference}{region\_area} \tag{4.1}$$

The proposed architecture solutions

We propose two solutions that can make the direct 3D clustering possible to implement. First, we can have an interface layer between the FE and the BE, which can solve the non-projectivity problem (Figure 4.20). It can re-order the input data in a projective manner, so that we can accomplish projective regions no matter of the sector cut. It is also possible to concentrate the data before sending it to the BE and transform the data into a format that is more suitable for 3D clustering. This can reduce the amount of work for the next stage but it adds additional latency to the system as well as more hardware.

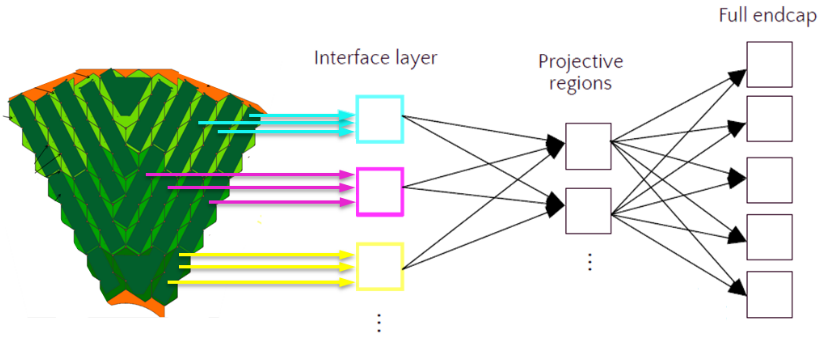


Figure 4.20: Concept of the interface layer to enable the projective regions.

Also, the communication between boards in stage 1 can be avoided by data duplication, which can again be accomplished with the interface layer. Namely, the same data can be sent to several neighboring FPGAs of stage 1 that share the same border. No additional latency is added, but more data transfer is required from the interface layer towards stage 1 (Figure 4.21).

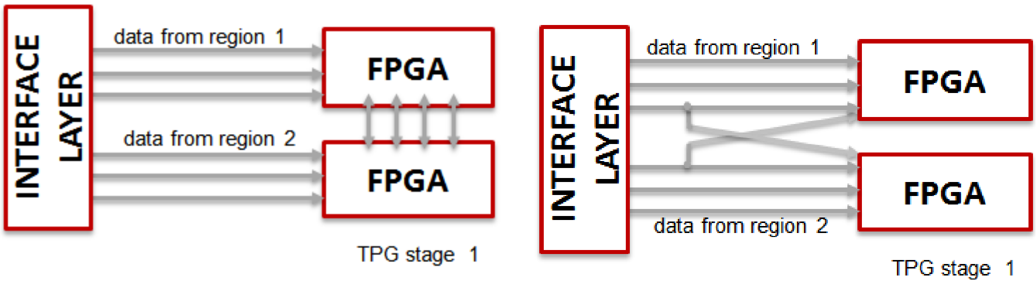


Figure 4.21: TPG architecture for a direct 3D clustering. The inter-communication between boards (left) and data duplication (right).

4.3.2 Two-step 3D architecture design

Once we have projective regions and the corresponding data, a first option can be to apply the 3D clustering directly inside the detector volume. However, it is rather resource-consuming to do 3D clustering in the whole detector at once, and it is also a matter of how to find 3D neighbors in memory. While in 2D we just navigate in (x,y), in 3D we need to find a projective neighbor in depth, so to navigate in three dimensions and not just in the horizontal line direction any more. Since this can be too complicated for the hardware, a second option could be to do the seeding (in the first stage) to identify the ROIs in the detector and then to apply a 3D clustering around the pre-selected seeds (in the second stage).

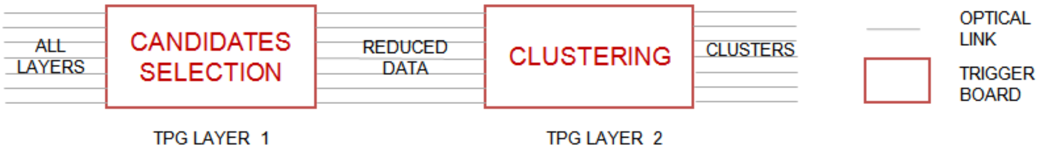


Figure 4.22: HGCAL two-step BE hardware architecture.

In this case, BE architecture can have 2 layers; one to select the seeds (seeding algorithm) and send them to the next layer that will perform the clustering (Figure 4.22). The main intention is to cluster the energy around seeds, so we need to send the seeds as well as their nearest neighbors to the next layer which "sees" the ROIs over one full end-cap and can do the clustering only on these ROIs. It is very important that the first stage "sees" the sectorized end-cap regions projectively (for example one $60°$ sector in depth,) while the second stage "sees" the whole end-cap layer. We would like to avoid any loss of information when being near the borders of the sector, so there should be some data sharing between FPGA boards. The strategy to avoid this is to duplicate the data such that a board "sees" a little bit more data than covered by its sector at the borders (it also "sees" some border data of the nearest neighboring sectors).

The problem with this approach may be to implement the efficiently pipeline data transfer in the first layer, because we cannot first input the data, and then process it while receiving another set of data at the same time. We have to keep all the input TC energies and their positions into FPGA memory while the seeding is performed. This is because the seeding tells us which data to select, so we cannot drop the data before we decide what to keep (our data needs to be in memory until we decide). Since the seeding has its latency, this may take a while and, on the other side, it may require a large memory to keep the whole data.

Another problematic thing to consider is the number of links or bandwidth required to send the selected ROI clusters. Naturally, it would depend on the size of the clusters, so in Section 4.3.3 we study the amount of data that is sent to the second layer when the selected seeds are considered together with a certain ring size around.

4.3.3 Study on data bandwidth between TPG stages in the two-step architecture

The bandwidth study presented in this section shows how much bandwidth and how many links do we need to send the selected ROIs defined by the selected seed candidates. What we want is not only the seed candidate, but also the ring of neighbors around it. Hence, the goal is to transfer the central TC candidate as well as a ring of TCs around the central one. We want to see what is the number of ROIs that we can send within the available bandwidth and number of links, where in the Xilinx description, the stage 2 FPGA has maximum of 96 input links available (VU7P FPGA from the Virtex Ultrascale family).

The formulas used in the study are modified formulas from [99]. The formula for the bandwidth B evaluation is:

$$B = N^{mean}_{candidates} * N_{NN\_rings} * v_{LHC} * N_{bits} * N_{layers} \tag{4.2}$$

where $N^{mean}_{candidates}$ is the mean number of selected candidates per event, N_{NN_rings} is the number of rings around a candidate, v_{LHC} is LHC collision rate ($v_{LHC} = 40MHz$), N_{bits} is the number of bits per candidate ($N_{bits} = 8$) and N_{layers} is the number of layers in depth ($N_{layers} = 40$).

The formula to calculate the number of links N_{links} is:

$$N_{links} = \frac{B}{v_{link} * \epsilon_{link} * T_{mux}} \tag{4.3}$$

where B is bandwidth, v_{link} is the raw link speed between stage 1 and stage 2 ($v_{link} = 16.4Gb/s$), ϵ_{link} is the link encoding ($\epsilon_{link} = 64b/66b$) and T_{mux} is the time multiplexing ($T_{mux} = 24$).

The results are given on Figure 4.23 and Figure 4.24, where the signal sample used is from the unconverted photons (pT = 25GeV, PU200), and the background is PU200. It means that in our study only EM-like showers are selected to be sent to the stage 2, which is not the case since also HAD showers need to be selected. The data bandwidth on Figure 4.23 presents the mean values for both end-caps and the number of links on Figure 4.24 presents the fraction of the total number of links that need for the transfer. Results are shown both without time multiplexing and with the time multiplexing period ($T_{mux} = 24$).

It is shown on the figures that a size of a ROI is varied for the number of rings around the candidate (up to 10 rings). Since the EM shower is rather narrow and small, we can select smaller number of rings around the candidate (maximally 5), being sure that our generated electron or photon are contained inside the ROI. The results on the data bandwidth per event show that for 5 rings of candidates around the seed and 98% SE the transfer rate is around $800Gbit/s$. For a 99.5% SE the needed bandwidth per event is around $1500Gbit/s$.

Considering the number of links without time multiplexing, for 5 rings of candidates around the seed and a 98% SE, the total number of $16Gb/s$ links required as input to the single layer-2 FPGA is around 50. For a 99.5% SE, the total number of $16Gb/s$ links is around 90. This means that a single stage 2 FPGA can receive the projective data in depth from a single BX because the VU7P FPGA has 96 input links in total. However, the data duplication can be problematic so more links could be needed in order to receive data from the border regions.

Hence, the time multiplexing can be used (ex. $T_{mux} = 24$), and the results show that for 5 rings of candidates around the seed and a 98% SE, the total number of $16Gb/s$ links required as input to the single layer-2 FPGA is around 2, while for a 99.5% SE the total number of $16Gb/s$ links is around 4. A corresponding BE TPG architecture concept is illustrated on Figure 4.25.

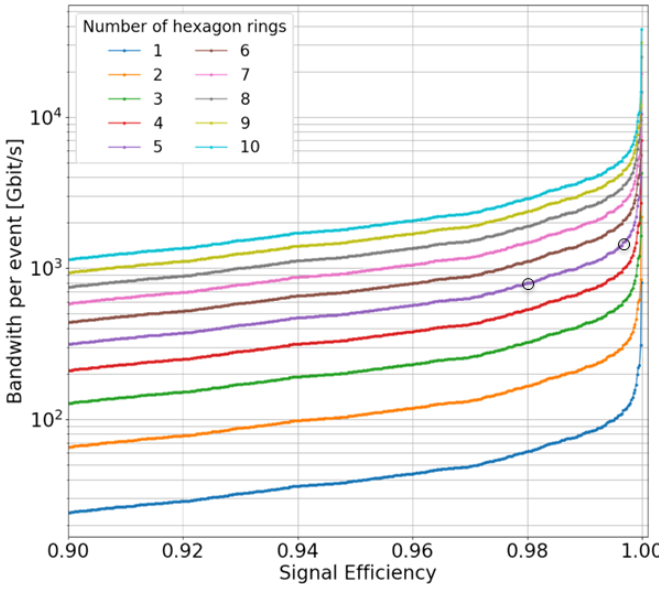


Figure 4.23: Results of the bandwidth study showing the mean bandwidth per event needed to transfer the ROIs which are defined as hexagon rings around the seed candidate. The 98% and 99.5% SE for 5 rings around the candidate are marked with circles. The particles used for the seed selection are the unconverted photons 25GeV with PU=200.

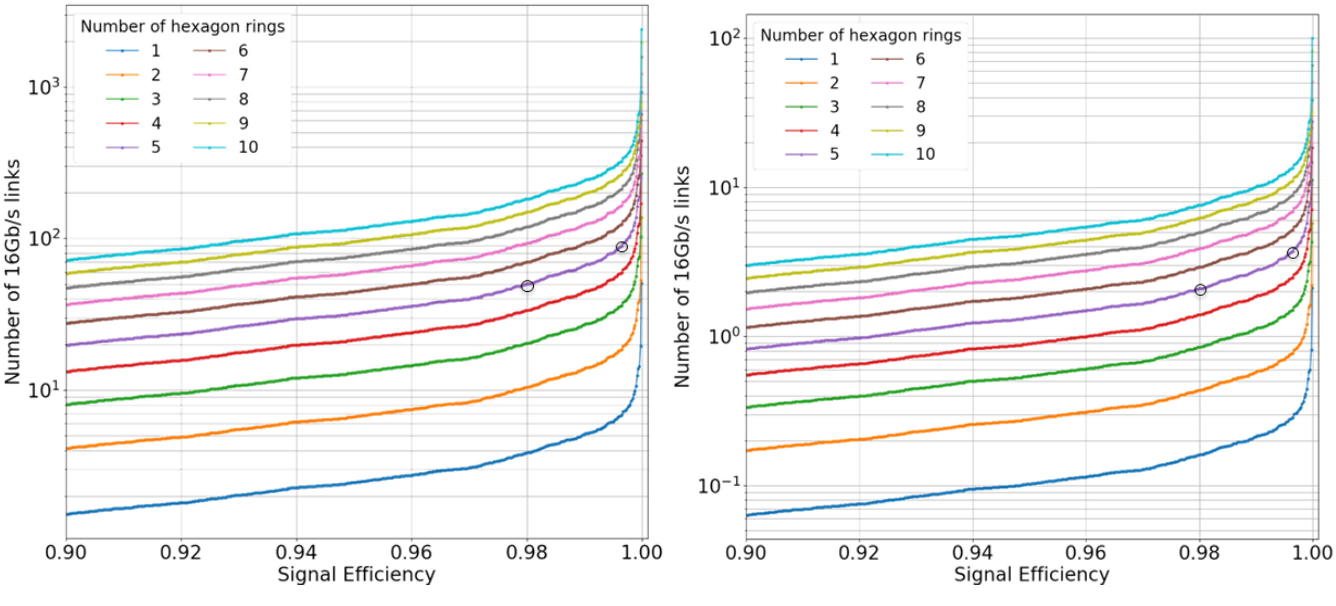


Figure 4.24: Results of the bandwidth study showing the number of links needed as input to a single stage 2 FPGA without time multiplexing (left) and with $T_{mux} = 24$ (right). The 98% and 99.5% SE for 5 rings around the candidate are marked with circles. The particles used for the seed selection are the unconverted photons 25GeV with PU=200.

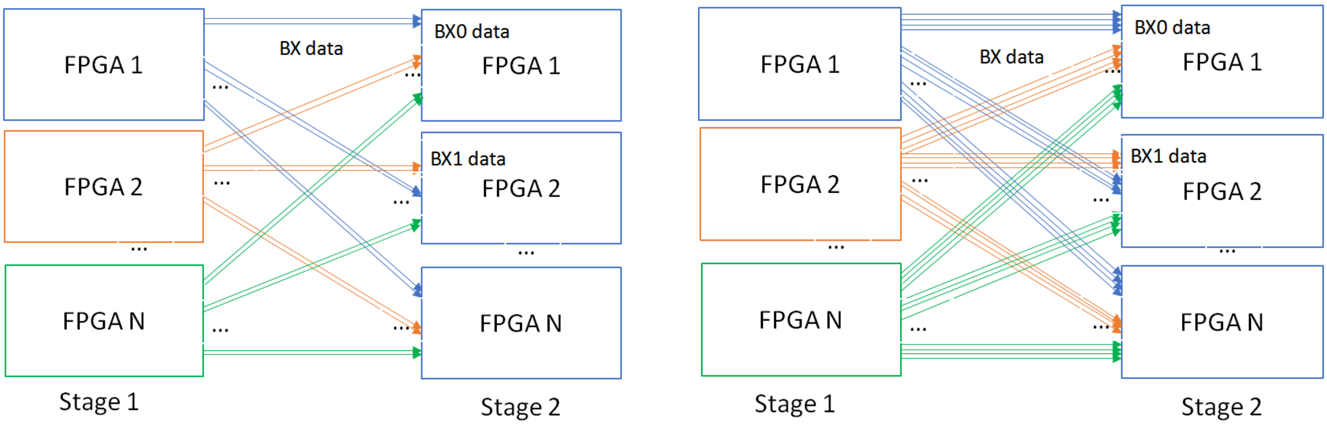


Figure 4.25: TPG architecture design where 2 input links (left) and 4 input links (right) are used for the stage 2 FPGA to receive the transferred ROI rings containing the TCs. The total number of stage 2 FPGA input links depends on the FPGA type and the links used for sending the ROIs are marked with color arrows.

To conclude, the preliminary study has potential, but it may be not feasible to select the ROI candidates and send them to the next stage. We would need at least 5 rings of TCs around to keep the high signal efficiency for the trigger, but the number of links is increasing very fast. This is especially true if we include all photons or electrons studies. The results on Figure 4.26 show that the number of links to a stage 2 FPGA for all photons and 5 rings around the candidate (for a 98% SE) is 15, and for electrons this is more than a hundred. Hence, we have to sacrifice the efficiency a lot to fit the data transfer to the available links number.

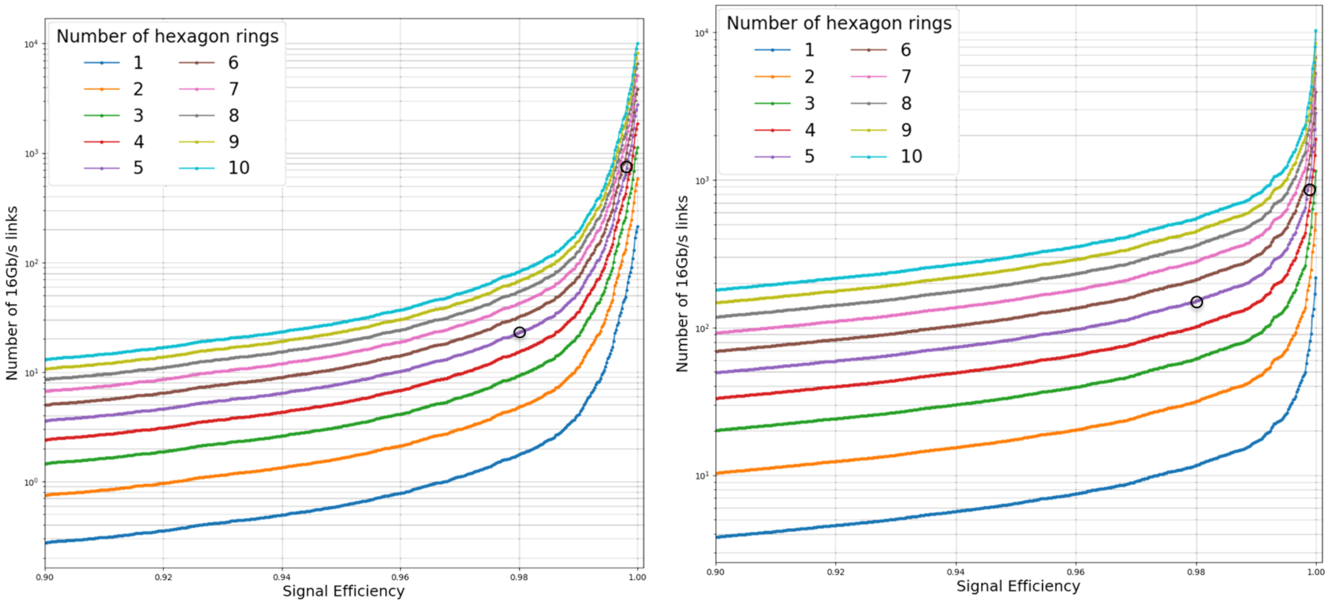


Figure 4.26: Results of the bandwidth study showing the number of links needed as input to a single stage 2 FPGA with the time multiplexing period $T_{mux} = 18$. The 98% and 99.5% SE for 5 rings around the candidate are marked with circles. The particles used for the seed selection are all photons 25GeV with PU=200 (left) and electrons with PU=200 (right).

Also, the hardware implementation may be complicated, due to the already mentioned problem with how to pipeline things. For example, let us consider that the stage 1 does the seeding with the input TCs, it waits until it finds the seeds and then selects the seed TCs and the TCs around. It is important to minimize the latency from waiting the seeding to be done. The data manipulation will add additional complexity and the latency to the system, and there is the transfer data bottleneck.

The limitation of the conducted study is that it lacks the selection of HAD shower, which are as important to select as the EM-particles in HGCAL L1 trigger. Also, simulation of the stages should be implemented in hardware, to more precisely evaluate the latency.

4.3.4 Discussion and evaluation

Once again, the data reduction is problematic along the whole trigger path, especially in the FE concentrator where only a fraction of TCs is selected. Then the two layers of the BE TPG allow to cluster and reconstruct the received TC energies. In the baseline TPG approach, this is done in the first layer that sees the individual HGCAL layers and performs a 2D clustering layer-by-layer. Afterwards, clusters are sent to the second layer which links the 2D clusters to form a larger 3D cluster. Going from the FE to the BE, a simple cone algorithm can be used when merging 2D to 3D clusters, to contain the full shower that needs to be reconstructed. Since the depth of the EM shower is an important characteristic as it develops throughout the layers of the calorimeter, the drawback of a 2D before the 3D clustering is the loss of the depth information. Since HGCAL has a layered design with fine granularity, which enables an efficient usage of the depth component, a direct 3D algorithm can be applied on the selected data.

We have studied possible architectures that can allow the implementation of the direct 3D clustering algorithms. We have shown that the main problem is that projective regions are mandatory, which can be solved with an interface layer. This may include additional hardware and bring additional latency to the system, but it will allow to rearrange the data so that each BE stage 1 FPGA receives the projective data from one region (part of a sector) in depth. Once we have projective regions, we can:

- Do the 3D clustering directly on these regions, with the advantage that there is no need for an additional stage by doing the clustering directly as a single step, so that there is less hardware in the trigger system. However, it requires more data sharing between neighboring FPGAs and it is more resource consuming to do clustering in the full detector volume.

- Design a tracking algorithm (TA) to identify ROIs in the detector and apply the 3D clustering directly as a single step only on these ROIs. This is less resource consuming than the previous approach, and also does not include additional stages, but again requires a lot of data sharing between boards.

- Identify ROIs with the TA in the first step (stage 1 FPGAs) and then send the pre-selected ROIs to the FPGAs of the next stage that will apply the 3D clustering on these ROIs.

The third option with a two-step TPG algorithm (tracking followed by clustering) is described in details in what follows (Chapter 5). We propose the TA that is independent of the architecture, which means that after it is designed, we can choose no matter which architecture we want with the tracking enabled. The main goal with the TA is to identify a track to find the ROIs in the detector, and to cluster energies along the track using the selected ROIs instead of at the whole detector at once. In order to accomplish this, we have to send the selected ROIs to the next layer that will build the clusters, taking into account the bandwidth limitation. A preliminary study shows that a possible architecture can be built with a rather low bandwidth consumption and a low number of links, especially if the time multiplexing mechanism is included in the system. However, the feasibility was studied only for the EM particles and it should be extended to other particles such as hadrons.

Another advantage of this two-step approach is that it requires less data sharing between neighboring FPGAs. This is because the first stage FPGAs only need to find seeds (the bins of the projected space) and associate the TCs to these seeds. Because of the tracking, all TCs which "fall" into the same bin are aligned along the track in depth. Therefore, when we associate the TCs to the seeds, we can select a certain ring of TCs around. However, it requires to keep all the input data into the FPGA memory while the seeding is performed. We don't know what will be selected exactly, so we have to keep everything in memory until we find seeds. This depends on the latency of the seeding, but it is possible that implementing an efficient pipeline concept may be too complicated for the hardware implementation. The solutions should be provided to enable the pipeline data processing with minimal waiting between clock cycles.

To conclude, the proposed architecture is promising, but the feasibility should be studied when other signal showers (besides EM) are sent to the next layer. The implementation of the architecture may be problematic, and it should be further studied in order to examine the complexity it brings to the hardware.

Chapter 5

Reconstruction of trigger signals

In Chapter 4, a study was presented on the possible TPG architecture designs that would enable that 3D clustering is applied on the reduced ROI data instead on being performed in the whole detector. The goal of the study in Section 5.1 is to describe the algorithm that we refer to as TA, which is used to select these "interesting" ROIs in the detector with tracking and seeding procedures, such that afterwards we can choose no matter which 3D architecture we want, because this algorithm is independent of the architecture used. We study the optimal TA parameters, such as bin size and bin space, in order to obtain the best potential energy reconstruction efficiency. Also, we present the identification mechanism that enables better background rejection by extracting more signal-containing ROI regions. Finally, we study different shower identification mechanisms to possibly simplify the TA hardware implementation.

5.1 L1 trigger data reconstruction studies

In this section, we propose a model for an EM shower track finding at the L1 trigger for the future CMS HGCAL. The aim of the work is the development of an efficient data reduction model, so we reduce the event input data to extract interesting regions or ROIs containing signal EM shower directly at the trigger level. The basic goal of the proposed model is a pre-processing step towards 3D shower reconstruction. Instead of performing the resource demanding clustering algorithm in the whole HGCAL volume, interesting detector ROIs can be extracted and sent for further processing. The methodology of the conducted research is based on the current state-of-the-art pattern recognition techniques which are used in the novel environment of HGCAL shower tracking application instead of the usual tracker sub-detector part.

Evaluation results clearly indicate benefits of the method proposed in Section 5.2. Also, we improve our basic model in Section 5.2.5 and show that larger data reduction can be accomplished by using the known EM shower pattern contained in the longitudinal energy profile. In addition, shower images can be produced as a result of the proposed procedure and are used for later ML study in Chapter 6 on the image-based classification.

5.1.1 Motivation

One of the major issues that comes along in HL-LHC is the increased number of PU events. This makes the development of reduction techniques very important, so in Section 5.1 we concentrate on that challenging issue of data reduction. The concrete target of our research is the reduction of the event data such that only EM signal is extracted, eliminating PU noise. We adopt the concepts from the current state-of-the-art pattern recognition algorithms in HEP that are commonly used in the tracker sub-detector of LHC experiments. This is the first intend to apply these methods in CMS HGCAL, and a proposed model for EM shower tracking in the upgraded detector environment is evaluated and its efficiency is reported. Also, further improvement of the model is examined for the additional noise reduction, accomplished by using the known patterns of signal and PU. The aim of the research is to extract positions of signal candidates revealing ROIs in the detector, to be sent for further processing. Also, ROI image generation is enabled with the proposed method, to generate data images used for the training and the classification with the neural network (NNet).

The baseline strategy for clustering and reconstruction consists in two steps (2D followed by 3D), and in Chapter 4, we have shown the advantages of a direct 3D clustering. Also, the implementation difficulties are examined, together with possible 3D clustering architectures. A main precondition for 3D clustering is the ability to define projective regions, so that one can apply one of the following strategies: do 3D clustering directly in the projective regions, or have a two-step algorithm (tracking followed by clustering). The latter implies that shower track is identified to find ROIs in the detector. Next, energies can be clustered along the track by using selected ROIs instead of the whole detector at once. To extract the ROIs, one needs to apply the seeding algorithm by following the shower energy tracks, as described in Section 5.2. When the ROIs are selected, we can target the architecture that we want to use for 3D clustering.

5.1.2 Background and related work

This section provides a short summary of the related work on pattern recognition techniques in HEP. Different tracking algorithms have been developed such as artificial retina, Hough transform or "tracklet" algorithm. The proposed solutions are usually optimized for a fast execution with parallel architectures and their goal is to offer an increased performance compared to the alternative approaches.

Pattern recognition with artificial retina algorithm in HEP

The goal of pattern recognition in HEP experiments is specific to the detector type. For example, track finding problem is related to tracker detectors, where interesting signal tracks must be identified [100]. The algorithm used for this task is named the artificial retina algorithm, and is based on a concept similar to the human visual system. Namely, groups of neurons work in parallel and reduce the data in the first stage of the image processing, as they receive signals from specific receptive fields. Next, an interpolation of the produced responses is performed to recognize patterns with minimal latency [101, 102, 103, 104]. Generated in parallel, the responses of neurons create a preview of the image edges or shapes in about 30 milliseconds [103].

The number of collisions is very large at the LHC, producing around 2 billion collisions per second (40 million per second * PU = 40 * 10^6 * 40 $\approx$ 2 * 10^9), where each one generates huge number of particle showers. Since most of them are uninteresting for physicists, it is necessary to develop an efficient reduction technique. That is what the retina does in human vision: it ignores huge volumes of data in the visual field but alerts the brain when interesting patterns appear [103]. To reduce the number of uninteresting tracks, a pattern recognition technique is used to detect their features (such as edges or shapes) and trigger data storage only when these tracks are "unusual".

Hence, the retina concepts can be used for track reconstruction, assuming that a tracking detector is made by a set of parallel layers. A particle passing trough the tracker material is leaving ”hits” through layers in the tracker space, providing the measurement of a single spatial coordinate x. In a detector volume without any magnetic field, the trajectories of charged particles are straight lines, intersecting detector layers, and they are identified by two parameters (m, q), where m is the angular coefficient and q is the intersection with the x-axis in the (z, x) plane. Next, the space of track parameters, (m, q), is discretized into cells, representing the receptive fields of the visual system. The centre of each cell identifies a track in the detector space, which intersects detector layers in spatial points that are called receptors. Therefore each (m_i, q_j)-cell of the parameter space corresponds to a set of receptors x_k, where $k = 1, ..., n$ runs over the detector layers, as shown in Figure 5.1. This procedure is called detector mapping and it is done for all the cells of the track parameter space [102].

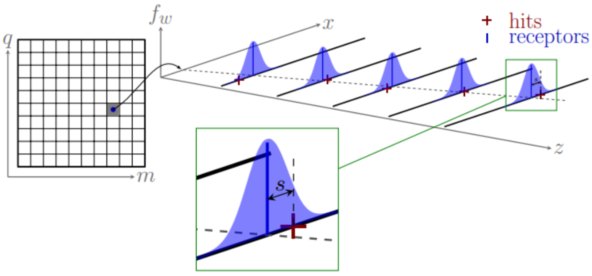


Figure 5.1: Mapping coordinates by using the receptors in the detector to form a grid in parameter space [102].

It is assumed a Gaussian response function R for each hit x_k, where a hit that is closer to the track centre point $\bar{x_k}$ contributes more to the parameter cell:

$$R = \sum_k \exp \frac{-(\bar{x_k} - x_k)^2}{2\sigma^2} \tag{5.1}$$

In Formula 5.1, σ is a parameter used to adjust the width of the Gaussian response to obtain a clearer mapping result. Another computation is done by averaging the nearest cells with a fixed kernel to additionally enhance the tracking efficiency [102, 104].

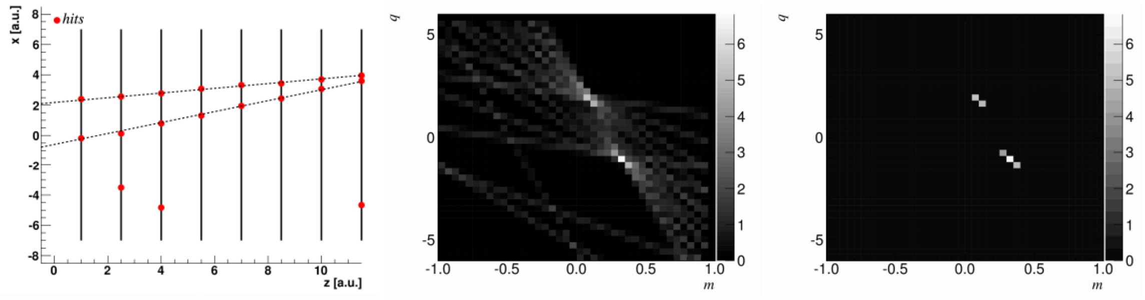


Figure 5.2: Retina response to an event with two tracks (above threshold) that are reconstructed [102].

Authors in [101, 103] provide a detailed design on the implementation architecture of the retina algorithm. Experimental results reveal a satisfactory hardware resource usage and high tracking efficiency, making the algorithm suitable for application in a real HEP environment. In a similar tracking study [105], the authors develop a modified retina algorithm with embedded fourth dimension of the particle hit. Namely, the hit position is extended with the precise track timing information. This results in an increased tracking accuracy. Another retina optimization that compares the algorithm performance and computational cost is proposed in [106]. When all hits are processed by the detector mapping discretization shown in Figure 5.1, tracks are identified as central local maximum elements calculated over the space of track parameters (Figure 5.2) [102]. Each cell represents the reconstructed particle trajectory line, where in the end only two tracks are selected. These two are the correct ones from the event, since the largest number of receptors have mapped to the same cell in the track parameter space. Authors in [106] substitute the standard procedure of finding all local maximums in the response function so that the brute-force grid local maximum search can be avoided.

Track finder algorithms based on Hough transform

Similar efforts besides artificial retina algorithm are found in the literature, again with the same goal to enhance the tracking capabilities for the trigger [107, 108, 109, 110]. Namely, the retina algorithm is based on the concept similar to the Hough transform (HT) [101, 104]. Retina assumes a weighted accumulation map built based on each incoming hit, so that a hit can contribute to a track with different percentages depending on the weight value. Next, a kernel is applied on the resulting map to calculate the mean of the neighboring weights. On the other hand, algorithm like Hough transform relies on a binary response [103], i.e. either a hit corresponds to the track or not.

The general Hough transform was first introduced rather long time ago in image processing [111] and it has been improved since then for different applications. Certainly, it can be applied in fast HEP triggers due to an efficient hardware implementation with low latency [108, 112, 23]. Hough transform is usually one of the various consecutive steps in track finder algorithms, including former projective data preparation and a following Kalman filter to eliminate the fake tracks [107]. The Hough technique is simple, based on a projective binning and mapping to the parameter space, where hits that are aligned along a track accumulate in a specific bin of the resulting map.

They all pass trough a single point in the Hough space, so the intersection of the lines is used to extract track candidates [107, 108].

The application of the Hough transformation in the tracker detector is illustrated on Figure 5.3. Basically, hits are present in the detector, which are bent in the magnetic field. The procedure is given with the following formulas [107]. Charged particles are bent in an homogeneous magnetic field B, and the radius of curvature R can be calculated as:

$$R = \frac{p_T}{0.003 * q * B} \tag{5.2}$$

Hence, the curvature is a function of the particles transverse momentum (p_T) and the charge value (q). Considering a hit defined by (r, ϕ), authors in [23, 107] show that the trajectory of the particle can be described by the relation $\frac{r}{2*R} = \phi - \theta$. The combination with the Formula 5.2 provides a transformation equation, such that the new parameter coordinates of the hit in the Hough space $(\frac{q}{p_T}, \theta)$ can be calculated as:

$$\theta = \phi - \frac{(0.0015 * B * r) * q}{p_T} \tag{5.3}$$

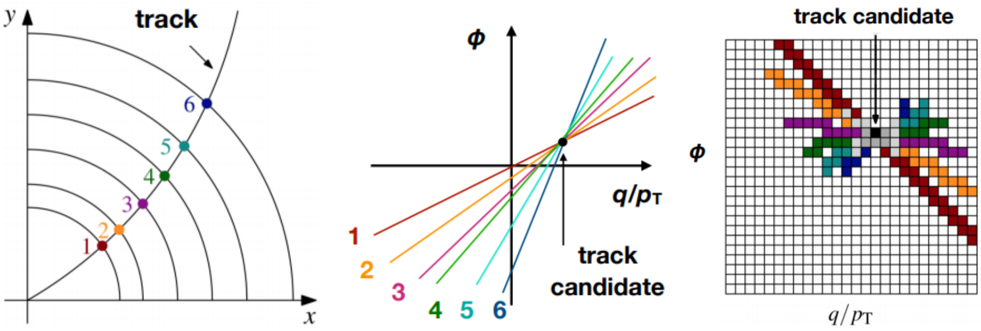


Figure 5.3: Schematic description of the HT technique [23].

A similar tracking algorithm is based on using track seeds or “tracklets” [108, 109, 110]. The algorithm is implemented in several consecutive steps like in the standard road searching strategy. First, a seeding process is performed to find seeds from each pair of hits on adjacent starting layers. Roads are formed by projecting the seeds to the next layers with a seed matching procedure. The trajectories are estimated assuming the interaction point at the center of the coordinate system [109]. Finally, the track parameters are calculated after removing duplicates.

5.1.3 Basic assumptions for particle energy track finding at the L1 trigger

The HGCAL trigger must perform an efficient energy reconstruction under the increased PU conditions. In Chapter 3, the geometry of the high granularity silicon-based sampling calorimeter is described, by using silicon sensor modules with TCs that enable a fine segmentation of the detector sensing plane. In order to handle the HL PU conditions, the tracking information can be used in the upgraded L1 trigger system. Unlike in the tracker detector, in a calorimeter trigger, the pattern recognition task is to group signal data to showers [100] and to perform shower classification or shower recognition tasks. Hence, in this context, tracking energy deposits similar to particles or seed tracks are extracted. A reduction technique can be performed to decrease the number of misidentified particle tracks, similar to rejecting low-momentum particles in the tracker detector [109]. Based on the extracted seeds, “hotspot” regions are be selected in the detector and these are interesting for further analysis.

By its nature, an EM shower is not just a track, but it has an extension with some known pattern. Hence, a better selection of the seeds in the seeding step can be done by using known the shower pattern information. It is based on the longitudinal shower energy profile, as a shower develops in depth and a fraction of the total energy is deposited in each layer. The EM shower parametrization is described in Section 5.2.3.

Like in the retina algorithm, the conversion of the parameter space from physical (x, y, z) to projected shower parameter space (r, c) is applied with the simple HT, where $r = \sqrt{x^2 + y^2}$ and $c = \pi * r$. Direct 3D detector information is used to project the data. However, unlike in retina, there is no averaging kernel applied on the result map, such that a simple HT concept is used whether the tracked energy deposit corresponds to the EM track or not. Also, all tracks are equally valid because, in the basic TA algorithm without EM shower identification, there are no weights applied on data. In the improved TA model, weights are applied on TC energies before the projection. Basically, different energy tracks contribute differently to the projected result, such that tracks initiated from an EM shower have larger weights and thus are "more" important. Furthermore, the TA weights in the improved model are extracted with the EM longitudinal shower profile. Next, seeds are found by using a 3x3 central local maximum filter. This is similar to how tracks are identified in retina algorithm, but we do not search for local maximums over a threshold in the space of track parameters. We apply here a simple central local maximum filter inside a 3x3 window on the accumulated map in the parameter space, to additionally reduce the number of seed candidates or ROI regions.

Several assumptions are considered in the TA design:

- The simple case with 2 track parameters is used, assuming that an EM shower can be approximated by a straight line coming from the centre of the detector (0,0,0). This is equivalent to assuming that particles are not charged and have a straight line trajectory, which is true for photons but not for electrons. Also, the beam spot is not precisely (0,0,0) [101, 22].
- In the basic version of the TA, accumulation of the energies is done within a 2D histogram map in the parameter space, such that hits aligned along a track accumulate more energy as they project to the same mapping bin.
- Unlike in the basic version, where the raw TC energy values are projected without any weights, in the improved TA model with EM shower identification, energy weights are applied for each detector layer, such that the energies where the shower is expected are "more significant".

5.2 Tracking algorithm design

To repeat, the goal of the TA is to follow an "interesting" energy track in the detector volume and to extract the interesting regions. In this Section, we describe the TA basic steps and the seeding method for the reconstruction of the EM shower. Also, EM shower parametrization is explained, which is used to improve the basic TA model.

5.2.1 Projective binning with Thales projection and Hough transform

The validity of the assumption used here, that shower develops inside the calorimeter in a direction that is a straight line following the particles path and joining the collision spot, is tested in [22]. As shown on Figure 5.5, a simple Thales projection of the hits in different layers is implemented. Compared to the baseline trigger reconstruction with a 2D layer-by-layer following by a 3D algorithm, with this seeding method, all the information of the different layers is used at once.

The detector mapping is performed with a virtual 2D grid, with bins of size 1cm^2, which is the typical size of a cell in the low-η region. Each TC is projected with the Thales algorithm as shown on Figure 5.4 and the projection procedure is as follows. Let us reconstruct a line from each TC centre position (x_i, y_i, z_i), where i is the corresponding detector layer. For this, we can use a standard canonical equation of a line in space, because we know the coordinates of the point that lies on the line - this is a collision spot or detector centre (x_0, y_0, z_0) - and the direction vector of the corresponding line $\vec{v} = l; m; n$. Then, the equation of the line can be written in the canonical form using the following formula:

$$\frac{x - x_0}{l} = \frac{y - y_0}{m} = \frac{z - z_0}{n} \tag{5.4}$$

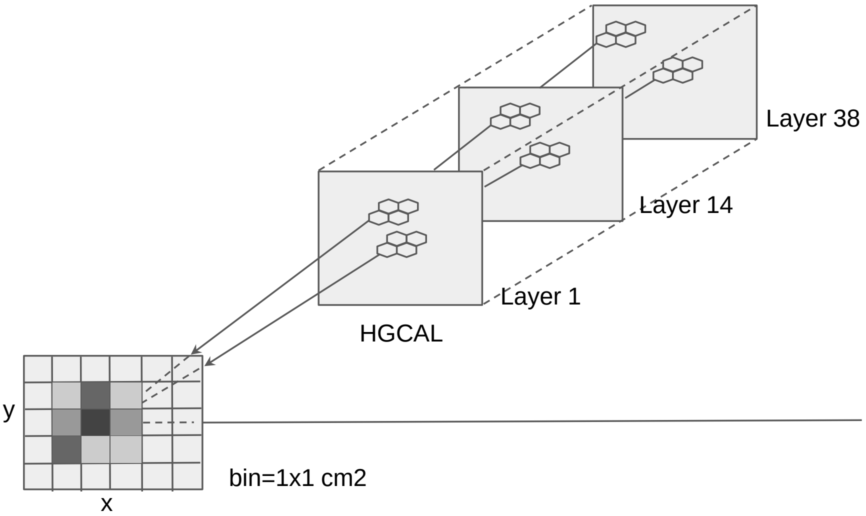


Figure 5.4: TA projection of TC energies.

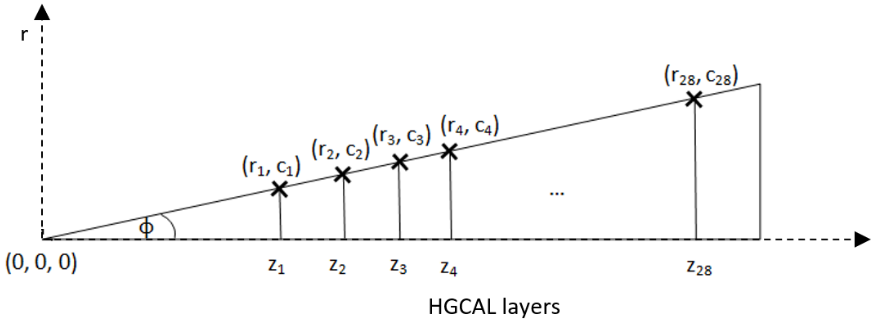


Figure 5.5: Thales theorem applied for the energy projection and reconstruction of the track initiated by the seed.

If the line also passes trough the TC centre $T_i = (x_i, y_i, z_i)$ at the layer i, the parameters l, m, n in Formula 5.4 can be expressed as:

$$l = x_i - x_0; \quad m = y_i - y_0; \quad n = z_i - z_0 \tag{5.5}$$

Following the assumption that $T_0 = (x_0, y_0, z_0) = (0, 0, 0)$, the direction vector is:

$$\vec{v} = \vec{0T_i} = \begin{bmatrix} x_i - 0 \\ y_i - 0 \\ z_i - 0 \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \tag{5.6}$$

Hence, following the Formula 5.5 and 5.6, the parametric equation of the reconstructed line $T = T_0 + t * \vec{v}$ can be written as:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + t * \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \quad or \quad \begin{cases} x(t) = t * x_i \\ y(t) = t * y_i \\ z(t) = t * z_i \end{cases} \tag{5.7}$$

Since the equation of the virtual projected plane is $z = 320$ cm, which is the coordinate value of the first HGCAL endcap layer, the intersection between the reconstructed line and the plane is:

$$z(t) = t * z_i = 320cm \Rightarrow t = \frac{320cm}{z_i} \tag{5.8}$$

Hence, the coordinates of the intersection points are:

$$x^{projected} = t * x_i = \frac{320cm}{z_i} * x_i; \quad y^{projected} = t * y_i = \frac{320cm}{z_i} * y_i \tag{5.9}$$

A simple variation of the Hough transform is used for the detector mapping during the projection procedure.

The concept is used to transform the coordinates from the detector space (x, y, z) to the parameter space with two (η, ϕ)-like parameters in cm, i.e. (r, c) (Figure 5.6):

$$r = \sqrt{x^2 + y^2}; \quad c = r * \phi \tag{5.10}$$

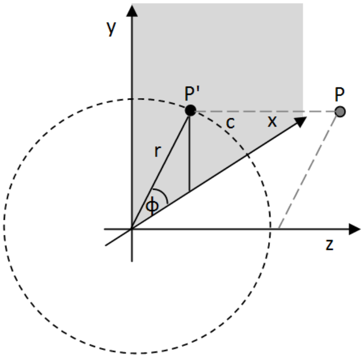


Figure 5.6: Conversion of the coordinate system to the parameter space.

The conversion between (x, y, z) and (η, ϕ, z) coordinate space is done with the following transformations:

$$r = \frac{z}{\sinh \eta}; \quad x = r * \cos \phi; \quad y = r * \sin \phi \tag{5.11}$$

The angle ϕ can be reconstructed from the coordinates x and y in the first quadrant:

$$\phi = \arctan \frac{|y|}{|x|} \tag{5.12}$$

Values ϕ' in other quadrants can be calculated from Formula 5.12 by using simple trigonometry, and the angle value is returned to the first quadrant with $\phi = \phi' \pm 2\pi$:

$$\phi' = \begin{cases} -\phi + \pi, & \text{if } x < 0 \text{ and } y > 0 \\ \phi + \pi, & \text{if } x < 0 \text{ and } y < 0 \\ -\phi, & \text{if } x > 0 \text{ and } y < 0 \end{cases} \tag{5.13}$$

The seeding in the projected parameter space is based on a 3x3 central local maximum filter, where a maximum is extracted if its position is at the center of the filter window). This way, a reduced set of seed candidates is extracted.

5.2.2 Reconstruction of the shower track initiated by the seed

After identifying the seed positions on the projected result in the parameter space, they represent the centres of "interesting" regions in the detector. We can extract these ROIs by reconstructing the shower line or shower tracks initiated by the identified seeds (Figure 5.5). The same procedure is used with Thales projection, where i is the layer number:

$$r_i = z_i * \frac{r^{projected}}{z^{projected}}; \quad c_i = r_i * \phi \tag{5.14}$$

Once the track is reconstructed, it provides a central axis for the definition of a cylinder ROI in the detector volume. The next step is to select the TC energies around the central axes inside the radius Δr (Figure 5.7).

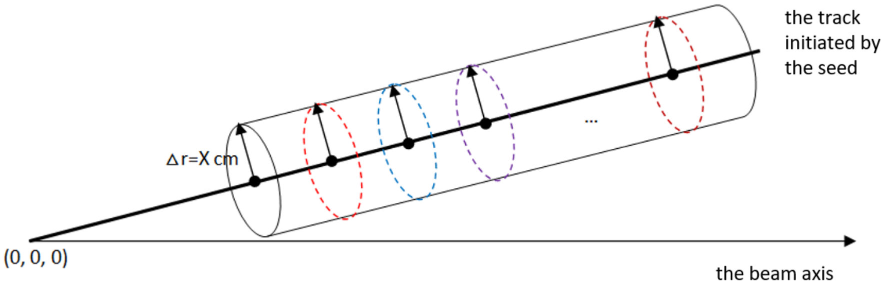


Figure 5.7: Selecting the ROI around the seed.

5.2.3 EM shower parametrization

There are two processes in HGCAL which are interleaved and they are absorption and detection (Chapter 2). When a particle reaches the calorimeter, it passes trough a series of layers of absorber material that initiate the EM shower. Electron or photon gets completely absorbed in the calorimeter, which means that total energy is absorbed, but only a fraction of initial energy is measured as a signal in the detector part. The development of the EM shower is a cascade process with two dominating effects: pair production and bremsstrahlung [20]. Each step in the development of the shower has on average the size of the radiation length X_0. The shower stops at the point when the critical energy E_c is reached at the shower maximum (t_{max}), when there is no multiplication any more, but the existing particles travel in the material and gradually lose their energy. This model suggests that the shower curve is a Gamma function, meaning that it should rise fast until a peak value is reached, after which it falls to zero.

The mean longitudinal profile of the energy deposition in an electromagnetic cascade (Figure 1.8) is well described by a Gamma function [22]:

$$\frac{dE}{dt} = E_0 * b * \frac{(bt)^{a-1} * e^{-bt}}{T(a)} \tag{5.15}$$

where the shower depth t is measured in radiation lengths X_0, E_0 is the initial energy and a and b are variable parameters, which depend on the atomic number Z of the absorber material. The maximum of the function is accomplished for $T = \frac{a-1}{b}$. To better describe what are parameters a and b, they relate to the first two moments of the Gamma distribution.

A simple approximation is that each time we multiply particles by two, we decrease the energy of the previous stage in halves:

$$E_t = \frac{E_{t-1}}{2} \tag{5.16}$$

This implies that the initial energy is shared into all the particles, whose total number in the cascade process follows the exponential rule $N(t) = 2^t$:

$$E(t) = \frac{E_0}{N(t)} \tag{5.17}$$

Also, shower lengths in the material t_{max} is predefined, and the maximal depth of the shower is:

$$2^t = \frac{E_0}{E_c} \quad \Rightarrow \quad t_{max} = \frac{\ln \frac{E_0}{E_c}}{\ln 2} \tag{5.18}$$

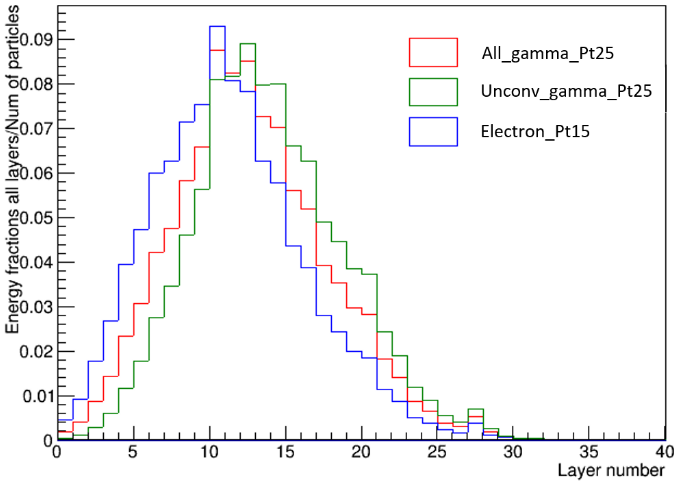


Figure 5.8: Longitudinal energy profile comparison between different particles.

Figure 5.8 shows examples of EM shower energy profile in HGCAL calorimeter for electrons, all photons and the unconverted photons. Electrons start their shower earlier in the detector compared to photons. All photons mean both, the converted and unconverted, whereas unconverted photon means unconverted before reaching the calorimeter. On the contrary, converted photon is equivalent to 2 electrons when reaching the calorimeter. EM shower of the unconverted photons is the latest because they first need to convert and create a pair of electrons.

5.2.4 Basic TA algorithm verification

Signal and background samples used in the TA studies are described in Chapter 2. The number of signal seed candidates extracted by the TA depends on the number of central local maximums in the projected map. The seed selection is done after an energy threshold of 150MeV is applied on the map (Figure 5.9). Since CMSSW simulation allows for the information of the simulated position of the true particle, immediate seed verification can be performed. Hence, signal seed candidate is associated with the particle by following a simple rule:

A signal seed is the one inside a cutoff distance $\Delta R = 3cm$ from the photon (gamma) or $\Delta R = 7cm$ from the electron. If there is more then one seed candidate inside the radius, the one with the maximal energy is selected.

The former is called the matching procedure and the selected signal seeds define the ROI positions in the detector. In case of the background sample, the basic TA model will find ROIs that are not interesting and they cannot be matched to particles. These background seeds are caused by PU contamination and the main intention in the trigger is to reduce this contribution. Hence, in Section 5.2.5, an improvement of the basic TA model is studied, with an implementation of a PU reduction mechanism that further reduces the number of extracted ROIs.

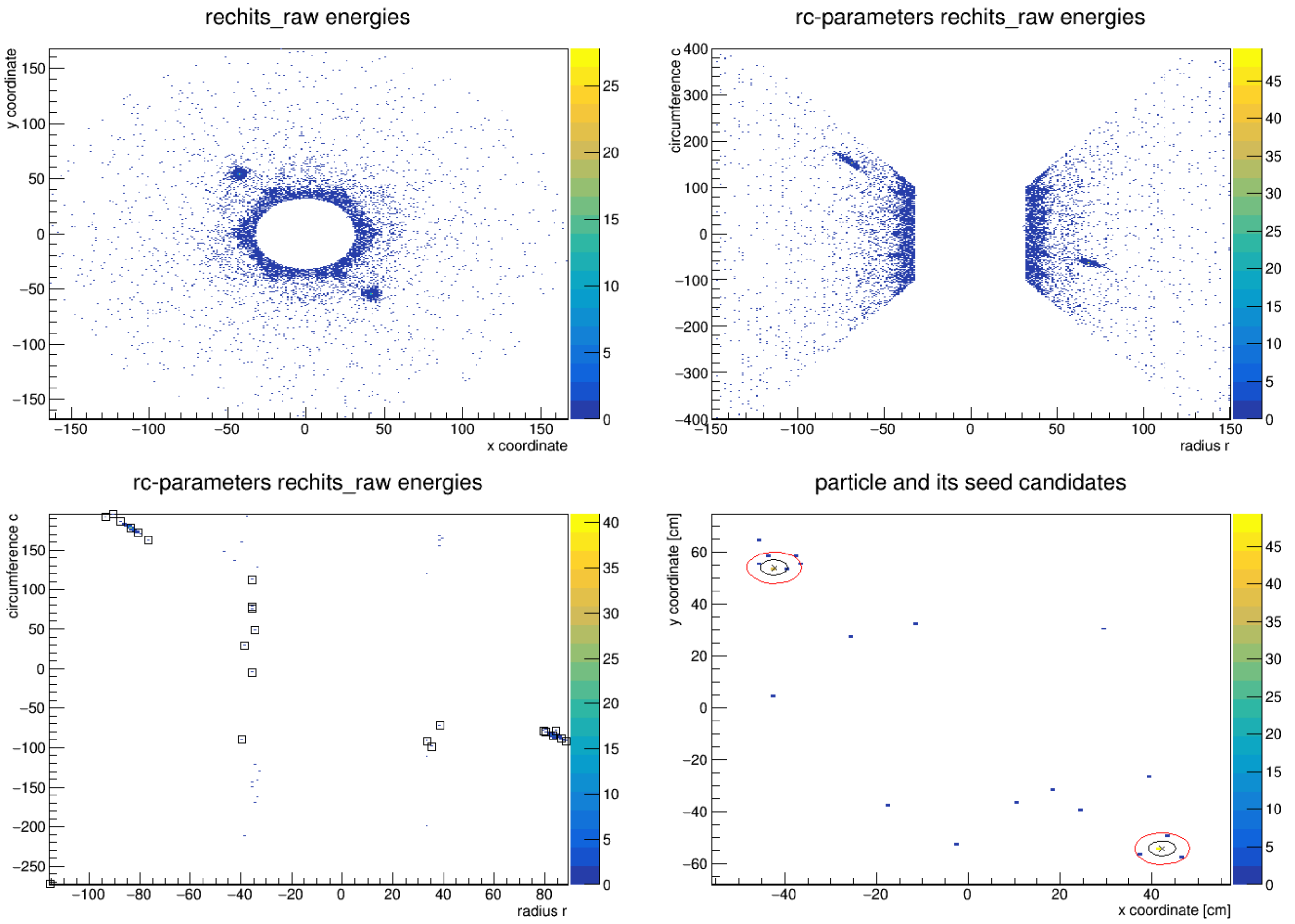


Figure 5.9: Projected event data (photons pT=25GeV, PU=0) in (x, y) and (r, c) parameter space (up). Seed candidates selection and matching particles to seeds (down). The black radius corresponds to $\Delta R = 3cm$ and the red radius value to $\Delta R = 7cm$.

5.2.5 Shower identification by using the EM longitudinal profile

To further reduce the background, we can combine tracking with some shower identification scheme. This can give us a better selection of the seeds. Namely, we encode the known EM shower pattern (in particular the longitudinal information) in terms of energy weights that are applied on TCs. Weight values depend on the HGCAL layer, such that layers 10 - 15 contain the most important energies. This is approximately where the EM shower peak is expected. On the other hand, the PU seeds distribution is mostly contained in the first detector layers, such that these weights are lowered and very close to zero.

The improvement of the basic TA model can further reduce the number of tracks (ROIs) by eliminating "uninteresting" tracks that are not coming from EM-like deposits. The identification can be applied on TC energy values directly, and the identification mechanism is applied with the following formula, where i is the index of the energy on the corresponding layer:

$$E(x,y) = \sum_{layer} w^{layer} * (\sum_{i} E_i^{layer}) \tag{5.19}$$

Algorithm performance with SC and TC granularity

It has been noted that TC energies are projected with the TA algorithm and this we refer to as TC granularity. Also, a bin size 1x1cm^2 is used. For the comparison, we have projected SC energies to study the SC granularity as well. Although unrealistic for a real trigger implementation, it can be indicative to see the bin size effect on the algorithm performance. The virtual grid, which is used for the event energy accumulation, introduces additional granularity factor, which can enhance the real granularity of the detector. For example, when projecting TCs instead of SCs, we can improve the granularity by using smaller bins on the virtual plane.

The trade-off between signal efficiency and the rate expressed as the mean number of background candidates per event (also referred to as bandwidth) is presented with the receiver operating characteristic (ROC) curves on Figure 5.10. It can be seen that the improvement of applying weights, i.e. the enhancement of efficiency with shower identification included in the basic TA model is significant. It gives a lower bandwidth while keeping the same signal efficiency. There is a background reduction in all cases with weights applied, but the background reduction is lower in case of electrons and all photons when compared to selecting unconverted photons candidates. This is because the longitudinal profile of unconverted photons is used for the shower model in all the cases.

Obviously, the selection of the longitudinal profile has an important role in the shower identification and profile differences could be taken into account in the identification algorithm. However, both electrons and photons are EM particles and their profiles are very similar. Hence, there could be a small reduction improvement when electrons profile is applied on electron data instead of the unconverted photons profile as in Figure 5.10. Figure 5.11 shows that this reduction is negligible.

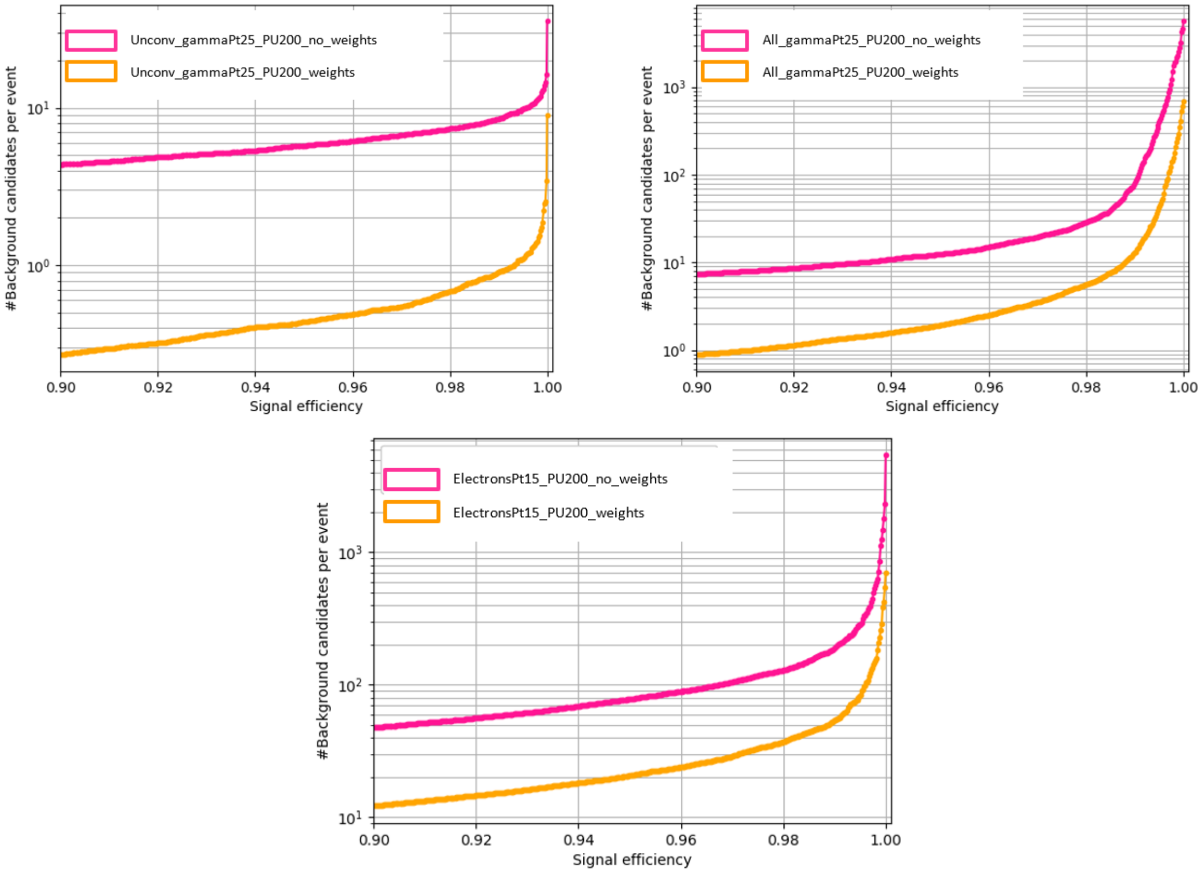


Figure 5.10: Improvement of the basic TA model with SC granularity. The unconverted photons and all photons (up) and electrons (down).

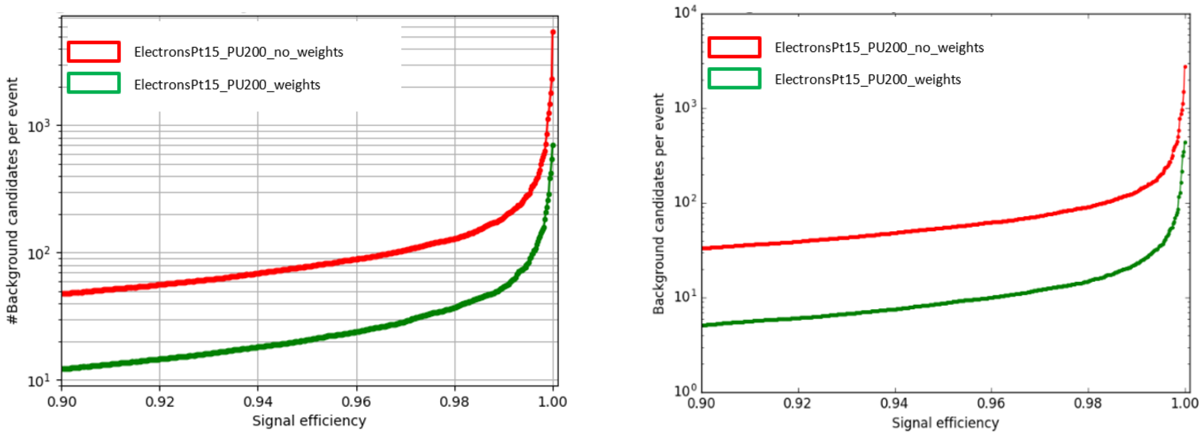


Figure 5.11: Comparison of different profiles applied on electron events. The profile used for identification is extracted from the unconverted photons (left) and from the profile of electrons (right).

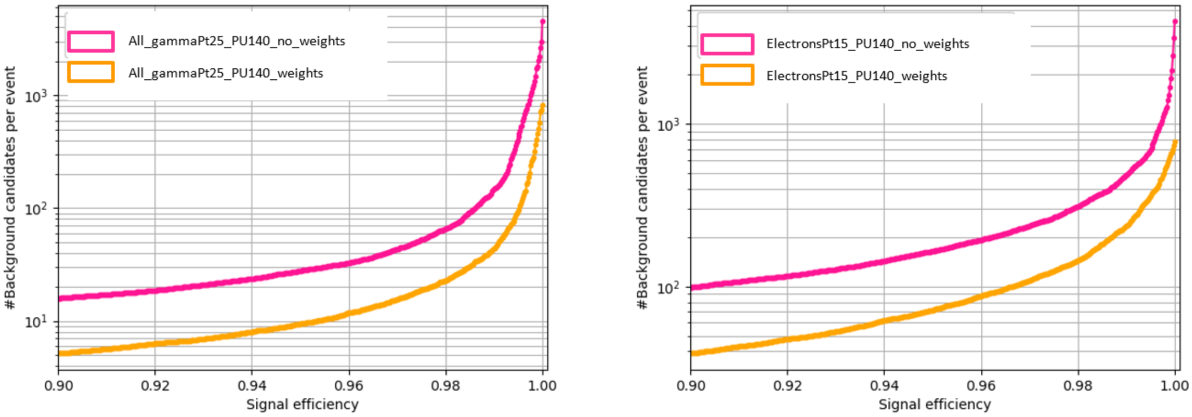


Figure 5.12: Improvement of the basic TA model with the TC granularity. All gamma (left) and electrons (right).

The ROC curves for the TC granularity are shown on Figure 5.12. Again, there is a background reduction in both cases with weights applied. It is more or less the same amount of reduction in both cases (as it was in the case of SG). However, it can be seen that a larger mean number of background candidates is present for the TC granularity. This is in both cases (electrons and photons), but one needs to take into account that a lower PU value is present for the TC granularity data. A possible explanation is that the bin size was not adjusted to TC granularity. Namely, the same bin size 1x1cm^2 was used in both TC and SC granularity, but the bin needs to be larger in case of projected TCs, since the TC area is larger. A bin size equivalent to the cell size (1x1cm^2) is too small, so the TC energy can be projected in the border between two bins, which causes energy to “split” between two neighboring bins instead of accumulating in one bin of larger size. This is a general well-known limitation of HT, as its efficiency depends on the correct binning. Namely, all hits passing trough the same line (or forming that line) must fall into the same bin in the Hough space, so that it can be easily detected. At the same time, a too large bin can include a lot of background noise. Obviously, bin size parameter should be further examined for the TA and the binning study is presented in Section 5.2.6 (it is the candidates selection performance that will tell us what is the best bin size to use in the TA algorithm).

Reduced EM shower longitudinal profile

We can use different longitudinal profiles to extract weights for shower identification. Besides the former full signal profile, a peak signal profile can be used, where we keep only the weights from the maximum energy layers in the profile and few layers around it, while all the other layers weights are set to zero. The full longitudinal profile of unconverted photons is filtered keeping only the 5 most energetic layers (Figure 5.13). It can be more suitable for the hardware (as it requires less multiplication of weights with energies) and it reduces the background. However, maximum profile is going to reduce the signal as well, so it can cause a degraded signal efficiency.

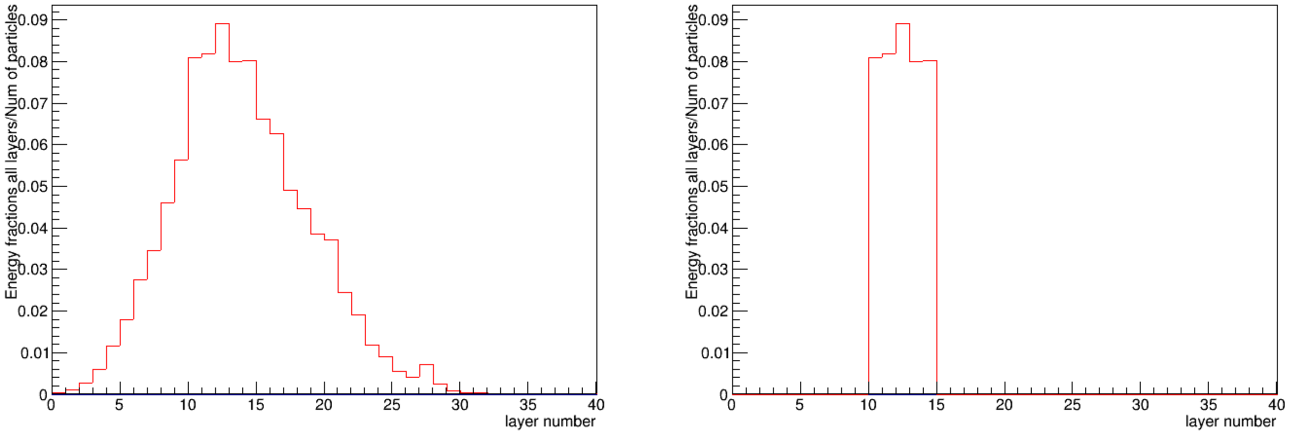


Figure 5.13: Full and reduced EM shower longitudinal profile for the unconverted photons of pT=25GeV.

5.2.6 Studies on selecting the optimal TA parameters

There are three important parameters for the 2D binning with the TA:

- Bin size - the projection of the TC energies can be done by using different bin sizes on the histogram map, where an optimal bin should be selected (not too small or too large).
- Study on the full vs. reduced EM profile used for the shower identification.
- Study on threshold applied on TC energies before projecting data in the accumulation map.
- Bin space - different coordinate systems like (r, c), (x, y) or (η, ϕ) can be used for the 2D histogram binning.

Binning study: histogram bin size

In this section, we present a binning study. While moving from SC to TC granularity, more energy should have been collected, but the bin size 1x1cm^2 (the size of the SC) was too small for accumulating TC energies on the virtual layer. Thus, we vary the bin size from 1x1 (one SC), 2x2 (four SCs = one TC) to 6x6 (one ring of TCs = 36 SCs). We can see on the signal efficiency (SE) curve (Figure 5.14) that we accumulate more signal as we increase the bin size, so the SE is higher. Also, since more energy is accumulated in the bins, we can apply larger TC energy cuts.

But, when we increased the bin size, the number of background candidates per event reduces up to the bin 4x4cm^2, and then starts to grow for the larger bin size of 5x5 and 6x6. We refer to this as "the bump effect" which is present in the background candidates energy distribution. As shown on Figure 5.15 the bump in the distribution appears for larger bin sizes somewhere around 25GeV and it is almost fully visible for the bin size of 6x6. This effect is caused by a sharp transition between the level of PU in the low ($|\eta| < 2.5$) and high eta region ($2.5 < |\eta| < 3.0$). Different resolutions are expected in the two separate η regions, while we used a constant bin size for the full η range.

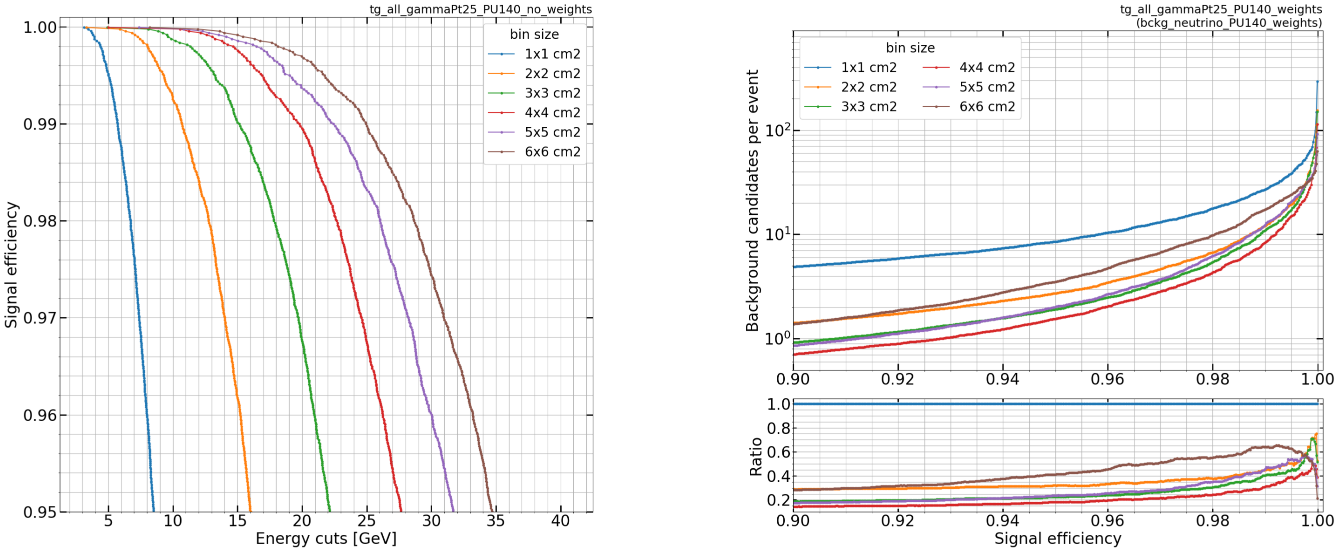


Figure 5.14: SE curve for photons of 25GeV with PU=140 and different bin sizes, where energy cuts are derived without weights applied (left). TA performance ROC curves for the improved tracking with identification (right).

Profile study: the full and the reduced EM profile comparison

In this section, the improved TA model is shown with the goal to compare the impact on the identification power when the full and the reduced EM shower profile is used. In the reduced version, the longitudinal profile of the unconverted photons is filtered leaving only 5 the most energetic layers. Hence, a reduced set of energy weights is extracted to be applied on TC energies, such that only TCs from these 5 ECAL layers are used, and all data from the remaining layers is set to zero. We are interested in the trade-off between signal and background in this scenario.

The results with photons of 25GeV (PU=140) shown on Figure 5.16 indicate that there is not a large difference in the background candidates reduction with the two profiles used. Filtered profile eliminates more background, and it reduces the signal as well, but the trade-off curves are more or less the same. The impact is slightly larger for the increased bin. The conclusions are the same of electrons 15GeV (Figure 5.17).

Threshold study: apply threshold on TC energies

In order to decrease the PU related "bump effect" seen in the binning study, we attempt to reduce the number of background candidates before accumulating the data on the virtual grid. We apply a threshold on the TC energies, and these are different from the thresholds applied on the candidates. Namely, the energy cuts defined for the ROC curve metrics (Figure 5.14) are signal efficiency derived thresholds, which are applied on the seed candidates. Here, thresholds are applied on TC energies before the TA projection, and they are derived by using a fixed quantile from the TC background energy distribution. For instance, a quantile regression parameter $\alpha = [.25, .5, .75, .9, .95, .99]$ can be selected whereas a set of thresholds is extracted removing α% of the background TCs. These thresholds are automatically adapted to the level of PU as they depend on the η value and on the layer number, because a

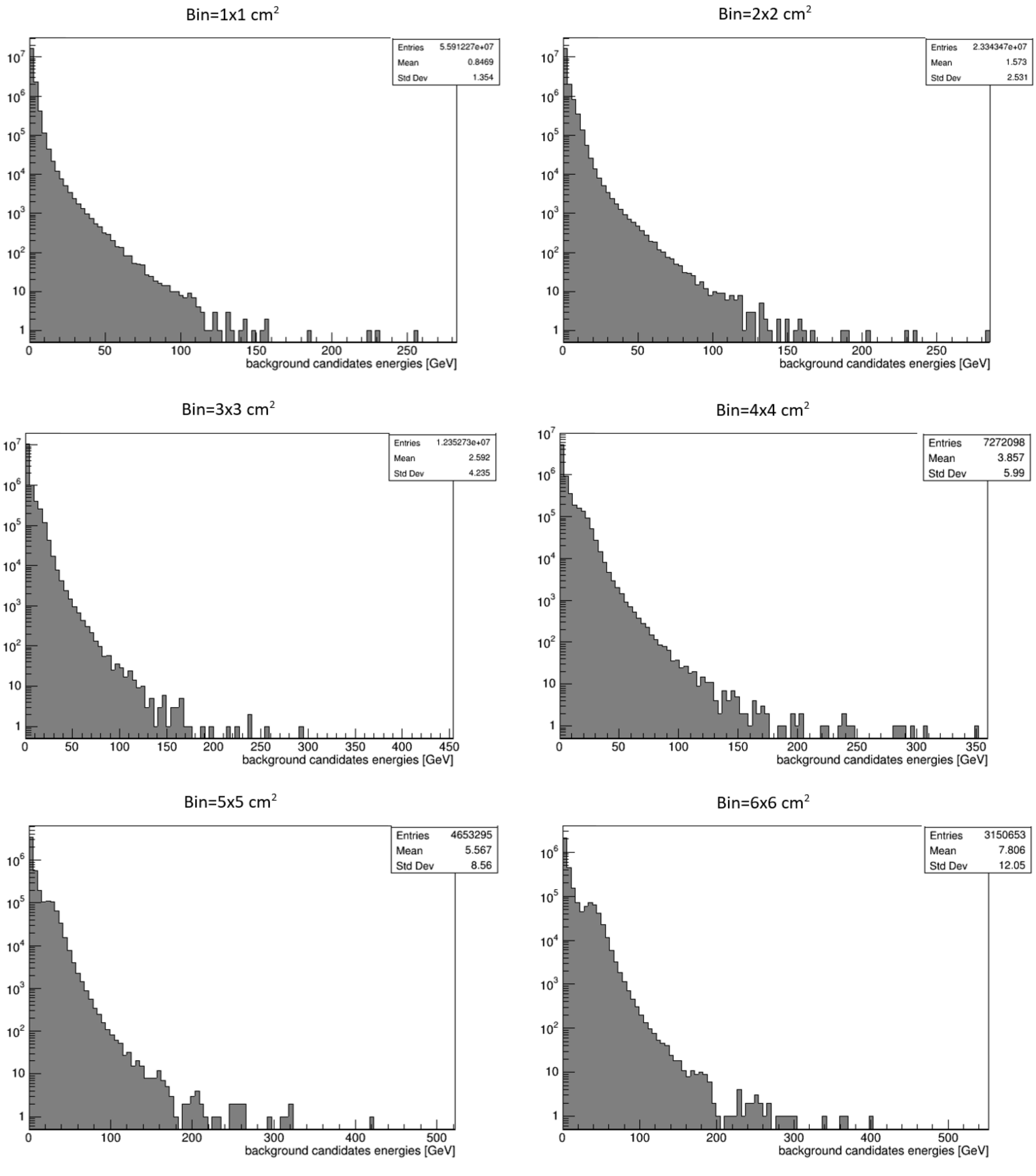


Figure 5.15: Background candidates energy distributions for different bin sizes (the sample is neutrino PU140).

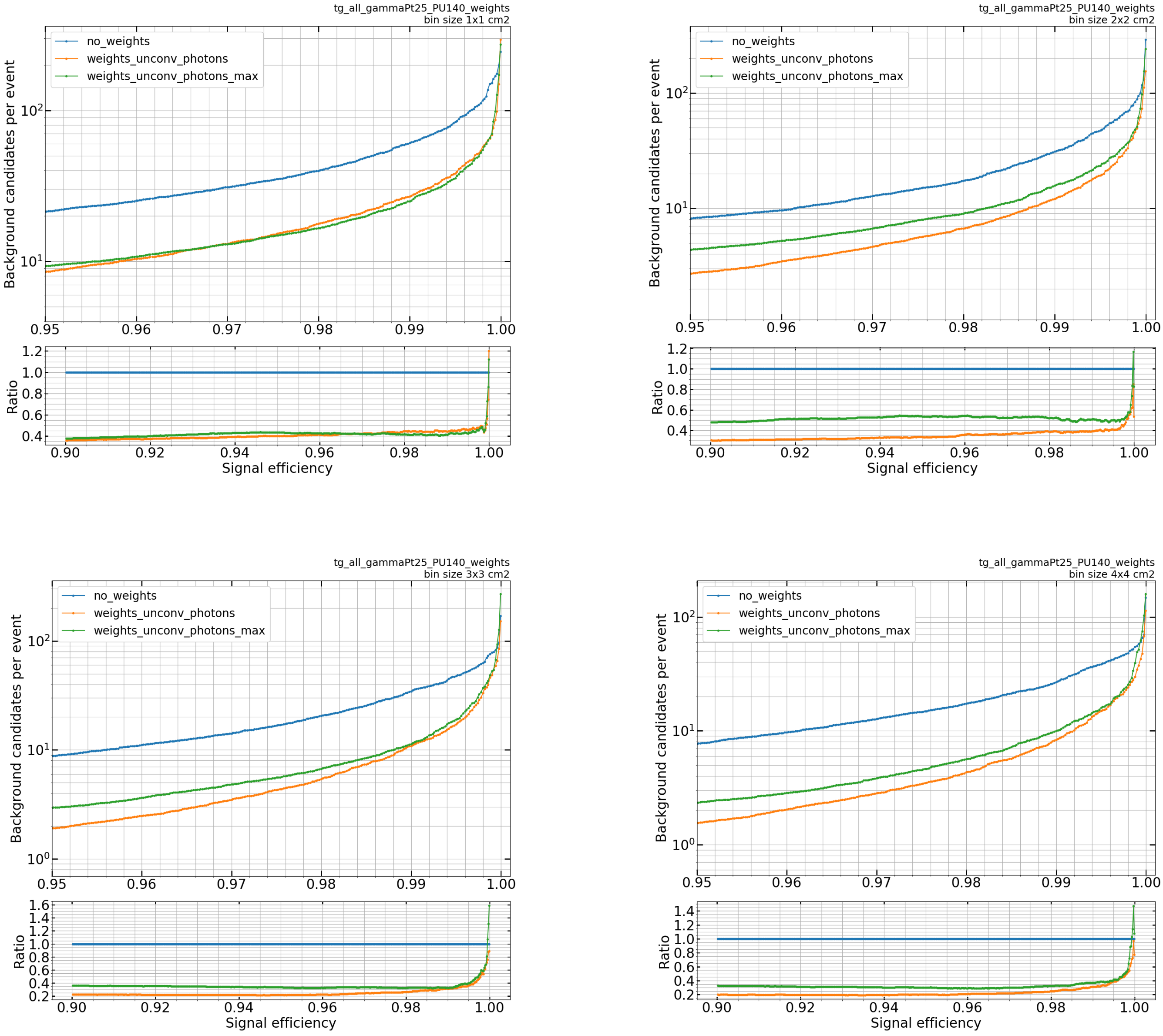


Figure 5.16: TA profile study results for photon EM shower identification (pT=25GeV, PU140). The bin size is varied in each plot.

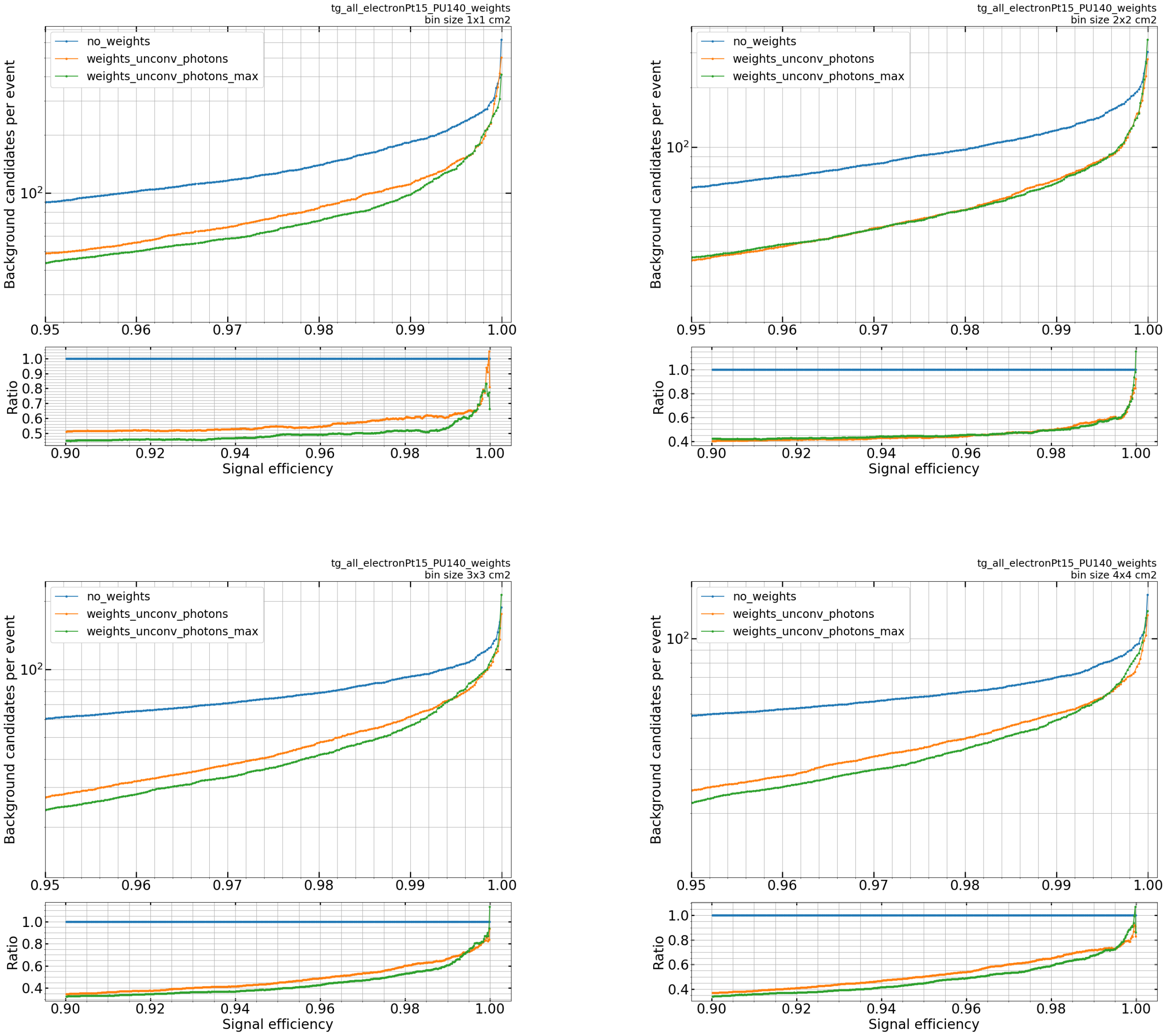


Figure 5.17: TA profile study results for electron EM shower identification (pT=15GeV, PU140). The bin size is varied in each plot.

quantile regression [113] of the background TC energies is done versus eta and the layer feature pairs.

Similar to the standard regression technique, goal is to predict our response variable (its quantile) based on the predictor variable, where in our case it is a pair of values. We predict the α quantile of our response variable based on a training that we perform giving all the responses that we have for a given predictor variable. Here, the predictor variable is a pair [eta,layer] and the response variable is the TC background energy for each [eta, layer]. So, when $\alpha = .9$, it will return a set of thresholds that is going to eliminate 90% of the TC background energies.

In order to choose the best quantile, different α values are varied in the threshold study. The goal is to examine the effect that we get when applying different quantile extracted thresholds on the TC energies before the TA algorithm projection. The results for photons of 25GeV (PU=140) with using the largest bin size 6x6cm^2 and several quantiles is given on Figure 5.18. Again, as we can see in the results with ROC curves that there is a strong background reduction (for all quantiles) with identification included in the TA. Also, there are less background candidates for the same SE with the increased quantile, because thresholds are higher and we keep less data such that less PU noise is included in the projection.

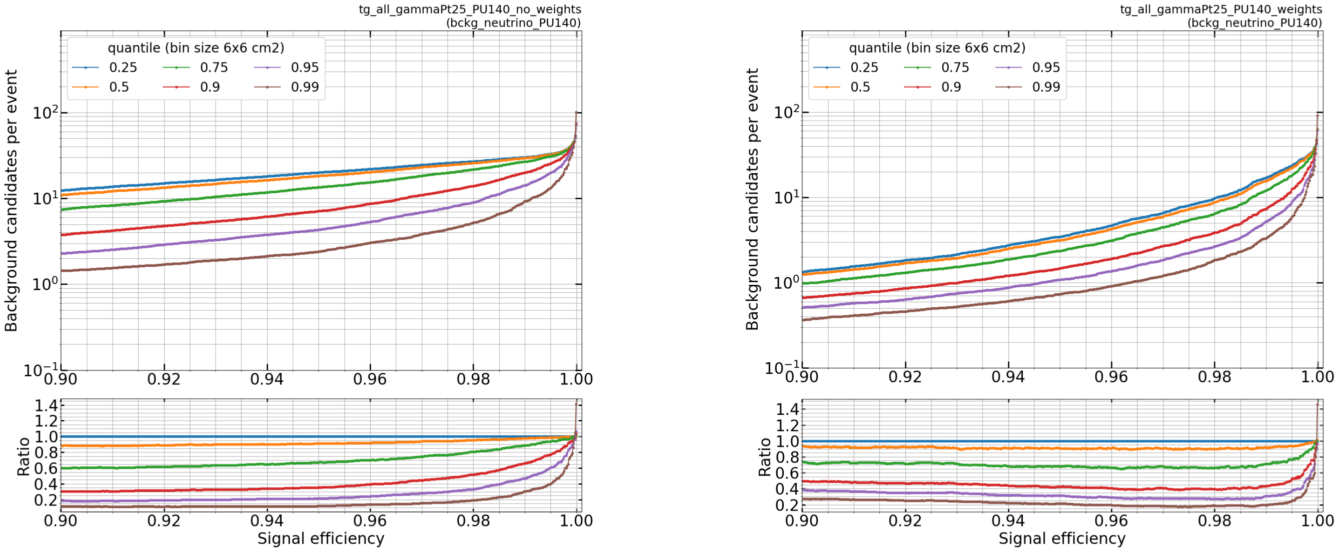


Figure 5.18: TA performance ROC curves for photons of 25GeV (PU=140) with different quantiles and bin size 6x6cm^2. Basic TA model (left) and the improved tracking with identification (right).

Figure 5.19 shows that the SE reduces faster with increasing quantile, because we don't know where our signal is, so we apply a threshold on TC energies everywhere in the detector. Since the same fraction of PU is eliminated for both signal and background TC energies, it means that we cut on the signal TCs as well, which decreases the efficiency. Naturally, the highest cut is applied for the largest quantile, and we are applying these highest cuts in the high eta region and in the first layers (where more PU is expected). Also, we cut on TC energies between layers where signal is expected (layers 10 to 15), which can be seen on the eta-layer maps of the TC cuts (Figure 5.20).

The threshold study is also performed for electrons of 15GeV (PU=140) and we can see the same effects (Figure 5.19). However, when compared to the photons shower tracking with identification included (Figure 5.18), one can

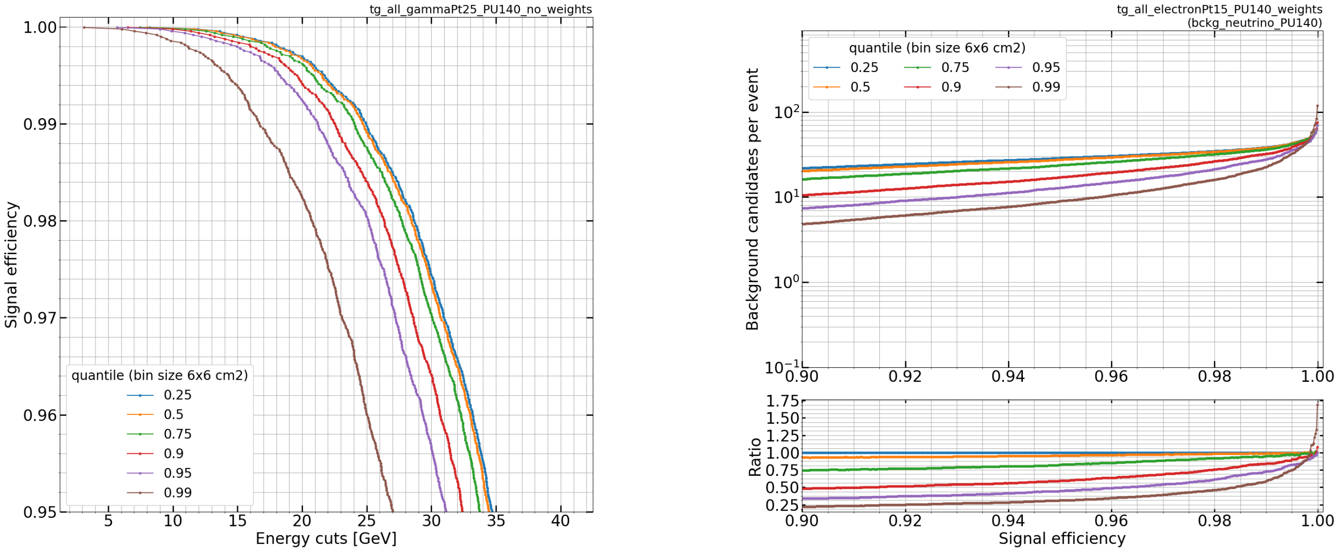


Figure 5.19: SE for photons of 25GeV (PU=140) where different quantiles are applied (left). ROC curves of the improved tracking with identification for electrons of 15GeV (PU140) with different quantiles and bin size 6x6cm^2(right).

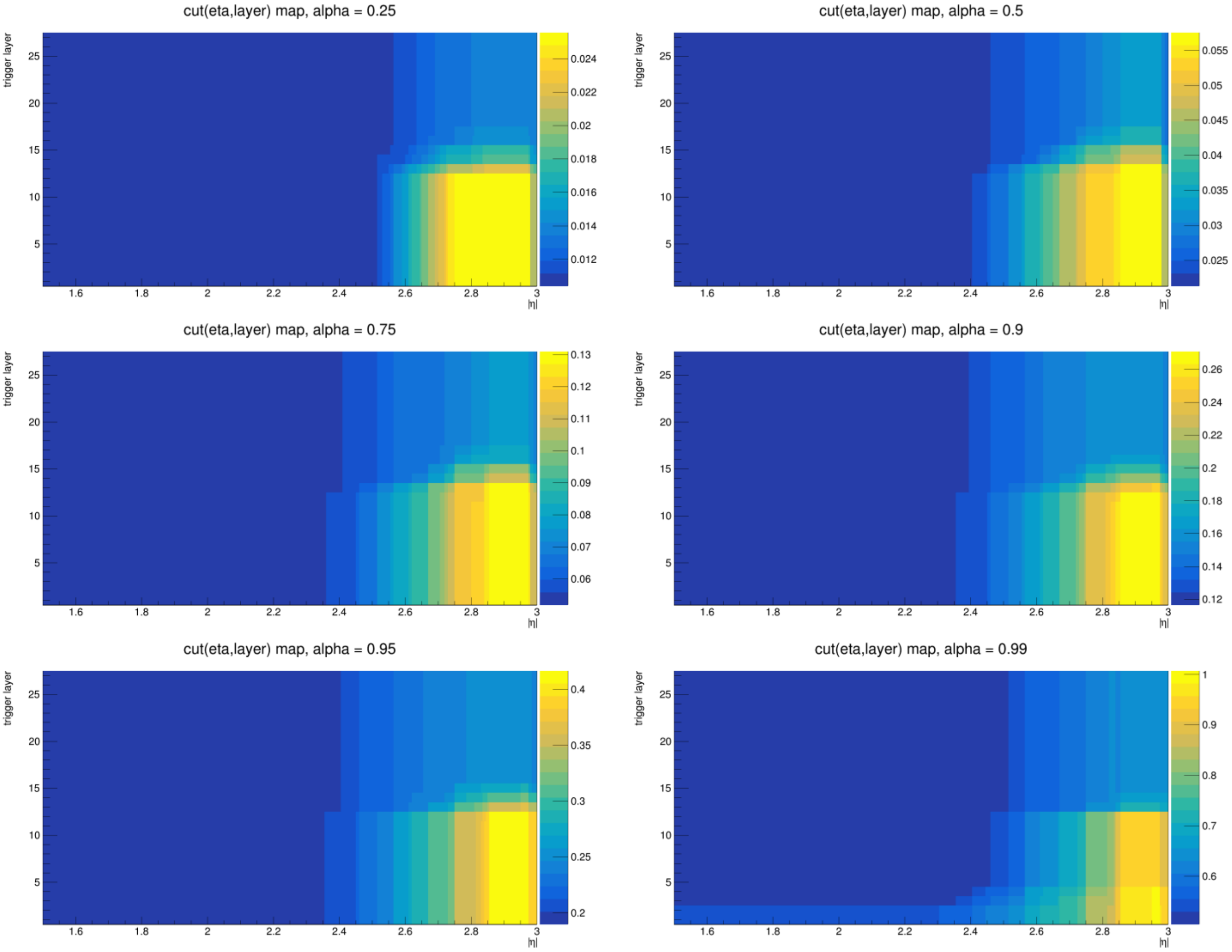


Figure 5.20: TC threshold maps for different quantiles. The z axis label (color) is the TC energy cut value. These plots show the cut values applied in different layers and eta regions.

notice the lower background reduction in the case of electrons. This doesn't have to be only due to the different profile used (as we believe that the effect of the profile is negligible as long as it is EM-like) but it can be due to to the lower energy of the electrons. Basically, the SE is degraded more rapidly by the cuts when the energy is lower.

The conclusion of the study is that the method with TC threshold before the TA accumulation removes the "bump effect". We went on with $\alpha = .99$ quantile because this one eliminates the bump completely (Figure 5.21). For this quantile, we have repeated the binning study (ROC curves are produced for all bin sizes) to see if there is a reduced number of background candidates as we increase the bin size (as expected). The efficiency for photons pT=25GeV (PU=140) is given on Figure 5.22. Overall, this study shows that indeed when we remove the noise before the TC energy accumulation, we decrease the mean number of background candidates per event (for a fixed quantile used). The reduction is stronger when we increase the bin size. Again, due to the basic TA model improvement with shower identification, there is a strong background reduction (for all bin sizes) with identification included in the TA.

Bin space study: (r,c) parameter space

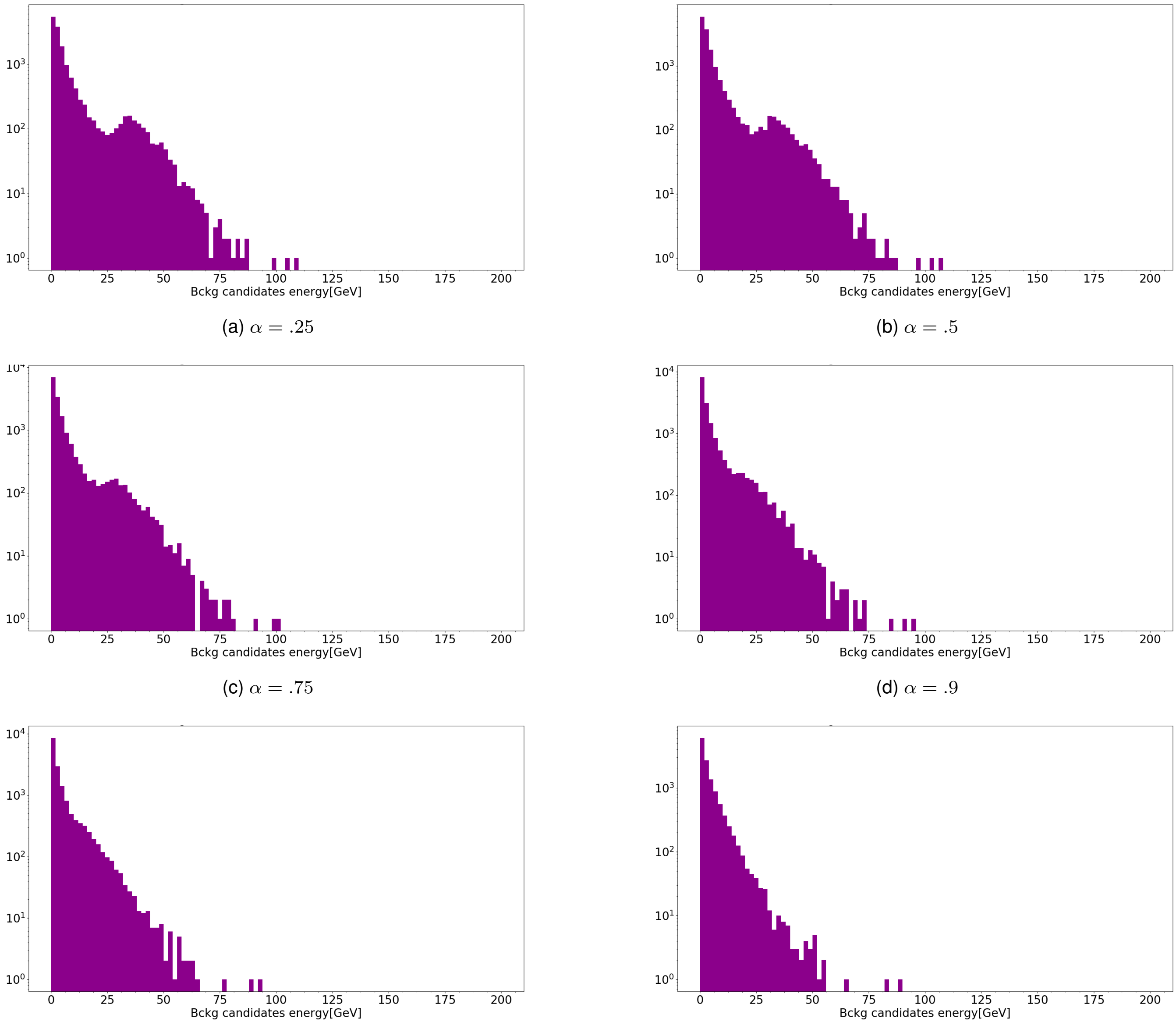
We can see on Figure 5.22 that increasing the bin size to 5x5 or 6x6cm^2 does not bring much enhancement, especially in the high SE range. Also, with using these bins we apply stronger cuts on the TC energies which can be avoided. The goal of the study described here is to qualify the drawback of using (r,c) parameter space for binning. The main question is how using the large bin size (6x6) affects the SE in the whole eta and phi range (with no PU).

We refer to the (r,c) coordinate space drawback as the "bin edge" effect. It is not specific to (r,c), possibly any binning will drop the efficiency at the edges of bins, but here we study the effect on our selected binning option. In the experimental setup, we select working points with SE=90%, SE=95% and SE=99% and we apply the corresponding thresholds to the seed candidates. We compare the SE as a function of η and as a function of ϕ, i.e. SE(η) and SE(ϕ), and two bin sizes are compared (3x3cm^2 and 6x6cm^2).

It can be seen on Figure 5.23 that SE(η) is more continuous for the bin size of 3x3, while for the bin 6x6 there are "holes". The SE is growing with η because there is more energy in the high eta region. Figure 5.23 shows the "bin edge" effect in action; one can see a large difference between SE at the bin center and at the bin edges for 6x6, caused by binning that is too large. The effect depends of the particle position and the energy spread (Figure 5.24). For a fixed particle spread and a position close to the large bin center, the energy is fully contained in one bin if the bin is too large, while for the smaller bin size the energy is spread in more bins causing a more continuous SE curve. Also, for a fixed particle spread and a position close to the large bin edge, the energy is shared between the two large bins while for the smaller bin size we have once again a more continuous SE.

A possible solution to the "bin edge" effect is thus to use smaller bins (Figure 5.25). It means to simulate 6x6cm^2 bins by using e.g. 2x2cm^2 bins, and the seed candidate is a sum of 3x3 bins (3x3 window) around the local maximum. Hence, the 3x3 window of 2x2cm^2 bins accumulates the energy of the candidate on the projected map.

Results are given on Figure 5.26 where we plot the $pT_{matched}(\eta)$ and $pT_{matched}(\phi)$, i.e. the pT of the seed



(a) $\alpha = .25$

(b) $\alpha = .5$

(c) $\alpha = .75$

(d) $\alpha = .9$

(e) $\alpha = .95$

(f) $\alpha = .99$

Figure 5.21: Background candidates distribution with thresholds that correspond to different quantiles applied (bin size 6x6cm^2). The quantile value α is given.

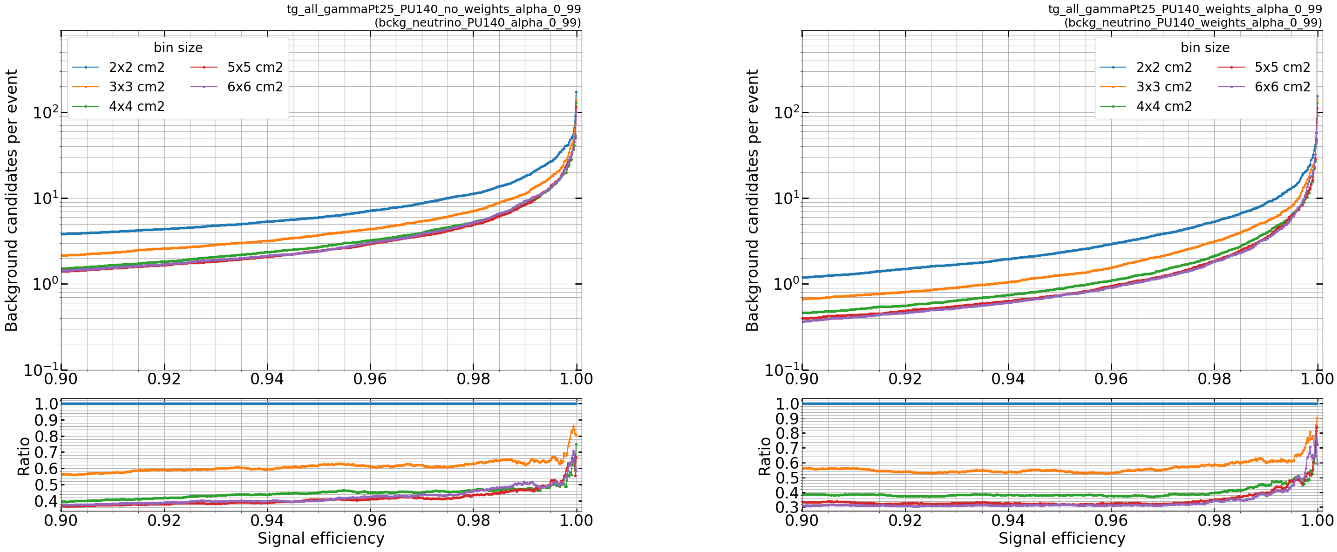
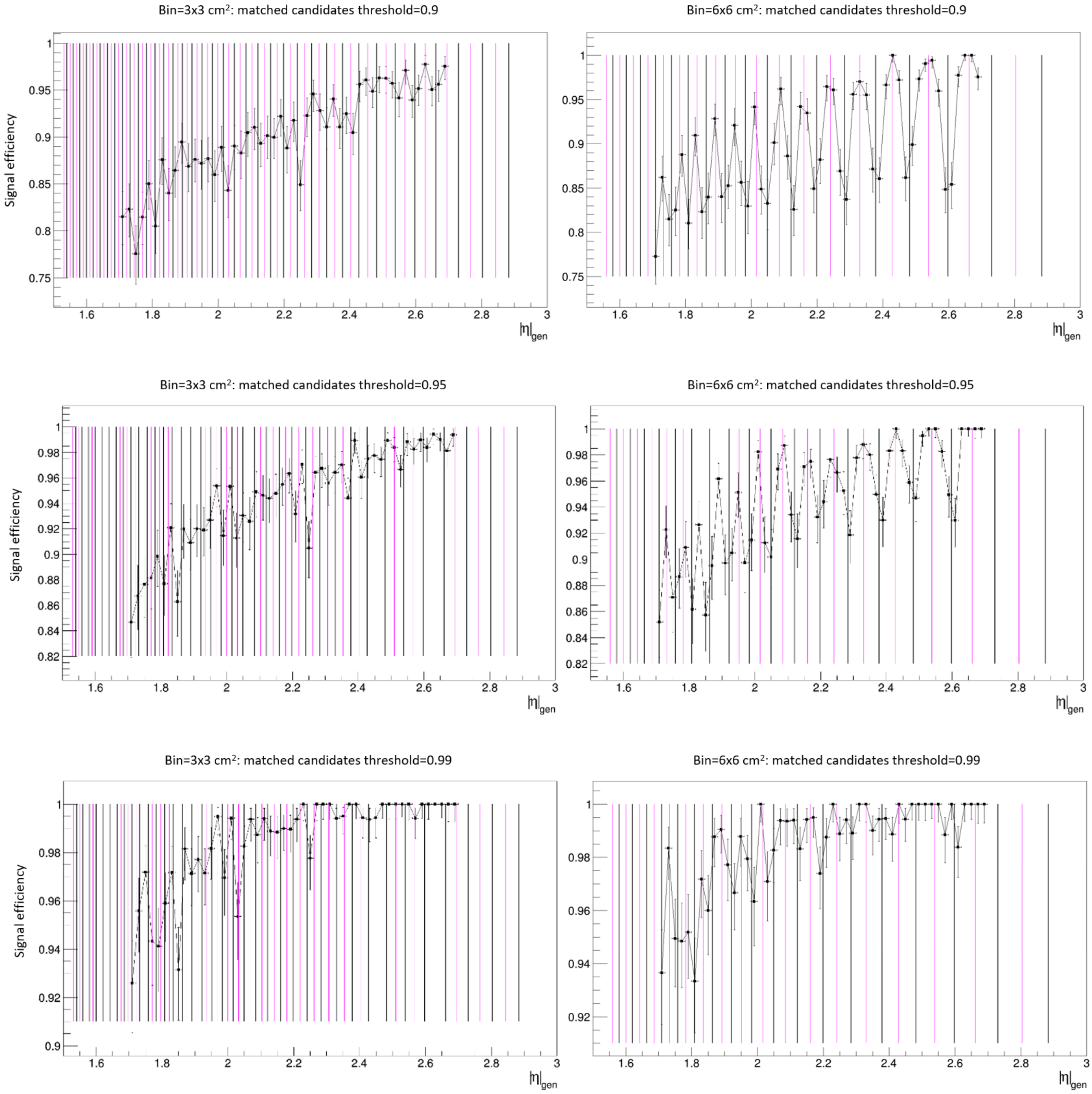


Figure 5.22: TA performance for photons of 25GeV with PU140 with the fixed quantile ($\alpha = .99$) and different bin sizes. Basic TA model (left) and the improved TA model with identification (right).

candidate that is matched to the particle. The sum of the smaller bins window is now used as a seed candidate energy to make the SE curve more continuous, with no "holes", as shown on the left plot in Figure 5.26. Also, we can notice larger energy for windowing applied, i.e. with smaller bins used than 6x6cm^2. The reason is that even if the area of the bin is the same in both cases, the center of the window is always in the central local maximum position of the seed. Probably, the window size 9x9 is too large, as it manifests in larger fluctuations. The results shown on the right plot of Figure 5.26 reveal that the seed candidate pT is lower for the TCs accumulated in the central bins (or bins positioned at the center of the detector), compared to the bins at the detector border. Since the (r, c) parameter space is based on the transformation Formula 5.10, the border is located at $\pi * r$, as $\phi \in [-\pi, \pi]$. Thus, central bins are at $\phi = 0$ and the border bins are $\phi = \pm\pi$. This plot indicates that the same event is reconstructed differently in the central detector part and at the borders, which is a major drawback of this parameter space. Naturally, if the window is large enough, it becomes more constant, so the effect is less evident.

In order to better visualize the effect, we rotated the (r, c) histogram such that we can look at the same event reconstructed in the center and at the borders. If the (r, c) binning space was rotation invariant, we would see the same pT of the matched candidate in both cases and the same bin energies. This would mean that the same event is reconstructed the same way. However, it can be seen on Figure 5.27 that this is not the case. The sum of the event energy is indeed the same, but the energy spread is larger in the binning at the borders, which causes the pT of the matched candidate to be smaller.


Bin=3x3 cm^2: matched candidates threshold=0.9

Bin=6x6 cm^2: matched candidates threshold=0.9

Bin=3x3 cm^2: matched candidates threshold=0.95

Bin=6x6 cm^2: matched candidates threshold=0.95

Bin=3x3 cm^2: matched candidates threshold=0.99

Bin=6x6 cm^2: matched candidates threshold=0.99

Figure 5.23: Efficiency graphs: SE as a function of η for working points 90%, 95% and 99%. Left: bin 3x3 and right: bin 6x6. The bin center is marked with a pink line and the bin edge is marked in black.

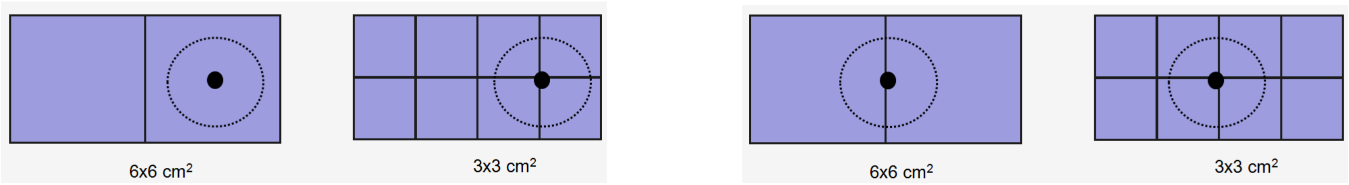


Figure 5.24: Schematic of the "bin edge" effect.

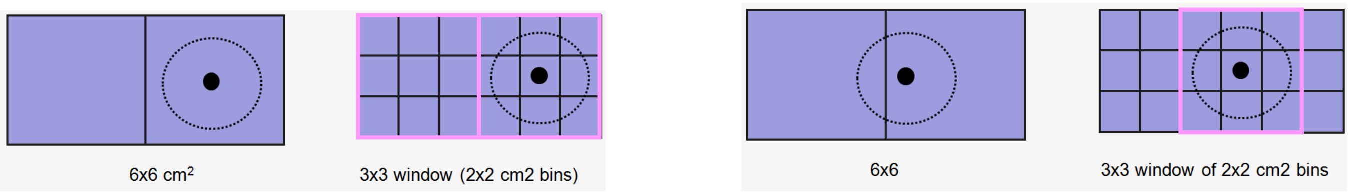


Figure 5.25: The schematic of the "bin edge" effect solution with windowing.

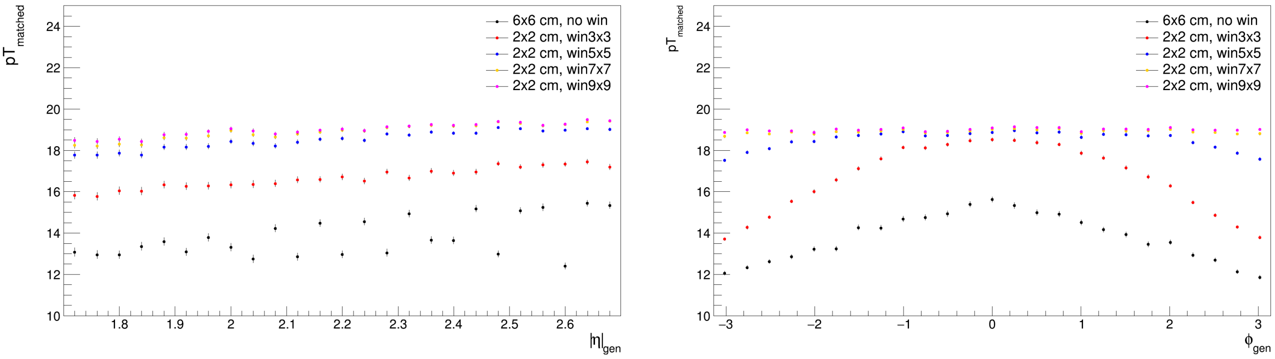


Figure 5.26: Window mechanism applied to solve the "bin edge" effect.

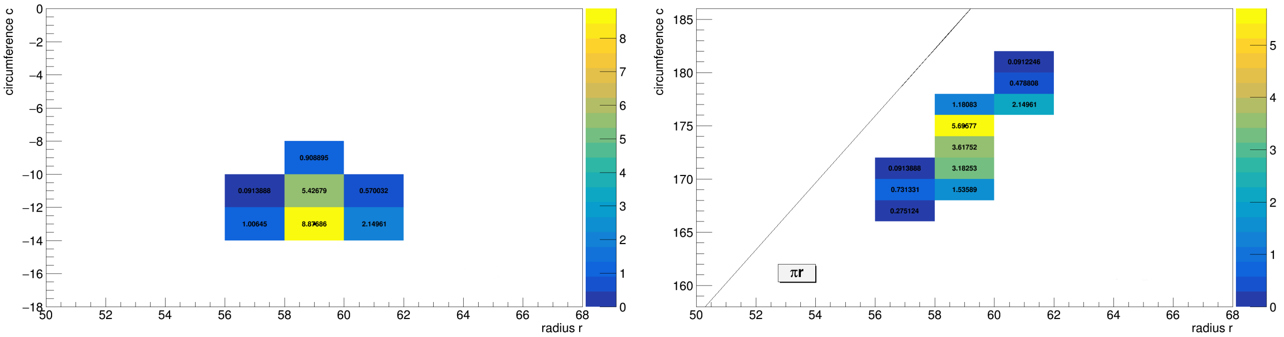


Figure 5.27: Visualization of the (r,c) drawback. Event reconstructed at the center (left) and at the border (right).

Bin space study: the mixed binning in (eta, phi)

As shown previously, there is a large energy spread between bins at the edge causing the smaller pT of the matched seed candidate which decreases the SE. Hence, it would be better to use (x,y) or (η, ϕ) binning space. The advantage of (x,y) is to stay in [cm], while (η, ϕ) is standard for a cylinder detector structure. In this section, we examine the (η, ϕ) binning used in the TA algorithm.

We decided to simulate the bin size 2x2cm^2 which corresponds to the TC size, in order to apply the TC granularity projection. Hence, since we are not in [cm] any more, we calculate $\Delta\eta$ and $\Delta\phi$ that correspond to 1cm. The values $r_{min} = 33.8, r_{max} = 236.04, z_{layer1} = 320.755$ are extracted from the CMSSW simulation in [cm]. The bin size calculation is given below:

$$\eta = \sinh^{-1}(\frac{z}{r}); \quad \Delta\eta = \frac{z * \Delta r}{(r * \sqrt{(r^2 + z^2)})}; \quad \Delta\phi_{min} = \Delta c / r_{max} \quad ; \Delta\phi_{max} = \Delta c / r_{min} \tag{5.20}$$

It follows for $\Delta r = 1cm$ and $\Delta c = 1cm$ that:

$$\Delta\eta_{min}(1cm) = 0.005; \quad \Delta\eta_{max}(1cm) = 0.025; \quad \Delta\phi_{min}(1cm) = 0.004237; \quad \Delta\phi_{max}(1cm) = 0.02957 \tag{5.21}$$

For this study, we derive 5 binnings that correspond to $2cm$, i.e. $\Delta\eta = [0.01, 0.02, 0.03, 0.04, 0.05]$ and $\Delta\phi = [0.01, 0.02, 0.03, 0.04, 0.05]$ and we calculate the mean number of TCs per bin for each bin size, where the minimal number of empty bins is required. The maps are shown on Figure 5.28, and they confirm that mean number of TCs per bin is increasing with the increased (η, ϕ) bin size, and that the number of empty bins is smaller when the bin is large enough. This analysis enables us to choose the bin size, since we would like smaller bins to keep a fine granularity, but also bins should not be too small otherwise "holes" appear in the accumulation space. The optimal bin sizes are extracted:

- Low η: 0.01x0.01, 0.02x0.02, 0.03x0.03; High η: 0.04x0.04, 0.05x0.05

The TA results are repeated and, again, we observe a reduced number of background candidates for the same SE for the increased bin size, and the reduction is stronger with the improved TA model (Figure 5.29). Figure 5.30 shows the matched candidate pT plotted as a function of η and ϕ. In order to solve the "bin edge effect", a 3x3 window sum is implemented. There is a decrease in efficiency for the lowest bin size with the increased η value. This is because in TC projected maps there are a lot of empty bins for lower bin sizes used in high η range, which means that a higher η region must have larger bin sizes. For the pT(ϕ) curve, it can be seen that there is no drawback of accumulating less energy at the borders of the detector than at the detector center, which means that the (η, ϕ) binning space is better to use than (r, c).

Finally, we examine the advantage of having two bin sizes depending on η; smaller bins in low η ($|\eta| < 2$)

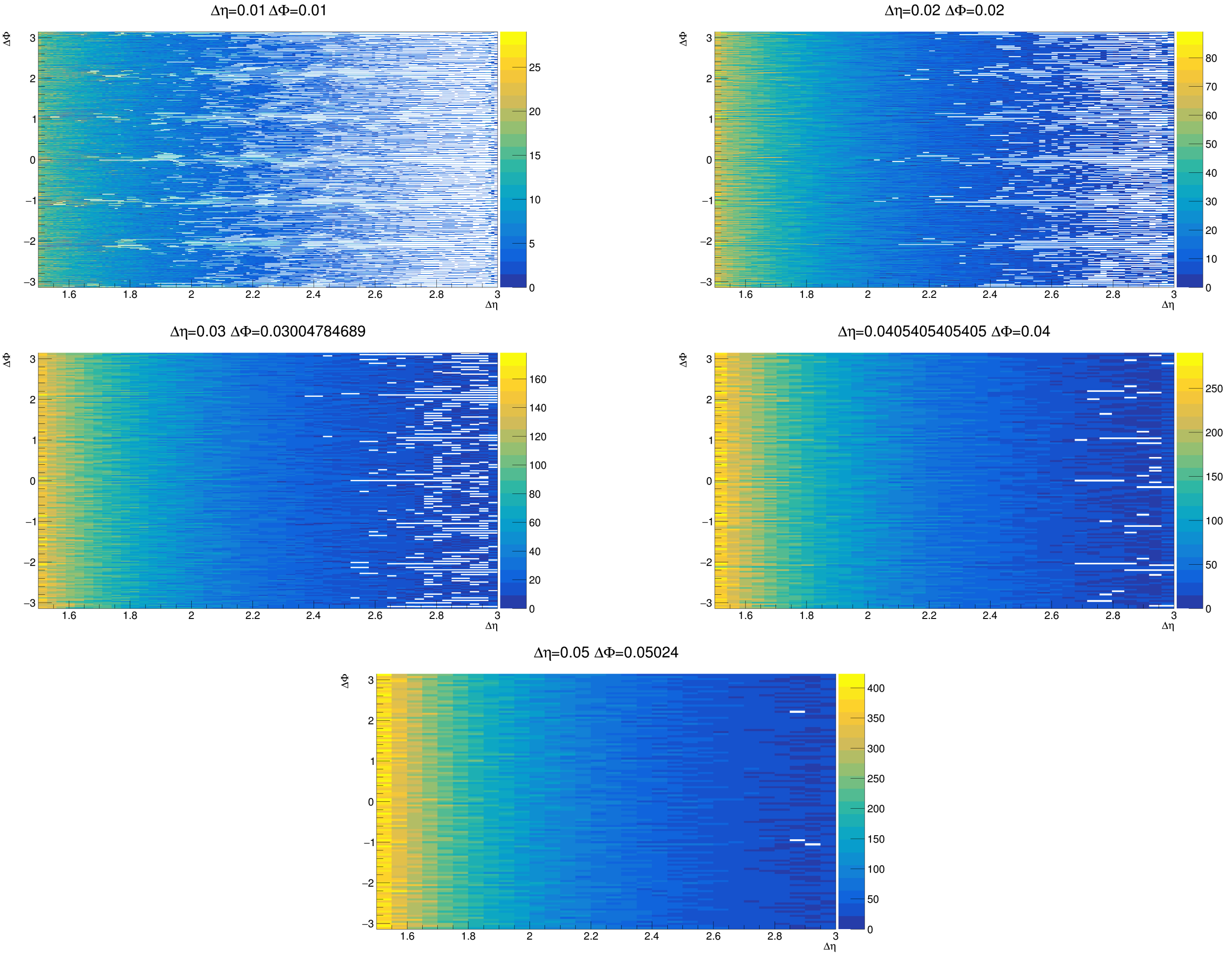


Figure 5.28: Projected TCs maps for ECAL showing the mean number of TCs per bin (the z axis label (map color)).

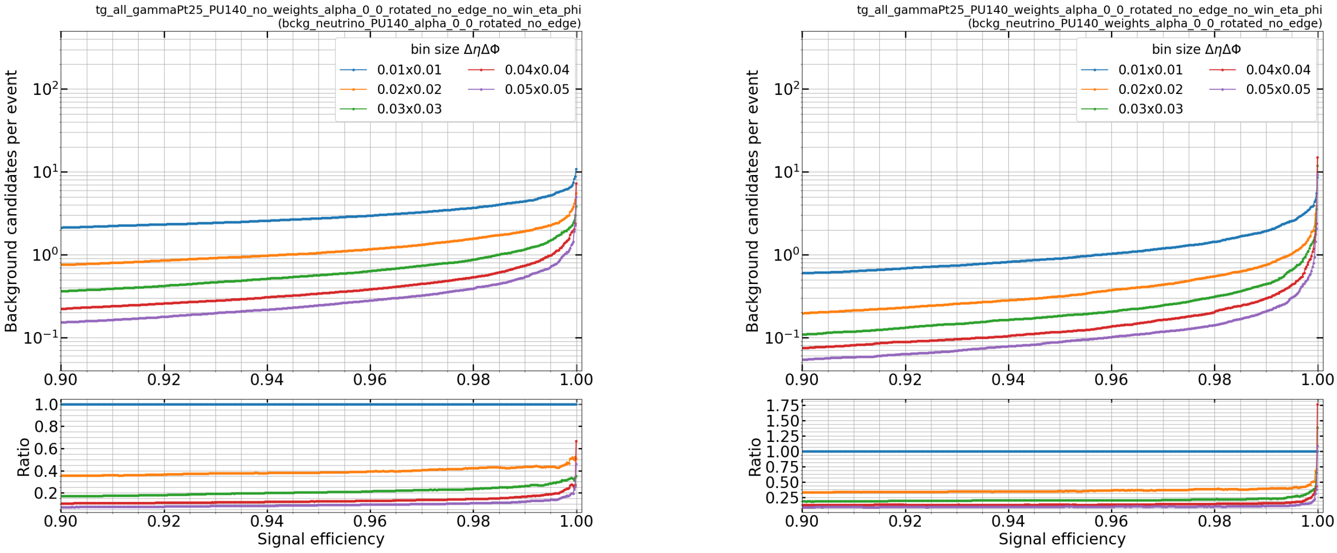


Figure 5.29: TA binning study repeated in (η, ϕ) binning space for photons of 25GeV with PU140. Basic TA model (left) and the improved TA model with identification (right).

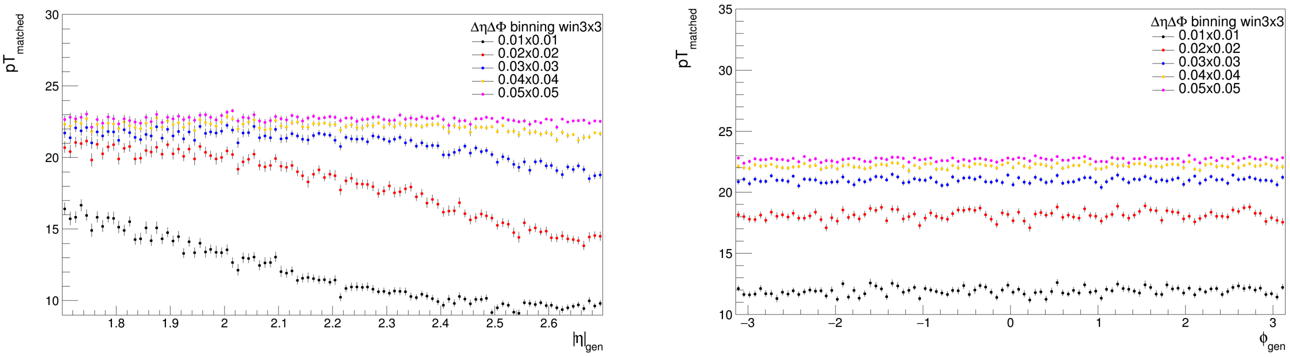


Figure 5.30: Matched seed candidate pT, plotted as a function of η (left) and ϕ (right). The (η, ϕ) binning is used.

and larger bins in high η region ($|\eta| > 2$). The mixed size (η, ϕ) binning efficiency is given on Figure 5.31. We select the bin sizes 0.02x0.02 and 0.04x0.04 to be used in the mixed binning study and we compare candidates pT distribution for different photon energies (pT=25GeV, 35GeV and 50GeV). As the energy is increased with larger pT, larger efficiency is found, and we can see an almost constant dependency in pT(ϕ). Overall, the conclusion is that a mixed (η, ϕ) binning can be used for an efficient photon EM shower energy reconstruction.

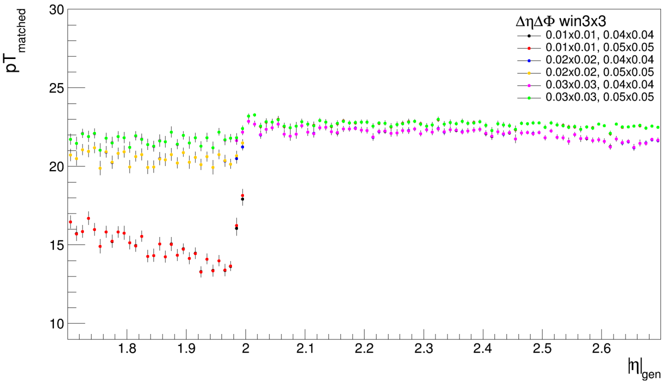


Figure 5.31: Mixed (η, ϕ) binning results given as pT(η).

5.2.7 Discussion and evaluation

The basic goal of the reconstruction studies was to examine how a simple but efficient seeding and tracking of EM showers can be performed in the detector volume. The idea is to use the full 3D information from the detector layered structure and to do a direct 3D clustering instead of the 2D layer-by-layer approach. The main motivation for the work comes from the L1 trigger main task, which is to decide whether the detector data is "interesting", and to categorize it as signal or background. It means that the data selected and reduced by the FE should be reconstructed in the BE part of the L1 trigger chain. The main intention is oriented towards developing efficient reconstruction algorithms such that the TC energies are recognized as part of the EM shower, forming the EM cluster shape.

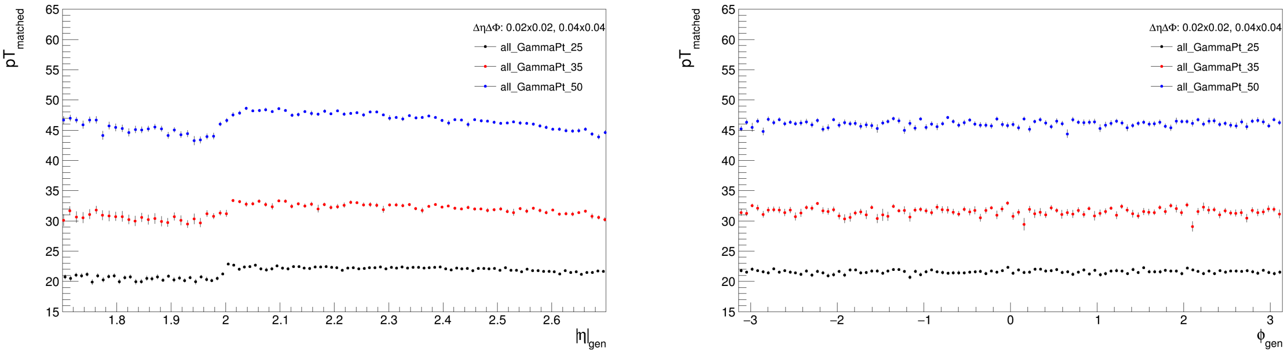


Figure 5.32: Mixed (η, ϕ) binning with two bin sizes 0.02x0.02 and 0.04x0.04 (photons 25GeV, 35GeV and 50GeV). pT(η) (left) and pT(ϕ) (right).

Performing the 3D clustering in the whole detector at once is resource-consuming, so it is better to identify "interesting" regions in the detector and further process only these ROIs. Hence, in our studies we designed an algorithm, which we call the TA, and we examine how efficient it is in finding these regions. During the design of the algorithm, we apply ideas and strategies from the existing state-of-the-art, which have already shown a great potential in HEP applications. For instance, we apply a technique similar to the classical HT to transform the coordinates for an efficient TC energy projection. We show that it is possible to successfully transform the coordinates and to perform a detector mapping by using the information from all the detector layers at once. Similar to what is used in artificial retina algorithm, but much simpler procedures are implemented, as there is no additional averaging kernel applied on the accumulation map. There is only a central local maximum filter used to recognize the positions of the reconstructed seeds.

Also, another idea inspired by retina is adopted, where every hit does not have to contribute to the receptor with the same weight factor. Following a similar logic, every TC energy in HGCAL can be weighted in the L1 tracking algorithm, depending on the layer it belongs to. In that case, the EM shower identification can be accomplished using the longitudinal energy profile of the signal, which can enhance the performance of the data reduction algorithm. Also, it can intelligently decrease the data volume when using this heuristic approach. Our study has shown that the former identification strategy enables an enhanced seeding algorithm with a reduced number of background seeds with respect to the non-weighted case where the basic TA model is applied. This is very important in the high PU scenario foreseen in the HL-LHC.

This research presents the first (seeding) step in the possible strategy for 3D reconstruction algorithm. The EM shower can be approximated by a straight line coming from the centre of the detector and, after identifying the seed or ROI positions, one can reconstruct the shower line and select the energies around the seeds in depth.

The TA seeding efficiency is explored by choosing the optimal algorithm parameters. During the TA design, we were constantly motivated by keeping its implementation in hardware as simple as possible. For example, seeding and binning (mapping TCs to bins) can be defined in an FPGA lookup table (LUT). Once the coded TC energy is received at the FPGA input link, and considering that the code word consists of module and TC address, we can read from the LUT to which bin this TC belongs to. Also, the EM profile can be encoded in another LUT, where layers can be mapped to the corresponding energy weights. The TA bin size importance in hardware is obvious. For the logic resources used in a FPGA, it would be better to have larger bin size leading to lower number of bins in the accumulation map and thus reducing the memory usage (smaller LUT). Also, an important conclusion from the profile study is that there is not much difference in SE when applying full or reduced profile for the EM shower identification. Hence, less multiplications can be needed in hardware with using the filtered profile.

We discussed which parameter space to use in the 2D binning; (r, c), (x, y) or (η, ϕ). In general, (r, c) can be better than (η, ϕ) as all the bins are the same in size on the whole endcap and both parameters are in centimeters where the EM shower "lives". The Cartesian (x, y) coordinates can be also as good as (r, c) for the single shower, while (η, ϕ) is better for multiple showers or bremsstrahlung initiated by electrons whose trajectory is bending in the ϕ direction. One more benefit of (η, ϕ) is that we have the same number of bins in the whole ring (for some η), which means that the number of bins is the same in the inner-most and in the outer-most ring (although different in size).

Our analysis showed that (r, c) is not the best option to use, since due to the rotation invariance and "the bump effect" it would be better to use the (x, y) or (η, ϕ) coordinate systems. There is no need to apply a threshold on the TC energies before accumulating on the virtual grid as in the former case of (r, c) used. However, it can be applied to accumulate less PU with larger bins. To finalize the TA parameters selection, (x, y) is considered to be the optimal parameter space to go on with and this one is applied in the process of generating the database of images in the ML study of Chapter 6. We consider that the mixed (η, ϕ) binning with smaller bins in low $|\eta|$ and larger bins in high $|\eta|$ region could complicate the image generation procedure and we decided to omit this. Also, a limitation of the study is that there is always a single quantile used, but one could define two separate quantiles (depending on $|\eta|$) and in this way remove more PU in the high $|\eta|$ region. The profile study showed that there is a great potential of separating signal TC energies from the background TCs with using the known EM shower profile. However, further study is done in this direction, to better discriminate between signal and background by using the ML techniques with the neural networks.

Chapter 6

Shower data classification with machine learning

ML techniques have shown a great potential for event classification and object classification tasks in HEP. There are two main reasons for this; first, convolutional neural networks (CNN) outperformed the traditional approaches, and they are robust to noise, which is a usual environment in HEP experiments. Another crucial factor for a ML application is the data reduction, as experiments such as those at the LHC are one of the largest big data sources. In this context, this chapter deals with data reduction schemes that utilize the image-based representation of data originated from collision events. Physics detectors are regarded as cameras, where the high-dimensional sensor data captured from the event is converted into the summarized form of a digital image. The sets of images are given to ML models to perform classification tasks.

A Section 6.2, is a review on the ML techniques used in HEP. The main goal is to explore the image-based event classification reported, and how event images can be generated and used in efficient ML models and frameworks. There are many research directions revealed, needed to fill the gaps in the current literature. Some of these directions are followed in the ML study described in Sections 6.3, 6.4, 6.5 and 6.6. The goal is to test the classification functionality between EM (signal with PU) and PU (background) event images. We are motivated by the possible ML application in the trigger, where the main restriction is the very limited real time processing. The trigger context requires that the trained ML network is robust, fast and simple enough to fit in the BE FPGA. Hence, we avoid CNNs (as they are much more complicated in hardware) and reduce the NNet to only few dense layers. Also, we prepare the database of the event data images by using the designed TA from Chapter 5. The main intention with the ML study it is to examine if the successful functionality of EM-like versus PU-like classification can be obtained with the reduced NNet. This would be very impelling, providing motivation to go on and implement this NNet in hardware. Finally, we would test the potential of using the ML techniques in the very early trigger level.

6.1 Theoretical background on the neural networks

In this section, we provide a theoretical background needed to introduce the basic ideas and principles of the NNets. The NNet terminology with the technical terms will be shortly described, used throughout the whole chapter.

6.1.1 The terminology

A NNet is a simplified model imitating the functionality of the human brain [114]. The brain functionality itself is a very complicated mechanism that consists of millions of interconnected neurons, where each of them learns by its own experience as well as the received experiences of other neurons on the input. Thus, simulating the human brain in terms of the NNet is a complex task, especially when we need to define a set of rules that classify something into a specific class. The definition of the former set of rules can be considered "learning", and the supervised type of learning is connected to the term "learning by examples". This is accomplished with the use of NNet, where each of the parallel input-output transformations is used to set up the parameters of the NNet during the training phase by using the known images, such that the NNet would be taught to generalize and decide on the new unseen data. The NNet is based on neurons, and a special type of neuron is called a perceptron. It calculates the sum of all the inputs in the input vector $X = x_1, x_2, ..., x_n$, multiplied by the vector of weights $W = w_1, w_2, ..., w_n$.

Thus, the output of each neuron y is calculated by using a formula $y = X \cdot W + b$, where b is a neuron bias. Hence, the output of each neuron y_i is the weighted sum, which is transformed by applying a non-linear activation function. One of the commonly used activation functions are Rectified Linear Unit (ReLU):

$$R_{y_i} = max(0, y_i) \tag{6.1}$$

A softmax function is commonly used in the last classification layer, normalizing the outputs to a range [0, 1]:

$$S_{y_i} = \frac{e^{y_i}}{\sum_j e^{y_i}} \tag{6.2}$$

The NNet consists of minimally three layers: the input layer, the hidden layer, and the output layer. The input layer is defined by the size of the input data, and the output layer nodes correspond to the number of classes the input data is classified in. The effect of overfitting (with large number of hidden layers) should be reduced by the carefully chosen set of network hyper-parameters, because it causes the over-trained NNet that does not generalize well. The way the NNet "learns" is that it iteratively calculates the loss function, which measures the error (the difference between the actual output and the predicted output), and tends to minimize it. The learning algorithm is a gradient descent, which updates the parameters during the back propagation through the network. We start from the initial set of weights and bias and calculate the error from these internal model values. Next, the error is propagated from the current neuron to the neuron of the previous layer, which updates the values of weights and biases to minimize the error. For this, the NNet uses the optimization algorithm which calculates the gradient and updates the values in several iterations until the minimal loss is reached.

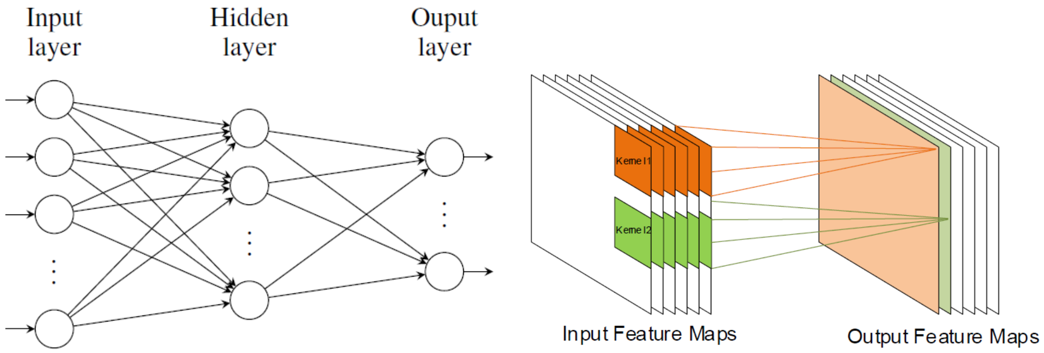


Figure 6.1: Fully connected neural network model (left) and the input and output feature maps of a convolutional layer in CNN [115, 116].

The cross-entropy loss function calculates the difference between the true distribution and the predicted distribution. The true distribution is a set of probability values for the target (one-hot encoded vector with a single 1 for the true class and 0 for other classes) and the predicted values are n probabilities for n classes. A perfect model would have the lowest possible difference between targeted and predicted output values. For the multi-class classification, the categorical cross-entropy is used, which calculates a loss for each class and provides the sum as the result.

A process of training the NNet is done in the selected number of epochs, where each epoch consists of batches of the training data. The number of epochs defines how many rounds of learning would be performed on the training data. In each round the batches of the input data set are used to update the model parameters, and the process is repeated until all the training data is used. Setting a large number of epochs allows the learning algorithm to run until fully minimizing the error, but again the overfitting should be prevented.

The learning rate defines the amount that the weights are updated during training. This parameter is set initially to a small value between 0 and 1, and it stays constant if a classical gradient descent is used. In the case of the optimizers such as Adam, the learning rate is adaptive and it changes during training.

6.1.2 The hardware implementation challenges

The CNNs are similar to NNets from the previous section, but includes one or more convolutional layers with respect to the fully connected neural network (FCNN). The convolutional layers use filters in order to extract the low-level features from the input image. The filter results are given as input to an activation function, whose output is passed to the next layer. Usually, the next layer is pooling performed by a downsampling operation (maximum or average). In the fully connected (FC) layer each input is connected to all neurons, and each neuron is connected to all others. These FC layers are used to optimize the class scores, as higher level features are processed, which ultimately leads to a complete understanding of the image.

In the past few years, there is an increasing demand for real-time hardware implementation of ML applications, especially CNNs. They have shown great potential for numerous applications, where a high accuracy can be obtained. However, CNN exhibits the increased complexity being computationally expensive, so the main problem is the real-time implementation in hardware. Usually, the demand is to maximize accuracy while keeping low latency requirements [117]. Since in most cases the maximally available latency is a few hundred milliseconds, CNNs have been used in the offline processing in HEP. In Section 6.2, we provide a survey on such applications. Not much interest is devoted to using CNNs in the L1 trigger, where the timing is critical requirement, with the maximal latency of a few microseconds.

The computational complexity of CNNs is large because of the convolutional layers, where most of the multiplications are done, calculating the convolution of the windowed image subpart and the filter coefficients. The calculations are done following a sliding window procedure and it is repeated until the whole image is scanned. The hardware implementation of the 2D convolution is complex, and it grows with the increased number of convolutional layers, where each neuron performs multiplication and accumulation (MAC) function.

The computational cost of the FC layers is much lower than the convolutional layers, but the problem here is the huge number of parameters. The hardware implementation is easier since all neurons of a single dense layer can work in parallel and perform the MAC operations by using the independent set of weights as soon as the inputs to the layer become available. The approach is semi-parallel, as, for the input image 32x32=1024 and 512 neurons, they all share the same input pixels while having different weights, so there are 1024 clock cycles needed to process the full image [117]. Therefore, FCNN can be simply implemented in hardware, without the complex logic. Also, the number of multiplications is reduced compared to CNNs. While in the CNN the parameters are shared between neurons, causing a smaller total number of parameters in the model, each neuron in FC layers has its weights. The number of parameters is growing with the number of neurons in the dense layers. Therefore, the number of memory accesses is reduced in CNN, whereas FCNN suffers from high power consumption due to the numerous memory accesses. These are required to load the parameters and store the results that need to be loaded again by the neurons in the next layer.

The main problem in FCNN is how to reduce the number of parameters and the pruning techniques successfully addressed this issue, by removing the least important weights and neurons from the network model. This way, models can be small enough to fit the memory of the ASIC or FPGA [115]. Also, the memory requirements can be reduced by weight quantization, such that the memory width is smaller. The usual size of the inputs, outputs, weights and biases is a 32-bit floating-point used for training, about this one should be avoided for the FPGA. Using weights with reduced fixed-point precision and operations between them requires less logic resources and higher hardware efficiency.

6.2 State of the art and open research directions

ML techniques have been commonly used in the HEP event selection for many years. Guest et al. ([118]) have provided a systematic survey on this topic, explaining the advantages of ML with respect to the traditional approaches [119, 120, 121, 122].

There are several reasons that make ML important for HEP applications. The CNNs work well in noisy scenarios, which will be even more important in the new era of the HL-LHC, bringing the much more complex and noisy events to handle. Another reason for ML in HEP comes from their ability to improve data reduction [118], and the tendency is to convert the raw high-dimensional sensor data from the detector, into the selected forms of reduced dimensions. The latter can be referred to as event-data summaries, and based on them, the NNet is used for event classification and selection. In this section, we are interested in the possibility to reduce events into an image-based representation. Physics detectors are considered as cameras, and the event energy distribution is summarized in the form of a digital image, where pixel levels correspond to the accumulated energy. The advantage of using NNets to classify the generated images is that they are independent of the domain knowledge [123, 124, 125, 126, 127], since network can learn the structures in the input images on their own.

The literature survey presented in this section provides an overview of the existing studies and reveals gaps together with future possible research directions based on prior work. Other questions answered in this section are:

- RQ1: Which are the usual approaches and strategies for an image-based classification in HEP?
- RQ2: How to generate images from the event data, and which image pre-processing techniques are applied?
- RQ3: Which are the future research directions needed to fill the revealed gaps from the literature?

In total, 30 scientific papers are collected (Table 6.1) and classified in 5 conference ([128, 129, 130, 131, 132]) and 25 journal papers ([125, 126, 127, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154]) covering a time range from year 2014 to 2020.

Table 6.1: Selected papers in the review.

	Selected papers	Total
Conference	[128, 129, 130, 131, 132]	5
Journal	[125, 126, 127, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154]	25
Total		30

We categorized the papers depending on the CNN classification input, i.e. whether authors use single-channel or multiple-channel input data images. As obtained from the analysis of the publication years, it is shown on Figure 6.2 that the multi-channel approach gained a lot of attention in the last years, and its research has just continued at the beginning of 2020. Also, the larger number of papers in the last years reflects the high popularity of the current topic. It is shown in Table 6.2 that there are 15/30=50% of the reference papers in the single-channel class, and most of them have an image pre-processing applied. On the other hand, most papers in the second class are without input pre-processing (12/15=80%). This is very important since these techniques usually require a specific knowledge from physics. For instance, the classification can be improved if the image is enhanced in a correct manner, but the precondition of having the physics knowledge somehow makes the classification tasks not directly available to other computer science domain experts. Hence, the general idea in the literature is to avoid any pre-processing step to simplify the classification task and let the CNN learn the initial raw image structure on its own, without any physics-driven inputs [126].

The authors in [138, 148, 154] use a mixed approach, combining both, studies with single-channel and multi-channel images. It can be seen in Table 6.2 that not much work has been devoted to the specific parameters like image sparsity and events with PU added. For example, sparsity is considered in [124, 126, 128, 147, 151, 152]. Also, the authors in [126, 141] apply a threshold on the energy to reduce the impact of the PU before creating the detector image. Baldi et al. [127, 152] consider more realistic scenarios and compare PU effect on the CNN classification performances.

Table 6.2: Papers divided by input image and other parameters.

CNN input image			
Single-channel		Multi-channel	
No processing	Processing	No processing	Processing
[126, 134, 138]	[125, 127, 130, 140, 141, 142, 143, 146, 147, 148, 149, 150]	[121, 133, 136, 129, 131, 137, 144, 145, 152, 153, 132, 154]	[135, 139, 151]

Specific parameters considered	
PU added	Sparse images
[126, 127, 128, 130, 138, 141, 143, 152, 153]	[124, 126, 128, 147, 151, 152]

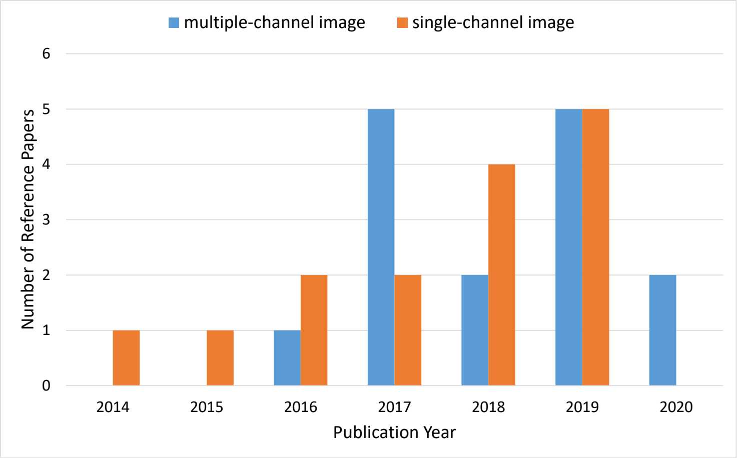


Figure 6.2: Distribution of reference papers by publication year.

6.2.1 Image-based shower data representation

Komiske et al. claim that a key question for ML approaches in particle physics is how to best represent and learn from event data [149]. Hence, the authors assume that a basic precondition to image-based classification in HEP is to represent physics event data as quality images. The images are obtained by using the detector geometry and specific design constraints that enable the data collection in the correct manner. The authors normally follow two approaches: they either project the data in one direction creating a single-channel event image, or they project the data in few dimensions to get different views of the same event in a multi-channel approach. We separate the papers using a special type of projection by means of a two-dimensional (2D) histogram, where a binning is done in a projection map such that all values that project to the same bin are summed together.

Both projection-based and histogram-based image generation procedures are systematized in Table 6.3 and Table 6.4. It is to notice that projections are usually done in Cartesian (x, y, z) coordinates, while for histogram-based accumulation map usually the $(\eta,\ \phi)$ cylindrical coordinate system is used. Also, the common number of channels in the multi-channel image approach is 3, to be close to the standard red, green and blue (RGB) scheme of coloured images.

Table 6.3: Generation of single-channel event images.

Ref.	Procedure	Coord. system	Size
[138]	projection	$(\eta,\ \phi)$, $(x,\ y)$	170x360 px, 100x100 px
[140]	histogram	(layer, channel)	12x60 px
[125, 126, 127, 130, 141, 142, 143, 146, 147, 148, 149]	histogram	$(\eta,\ \phi)$	25x25 px, 30x30 px, 32x32 px
[134, 150]	histogram	$(x,\ y)$	64x64 px, 25x25 px

Table 6.4: Generation of multi-channel event images.

Ref.	Procedure	Coord. system	Num. of channels
[132, 128, 139, 144, 131, 151, 129]	histogram	$(\eta,\ \phi)$	3
[137, 154]	projection	$(x,\ y)$, $(x,\ z)$, $(y,\ z)$	3
[133]	projection	$(x,\ z)$, $(y,\ z)$	4
[135]	projection	$(x,\ y)$, $(x,\ z)$, $(y,\ z)$	25
[145]	projection	$(x,\ z)$, $(y,\ z)$	2
[136]	histogram	$(\eta,\ \phi)$	10
[152]	projection	$(\eta,\ \phi)$	2

Single-channel images

The authors in [138] use the CMS detector geometry with three subdetector parts and develop three different detector images from the tracker, ECAL and HCAL detectors. Calorimeter images are constructed such that their pixels correspond approximately to physical ECAL crystals or HCAL towers. Tracker images are built as 2D histograms of the reconstructed (η, ϕ) tracks directions. In the case of ECAL and HCAL images, the low level detector feature is considered, i.e. energy deposits or reconstructed energy hits per calorimeter crystal (tower). These energy values are projected by summing over the ECAL crystals or the HCAL towers coordinates. This projection is based on eliminating the depth z-coordinate, and figures are reproduced from [155] using the CaloGAN tool [156]. The projections of a photon shower in three calorimeter layers is shown on Figure 6.3.

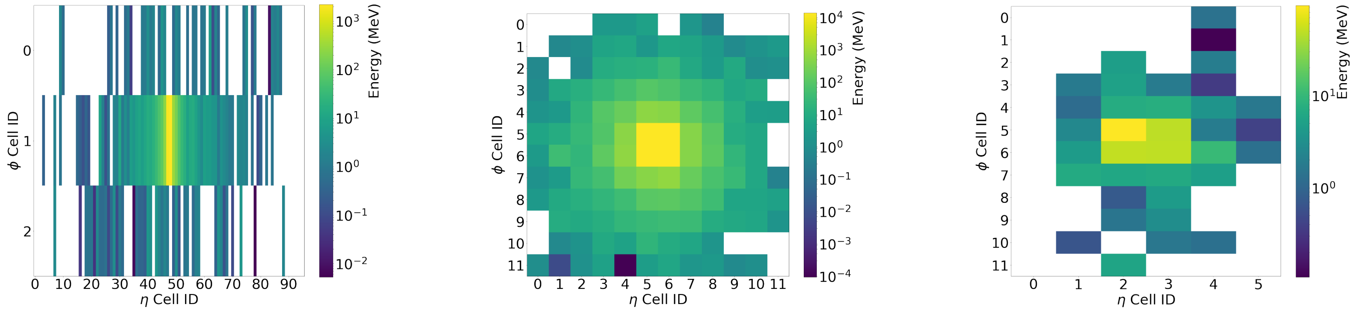


Figure 6.3: Projections of a photon shower per calorimeter layers. Reproduced with the code from [156].

The authors in [140] represent event data as histogram images, where each bin corresponds to a specific readout data channel per detector layer. The basic intention is to determine normal and faulty readout events in the context of anomaly detection. A 2D histogram is useful for this application, since bins with very low number of readout data (or a number close to zero) implies a faulty channel.

The authors in [125, 126, 127, 130, 141, 142, 143, 147, 148] create particle data images from the energy depositions, which are used to create image pixel intensities. Namely, an image is formed by discretizing the collection of particle energies into pixels in an (η, ϕ) map, such that the intensity of each pixel is the sum of the energy in a particular (η, ϕ) bin. Basically, the (η, ϕ) event data histogram is constructed by summing the energy deposited by the particles in each of the bins along the z-direction. In this way, the whole calorimeter is approximated by a single grid containing a 2D energy distribution.

Multi-channel images

The authors in [133, 145] extend the input image with an additional channel, to provide another view of the event data and improve the classification. In this case, two independent 2D images are provided based on the projections in (x, y) and (x, z) coordinate planes. Also, in [133], a context is added to each projection, based on the information from the whole event. For example, unlike for electrons that deposit their energy and produce the EM shower right after the entry point, photon-induced showers happen later in the detector. Measuring the gap between the entry point and the shower start can help us to distinguish between EM particles.

The authors in [135, 137, 154] capture the energy deposits in the detector 3D volume by combining measurements in three separate channels or an RGB-like image structure. Since the 3D detector image is a collection of voxels [137] defined by (x, y, z) of the energy sums of all hits that fall into the corresponding voxel, three independent projections can be defined to describe the event, i.e. (x, y), (x, z), (y, z). In the work of [154], another idea to apply a ROI-based classification is introduced, so that the classification is not done on the whole image but only on a specific image sub-region. Belayneh et al. [135] also propose a 3D shower image in a scheme with three separate planes, but unlike in [137, 154], another context of ECAL and HCAL calorimeters is considered when creating images. The authors create an event image by taking the 2D projection of the energy deposits in the ECAL and HCAL layer by layer, taking a cuboid slice into the detector volume around a predefined ROI.

Andrews et al. [131] use an image-based approach where the information from each CMS subdetector (tracker, ECAL, HCAL) is represented by the image histograms in the (η, ϕ) space. Images from all the three subdetectors are combined to form a single multi-channel event image.

Another way to partition the calorimeter deposits is to have images with an even larger number of channels, for example to have one or more input channels per particle [136, 129, 139]. Nguyen et al. [139] use the raw image of the detector hits and create an image of the event from the subdetector parts like the two forward regions, the barrel and the two endcap regions. One larger image is constructed in this way, binned into histograms where each bin is filled with the sum of the energies of the particles pointing to that bin. Three classes of particles (charged particles, photons, and neutral hadrons) are considered separately, resulting in three image channels.

Unlike previous works, where a larger image is created by capturing the whole event, Komiske et al. [144] use smaller images not covering the entire detector and use these to classify individual objects rather than entire events. The authors define a ROI in the detector and construct the images as square arrays in the (η, ϕ) coordinate space with each pixel value given by the total energy deposited in the associated region. This grayscale map represents the transverse momenta of charged particles given in the first channel, and the next two channels are constituted by the transverse momenta of neutral particles and the charged particle multiplicity.

Table 6.5: Image processing techniques applied.

Technique	Before projection	After projection
Noise reduction	[126, 127, 138, 141, 143]	[128, 152]
Normalization	[148]	[141, 143, 150]
ROI finding/cropping	[135, 136, 130, 151]	[131, 141, 143, 147, 150, 154]
Data augmentation	[148]	[127, 130, 141, 142, 143, 146, 147, 150, 151]
Zero-centred data	[148]	[127, 130, 142, 146, 149, 151]
Maximum finding/Edge detection	[148, 135, 139]	[140, 147]

Lee et al. [136] define a cone around the jet data as ROI, with the central bin aligned to the jet axis. A jet is a mixed combination of particle showers, mostly hadronic, but also with photons included. For each particle within the jet, the particle energy is added to the bin corresponding to the particle direction and relative to the jet axis. Several such channels are filled corresponding to the different particle types.

De Oliveira et al. [151] show that the EM shower can be represented by slicing the calorimeter ROI around the particle shower direction as a series of digital images. The authors find that it is not enough to treat each layer independently and propose to create a 3D image of the event by using three histogram maps. In this way, the three-dimensional particle energy signatures are presented by three 2D images in the (η, ϕ) space, where the pixel intensity equals the sum of the energies of all particles that are incident to that cell [155].

6.2.2 Image processing and pre-processing techniques

The importance of image pre-processing is well-known in ML applications, especially the noise reduction needed to enhance the recognition process [157]. Image denoising is important for HEP applications, and it is usually applied before creating the event data image [126, 127, 141]. The strategy described in the reference papers is first to apply some selection of the event data based on the application of a threshold and only the selected data subset is projected or binned to image maps.

Besides noise reduction, there are other commonly applied techniques in the literature, and we classify them in 7 classes as presented in Table 6.5. Image normalization is a standard technique in computer vision (CV), which eliminates the effect of lighting conditions changes on captured images [141]. Normalization is performed by dividing the event image by the maximal pixel value [150], or by the total event energy [148].

ROI finding allows to locate an object within a selected sub-region of an image [154]. Similar to locating eyes on a face prior to the iris recognition,the smaller ROI images can be extracted (cropped) from the full-detector 2D image [131]. A similar strategy is applied in [143, 147, 150], to reduce the amount of empty space or to reduce the image sparsity. The ROI selection can also be done prior to the image generation, by selecting the calorimeter region in volume around a particle shower [135, 136, 130, 151].

The data augmentation is well-known in image processing and typical transformations are scaling [150], translation [141, 147, 150], rotation [127, 130, 141, 142, 143, 146, 147, 150, 151], reflection [127, 141, 143, 150] and flipping [147, 148]. These techniques are used to instruct the network about object variations and to achieve better performance [150].

For example, learning translation invariance can be realised with shifts by an integer number of pixels in the left/right/up/down direction with respect to the image center [150]. The image reflection is done after the translation [127, 141, 143]. Reflection can be performed by flipping (mirroring) [147, 148, 150], either over a single axis to ensure that the maximal energy is in the desired plane, or by two axes so that the maximum is in the specific quadrant. Image scaling invariance is considered by zooming the image in and out [150].

It is noticed in the literature that authors usually apply a maximum finding and data centering prior to any data augmentation techniques or image alignments [127, 130, 142, 146, 149, 151]. The maximum finding procedure is described in [147, 148], and it can be considered an ROI-based technique, since it identifies characteristic points. We merge maximum-finding and edge detection techniques in a single image processing class, because these operations are similar, being both kernel-based though with different weights.

6.2.3 Main research challenges and opportunities

Based on the conducted survey of image-based physics event data representation, we summarize the topics where we believe that further progress can be done. Hence, we propose future research in the following directions:

- Image pre-processing
- Image parameters of the bin (pixel) - bin size, bin space and bin shape (square/hexagon)
- Image sparsity
- Image with and without PU added
- ROI-based classification
- Projection-based separate views of the event data in the CMS detector
- Per layer strategy projection
- Oblique or orthogonal projection type

We conclude that most image pre-processing techniques are applied after the projection, while some authors still process the data before creating the event image. The importance of the pre-processing itself is straightforward in CV and in HEP, with the goal to enhance images and to increase the efficiency of the classification model. It is to notice from the literature that pre-processing is certainly a must have in single-channel image representation, where it plays a crucial role. Some authors consider pre-processing steps to be optional, but in most cases, they are commonly applied in order to assist the model to solve a classification problem. This is emphasized as well in [149], and we consider it logical, because in 2D grid representation of an event there is a loss of information, so it needs to be done in the most efficient way.

On the other hand, since the idea of multiple-channel images is getting more and more attention in the last years, we notice how the pre-processing techniques are becoming somehow less important. The authors use a minimal image processing and do not optimize them too much, which is a good strategy to be as model-independent as possible [125]. We consider that in this way the gap can be filled between computer science and physics applications. Even though the applied pre-processing techniques are standard ones, taken from CV, usually their application is physics-driven, requiring high physics knowledge. Without pre-processing, the computer scientists can treat the image-based event representations as any raw images captured by a camera, considering the detector to be camera in this context [123, 135, 151]. The goal can be to understand the structure of the energy deposits in the calorimeter, so that in the case where a single-image approach is used, further improvement of the classification can be accomplished by pre-processing the image prior to the learning process, while the same enhancement might be obtained by a multiple-channel image approach without pre-processing. We believe that this could be further explored.

Generally, not many parameters of the input image are varied in the literature. For example, the projection parameters or bin space in the event histogram are usually (η, ϕ), i.e. cylindrical coordinates (used in 23/30=76.7% papers). Another coordinate system is the Cartesian system, where the (x, y, z) are in centimetres, but we did not find any comparison of the two, or why (η, ϕ) is better. We believe that these two choices should be compared in more details and we propose a future research in this direction. Also, if (η, ϕ) is better, maybe it would be useful to compare another (η, ϕ)-like coordinate space of the images that would be in centimetres. This could maybe be more appropriate for EM showers which develop in terms of radiation length [133]. The bin size on event histogram representations is also not varied much in the literature. The bin parameter size is studied in [137], where the lower bin size results in a better CNN classification accuracy. Considering other reference papers, only [144, 154] tackled on this issue in the context of image down sampling and several event image grid sizes are chosen to meet the network performance parameters. The authors in [144] report on a decreased performance for the lowest image pixelization. The bin size on the event image grid is also studied in [136] with state-of-the-art CNN models used for event classification.

The 2D grid in the event image is always square for the square image or rectangular if the height and width of the image histogram are not the same. However, authors in [141] also provide an insight into hexagonal event image histogram binning. Generally, hexagonal image pixels have a lot of advantages in computer vision but were not very successful in practice due to the lack of camera to produce such images. We believe that this opportunity in HEP application could be further explored, especially for the upgraded CMS HGCAL detector whose sensors are based on an hexagonal geometry.

The sparsity of event images is commonly mentioned in the literature as one of the problems for HEP applications [124, 126, 128, 143, 147, 151, 152]. Our survey shows that most authors just mention this issue and try to avoid it with ROI-based recognition, applied by carefully choosing the radius of the preselected data before projection [127, 135, 136, 130, 151] or by cropping the subpart of the image that contains the points of interest [131, 141, 150, 154]. A minority of authors deal with this directly in their studies, like [126], who vary the kernel size and find that a larger processing kernel needs to be used on sparse images. We believe that there is more room to explore image sparsity and develop strategies on how to handle it in CNN classification tasks.

It can be noted from the literature that not much work is devoted to PU impact on CNN image classification [126, 127, 128, 138, 141, 143]. This issue is studied in [127, 130, 138, 143, 152], while other studies either do not mention whether the PU-added data is used or explicitly say that they use samples with no PU contribution [119, 125, 141, 142]. We believe that studies from the literature can be repeated with more realistic input data. PU collisions are normally present in physics events at the LHC and influence a lot to the object or particle recognition tasks, making images noisier and making the detection of signal data more difficult.

Considering the projection of the event data, we notice that there are no projection-based approaches with several separate views of the detector data in the context of the CMS detector. These are mostly from other experiments

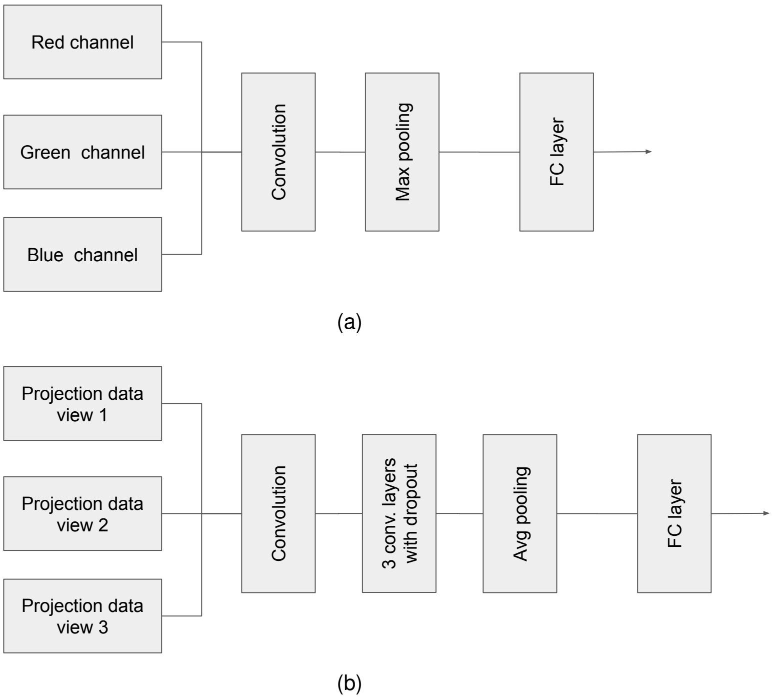


Figure 6.4: Multi-channel CNN schemes with 3D kernel. (a) Adjusted from [144]; (b) Adjusted from [135].

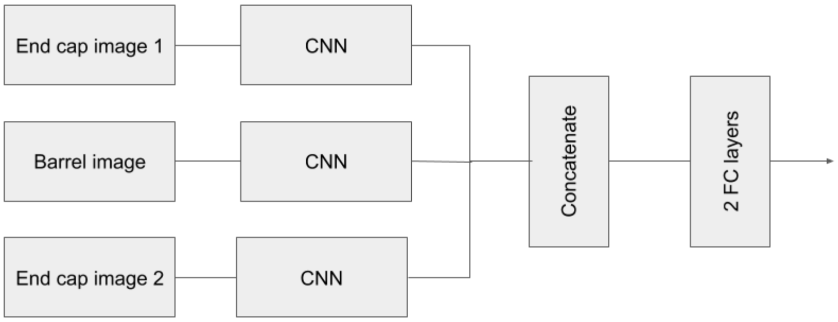
like NovA [133, 145], NEXT [137] or MicroBooNE [154]. Also, the CMS HGCAL detector is not considered in the papers, and we believe that the transfer of these image-based approaches, together with the adjustments of the presented models can be done for the new upgraded CMS detector, especially if hexagonal geometry is used in the new projection-based CNN learning scheme.

Also, there is one attempt of projection per layer [135] where CMS detector layers are used as image slices of the event data in a multiple-image classification approach. However, authors emphasize that only showers produced by particles traveling perpendicularly to the calorimeter surface are considered. We think that a future research direction could be to repeat their study but not only by considering perpendicular spread of the showers with respect to the detector layers like in the orthogonal projection, but also to explore the oblique data projections. An approach like [139] could be used in the absence of pointing information in the data, considering (0, 0, 0) as a point of origin.

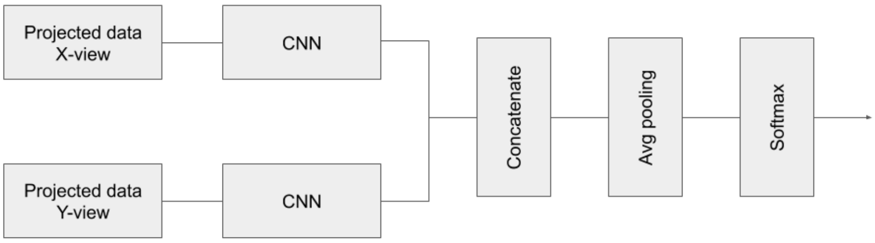
We have followed some of these above-mentioned research directions in our ML study which will be described in Section 6.3.

6.2.4 Existing multi-channel ML classification models

We divide the existing multi-channel CNN models in two classes. Namely, depending on the strategy applied, we consider the architecture to be either based on direct 3D processing with 3D tensor kernel applied [135, 128, 144], or based on several parallel CNNs with 2D kernels implemented as separate processing branches whose outputs are merged together to get the final decision [133, 137, 138, 145, 148, 152, 153, 154].



(a)



(b)

Figure 6.5: Multi-channel schemes with parallel CNN branches. (a) Adjusted from [138]; (b) Adjusted from [145].

The CNN in [144] has a filter on the first convolutional layer, which is a tensor of size 8x8x3 with 192 weights, i.e. 64 for each channel (Figure 6.4a). The concept of 3D kernel is explained in [154], where a local 3D volume around every pixel in the input multi-channel image is explored, and the weighted sum is calculated over the whole input tensor. Next, a feature map is generated by scanning the whole input image. The network in [135] is based on three input arrays of ECAL and HCAL energies and the total energy ratio ECAL/HCAL used to form a multi-channel image. It consists of four 3D convolutional layers, and the output of the final convolution layer is flattened and connected to a sigmoid classification (Figure 6.4b).

The architectures in the second class consist of two or more processing branches (one per channel), where in each branch a parallel CNN is implemented. The branches can be separate views of the event data, which are processed separately, as in [133, 137, 138, 145, 154]. For instance, two parallel 2D CNNs are shown in Figure 6.5b. The authors in [154] add another view of the event to obtain a full 3D experience in their multi-channel approach. They develop three parallel CNNs, where each one processes a single plane view, and the merged result of the three branches is processed by the final CNN. The authors in [138] use separate CNNs to process each image of the barrel and of the two endcaps (Figure 6.5a). Results of all processing branches are concatenated in the end, and the final layers are used to get the classification probability.

6.2.5 Main research challenges and opportunities

From by the survey on CNN architectures, we derive several open research directions for future work, where we believe that progress is needed:

- Multi-channel CNN in CMS detector
  - Using 3D kernels or separate 2D CNN processing branches in parallel
  - Detector layers used as CNN layers
- Avoid using CNN and simplify the ML model with few dense layers only without convolutional layers

Concerning the multi-channel input CNN approaches, we observe that mostly hybrid CNN models are present. They are based on combining additional variables besides image representation of event energy deposits, like the number of particles or the number of particles tracks [136, 128, 144, 148, 152, 132]. Some researches increase the number of input image channels depending on how many particles are classified [129, 139], or each channel corresponds to one subdetector image [131, 138]. However, not much attention is given to having a deeper input image tensor, such that the whole subdetectors are represented as 3D arrays of (x, y, z) positions filled with energy values. There are only few examples of this approach [135, 151]. We believe that this should also be studied in the context of CMS detector. One strategy would be to give the whole ECAL or HCAL data to the network to will learn its structure, so that the detector layers are used as CNN layers like in [153].

When we examine researches that design separate parallel CNN branches to process each input image channel, we can see that there are still not much papers in the literature about the projection-based strategy in the context of CMS detector. The separate 2D views of the detector event data are examined for other detectors [133, 137, 145, 154]. We believe that there is a gap and that more work is needed in this direction. It would be interesting to see the possible advantages from using separate CNN branches to process the CMS event data (particularly HGCAL), as well as the impact on the accuracy with respect to direct 3D learning network models.

Finally, it can be concluded that authors concentrate on CNNs and their application for the HEP classification tasks. Another possible strategy, which is not explored in prior work, is to avoid using CNN layers and keep only few dense layers for the ML classification task. Although a CNN is easier to train and it reduces the number of parameters in the network because the parameters are shared among several neurons, its implementation in hardware can be more complicated. Therefore, an approach with a FC network could have a smaller number of operations and use less resources with less number of multipliers in the hardware. We believe that this needs to be further explored and one needs to test whether simple FC approach in ML design can be enough to produce meaningful classification results.

We followed the above-mentioned strategy of using FC layers in the ML study that will be further described in Section 6.3. Also, the idea of having separate FC branches in parallel (for processing different independent views of the detector data) is adopted.

6.2.6 Discussion and evaluation

We evaluate the state-of-the-art and summarize the research findings in this section. We find that authors usually report on several topics that can be classified as follows:

- Parameters that impact the CNN performance
    - Image pre-processing
    - Increased number of image channels
    - CNN robustness to PU
- CNN optimization parameters
    - Image size/bin size
    - Filter number/kernel size
    - Sample size used for training

Considering HEP applications with image-based event data from the prior work, we conclude that still not much research is done on EM shower classification. It is only recently, in 2019, that people started to explore the possibility of applying prior work on jet images [126, 136, 129, 131, 141, 143, 148] to EM showers, to differentiate them from hadronic showers [139], or to classify the EM particles [138]. Most recently, people started to explore the potential of having a multi-channel classifications of EM data, with respect to the single-channel approach [135]. We believe there is space for future research in this direction, to compare single-channel and multi-channel strategies with the CMS event data and to explore the dependence on the detector hexagonal geometry.

Many authors show that a pre-processing makes the network training and classification accuracy more efficient, but there are also reports on degraded network performance with pre-processed images. We believe that it is needed to further understand this issue and possibly to obtain the higher CNN classification efficiency that does not depend on pre-processing. The authors in [144] elaborate on this and emphasize the need to avoid any pre-processing motivated by physical insights. Hence, they allow only generic pre-processing like normalizing the pixel intensities. Also, it is found in the literature that an increased number of channels added to the input image leads to an enhanced network performance. However, the authors in [137] highlight the potential of applying a direct 3D convolution rather than having separate CNN processing branches in parallel. We believe that it would be worthwhile to compare the two types of multi-channel schemes in more details.

In the context of having more realistic scenarios and images used for the network training, authors commonly investigate the effect of PU, which is a common source of noise within the event images. It is found in the literature that CNN can be trained on samples without PU to best represent the underlying physics on images without noise [141]. However, a key step is to test the networks on images with PU noise included, to see how robust they are and how they behave in a real environment.

6.3 Study on EM shower classification

In the previous section (Section 6.2), we provided a literature survey on image-based event representation and classification with deep learning techniques in HEP. Several main conclusions can be derived from the study, and the new-found answers to our posed RQs, which guided the conducted review. First, a key question for the event image classification is how to best represent the event data, and how to create high-quality images that contain discriminating features to be used in ML techniques ([129, 149]). We adopt the projection-based strategy with maps of accumulated energy and we describe in Section 6.3.1 how data is prepared, i.e. how the signal and background ROI images are generated.

Next, in Section 6.3.1, we describe the network architecture used for the ML classification task. We find that, like with using standard images in CV such as people, cats, dogs etc., FC networks can also be successfully trained on event images built from energy deposits in the upgraded CMS HGCAL detector. We optimize the network by varying the parameters like the number of layers in the network and the number of neurons on each layer. Finally, we provide the results of the ML study in Section 6.4, together with a discussion on the choice of the optimal ML design for an efficient image-based EM shower data classification as "interesting" (or signal deposits), towards "uninteresting" (or background deposits) caused by PU events.

6.3.1 Signal and background multi-channel ROI images generation

First, raw energy values from the CMS HGCAL detector layers are used, with the pre-processing in the form of the ROI selection in the detector volume, and the applied maximum-finding. Also, we transform the complex HGCAL geometry into simple projective squared grids. The idea is to make event images simple to produce, while keeping them correct from the physics side, because one cannot expect that the ML model output is accurate if it learns from incorrect images given for the training [126, 127, 144]. For this, the TA described in Chapter 5 is applied, so that a single ROI image is generated for each event by selecting the area around the reconstructed seed in depth. This way, a multi-channel ROI image is generated, providing a sliced view in depth of the detector data, and using directly the per layer projection strategy for the image generation. Consequently, data is projected in the oblique (EM shower projective) manner and not perpendicular to the detector layers. The strategy is described and visualized in the following.

Next, another approach to the ROI image generation by using three independent projection-based views of the

CMS HGCAL detector data in three dimensions is described. We have varied some of the ROI image parameters such as the image size, the pixel or bin size (image resolution), the image depth for a multi-channel approach and the study of the impact of shower identification with different profile types in 2D single-channel ROI image generation algorithm. This is described in details in the study given in Section 6.4.

Strategy by using trigger cell energies on detector layers

The raw TC energies from HGCAL layers are projected on a fixed grid with a predefined bin size, whose area corresponds to either a single SC ($1cm^2$), or a TC ($4cm^2$), or a cluster of four TCs ($16cm^2$). The procedure is based on the TA that has been described in Chapter 5. It is important to notice here that the TA is applied without any identification mechanism (by using raw TC energy values). All the HGCAL layers are compressed into a single layer, and the Cartesian coordinate system of the TCs is mapped from the detector space (x, y, z) into the binning space (x, y) eliminating the depth coordinate z (Figure 6.6a). The projection of the TCs is done in the "projective" manner on a virtual plane, which is located at the first detector layer, by following a straight line towards the center of the detector (0,0,0). The projection can be described by the following formula where where $z_{virtual}$ is the z position of the virtual plane:

$$x = \frac{x_{layer}}{z_{layer}} * z_{virtual}; \quad y = \frac{y_{layer}}{z_{layer}} * z_{virtual}; \quad E(x, y) = \sum_{TCs} E(x_{layer}, y_{layer}) \tag{6.3}$$

Next, the seeding procedure with the 3x3 bin window (Section 5.1.3) is applied on the resulting grid, in order to extract seed bins. The definition of an electron seed and how it is associated to the generated particle has already been described in Section 5.2.4. In the case of a single shower data sample, there is a single seed extracted for each event (originating from EM shower energy initiated by electron particle). Conversely, in the case of background data sample, there will be a large number of PU seeds on the projected plane, each of them originating from an additional PU shower contaminating the signal in the event. However, for simplicity, we have applied the similar procedure for the generation of the signal and background ROI images. The main difference is that a matching procedure from Section 5.2.4 is applied in the case of the signal. This means that, unlike for signal, where the maximum bin must be matched to the generated particle, the background ROI image is generated directly from the maximum bin extracted by the seeding. Hence, it is assumed that only a single PU seed with maximal energy is extracted, which is considered a limitation of the study, and elaborated in Section 6.7.2.

The second step of the ROI generation algorithm is depicted on Figure 6.6b. Namely, a straight line originated from the detector centre $(0, 0, 0)$ is reconstructed from the seed and propagated backwards into the detector volume, where each intersection point in depth corresponds to the centre of the ROI image slices. The reconstruction procedure is that described in Section 5.2.1, but in this case a rectangular box is extracted in depth instead of a cylinder. Finally, a multi-channel ROI image is generated by selecting a squared area around each reconstructed

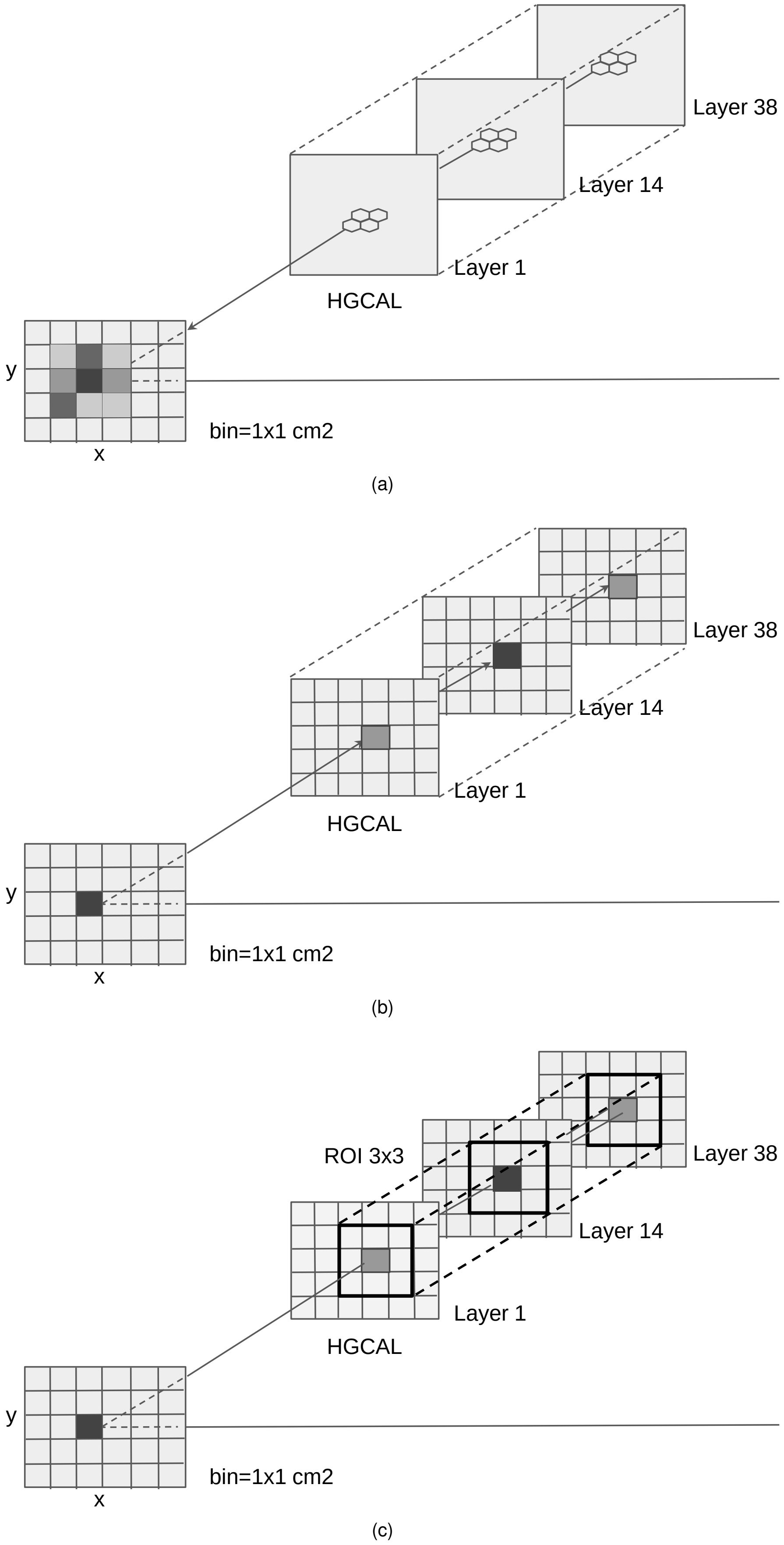


Figure 6.6: Multi-channel ROI image generation algorithm.

track. It is defined as a 3D matrix with dimensions NxNxM, where N is the width and height of the ROI and M is the number of layers along the z dimension (Figure 6.6c).

Projection-based separate views of the detector data

For the second set of ROI images, a concept similar to [133, 137, 145, 154] is adopted. Namely, the images are generated from the projections of the event energies in three independent planes or views (x-y, x-z and y-z). The ROI generation algorithm is the same as in the case of the multi-channel ROI image (Section 6.3.1), but a single-channel ROI image is created in each direction. It means that the same first 2 steps can be used for the ROI generation, but as a new step 3, the ROI is taken on the projected 2D result directly. Hence, the x-y summed ROI result is obtained by setting $z = 0$. Similarly, summing the data by keeping $y = 0$ and $x = 0$ results in the "side" oblique projections x-z and y-z. Finally, each event is represented by the 3 ROIs (one per projection) and each of them is processed separately by the neural network. Details on the ML architecture are given in Section 6.6.2.

6.3.2 ROI visualization

The mean energy fractions for 1000 signal ROI images of size 15x15x38 shown on Figure 6.7, where EM showers with PU added are compared to pure EM showers with no PU. We can notice the PU impact on the longitudinal EM shower profile in HGCAL, i.e. a higher energy accumulation on the first ECAL layers, as well energy deposits in HCAL, which are absent in the case of pure EM with PU=0. This supports the signal ROI visualization for randomly selected events given on Figure 6.8, 6.9 and 6.10. The ECAL layers range from 1 to 28, while other layers belong to the HCAL. For simplicity, only the first 28 layers of the ROI image are shown instead of 38, where every second layer in ECAL is used for the trigger. It can be seen that the signal (EM with PU200) ROI is more compact, and the shower maximum is contained in the layers 9 to 15, which is the usual EM shower footprint in ECAL.

The background ROI images (PU200) shown on Figure 6.11, 6.12, and 6.13 have a lower energy compared to signal ROIs, and the energy is more spread across layers. We can also see a higher contribution in HCAL layers, due to the hadronic components in PU showers. Generally, we can see how the projection bin size impacts the ROI images, where more spread is present for the lowest bin size 1x1cm^2 (since we are projecting TC energies on an SC-size grid). When we increase the bin size to 2x2cm^2 or 4x4cm^2, more energy is accumulated, and we obtain more centred and more compact ROIs. This will help to better discriminate signal from background showers in the study.

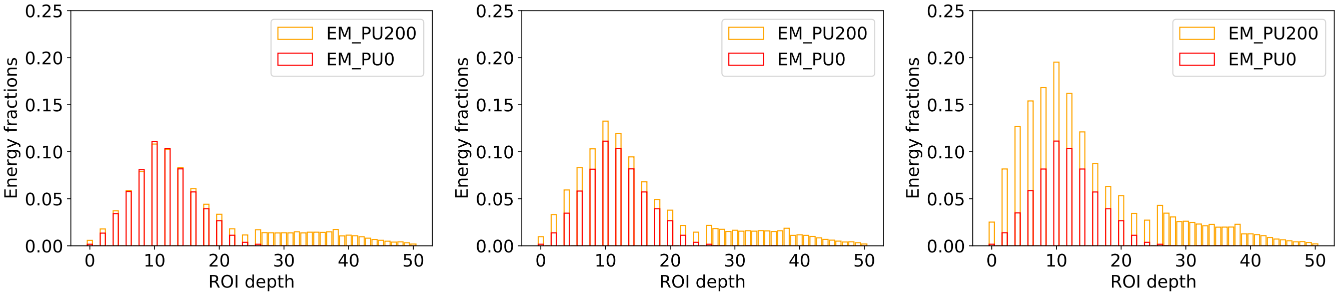


Figure 6.7: Mean energy fractions per ROI layer (1000 events, ROI size 15x15x38). Bin size 1x1 cm^2 (left), 2x2 cm^2 (middle) and 4x4 cm^2 (right). Every other layer in ECAL part of HGCAL is used for the trigger.

6.4 Model verification and ML classification results

In this section we describe our ML classification model. We elaborate on the results of the ROI image classification with the various approaches used: the multi-channel ROI image whose depth corresponds to the depth of HGCAL detector with TC energy deposits, the multi-channel ROI image with reduced depth and, finally, the three single-channel 2D ROI images that correspond to the three independent views of the detector data.

6.4.1 The neural network architecture and training parameters

The ROI image described in the Section 6.3.1 is used as input to the network design presented in this section. Similarly to [135, 151, 153, 158], the whole sub-detectors are structured as 3D arrays of (x, y, z) positions filled with energy values. In this way, we provide the full HGCAL detector energies to the network that will learn the signal and background data structures. However, unlike in prior work, the FCNN is used.

The FCNN is a type of artificial neural network where all the nodes or neurons of each layer are connected to the neurons of the next layer. The concept is illustrated on Figure 6.14. First, our FCNN consists of an input layer with the ROI which is a flattened multi-channel image, where the number of layers corresponds to the number of channels, but the image is flattened to a vector format. The FCNN has three dense layers with N neurons, and an output classification layer that separates in two classes (the signal and the background). The number of neurons is varied within $N = 32, 64, 128$ and we defined nine ML FCNN models, denoted as $3D\_N_1\_N_2\_N_3$, where $N_i, i \in 1, 2, 3$ is the number of neurons on each of the three dense layers. The size of the network is evaluated as the total number of parameters or coefficients that need to be calculated for a ROI of size 15x15x38, as given on Figure 6.15. The FCNN models are denoted as: m1 (3D_128_128_128), m2 (3D_128_64_64), m3 (3D_128_64_32), m4 (3D_64_128_64), m5 (3D_64_64_64), m6 (3D_64_32_32), m7 (3D_32_128_128), m8 (3D_32_64_64) and m9 (3D_32_32_32).

The training parameters are selected by using a grid search. The activation function ReLU is used for the computation of the weights in the dense layers of the network, and it enables neurons to interact and to learn

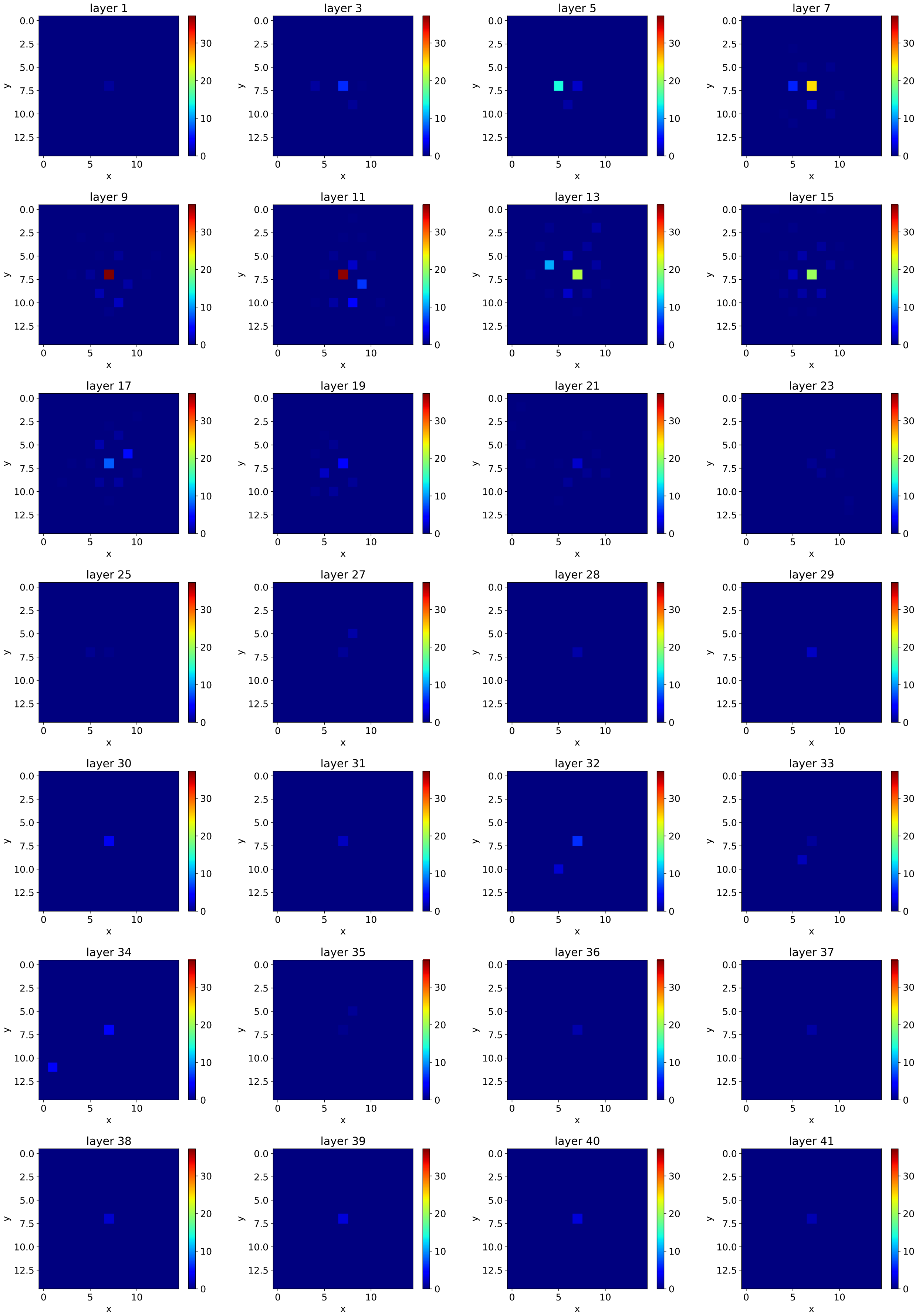


Figure 6.8: 15x15x38 signal ROI image for the event 3, with the first 28 layers visualized. Bin size: 1x1 cm^2.

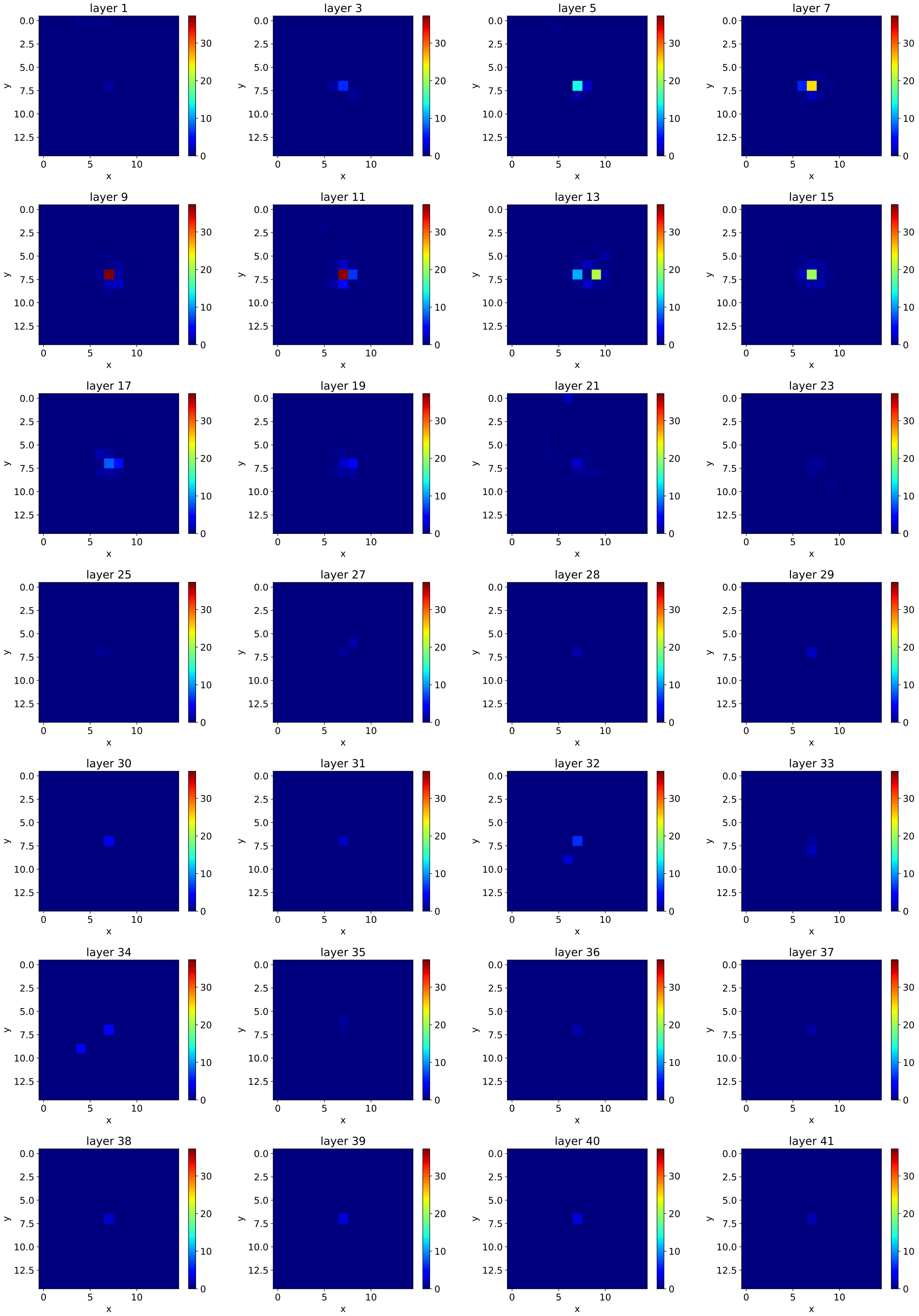

layer 1
layer 3
layer 5
layer 7
layer 9
layer 11
layer 13
layer 15
layer 17
layer 19
layer 21
layer 23
layer 25
layer 27
layer 28
layer 29
layer 30
layer 31
layer 32
layer 33
layer 34
layer 35
layer 36
layer 37
layer 38
layer 39
layer 40
layer 41

Figure 6.9: 15x15x38 signal ROI image for the event 3 with the first 28 layers visualized. Bin size: 2x2 cm^2.

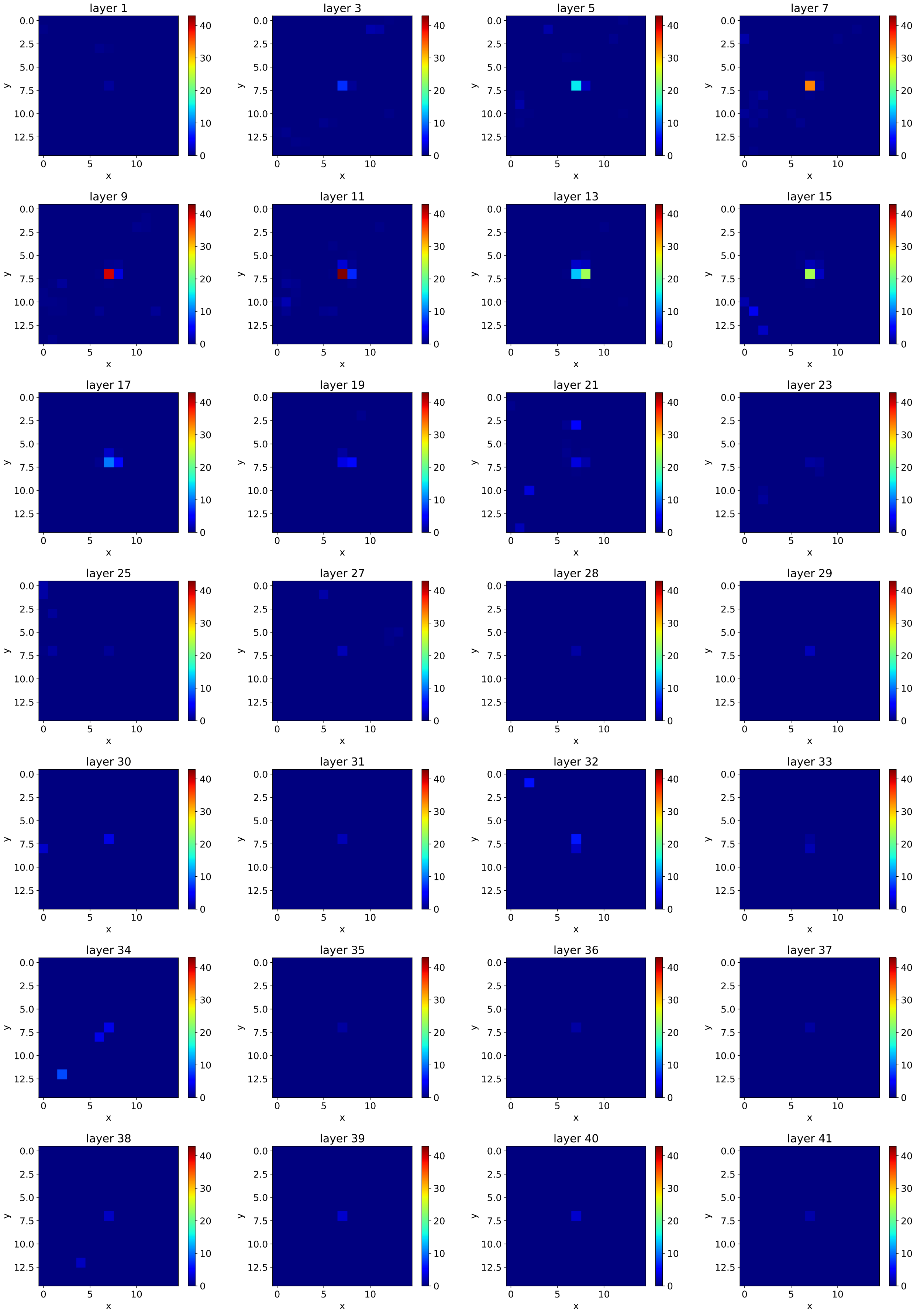


Figure 6.10: 15x15x38 signal ROI image for the event 3 with the first 28 layers visualized. Bin size: 4x4 cm^2.

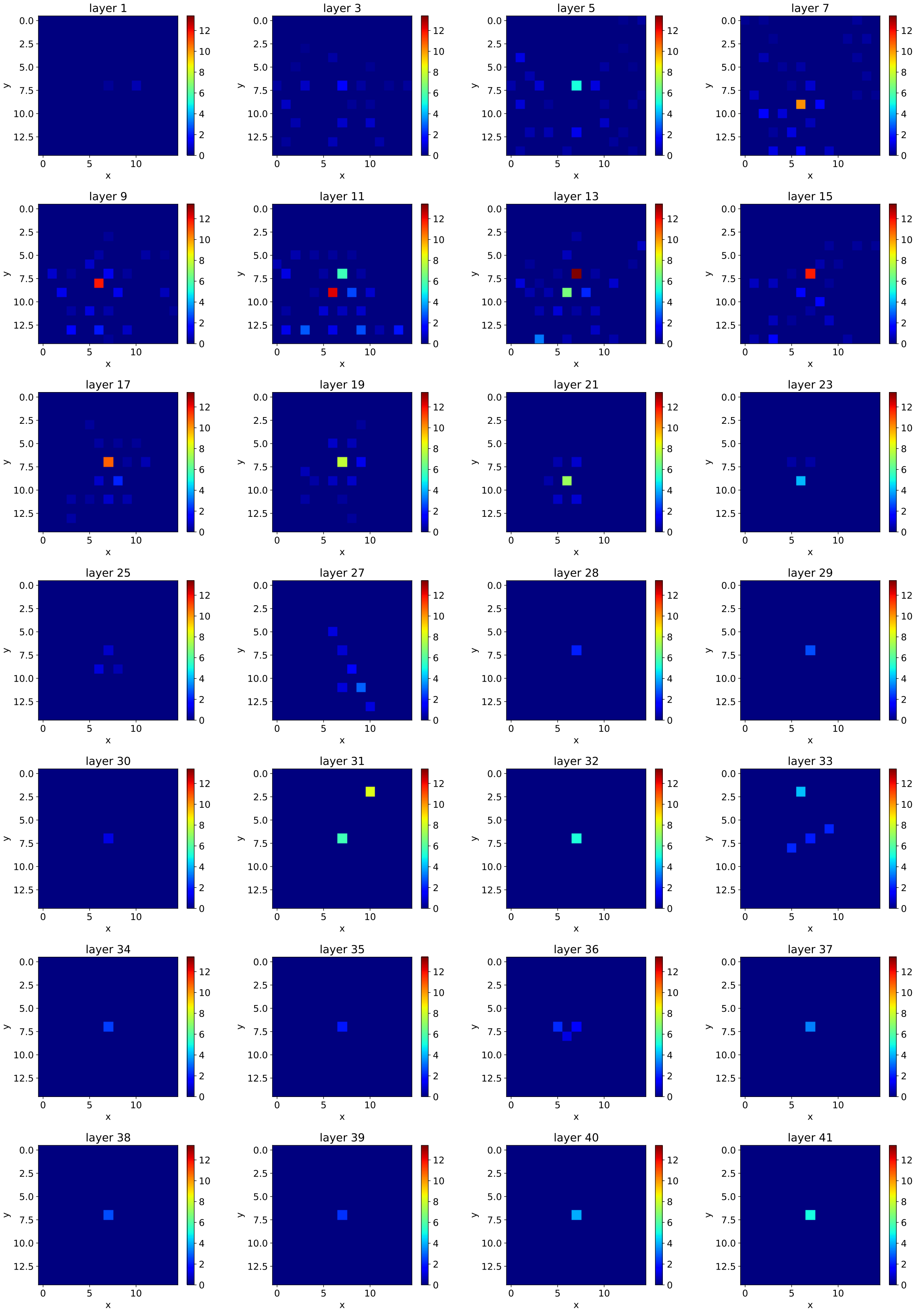


Figure 6.11: 15x15x38 background ROI image for the event 2 with the first 28 layers visualized. Bin size: 1x1 cm^2.

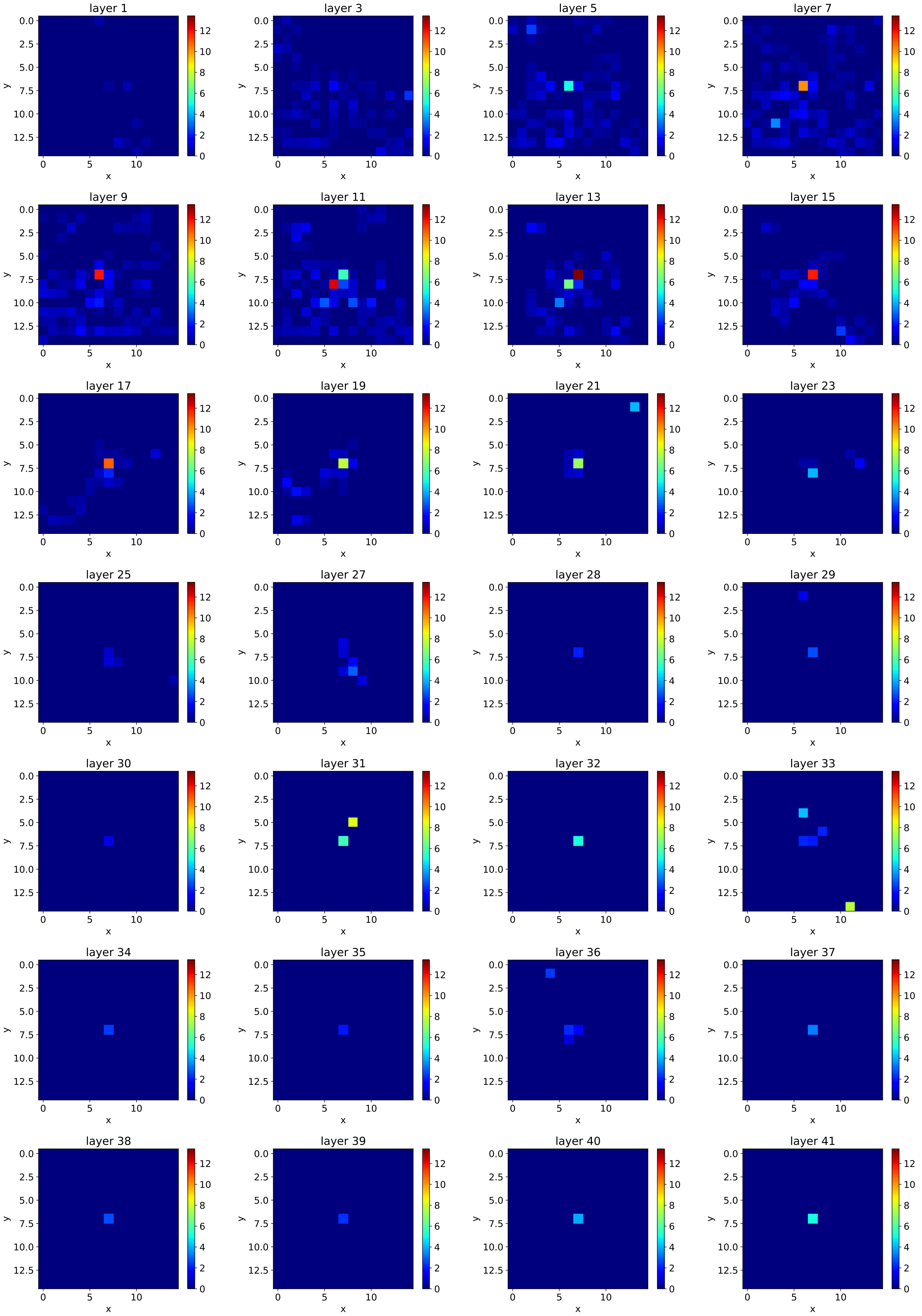


Figure 6.12: 15x15x38 background ROI image for the event 2 with the first 28 layers visualized. Bin size: 2x2 cm^2.

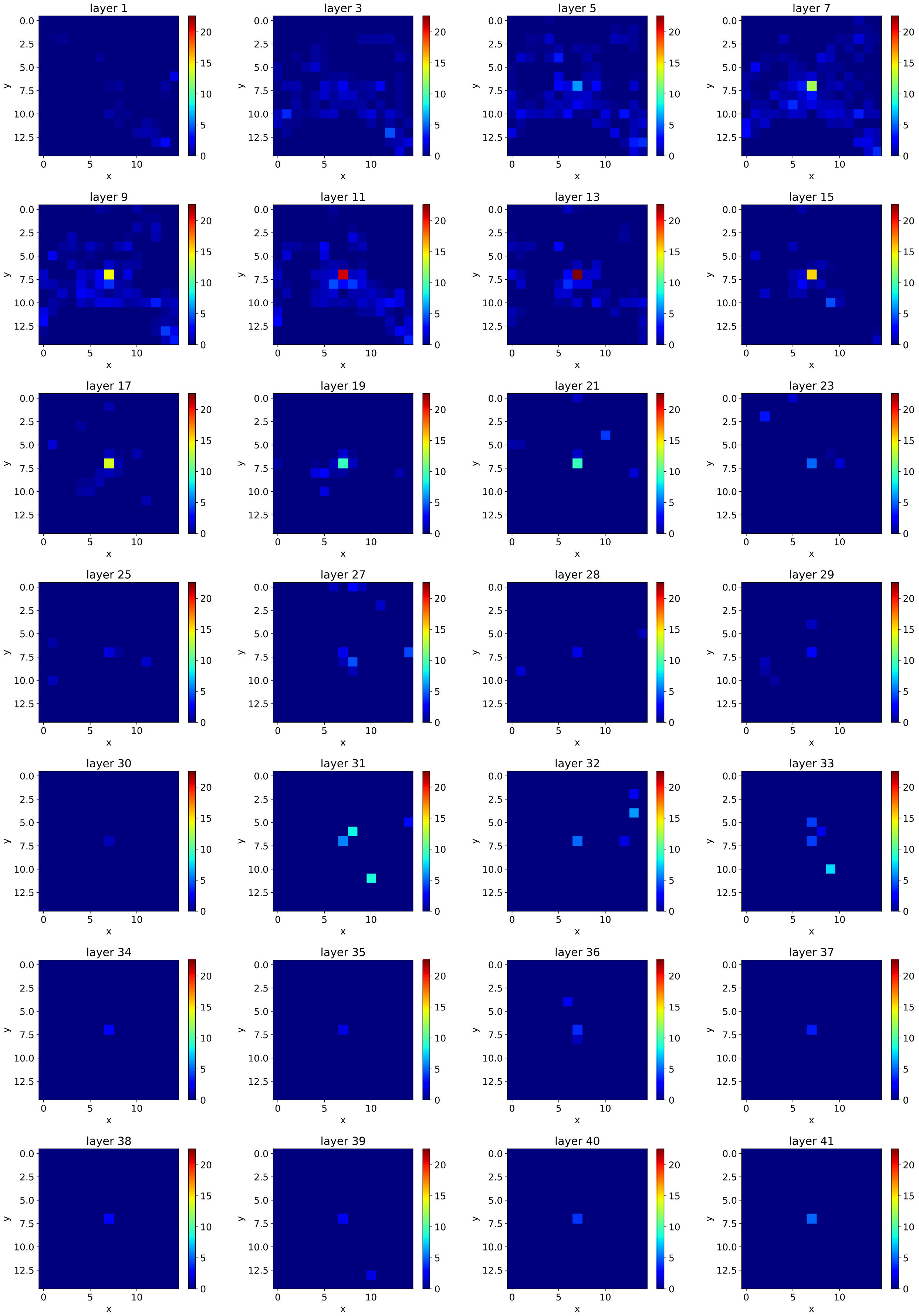


Figure 6.13: 15x15x38 background ROI image for the event 2 with the first 28 layers visualized. Bin size: 4x4 cm^2.

connections between the input data and the class labels. At the end of the network, a softmax activation is used for the output layers, which predicts the final probabilities of the input to be in each class. Loss functions are important as they ensure that the NNet learns something that is as close as possible to the true result. In our case, it is set to categorical crossentropy, which is commonly used for the multi-label classification or a special case of binary classification when there are only two classes. The Adam optimizer in the loss function is chosen to minimize the loss. The initial learning rate is 10^{-3}, the batch size is 64 and we train on 100 epochs with an early stopping mechanism applied if no progress is seen beyond 10 epochs.

Concerning the composition of the signal and background events used for the classification task, it is a balanced dataset of 5000 signal and 5000 background ROI images. The fraction of the events used for the training and validation are 20% and 80% respectively, so there are 8000 training and 2000 validation images. We train on signal ROI images containing an EM shower, i.e. the energy deposited by the electron particle in the calorimeter material and with PU added, to obtain a more realistic data simulation. On the other hand, there are background ROI images containing only PU showers, where no signal or "interesting" deposits are present. The basic goal of

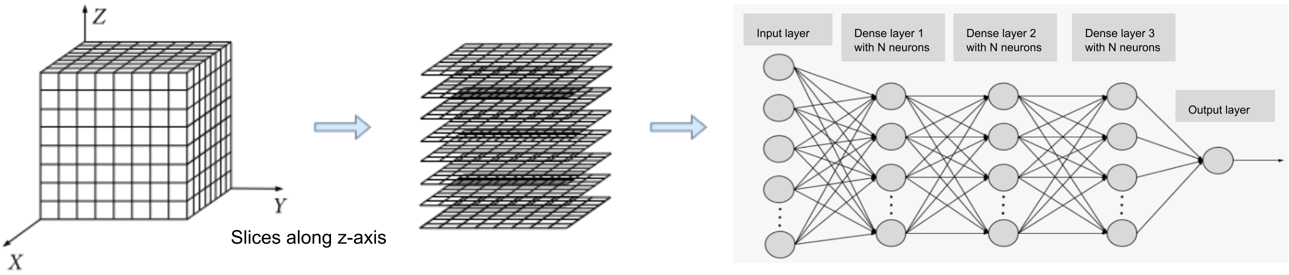

Z
Y
X
Slices along z-axis
Input layer
Dense layer 1 with N neurons
Dense layer 2 with N neurons
Dense layer 3 with N neurons
Output layer

Figure 6.14: Schematic view of the FCNN model used in the study. Inspired by [158].

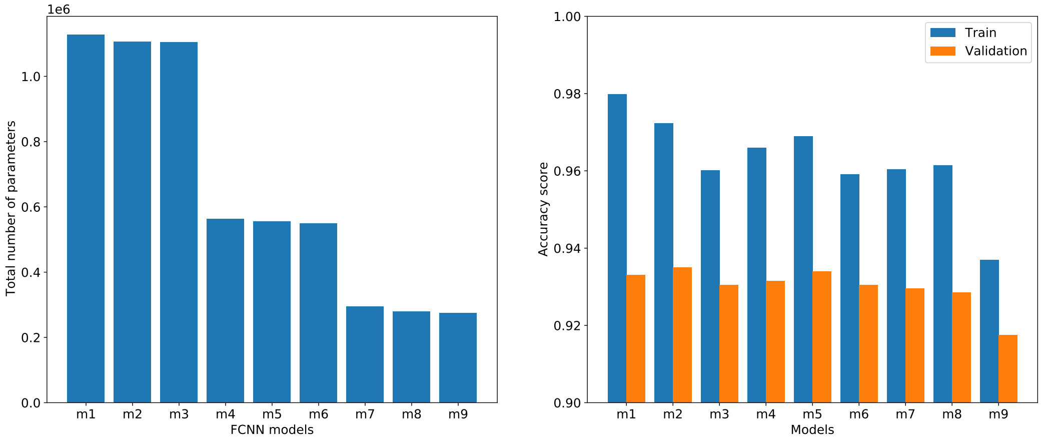


Figure 6.15: Model comparison. The total number of parameters (left) and the training and validation accuracy (right). The training accuracy is the percentage of the correctly classified training images (to see how the model is progressing in terms of training), and the validation accuracy shows the model behavior on the unseen images (it is a measure of the quality of the model with the selected set of parameters).

the classification is to conclude whether the ROI image contains "interesting" data or not.

We use the area under the curve (AUC) of the ROCs as the basic metrics for the evaluation of the results. As is common in HEP, the ROC curve consists of true positive (signal) versus false positive (background) rates, to describe the signal efficiency versus the background rejection [138]. The false positive rate is calculated as 1 - true negative rate and false negative rate is 1 - true positive. The standard evaluation metrics are applied [159]:

- Accuracy - the ratio of correctly classified items: $accuracy = \frac{truepositive+truenegative}{truepositive+truenegative+falsepositive+falsenegative}$
- Precision - how many of the predicted positives are true positives: $precision = \frac{truepositive}{truepositive+falsepositive}$
- Recall - how many of the actual positives are labelled as positive: $recall = \frac{truepositive}{truepositive+falsenegative}$
- F1 - weighted combination of precision and recall: $F1 = \frac{2\times(recall\times precision)}{recall+precision}$

Again as in the training procedure, the evaluation (testing) is performed on a balanced mix of the ROI images from both signal and background class sets (500 signal and 500 background test samples). Training is performed on Nvidia DGX-1 system, provided by College of Information Technology Zagreb. System is partly funded through European projects.

6.4.2 Impact of the ROI image resolution on classification performance

The ROI resolution is expressed as the bin size of the accumulated space defined before the projection, as was already shown on Figure 6.6. In order to select the ML model for the study, a classification is performed with the ROI input image 15x15x38 and the default ROI resolution where the bin area is 1cm^2. This bin size is approximately equal to the area of an hexagonal SC in the HGCAL. The result on Figure 6.15 shows that the accuracy is satisfactory and the model correctly learned to classify on average 96.3% of the training images. The mean percentage of correctly classified images from the validation set is 93%. The effect of overfitting is visible on the figure, and we can see that a larger number of neurons on the first layer does not necessarily mean a better model, while on the other hand it leads to a larger total number of parameters. We decide to extract three models to further study; 3D_128_64_32, 3D_64_32_32 and 3D_32_32_32, where the effect of overfitting is minimal (the smallest difference between the training and the validation accuracy score). The training history curves with the small overfitting effect are shown on Figure 6.16.

We vary the ROI resolution with three different bin sizes: 1x1cm^2, 2x2cm^2 and 4x4cm^2. The result shown on Figure 6.17 reveals that the classification accuracy is better for a larger bin size, while it degrades with the lower bin size of 1x1cm^2. This is straightforward and it can be explained by looking at the ROI visualizations presentedin Section 6.3.2, where we saw that a larger granularity (or spread) is present with the 1x1cm^2 bin size. This is because we are projecting TC energies on a bin that is too small and corresponds roughly to the area of a single hexagonal SC that is only $1/4$ of a TC area. On the other hand, a 2x2cm^2 bin size equals to the TC size, so that the ROI

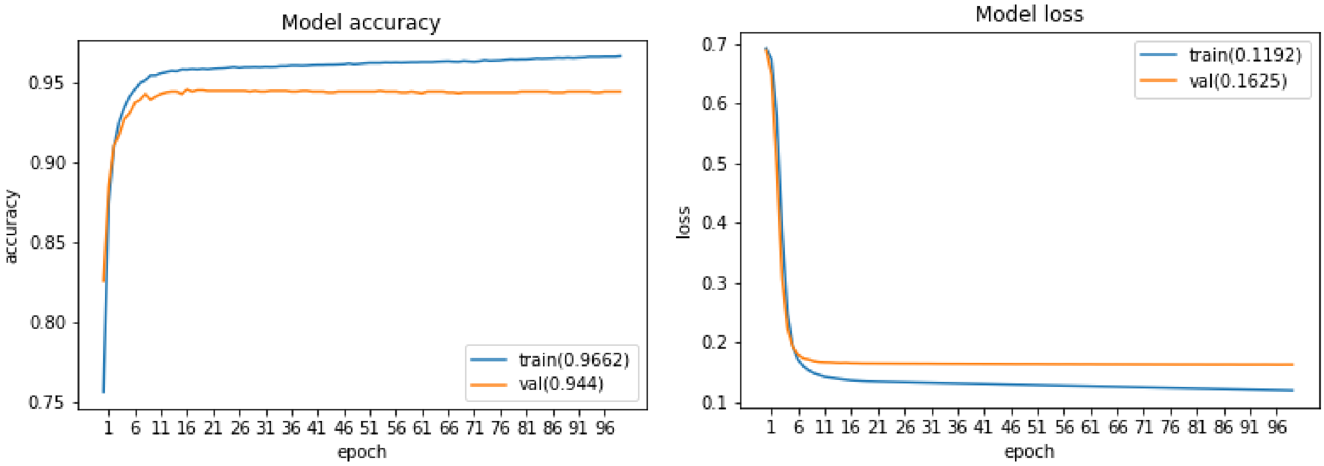


Figure 6.16: Example of training history curves for the ML model 3D_64_32_32.

image is more compact and the discriminatory power of the accumulated energy is better. The result for a bin size 4x4cm^2 of ROI visualization is even more compact, because we accumulate more energy with a larger bin size (area of four TCs clustered together). Hence, we obtain a slightly better classification result than with a bin size of 2x2cm^2. The evaluation results when testing the classification models on unseen ROI images of EM and PU showers is presented on Figure 6.18 and Table 6.6. They confirm the above conclusions, and we see a significant improvement for the AUC with the larger bin sizes than 1x1cm^2, as well as for the other evaluation metrics. Also, there is not much difference between the results for the three ML models, which indicates that we can use a model with a lower number of parameters (reduce the model complexity) and achieve similar performances.

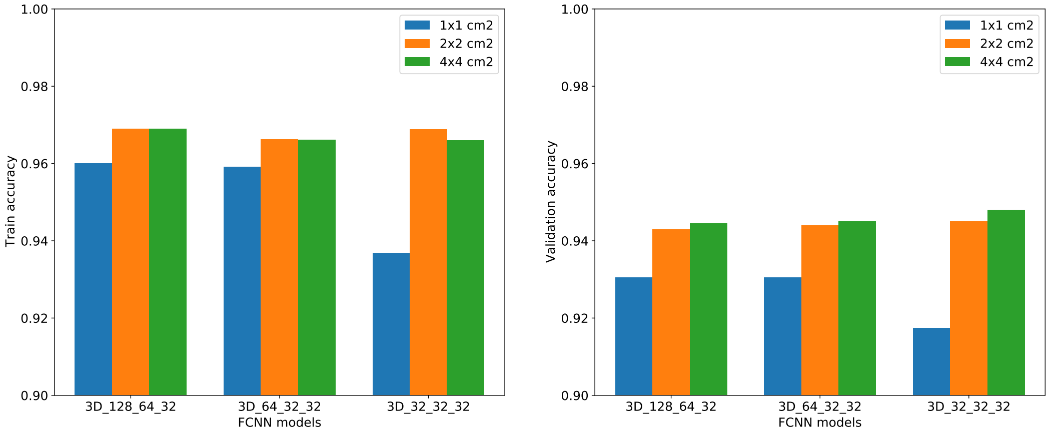


Figure 6.17: ROI resolution impact on the classification accuracy (ROI size: 15x15x38).

Table 6.6: Results on accuracy, precision, recall, F1 and AUC scores (ROI size: 15x15x38).

model	bin size	accuracy	precision	recall	F1	AUC
3D_128_64_32	1x1	**0.941**	0.966	**0.914**	**0.939**	**0.973**
	2x2	0.954	**0.963**	0.944	0.954	0.984
	4x4	0.958	0.965	0.95	0.958	0.982
3D_64_32_32	1x1	**0.939**	**0.964**	**0.912**	**0.937**	**0.974**
	2x2	0.959	0.969	0.948	0.959	0.984
	4x4	0.958	0.965	0.95	0.958	0.984
3D_32_32_32	1x1	**0.927**	**0.953**	**0.898**	**0.925**	**0.973**
	2x2	0.956	0.965	0.946	0.956	0.984
	4x4	0.959	0.967	0.95	0.959	0.984

6.4.3 Impact of the ROI size on the classification performance

Here we vary the ROI size in width and height, while keeping the ROI depth constant (M=38 layers). The goal is to have a squared ROI per layer and to enable having a central bin, so that odd values are selected for width and height values. We define the following ROI sizes NxNxM: 15x15x38, 13x13x38, 11x11x38, 9x9x38, 7x7x38, 5x5x38 and 3x3x38. For example, the ROI size 3x3x38 corresponds to 3x3 projection bins that are extracted on each of the 38 detector layers in depth. Also, the variation of the ROI resolution (bin sizes 1x1, 2x2 and 4x4cm^2) enables us to measure the impact of the accumulated ROI energy on the classification accuracy. The results on Figure 6.22 show that, for a particular ROI size like 3x3x38, the larger the ROI resolution, the larger the classification accuracy with the corresponding ROI images. The reason for this is that 3x3 for 1x1, 2x2 and 4x4cm^2 bin sizes equal 9, 36 and 144cm^2 ROI area per layer, so that we accumulate more and more energy in the ROI when we increase the bin size. Again, the mean validation accuracy score is the lowest for the 1x1cm^2 bin, while it still rather satisfactory (91% compared to 94% for the other bin sizes).

Generally, considering the ROI size variation, more fluctuations are present with the smallest bin size 1x1cm^2, while the results are rather flat for the validation accuracy with the largest 4x4 cm^2 bin scenario. This is due to the

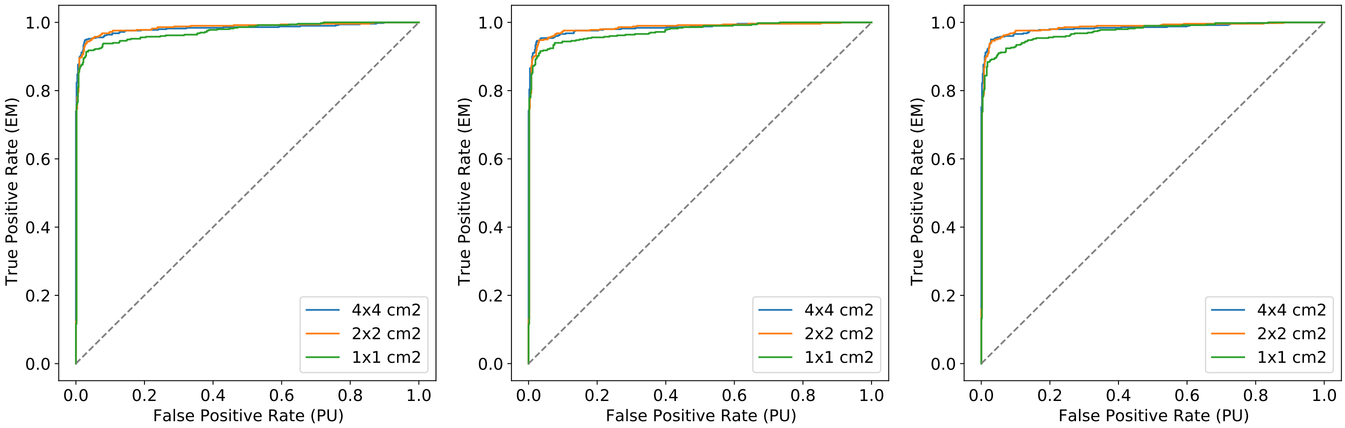


Figure 6.18: ROC curves for binary classification of the EM vs.PU ROI images (ROI size: 15x15x38). Model 3D_128_64_32 (left), 3D_64_32_32 (middle) and 3D_32_32_32 (right).

granularity problem in small bin that we already discussed. In case of a too large bin, the EM shower is always contained in each ROI area. The EM shower is rather small, and almost 90% and 95% of it is contained in a cylinder of 1 and 2 Moliere radius respectively, where $R_M = 2.19cm$ [25, 160]. Hence, the base of a cylinder containing most of the shower ($r_{base} = R_M$) is approximately $15cm^2$ in area, so the core of the shower is always fully contained for bin sizes 2x2cm^2 and 4x4cm^2, which leads to a good accuracy result.

On Figure 6.19, we examine the effect of decreasing the ROI size, i.e. how much EM shower energy is lost with if using a smaller ROI. Naturally, the EM sample with no PU is used in this case, and the mean energy fraction is calculated for 1000 EM shower events without PU. In our calculation, we consider the maximal ROI size of 15x15x38 to correspond to 100%. Then, we measure the amount of energy contained in smaller ROI compared to this "maximal" one. The results confirm that indeed, in the case of the larger bin sizes 2x2 and 4x4, almost 90% of the shower is always contained no matter of the ROI size variation, while in the case of the smallest bin size 1x1cm^2 a degradation is observed. Namely, almost 50% of the energy is lost with the smallest ROI 3x3x38 (the projected area is approximately 9cm^2 for 1x1cm^2 bin, which is approximately half of a cylinder containing most of the shower ($r_{base} = R_M$, $P = {R_M}^2\pi \approx 15cm^2$).

To visualize the discriminatory power between signal and background ROI images, the mean energy fractions are given on Figures 6.20 and 6.21, where the mean longitudinal profiles of EM with PU200 and pure PU200 showers are calculated for 1000 ROI images of each size and for the three resolutions. We can see the effect of the ROI resolution on the classification accuracy, because a very low amount of PU energy is included with the smallest bin size of 1x1cm^2. The images show how more and more PU is included with the largest bin size of 4x4cm^2, especially in the first detector layers, because the mean PU shower profile is exponential-like, with the largest contribution in the first layers. The discrimination power between the signal and the background remains high for all bin sizes, while the largest profile differences are present for the bin sizes of 2x2 and 4x4, where the PU takes its natural shape.

Concerning the ROI size impact, we can see that PU showers are more spread across HGCAL, while EM showers remain only in the ECAL with the small PU tail visible in the HCAL. We reduce the PU contribution in the

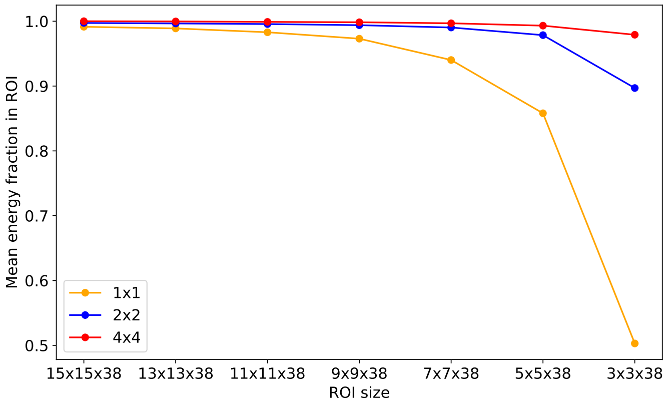


Figure 6.19: Comparison of the mean EM shower energy fractions contained in ROIs of different sizes.

signal when we decreased ROI size (ECAL layers), as elecron showers become more and more narrow with smaller ROI sizes. Also, a decrease of the ROI size impacts background, because we reduce PU contributions in ECAL with a smaller ROI, while the HCAL part remains almost intact. The ECAL part of the background energy deposits gets shifted towards EM-like, which might slightly decrease the classification score.

We present the ROC curves on Figure 6.23, to evaluate the different ML models and ROI resolutions/sizes. We can see on the figures that the classification power is a little bit degraded for the smallest ROI 3x3x38, which results in more false positives for all the three ROI resolution scenarios and all the ML models. However, the results are very good (around 92% fo all the evaluation metrics given in Table 6.7, Table 6.8 and Table 6.9). The signal efficiency is higher with larger ROI sizes, since there is much more true positives for the same false positive rate. Again, it is confirmed that the classification result is almost the same with all the three ML models used, and that the lowest classification result is obtained for smallest bin size of 1x1cm^2.

Table 6.7: Results for the different ROI sizes. ML model: 3D_128_64_32.

ROI size	4x4 cm2					2x2 cm2					1x1 cm2				
	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC
15x15x38	0.958	0.965	0.950	0.958	**0.982**	0.954	0.963	0.944	0.954	0.984	0.941	0.966	0.914	0.939	0.973
13x13x38	0.960	0.969	0.950	0.960	0.985	0.956	0.967	0.944	0.955	0.984	0.941	0.966	0.914	0.939	0.974
11x11x38	0.956	0.965	0.946	0.956	0.985	0.950	0.965	0.934	0.949	0.984	0.938	0.960	0.914	0.936	0.973
9x9x38	0.954	0.969	0.938	0.953	0.985	0.947	0.963	0.930	0.946	0.985	0.925	0.949	0.898	0.923	0.972
7x7x38	0.953	0.967	0.938	0.952	0.984	0.949	0.967	0.930	0.948	0.985	0.922	0.943	0.898	0.920	0.968
5x5x38	0.953	0.963	0.942	0.952	0.984	0.950	0.967	0.932	0.949	0.983	0.919	0.941	0.894	0.917	0.967
3x3x38	**0.937**	**0.938**	**0.936**	**0.937**	0.984	**0.925**	**0.936**	**0.912**	**0.924**	**0.975**	**0.911**	**0.925**	**0.894**	**0.909**	**0.965**

Table 6.8: Results for the different ROI sizes. ML model: 3D_64_32_32.

ROI size	4x4 cm2					2x2 cm2					1x1 cm2				
	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC
15x15x38	0.958	0.965	0.950	0.958	0.984	0.959	0.969	0.948	0.959	0.984	0.939	0.964	0.912	0.937	0.974
13x13x38	0.957	0.969	0.944	0.956	0.984	0.954	0.965	0.942	0.953	0.984	0.940	0.962	0.916	0.939	0.974
11x11x38	0.957	0.967	0.946	0.957	0.985	0.951	0.967	0.934	0.950	0.984	0.931	0.954	0.906	0.929	0.973
9x9x38	0.952	0.969	0.934	0.951	0.985	0.948	0.963	0.932	0.947	0.985	0.923	0.943	0.900	0.921	0.971
7x7x38	0.956	0.967	0.944	0.955	0.984	0.945	0.965	0.924	0.944	0.985	0.919	0.941	0.894	0.917	0.966
5x5x38	0.953	0.963	0.942	0.952	0.984	0.948	0.965	0.930	0.947	0.983	0.918	0.939	0.894	0.916	0.966
3x3x38	**0.933**	**0.934**	**0.932**	**0.933**	**0.983**	**0.923**	**0.938**	**0.906**	**0.922**	**0.974**	**0.902**	**0.915**	**0.886**	**0.900**	**0.963**

Table 6.9: Results for the different ROI sizes. ML model: 3D_32_32_32.

ROI size	4x4 cm2					2x2 cm2					1x1 cm2				
	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC
15x15x38	0.959	0.967	0.950	0.959	0.984	0.956	0.965	0.946	0.956	0.984	0.927	0.953	0.898	0.925	0.973
13x13x38	0.957	0.969	0.944	0.956	0.984	0.952	0.973	0.930	0.951	0.984	0.937	0.962	0.910	0.935	0.974
11x11x38	0.959	0.973	0.944	0.958	0.983	0.949	0.967	0.930	0.948	0.984	0.926	0.949	0.900	0.924	0.971
9x9x38	0.952	0.967	0.936	0.951	0.984	0.945	0.965	0.924	0.944	0.985	0.920	0.939	0.898	0.918	0.969
7x7x38	0.954	0.965	0.942	0.953	0.984	0.949	0.967	0.930	0.948	0.985	0.920	0.943	0.894	0.918	0.966
5x5x38	0.952	0.963	0.940	0.951	0.984	0.944	0.962	0.924	0.943	0.982	0.917	0.939	0.892	0.915	0.966
3x3x38	**0.932**	**0.935**	**0.928**	**0.932**	**0.982**	**0.926**	**0.935**	**0.916**	**0.925**	**0.975**	**0.901**	**0.910**	**0.890**	**0.900**	**0.964**

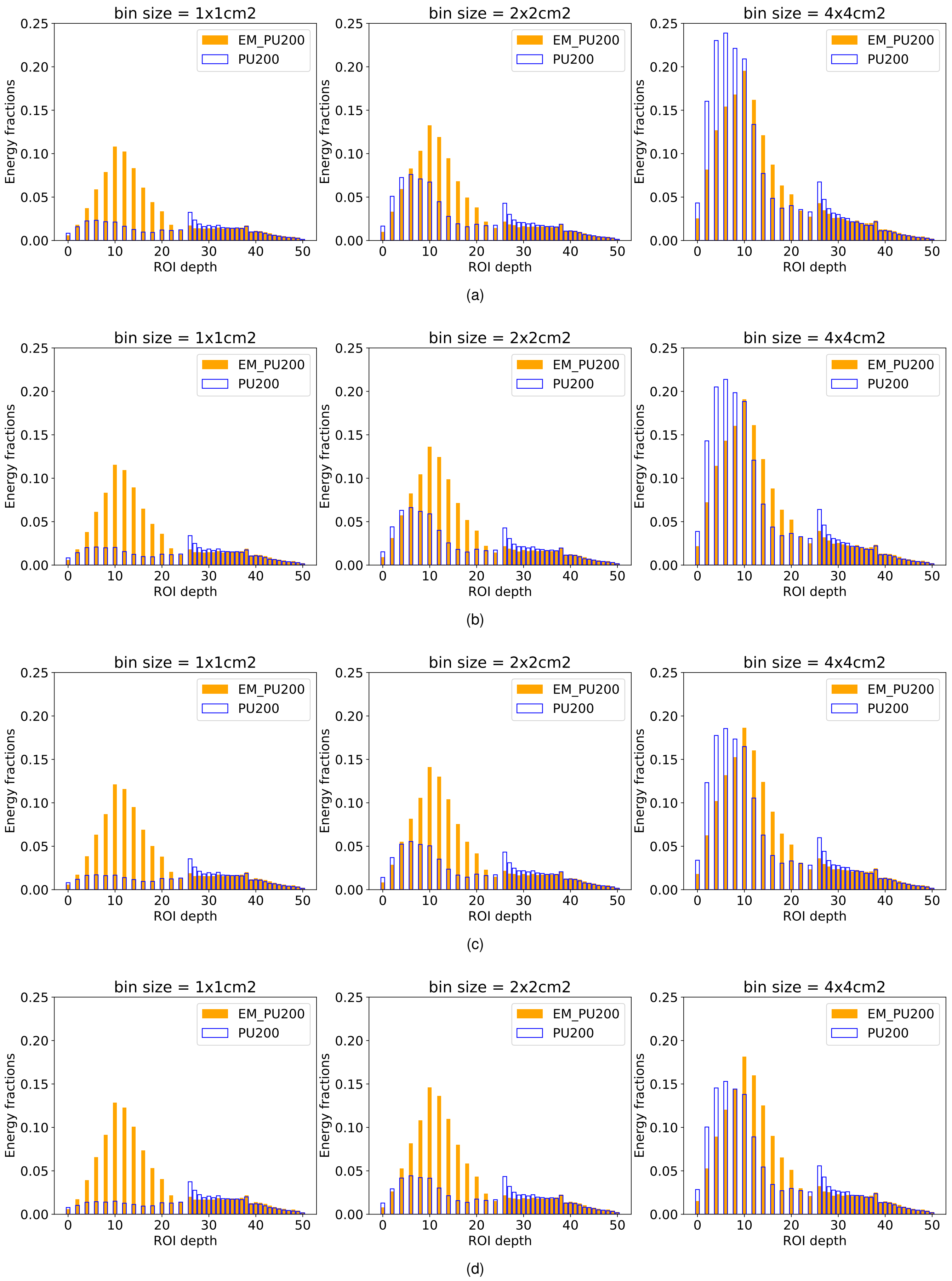
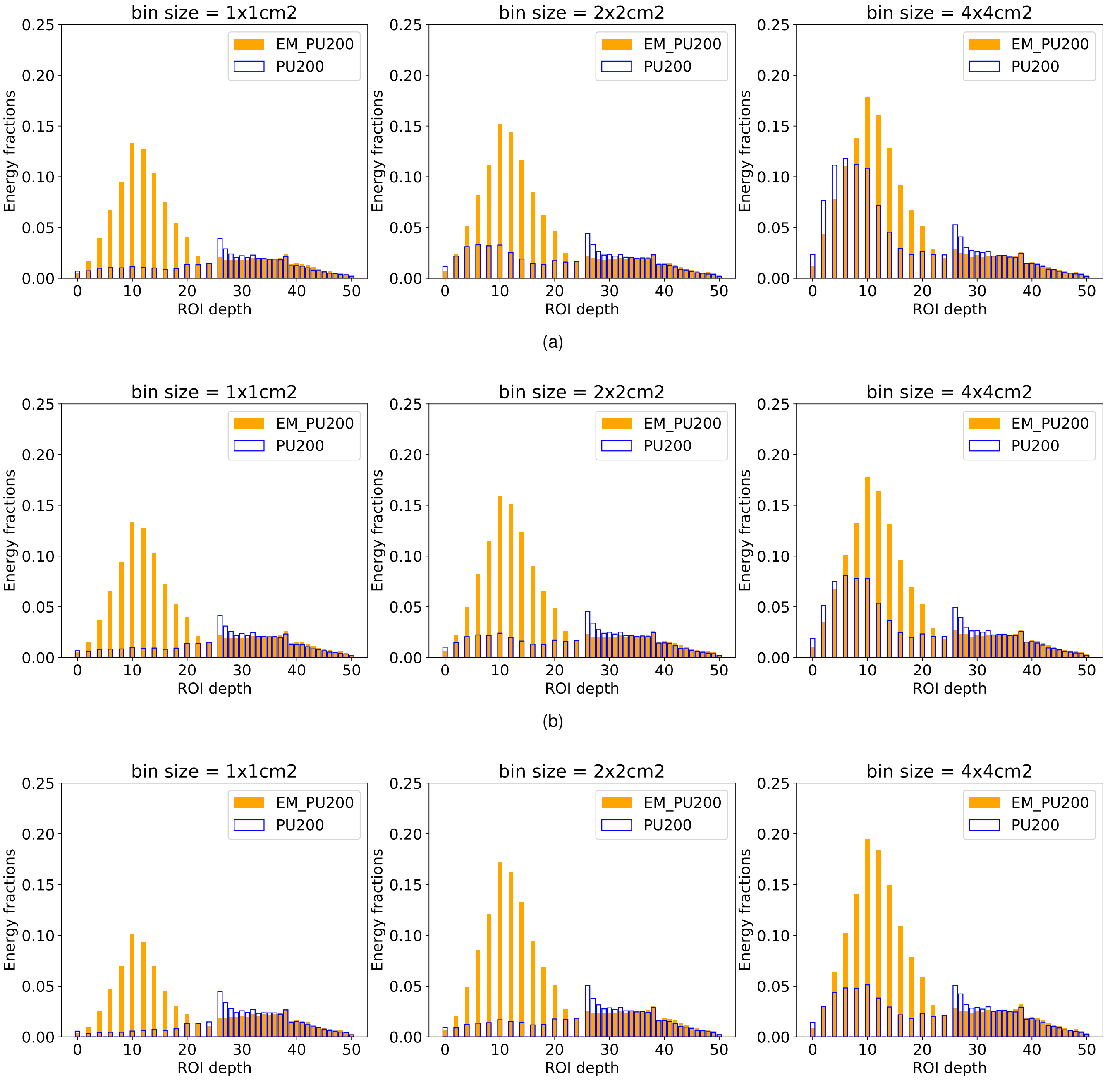


Figure 6.20: Mean energy fractions per ROI layer (1000 events). The bin sizes used are: 1x1cm^2 (left), 2x2cm^2 (middle) and 4x4cm^2 (right). The ROI sizes are: 15x15x38 (top row), 13x13x38 (second row), 11x11x38 (third row) and 9x9x38 (bottom row).

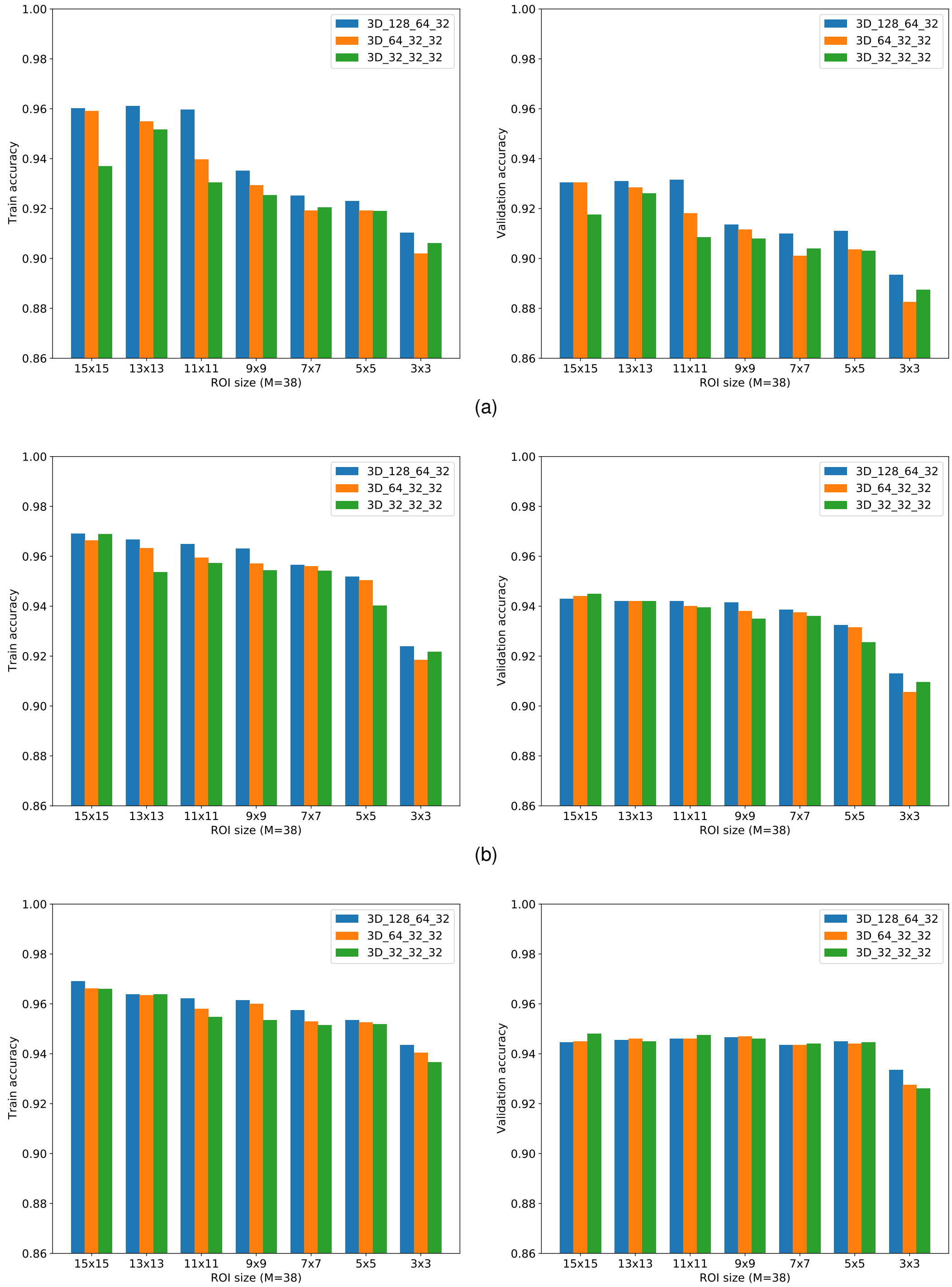


(a)

(b)

(c)

Figure 6.21: Mean energy fractions per ROI layer (1000 events). The bin sizes used are: 1x1cm^2 (left), 2x2cm^2 (middle) and 4x4cm^2 (right). The ROI sizes are: 7x7x38 (top), 5x5x38 (middle) and 3x3x38 (bottom).

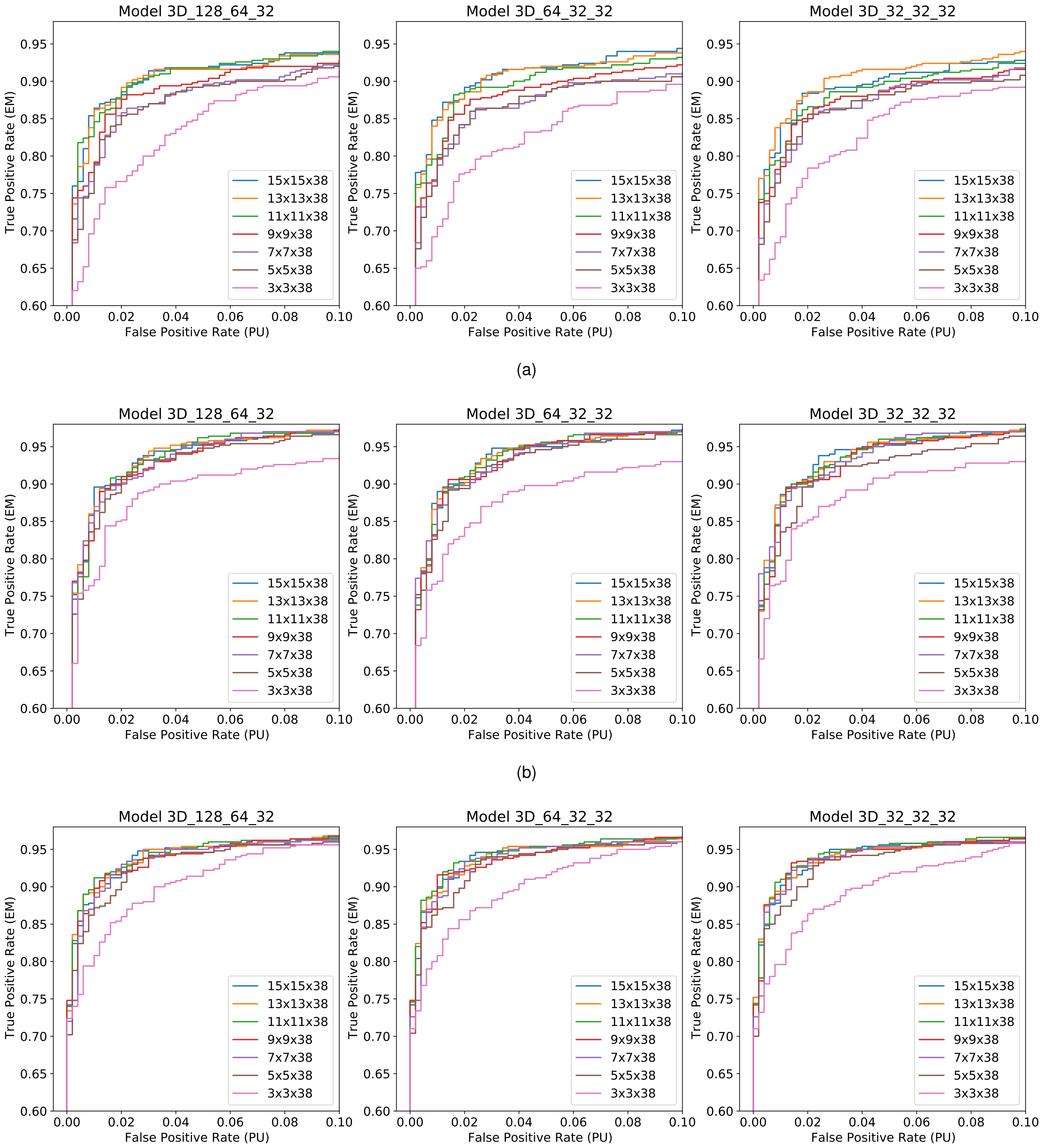


(a)

(b)

(c)

Figure 6.22: ROI size impact on the classification accuracy. ROI resolutions are: a) 1x1cm^2; b) 2x2cm^2; c) 4x4cm^2.



(a)

(b)

(c)

Figure 6.23: ROI size impact on the classification accuracy. ROI resolutions are: a) 1x1cm^2; b) 2x2cm^2; c) 4x4cm^2.

The results show that differences in classification power when different ML models are used are negligible, so that there is not much improvement with the 3D_128_64_32 model compared to the other two, while the total number of parameters for this model is huge (Figure 6.15). Hence, we decide to keep the model with the lowest number of parameters for further studies, i.e. 3D_32_32_32, because this one offers a good compromise between the model complexity and the performance result. Moreover, the number of parameters can be further reduced by using the lower ROI size, because we saw that the classification score as well as the other evaluation metrics such as model precision, recall and AUC are very satisfactory with the minimal ROI of size 3x3x38. Small ROI does not degrade the efficiency of the classification that much, while at the same time it reduces the ML model complexity.

Finally, we decide to omit the lowest and the largest ROI resolution of 1x1cm^2 and 4x4cm^2, and to go on with the ROI resolution 2x2cm^2. Its bin area corresponds roughly to the size of a TC which makes it a natural choice for the reconstruction algorithm in the trigger. It gives better result than the lowest bin 1x1 (accuracy score of 90%), and there is no significant improvement when the classification is performed with the larger bin size 4x4 (Table 6.9).

6.4.4 Impact of the ROI depth on the classification performance

Although the full information on energy deposits is used for the ROI image generation when all detector layers are included in the multi-layer ROI concept ($M = 38$), for a classification between EM showers and PU, only ECAL layers would be sufficient. The full ROI depth with all the layers can serve as a good starting point when the hadronic component will be included in the multi-class classification (for ex. to distinguish between EM, hadronic and PU ROI deposits). However, in our simpler case, we can limit the ROI to ECAL layers only, and to use the first 14 slices of the ROI (since every second ECAL layer is used in trigger). Hence, we decrease the ROI depth to NxNxM where $M = 14$ and examine the impact on the accuracy score. Our main goal is to reduce the total ML model complexity since with a smaller number of inputs, a lower number of parameters will be required for the network (Figure 6.24).

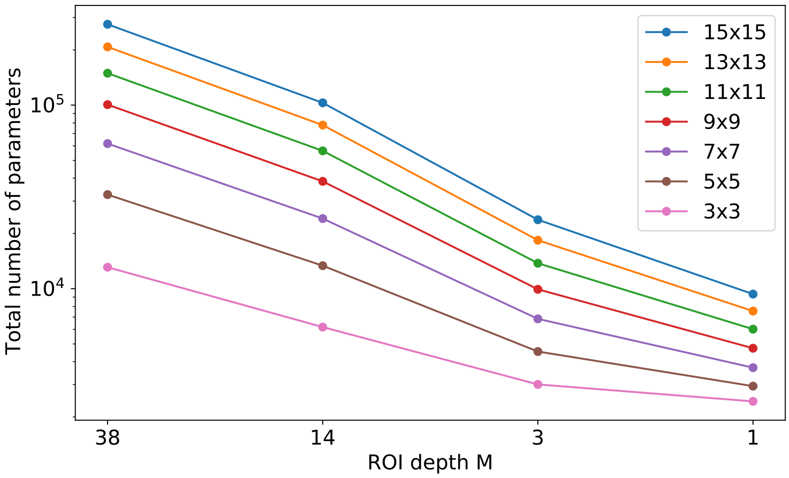


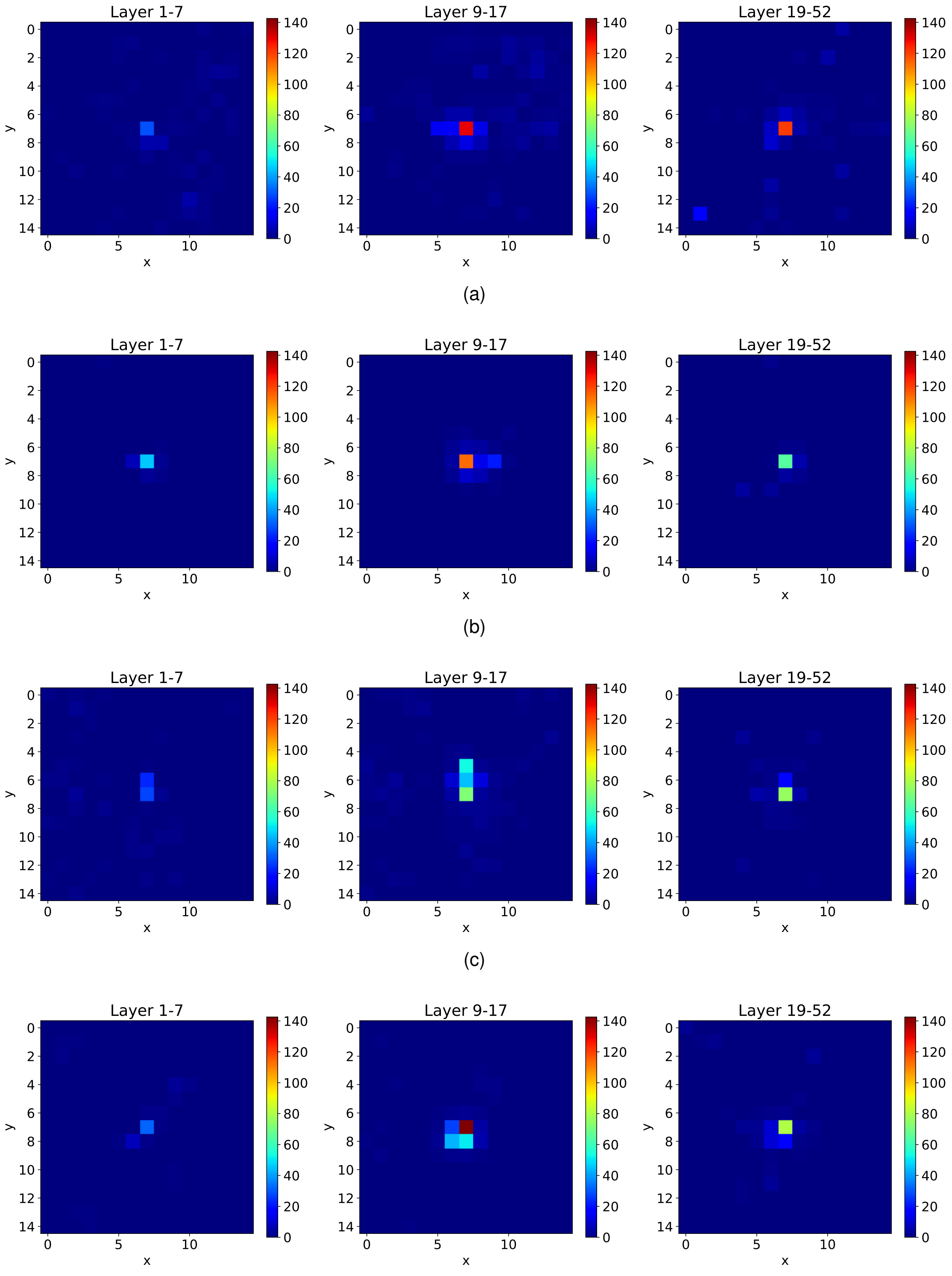
Figure 6.24: Comparison of the total number of parameters for ROIs of different sizes and depth ($M = 38, 14, 3, 1$). The results correspond to the 3D_32_32_32 ML model.

Another mechanism is used to decrease the ROI depth in the z dimension. Namely, we sum energies on layers by taking into account to which parts of the detector these layers belong to. The concept is similar to [131, 138], where authors create and process event images for each part of the detector separately. In our case, the layers are summed in the following manner: first image is the sum of the ROI layers 1 to 7 (where mostly PU is expected and EM shower is in its early stage), second image is the sum of layers 9 to 17 (where the maximum of EM shower is expected), and a final (third) image is the sum of the remaining ECAL layers (19 to 28) including all HCAL layers. Unlike in [131, 138], our images are mostly ECAL-based, since our goal is to classify only EM and PU. The examples of ROI images for the signal and the background are given on Figure 6.25 and Figure 6.26. Naturally, the signal image is more structured, with the maximum energy contained in the middle ROI image. Since PU is included in the signal sample and it mostly originates from hadrons and covers a larger area in the detector, the HCAL part of signal ROI image also contains energy, together with the PU energy present in the front layers. On the other hand, the background ROIs are more spread and the largest amount of energy is in the last HGCAL layers.

The former approach with $M = 3$ can be interpreted as an RGB-like image with three channels. We also choose to further decrease the ROI image depth to $M = 1$ by summing all the detector layers in a single 2D image, as shown on Figure 6.27 and Figure 6.28. The approach is like in [126, 134, 138], such that the event ROI image is single-channel instead of multi-channel, as already discussed in Section 6.2.1.

The result on the classification accuracy for the training and validation with different ROI depths $M = 38, 14, 3, 1$ is given on Figure 6.29. The result of the evaluation after testing the trained model with the trained model on the unseen images is presented on Figure 6.30. We can see that the conclusions from Section 6.4.2 for the impact of the ROI size on the classification remain the same - a small decrease in accuracy score is present with the lowest ROI size. We can see on Figure 6.30 that the larger degradation is present when using single-channel ROI images (The ROC curve is lower, causing a smaller number of true positives, and also it is shifted to the right, so that the false positive rate is higher). The result for the ROI of depth $M = 1$ is worse than for $M > 1$ for all ROI sizes as shown on Figure 6.29 in the validation, so it is better to use multi-channel than single-channel ROIs, which is consistent with findings in [125, 128, 131, 133, 136, 144, 148].

The validation score on Figure 6.29 shows that using ROIs of depth $M = 14$ with only ECAL layers gives an almost as good classification model as for $M = 38$. In addition, the ROI size with $M = 3$ is always worse than with $M = 14$ and it is worse than with $M = 38$ in almost all cases. On Figure 6.31 and in Table 6.10, we compare the classification results between the different ROI depths for each ROI size NxNxM, when the ML model is tested with the unseen images. We notice that the difference between the efficiencies with the AUC metric for different ROI depths is negligible, and the minimal AUC result is always larger than 93%. Also, the AUC and accuracy results are slightly decreasing for a lower number of channels in the input ROI image. The mean AUC score and other metrics except the precision of the model are the best for $M = 38$, while it is slightly degraded for a reduced depth. In addition, the lower number of channels with summed layers ($M = 3$) is slightly less efficient for the classification



(a)

(b)

(c)

(d)

Figure 6.25: 15x15x3 signal ROIs (bin size 2x2cm^2). Randomly selected events with the identification number: a) 0; b) 3; c) 7 and d) 10.

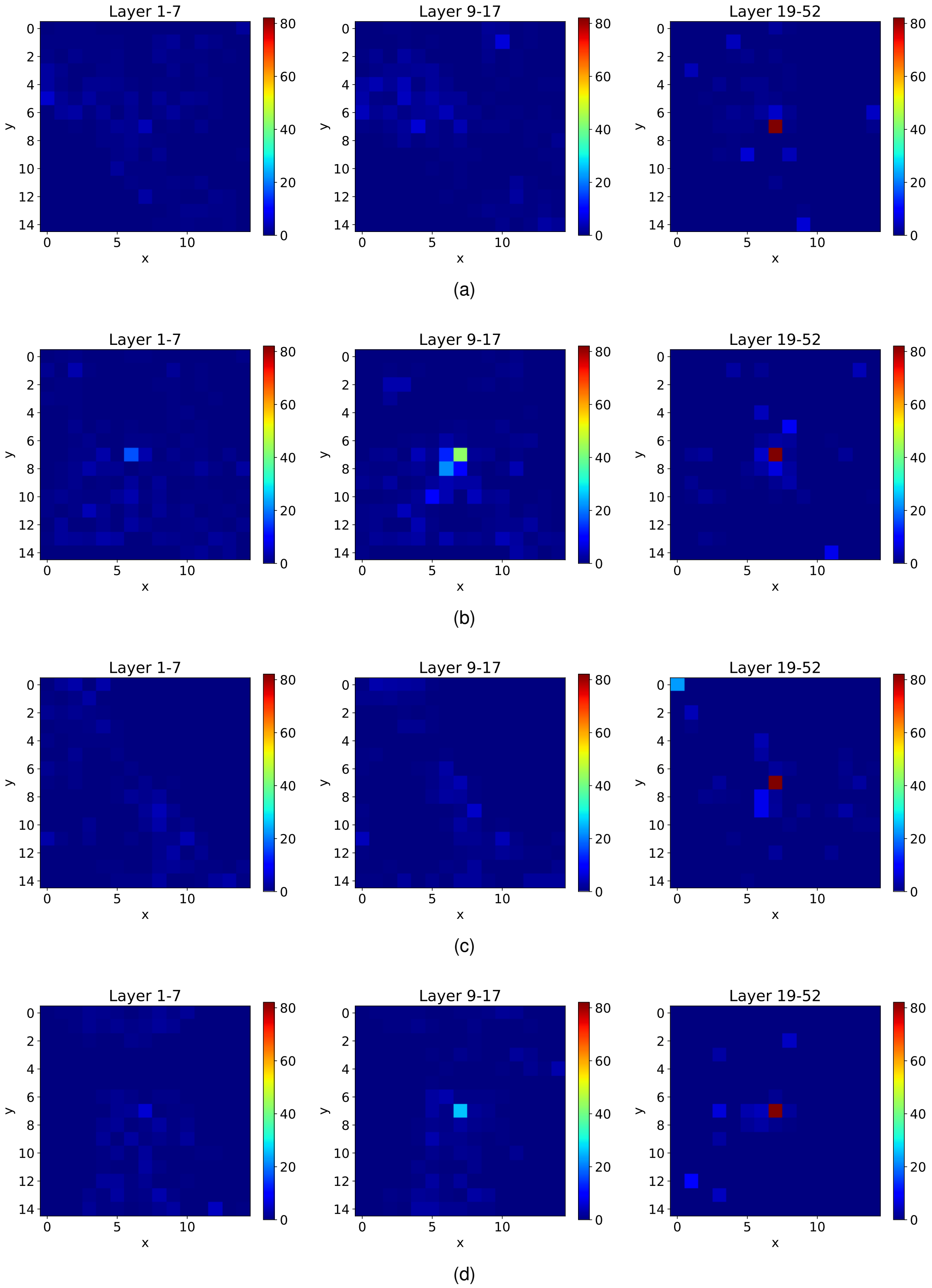
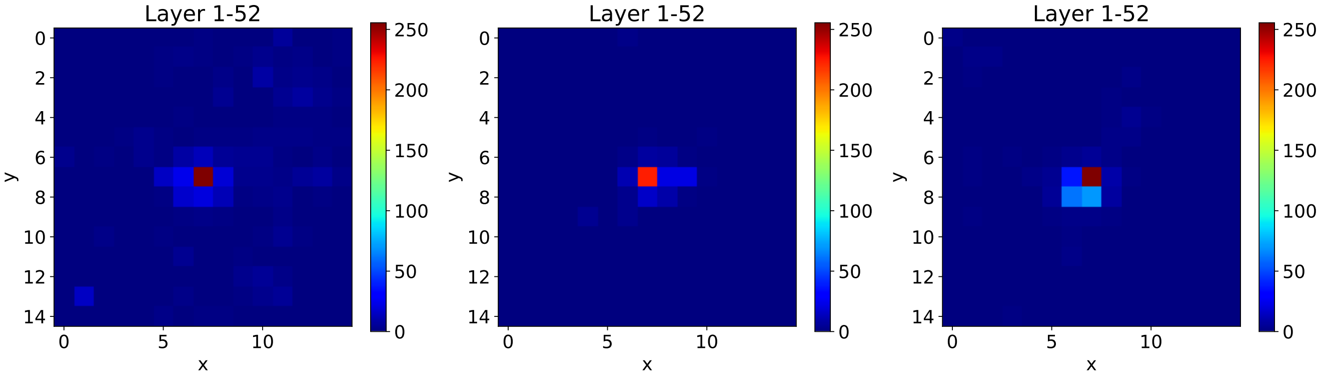


Figure 6.26: 15x15x3 background ROIs (bin 2x2cm^2). Randomly selected events with the identification number: a) 1; b) 2; c) 3 and d) 7.

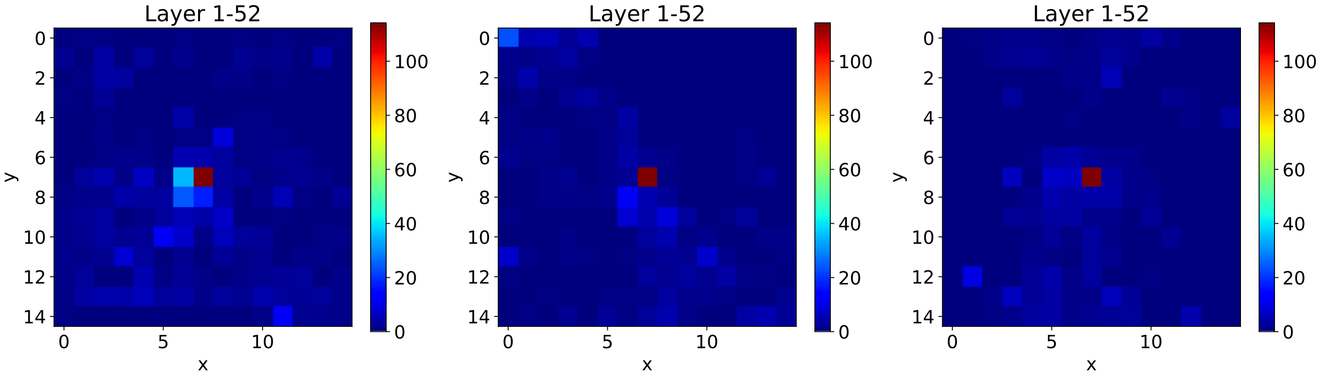


(a)

(b)

(c)

Figure 6.27: Projection of the signal ROIs in x-y plane (ROI size is 15x15x1, bin size is 2x2cm^2). Randomly selected events with the identification number: a) 0; b) 3 and c) 10.



(a)

(b)

(c)

Figure 6.28: Projection of the background ROIs in x-y plane (ROI size is 15x15x1, bin size is 2x2cm^2). Randomly selected events with the identification number: a) 2; b) 3 and c) 7.

than $M = 14$ and $M = 38$. The same effect is present for $M = 1$, which is the least efficient. However, it can be seen that the degradation with lower depth ROIs is rather small and the classification results are still very satisfactory with accuracy $> 91\%$.

Concerning the precision of the classified images, it is better when there is no summation over the layers and we use only ECAL or the full detector information. This is expected since we lose information on the raw TC energies by summing the consecutive layers.

Table 6.10: Results for the different ROI depth of $M = 38, 14, 3, 1$ (bin size: 2x2cm^2, ML model: 3D_32_32_32).

ROI	M=38					M=14					M=3					M=1				
	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC
15x15	0.956	0.965	0.946	0.956	0.984	0.950	0.969	0.930	0.949	0.983	0.946	0.959	0.932	0.945	0.981	0.924	0.945	0.900	0.922	0.966
13x13	0.952	0.973	0.930	0.951	0.984	0.955	0.975	0.934	0.954	0.982	0.945	0.957	0.932	0.944	0.982	0.926	0.946	0.904	0.924	0.966
11x11	0.949	0.967	0.930	0.948	0.984	0.951	0.971	0.930	0.950	0.982	0.937	0.958	0.914	0.936	0.981	0.924	0.949	0.896	0.922	0.966
9x9	0.945	0.965	0.924	0.944	0.985	0.950	0.965	0.934	0.949	0.983	0.939	0.956	0.920	0.938	0.982	0.923	0.949	0.894	0.921	0.965
7x7	0.949	0.967	0.930	0.948	0.985	0.938	0.964	0.910	0.936	0.981	0.940	0.962	0.916	0.939	0.983	0.922	0.949	0.892	0.920	0.966
5x5	0.944	0.962	0.924	0.943	0.982	0.938	0.966	0.908	0.936	0.980	0.936	0.952	0.918	0.935	0.980	0.900	0.937	0.858	0.896	0.958
3x3	**0.926**	**0.935**	**0.916**	**0.925**	**0.975**	**0.924**	**0.945**	**0.900**	**0.922**	**0.964**	**0.926**	**0.944**	**0.906**	**0.924**	**0.970**	**0.872**	**0.917**	**0.818**	**0.865**	**0.932**
Mean	0.946	0.962	0.929	0.945	0.983	0.944	0.965	0.921	0.942	0.979	0.938	0.955	0.920	0.937	0.980	0.913	0.942	0.880	0.910	0.960

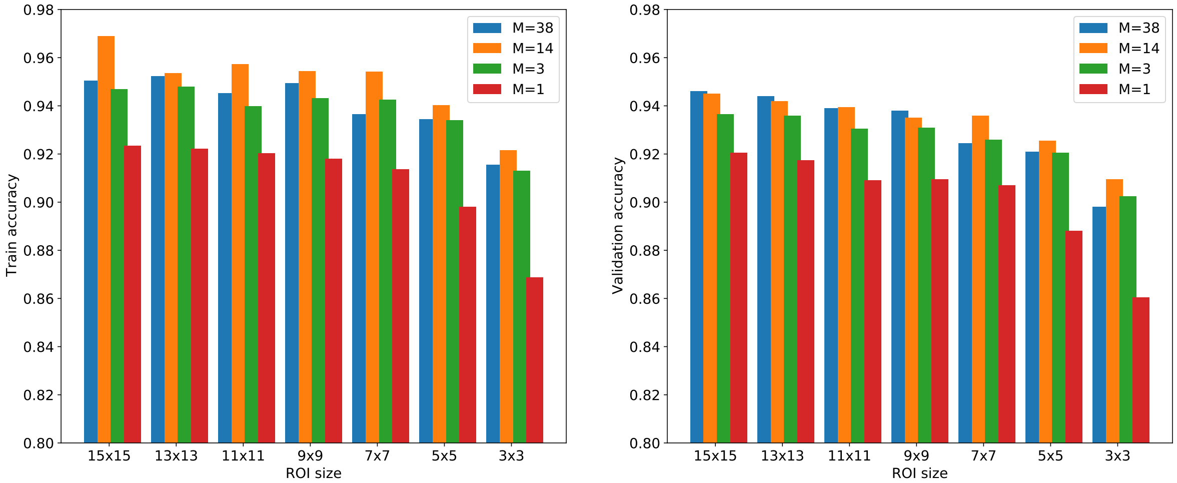


Figure 6.29: ROI depth impact on classification accuracy (ROI resolution: 2x2cm^2, ML model: 3D_32_32_32).

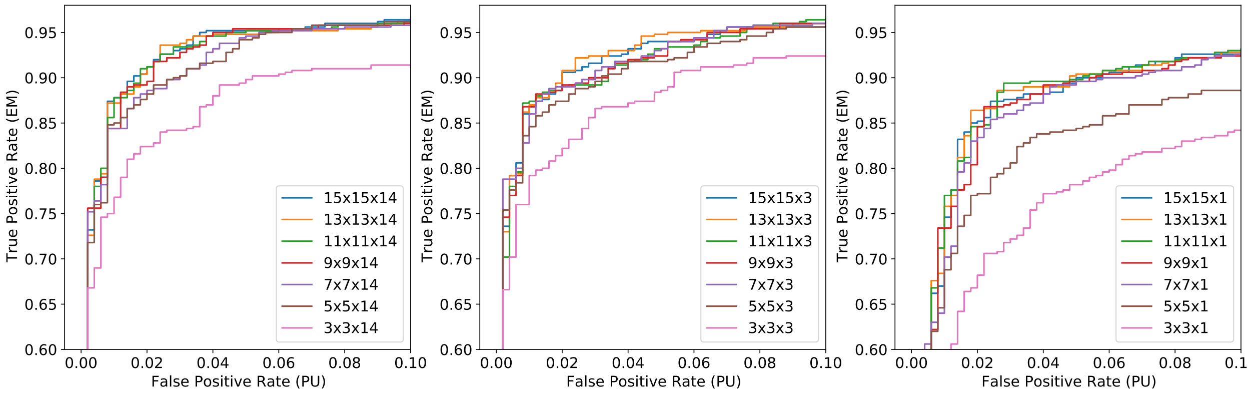


Figure 6.30: ROI depth impact on classification accuracy (ROI resolution: 2x2cm^2, ML model: 3D_32_32_32). ROI depth: $M = 14$ (left), $M = 3$ (middle) and $M = 1$ (right).

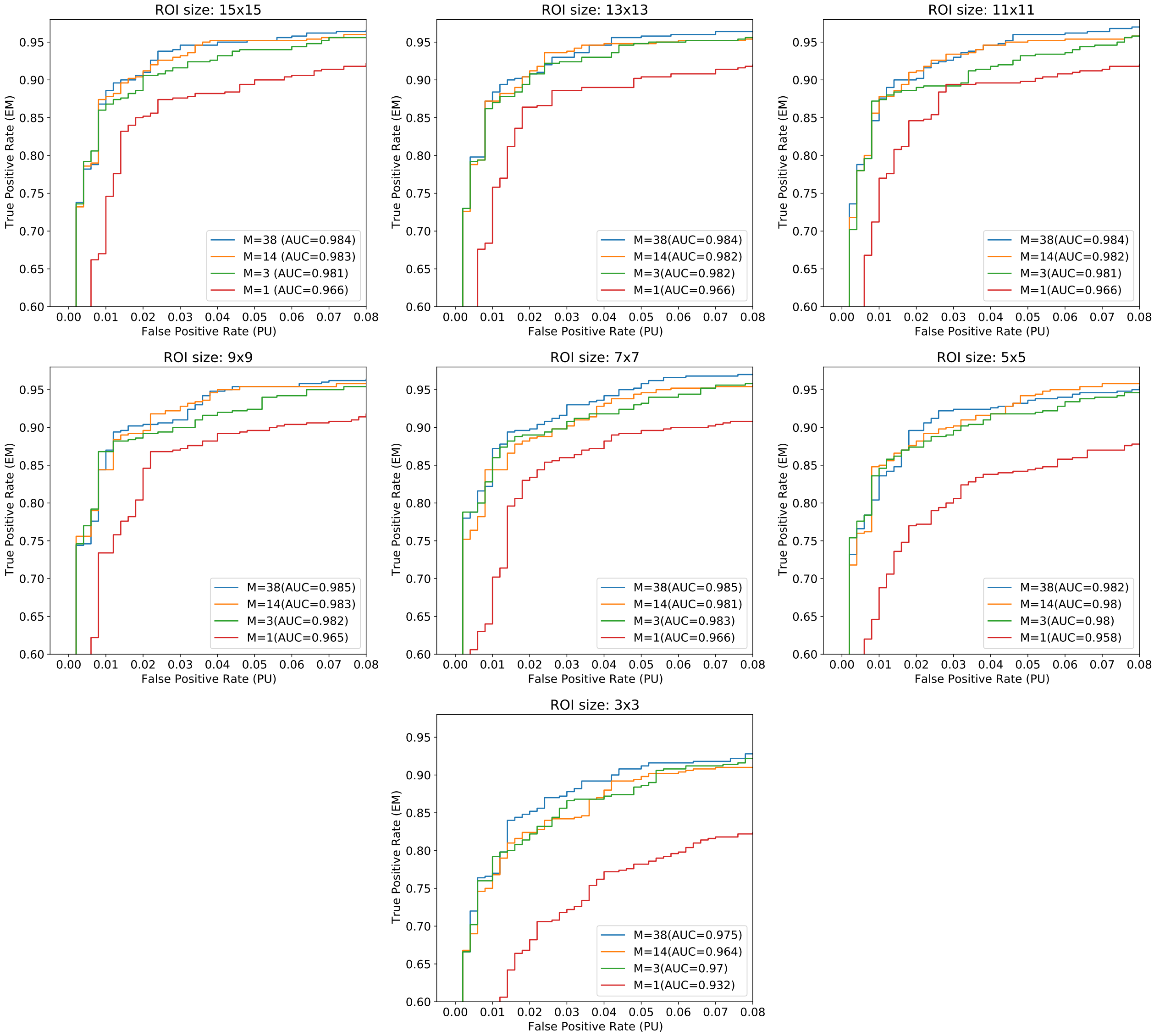


Figure 6.31: ROI depth impact on the classification accuracy for different ROI sizes $NxNxM$, $M = 38, 14, 3, 1$ (ROI resolution: 2x2cm^2, ML model: 3D_32_32_32).

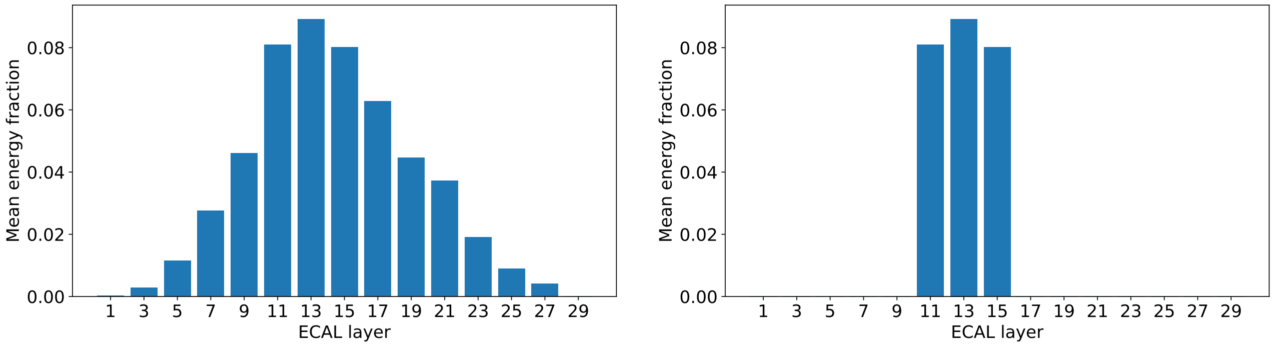


Figure 6.32: Illustration of a pre-processing of ROI images. Full EM profile (left) and maximum EM profile (right).

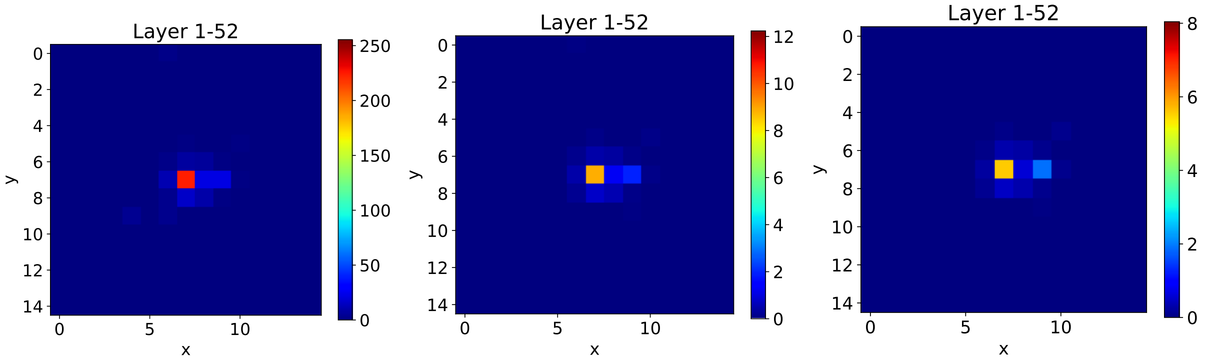
6.5 Classification with pre-processed single-channel 2D ROI images

We have seen in Section 6.4.4 that the single-channel ROI image where $M = 1$ is less efficient than multi-channel ROI variants. It is to notice that there was no specific type of "pre-processing" applied in any of the previous results, besides the ROI selection in order to generate the image. However, the state-of-the-art review on ML in HEP in Section 6.2 revealed that using single-channel ROI images is usually combined with some more specific pre-processing technique like normalizing the energy values or applying a threshold for the noise reduction, unlike for the multi-channel approach. Hence, we decided to apply a direct pre-processing on energy values by profiling the data based on the expectation for EM shower energy deposits. The main goal is to possibly enhance the classification results. The concept is already described in Chapter 5, and we have applied two types of EM profile on energies, as shown on Figure 6.32.

The pre-processing results for randomly selected events are shown on Figure 6.33 and Figure 6.34. It can be seen that there is not much difference for signal ROI images, since using the full EM profile or the maximum EM profile is more or less the same. The energy weights are different and they affect the signal but not that much, since weights that are zero or very close to zero produce almost the same effect. The image is just cleared from PU in the first layers and HCAL deposits, such that maximum of the shower is more clear and with less low-energy PU contributions. On the other hand, the background ROI images are different with and without pre-processing. The reason for this is that, unlike in the signal sample which contains EM shower but with the signal-like characteristic (Figure 6.32 a)) and PU included, so when we apply the weights we emphasize the signal, in the case of the background sample, we just take a random part of the sample. Since PU is background and does not include the signal we are interested in, applying the weights (any of the two profiles on Figure 6.32), does not mean much difference. Since PU contains several contributions (low-energy EM particles in the first layers, hadronic contribution in the layers around 12-38, etc.), applying any weights on the energies before projection means just taking a random part of the sample making the resulting image more random.

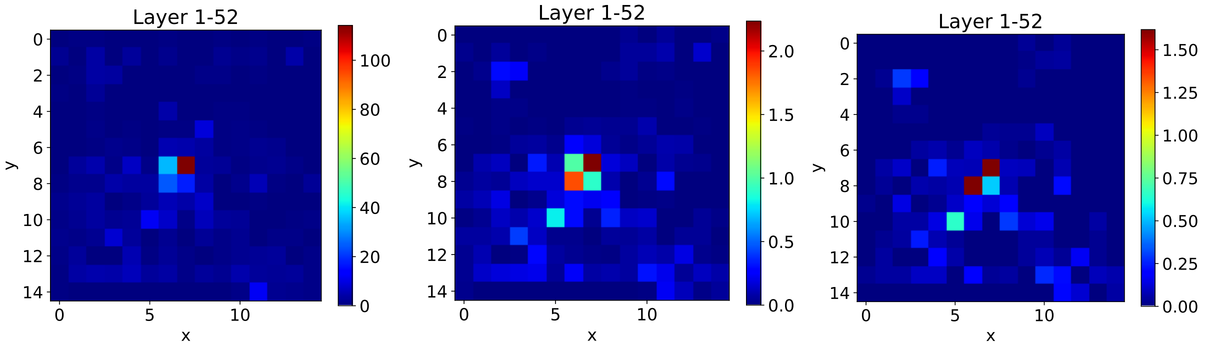
The impact of processing on the training and validation accuracy is given on Figure 6.35. The evaluation of the predicted classes with the unseen images is given with ROC curves on Figure 6.36. It can be seen that there is not much difference with and without processing, and also with the two profile types. A degradation is visible with the lower ROI size, which is in line with previous findings. The numerical details in Table 6.11 show that the minimum accuracy score of the model is around 91%. Applying the EM profile results in a slightly better precision than without pre-processing, which means that more predicted signal ROIs are relevant (actual signals), and also that the sensitivity of the model is higher, with more relevant results classified correctly. We can see from the mean AUC score that pre-processing techniques slightly increase the efficiency of our binary classifier.

There is another advantage from the maximum EM profile: there are less multiplications needed to multiply the energies with the layer weights. Consequently, the hardware implementation of the system can be simplified. In



(a) (b) (c)

Figure 6.33: Projection of the signal ROIs in x-y plane, (event number 3, ROI size 15x15x1, bin 2x2cm^2). Preprocessing techniques: a) none; b) full EM profile and c) maximum EM profile.



(a) (b) (c)

Figure 6.34: Projection the background ROIs in x-y plane, (event number 2, ROI size 15x15x1, bin 2x2cm^2). Preprocessing techniques: a) none; b) full EM profile and c) maximum EM profile.

addition, the maximum EM profile can be used as another version of multi-channel ROI image, such that an RGB-like structure is accomplished with only three maximum energy layers. Using these 3 layers increases the ROI depth to $M = 3$ instead of a single layer $M = 1$ with all layers summed together. This can enhance the classification result for the single-channel approach, and the model can be simplified with a lower total number of parameters compared to cases where a larger number of layers ($M > 3$) is used.

Table 6.11: Results for different processing (ROI depth: $M = 1$, bin: 2x2cm^2, model: 3D_32_32_32).

ROI size	No processing					Full EM profile					Max EM profile				
	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC	acc	prec	rec	F1	AUC
15x15x1	0.912	0.935	0.886	0.910	0.965	0.927	0.973	0.878	0.923	0.976	0.923	0.949	0.894	0.921	0.975
13x13x1	0.926	0.946	0.904	0.924	0.966	0.929	0.971	0.884	0.926	0.976	0.917	0.946	0.884	0.914	0.974
11x11x1	0.924	0.949	0.896	0.922	0.966	0.937	0.960	0.912	0.935	0.975	0.916	0.939	0.890	0.914	0.973
9x9x1	0.923	0.949	0.894	0.921	0.965	0.928	0.957	0.896	0.926	0.974	0.919	0.941	0.894	0.917	0.974
7x7x1	0.922	0.949	0.892	0.920	0.966	0.896	0.965	0.822	0.888	0.958	0.912	0.940	0.880	0.909	0.967
5x5x1	0.900	0.937	0.858	0.896	0.958	0.889	**0.964**	0.808	0.879	**0.944**	**0.864**	0.979	**0.744**	**0.845**	**0.949**
3x3x1	**0.872**	**0.917**	**0.818**	**0.865**	**0.932**	**0.874**	0.984	**0.760**	**0.858**	0.947	0.898	**0.934**	0.856	0.894	0.952
Mean	0.911	0.940	0.878	0.908	0.960	0.911	0.968	0.851	0.905	0.964	0.907	0.947	0.863	0.902	0.966

6.6 The classification with the three separate views of the detector data

Here, we present a study with a different projection-based concept, where event energies are projected in three independent directions (x-y, x-z and y-z). Hence, 3 images are generated per event, and these are single-channel ROIs where $M = 1$. A visualization of this strategy with 3 projections is presented in Section 6.6.1. Next, the ML architecture with 3 parallel FC processing branches that extract features that are merged at the end and processed by another FC layer is described in Section 6.6.2.

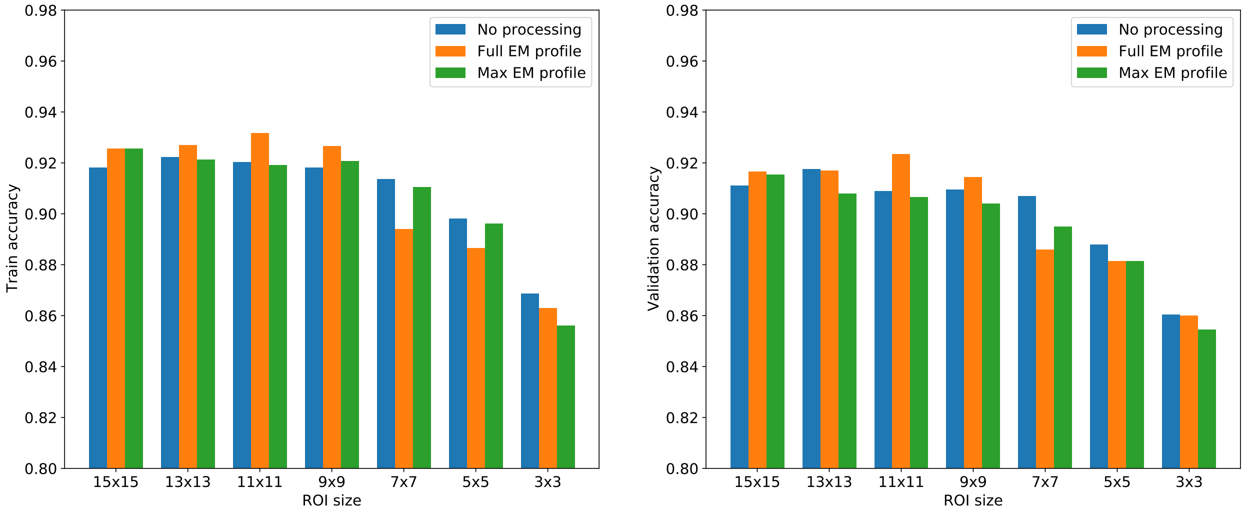


Figure 6.35: Pre-processing impact on the classification accuracy, ROI size NxNx1 (bin: 2x2cm^2, model: 3D_32_32_32).

6.6.1 Visualization of the ROI data projections

Figure 6.37 and Figure 6.38 show visualizations of several data events in the 3 directions. The event selection is random, the visualization is performed with randomly selected event numbers.

We have already seen that, for the x-y projection of the ROI, the maximal energy is contained inside the central bin and the nearby energies are located around the center. This is due to the "projective" energy accumulation trough the plane, where the maximal energy deposits follow an imaginary straight line in z-direction. Like we can notice that the x-y image is compact, the same is valid for the x-z and y-z projections. Figure 6.37 clearly reflects the effect when the 3D cluster of the EM shower energies is sliced along the specific dimension.

Background ROI projections from Figure 6.38 are more spread in the x-y projection, and also the shower is wider and longer along the both x-direction and y-direction. In addition, we can see in the signal images that there is a

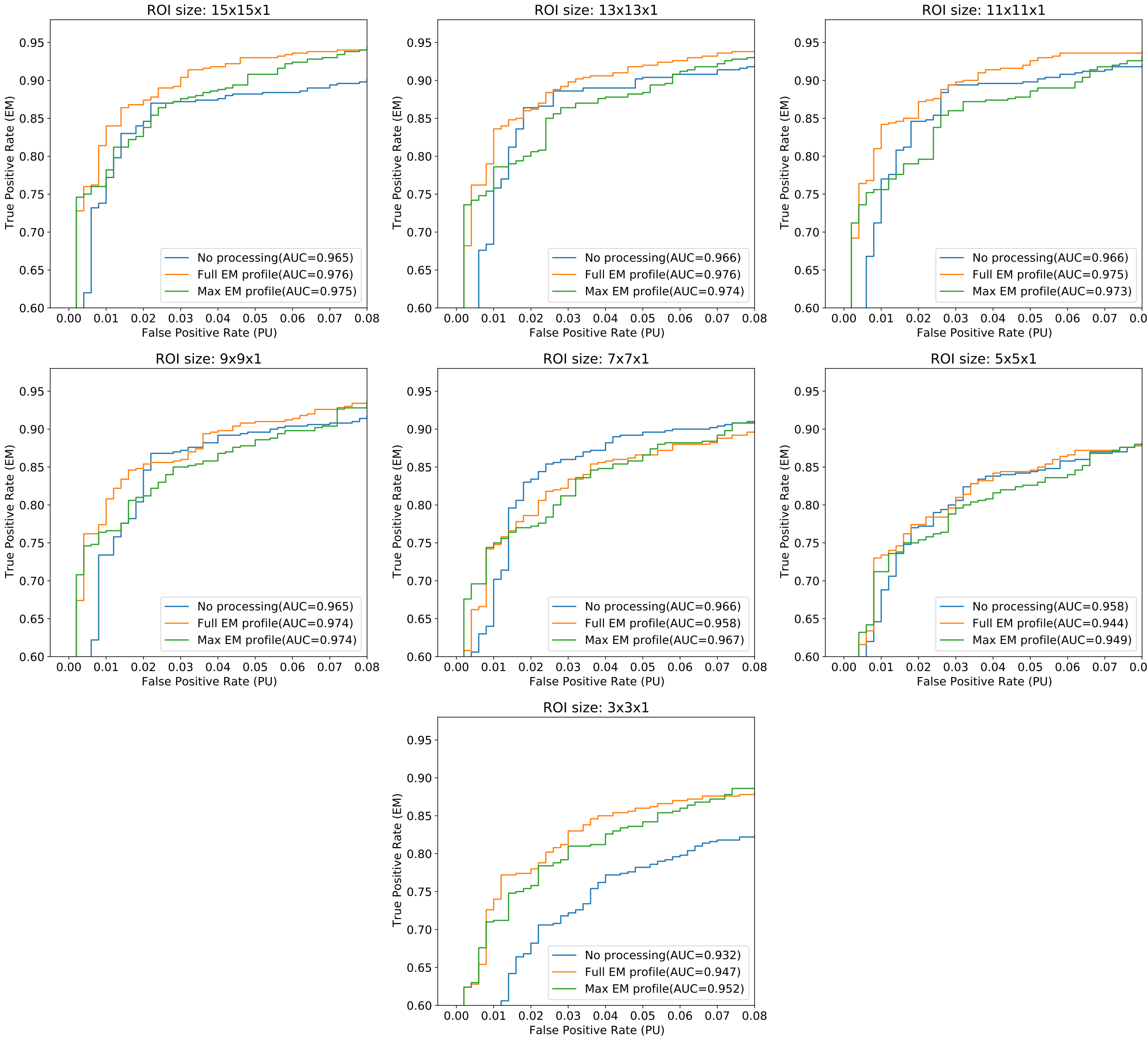


Figure 6.36: Pre-processing impact on the classification accuracy, ROI size NxNx1 (bin: 2x2cm^2, model: 3D_32_32_32).

Table 6.12: Results with the projected 3 views scheme (parallel FCNN architecture, bin size: 2x2 cm^2)

ROI size (3 views)	acc	prec	rec	F1	AUC
15x15, 15x38, 15x38	0.956	0.967	0.944	0.955	0.985
13x13, 13x38, 13x38	0.956	0.969	0.942	0.955	0.985
11x11, 11x38, 11x38	0.953	0.969	0.936	0.952	0.985
9x9, 9x38, 9x38	0.951	0.967	0.934	0.950	0.985
7x7, 7x38, 7x38	0.946	0.965	0.926	0.945	0.985
5x5, 5x38, 5x38	0.941	0.957	0.924	0.940	0.983
3x3, 3x38, 3x38	**0.929**	**0.937**	**0.920**	**0.928**	**0.976**
Mean	0.947	0.962	0.932	0.946	0.983

rather highly energetic EM contribution, together with the lower energies that originate from PU. On the contrary, in the background PU ROI projections, there is a mix of several contributions. For example, mostly EM contribution on Figure 6.38 b) and d), but mixed with the hadronic energies later in the detector (throughout the layers), or mostly hadronic contribution on Figure 6.38 a) and c).

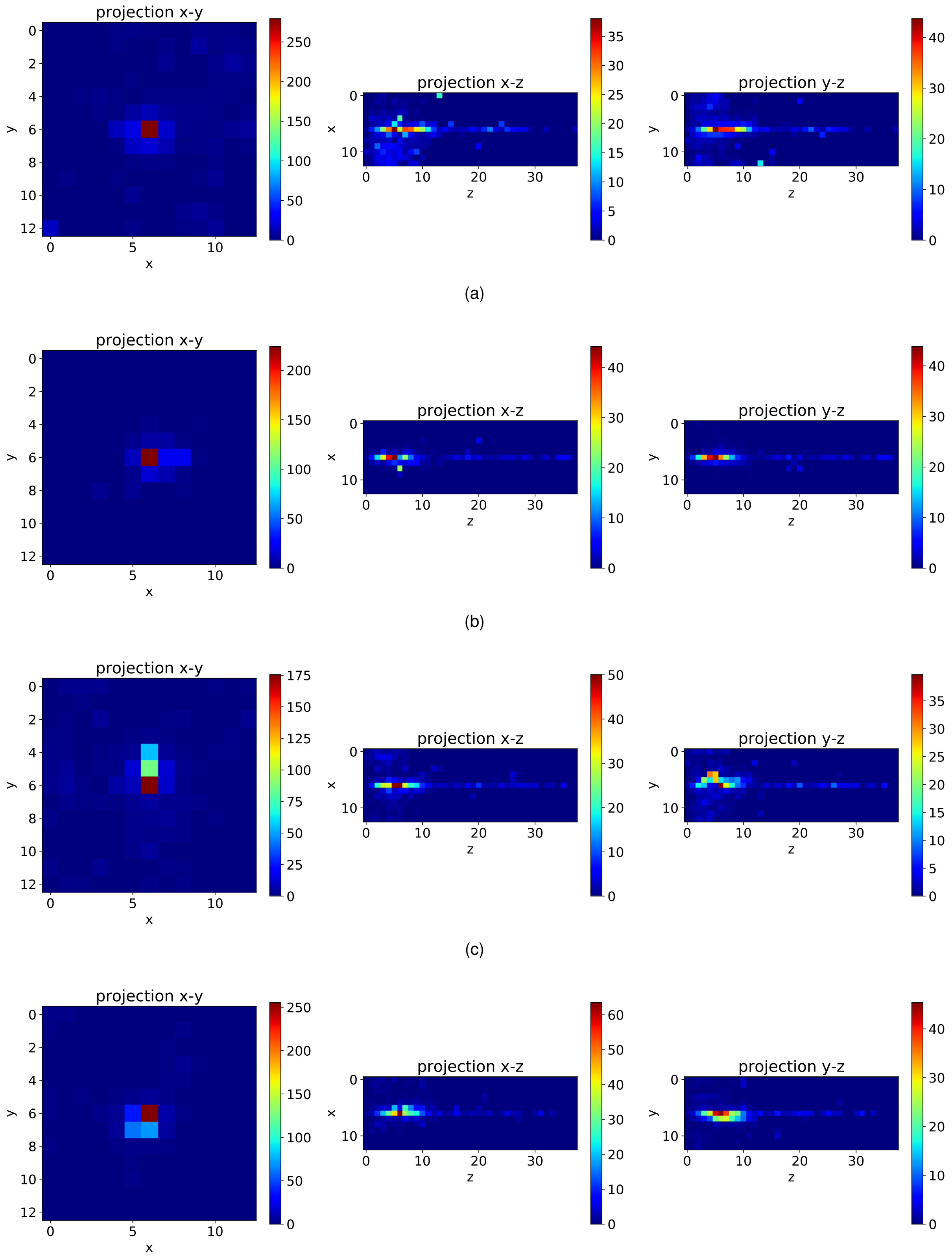
Concerning the dimensions of the generated images, they are NxNx1 in the x-y projection and Nx38x1 in both, the x-z and y-z projection, where $N = 15, 13, 11, 9, 7, 5, 3$. One can see that the depth of the images is $M = 1$ due to the single-channel 2D ROI projected results, and the width of the x-z and y-z is 38 because this was the depth of the multi-channel image $M = 38$ that is now projected.

6.6.2 Classification with ROI images for 3 projections

The ML architecture used for the training with ROI images in 3 separate projections on different planes is similar to [133, 137, 145, 154], with parallel FCNN branches to process each of the event views. As already mentioned, each plane of the projected 3D detector TC energies is actually a single-channel 2D image provided by summing the slices along a specific dimension (Figure 6.39). The 3 projections are then input into 3 independent 2D FCNNs, which work in parallel and extract the corresponding features. These FCNNs are designed as the ML model 3D_32_32_32. Finally, the outputs of these 3 FCNNs are concatenated and the merged result is given as input to another FCNN, which consists of a single dense layer with 32 neurons. Training parameters are kept the same as described in Section 6.4.1.

It is shown on Figure 6.40 that the training and validation results with this new FCNN architecture is satisfactory ($> 94\%$ efficiency). The results with the trained model on the set of unseen images is given on Figure 6.41. It is shown that again a degradation is present with the smallest ROI size, while there is not much difference for the AUC metric with larger ROIs. Also, when we compare those ROC curves with the ones on Figure 6.31, it is noticed that a slightly more efficient AUC is obtained. The numerical details on other prediction metrics is reported in Table 6.12.

In summary, the proposed parallel FCNN architecture over performs other models as it achieves the best classification performance among all compared ML solutions (Table 6.13). Also, it is more cost-effective for the classifica-



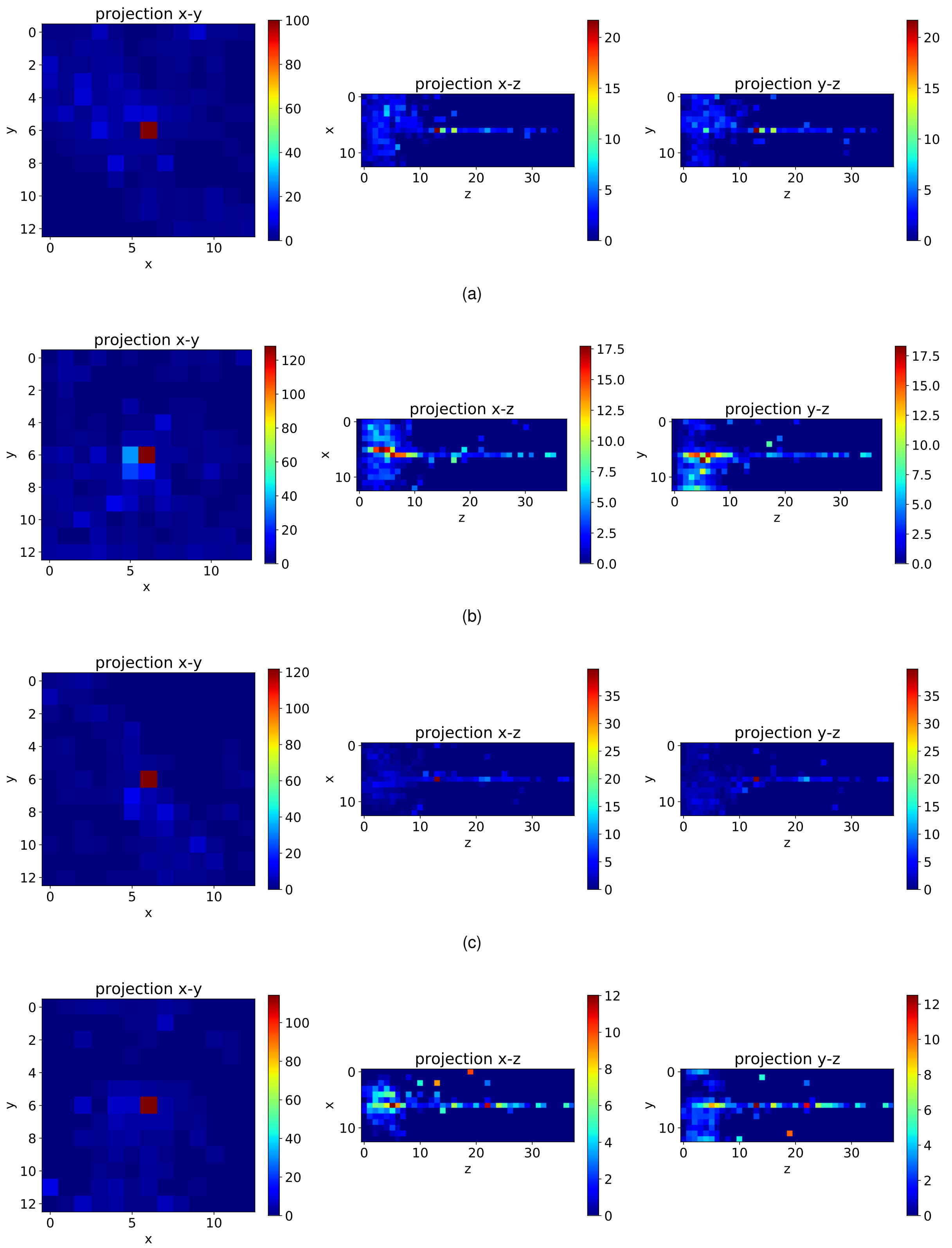
(a)

(b)

(c)

(d)

Figure 6.37: Signal ROIs with the 3 separated views of the event data (bin 2x2cm^2). Randomly selected events with the event numbers: a) 0; b) 3; c) 7 and d) 10.



(a)

(b)

(c)

(d)

Figure 6.38: Background ROIs with 3 separated views of the event data (bin 2x2cm^2). Randomly selected events with the event numbers: a) 1; b) 2; c) 3 and d) 7.

tion of event ROI images as signal (EM) or background (PU) than the other tested models in this study. It offers a higher efficiency in terms of the total number of parameters as well, so that a less complex NNet model is designed (Figure 6.41). The improvement is more important for the larger ROI sizes, whereas for ROI sizes of $N \leq 7$, the proposed architecture is less parameter-efficient than $M = 14$, though still better than $M = 38$. However, we can say that again the slightly degraded parameter efficiency is alleviated by the better classification performance. On the other hand, models like $M = 3$ and $M = 1$ can be useful when the complexity of the model is the most important requirement, because the lower number of parameters is alleviating the lower classification AUC score.

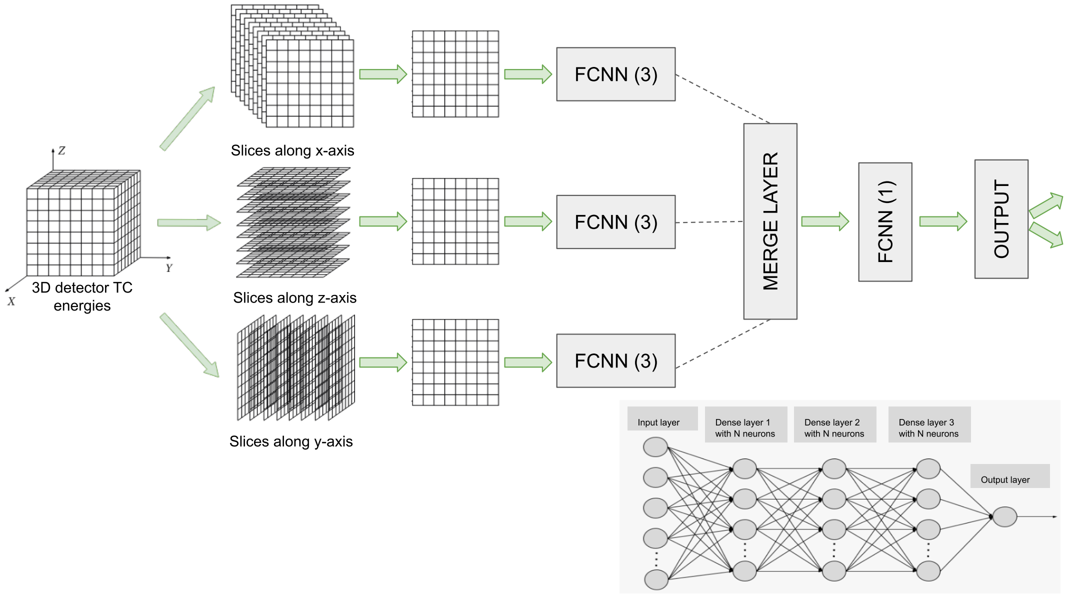


Figure 6.39: Parallel FCNN architecture used in the study. Inspired by [158].

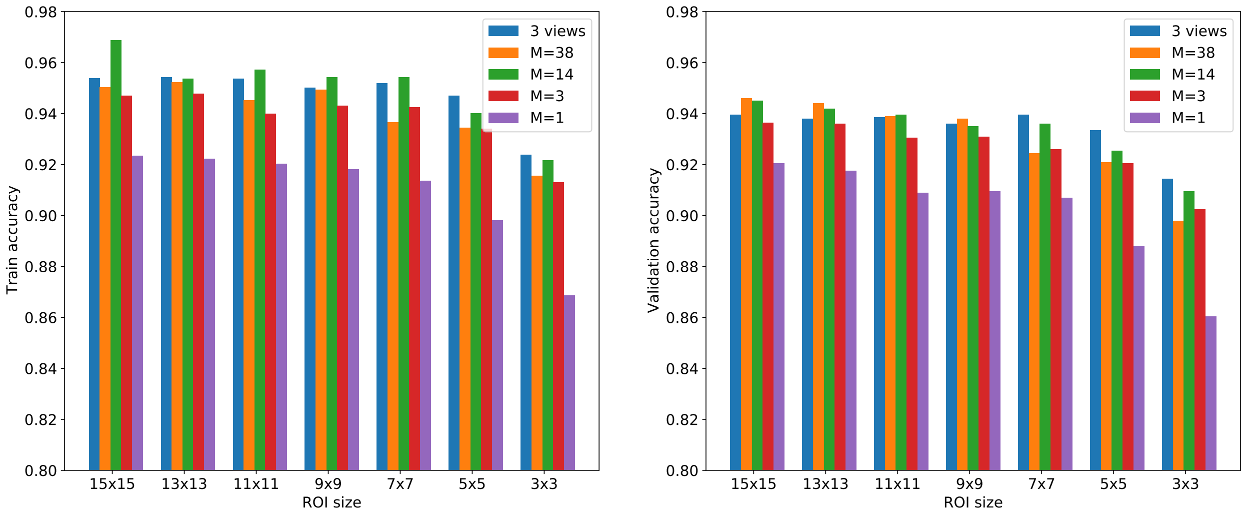


Figure 6.40: Comparison of classification accuracy with the projected 3 views and other NNet models (bin size: 2x2cm^2).

Table 6.13: Comparison of all classification models evaluated in the ML study.

metrics	projected 3 views	M=38	M=14	M=3	M=1 No processing	 Full EM profile	 Max EM profile
accuracy	**0.947**	0.946	0.944	0.938	0.913	0.911	0.907
precision	0.962	0.962	0.965	0.955	0.942	**0.968**	0.947
recall	**0.932**	0.929	0.921	0.920	0.880	0.851	0.863
F1	**0.946**	0.945	0.942	0.937	0.910	0.905	0.902
AUC	**0.983**	0.983	0.979	0.980	0.960	0.964	0.966

6.7 Discussion and evaluation

This section summarises the research findings and contributions made in the conducted ML study. A comparison of the ROI models and strategies is evaluated in the context of research findings and comparison with prior work. The main goal was to fill in some of the gaps revealed in state-of-the-art and to examine the possibility of multi-channel ROI image classification of EM showers (signal) versus PU (background) events representations in CMS HGCAL. Also, the limitations of the current study are derived to better explain the assumptions for the ML design and methods that may impact the interpretation of the key findings.

6.7.1 The summarized research findings and the comparison with prior work

The following concepts that emerged from the state-of-the-art (Section 6.2.6) are examined in this section:

- Image-based shower data representation
    - Image parameters like bin size are varied
    - Image pre-processing is applied based on the EM shower profile in data (physics heuristics)
    - Images with PU added are used (for both the signal and the background)
- Data pre-selection procedure

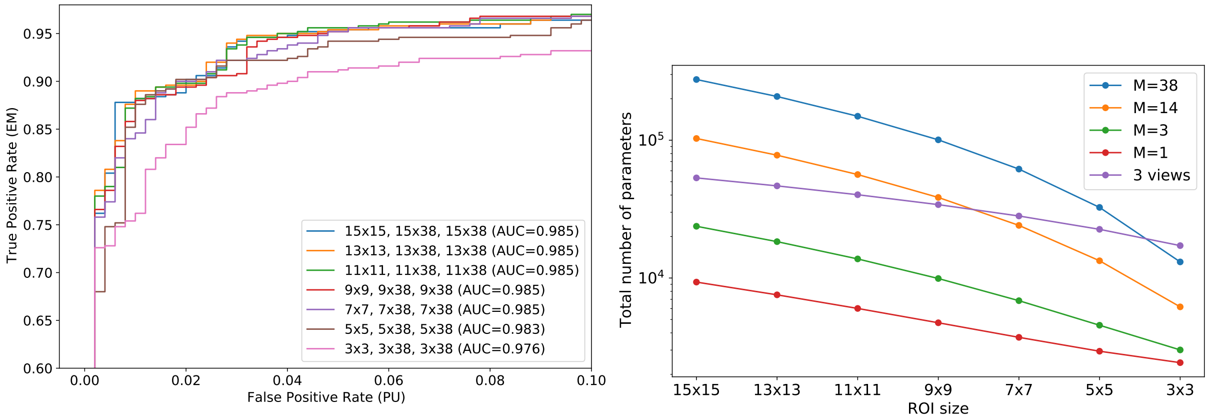


Figure 6.41: Results with ROC curves (left) and total number of parameters (right) for the projected 3 views ROIs (bin size: 2x2cm^2)

  - A ROI generation algorithm is defined for the ROI-based classification
  - Oblique projections or "projective" data ROIs are used for event images
  - The TA is defined for the ROI data extraction
- Training architecture design
  - Using only FCNN instead of the CNN layers
  - Multi-layer projections with the raw TC energies
  - Projection-based separate views of the event data in the CMS HGCAL
- Evaluation
  - Impact of the ROI resolution (bin size), ROI size and ROI depth on the classification performance
  - The evaluation of the model complexity (the total number of parameters)
  - Summing the detector layers energies to increase the classification efficiency
  - The comparison between single-channel and multi-channel ROIs
  - The classification performance with the multi-view approach in three directions

Our results on the image bin size (pixel) variation demonstrate that better discriminatory features in the NNet are gained with a larger bin size. We refer to this as the larger ROI resolution, and the accuracy is better since more energy is accumulated with the larger bin. This complies with the findings in [144, 148] which show the degraded network performance when training with images of the smallest bin size. On the contrary, in the work of [137, 154], the authors report on a decreased performance for the lowest image pixel size. However, the reported network is completely CNN-based with convolution kernels applied to extract the lowest features, so it is reasonable that these features would be lost when increasing the pixels sizes of the real image. In our case, it is better to accumulate more energy, so the bin size should approximately correspond to the size of a TC ($4cm^2$). It was shown that a lower bin size would correspond to the size of a SC ($1cm^2$), and this is not suitable for projecting the TC energies due to the loss of granularity. However, very large bins are not suitable neither, since the EM shower is rather small, so there will be no increase or decrease in the performance, and the efficiency will remain more or less same.

Next, we measure the impact of pre-processing when a single-channel image $M = 1$ is used. The results match that from the state-of-the-art methods, and we gain better performance for the ML recognition when applying pre-processing [148, 144]. We propose our own strategy for the pre-processing, which is based on the heuristics from physics, and we refer to it as the EM shower identification. For this, a longitudinal energy profile of the EM shower is encoded as energy weights so that a profiled data recognition can be performed. In addition, we have developed two a two alternatives of such profiles, to possibly simplify the hardware implementation by keeping only the layers of the maximum EM deposits.

Concerning the event data representation, we have used signal images with PU added. Hence, a realistic scenario is obtained for training, which makes our network robust to PU conditions [127, 130, 138, 143, 152].

The data pre-selection is performed by selecting a ROI in the detector volume based on a predefined radius. Authors in [127, 130, 135, 136, 151] have already tested the efficiency of the procedure. Finding interesting regions in HGCAL could be effective, since the processing volume can be reduced and one can do the clustering only around pre-selected seeds instead of in the whole detector. To enhance the ROI extraction, "projective" energy deposits are selected, since the EM shower is modelled by a straight line coming from the centre of the detector. Unlike in [139, 135], an oblique projection is applied compared to a simple orthogonal one, which only considers particles whose energy shower spread is perpendicular to the detector layers. We have defined a TA which is based on extracting the pattern of energy deposits and enables an efficient seed extraction for further processing.

The ML architecture design in the experiment demonstrates two things. First, only few dense layers can be used for an efficient training of the model, without convolutional layers. CNNs have smaller number of parameters than FCNN, but they are more difficult to implement due to larger number of operations and resources required. Second, the strategy of presenting the whole detector as 3D arrays of TC energies is used as tensor slices for the ROI generation, similar to the attempt in [151, 131]. The present study confirmed the findings about the efficient classification which can be accomplished, as well as using projection-based separate views of the HGCAL event data to implement a multi-view approach. We adjust this strategy for CMS detector, unlike the usual applications in other HEP tasks [133, 137, 145, 154].

The findings in the evaluation section are in line with previous reports. Besides the ROI resolution, we have also tested the impact of the ROI image size on the classification accuracy. The studies in [131, 141, 150, 154] have pursued image cropping to reduce the region by keeping only the points of interest and thus reduced image sparsity. Our efforts were oriented towards reducing the ROI width as much as possible, to still contain the "interesting"

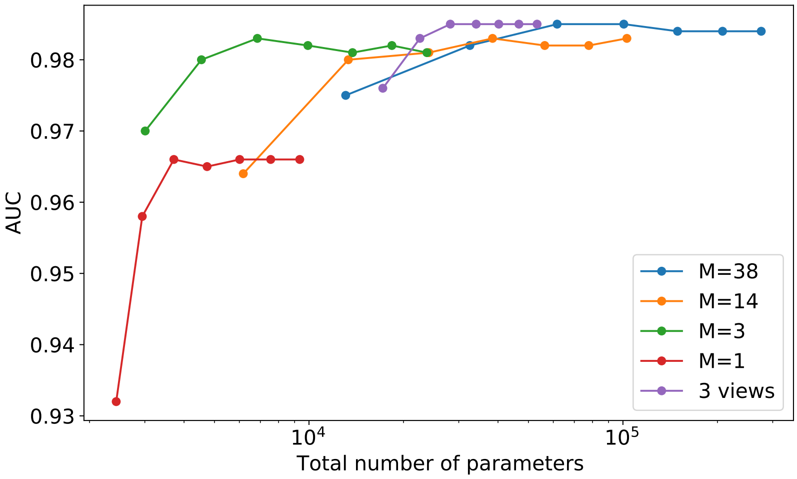


Figure 6.42: ROC curves comparing the performance of different models (the accuracy vs. the complexity).

shower data, while providing a higher parameter-efficiency for the recognition model. Hence, our findings go beyond previous reports, and the model complexity is reduced not only by the ROI size, but also by the ROI depth. A promising finding was that we can reduce both ROI dimensions and our classification performances are still very satisfactory ($> 90\%$), even using the single-channel ROI images ($M = 1$). We have verified that using an increased number of channels in the input ROI image enhances the network performance [125, 128, 153, 133, 136, 144, 148]. Finally, a multi-view in 3 separate planes (x-y, x-z and y-z) produces better results than single-view [154].

The final comparison between the presented models is shown on Figure 6.42, where the trade-off is measured between the classification accuracy (AUC) and the model complexity (the total number of parameters). It can be seen that the model with $M = 38$ is the most complex, while it is more accurate than the other multi-channel ROI approaches ($M = 14$, $M = 3$, $M = 1$). On the other hand, the model with three views is even more efficient in terms of the provided accuracy, while it offers the lower complexity than $M = 38$. The lowest complexity is accomplished with $M = 1$, but it is not as accurate as others. We can consider the model to be satisfactory in terms of accuracy because it is very high ($AUC > 96\%$) and it has the lowest complexity, but nevertheless, other models are better. The model with $M = 3$, where the three ROIs are summed along the z axis over performs all the others concerning both the AUC score and the total number of model parameters. Hence, we propose it as the best compromise between accuracy and model complexity.

6.7.2 Limitations of the study and future work

Regarding the limitations of the current study, it could be argued that not the same image generation procedure should have been used for the signal and background ROIs. Namely, as shown in Section 6.3.1 with the input data preparation, the pre-selected TC energies are projected on the virtual plane, after which a central local maximum bin with accumulated energy is extracted as a potential seed. The same process is applied for the signal and for the background, which means that a single PU seed is used for the background ROI generation. In the case of the signal, this is satisfactory since single-shower signal events are used for the signal ROI generation. Thus, only a single seed is contained inside the EM shower images, while in the case of PU events this is not realistic. Instead of a single maximum, each of the central local maximums inside a 3x3 window should be used as a potential seed for a background ROI image. A threshold should be studied in order to see how many seeds to select. Besides the PU ROI generation algorithm optimization, the classification procedure could be further improved:

- Instead of just classifying the event data as "interesting" with containing the EM signal or not, the model should also be able to reconstruct the total energy of the shower and its position in the detector volume. Then, those regions should be further analyzed to form the output of the Level 1 trigger with the TPs.
- It should be noted that a more realistic HEP event context is present in the CMS HGCAL, where the detector ROI image is not just a single-shower capture. There are many EM showers generated in HGCAL at the same

time, which usually overlap and are difficult to separate.

- It is necessary to investigate the classification of different particle showers, such as photons, electrons and pions. Possibly, a three-class multi-label problem would arise with a categorization of EM, PU and hadronic, whereas there could be sub-classes with EM_{gamma} and $EM_{electron}$ showers.
- The study could be extended by adding polar (η, ϕ) coordinate maps which are widely accepted in HEP, and to compare with the Cartesian (x, y) coordinate space used for TC energy binning.

The list provided above is a starting point for further research. First, our results with the ML study can be compared to a simple cut-and-count approach [161] based on energy cuts applied on the data and demanding that most of the data passes the cut to be labeled as signal. Again, a threshold should be previously studied that separates the best between signal and background. Another future research is to investigate how to find total shower energy and position. One strategy could be to "go back" into detector volume after the event is recognized as "interesting" and to cluster the TC energies around the reconstructed 3D seed. However, this procedure requires that very efficient structures are developed to keep the neighbor information in three dimensions and to keep the neighbor-finder algorithm feasible for a real time execution. Another possibility is to keep the 2D clustering concept but to cluster energy bins in the accumulated maps instead of clustering the raw TC energies. The idea is to make everything a 2D problem and to do the 2D clustering. Finding neighbors in memory in 2D is simpler than in 3D because in 2D one just needs to navigate in (x, y) while in 3D we have to find projective neighbors in depth. Also, hardware implementation is simpler as one could make a fixed but a rather large LUT for 2D algorithms. This way we still keep the 3D longitudinal information, even though reduced in the transversal maps, but we know for the successful ML decision whether the data is more EM-like, PU-like or hadronic-like. Then, we can use a corresponding map based on the detected paticle type to extract the seed bin and reconstruct the shower position, while at the same time the total energy map could be used to reconstruct the shower energy.

Future research should consider the potential effects of the hardware implementation of the trigger architecture and the compromise between the algorithm accuracy and the resource usage. In addition, the possibility of using CNN in ML network could be explored to examine the trade-offs. Since we have shown that the proposed multi-view architecture from different sides of the detector plane (x-y, x-z and y-z) is the most efficient, one can apply a multi-channel approach with sliced view of the detector data in three directions. Hence, the classification algorithm can be further improved making the decision-making process more powerful.

Overall, the successful functionality of EM-like versus PU-like classification obtained with the reduced NNet architecture with only few dense layers is very motivating for further research with the trigger goal in mind. This motivates us to explore in the new direction and to implement the reduced NNet in hardware, and test the required latency and the area usage. Finally, the potential of using the ML techniques in the very early trigger level can be revealed, enabling a successful recognition of the signal pattern in the 3D detector volume.

Chapter 7

Outlook and perspectives

In this chapter, we discuss the link of the conducted studies with the latest TPG architecture that enables the direct 3D clustering. Section 7.1 summarizes the details of the newest baseline architecture, with emphasis on the concepts from the studies presented in this thesis that are included in the current algorithm. It is shown how our work is inserted in the evolution of the algorithms and the architectures, and our work is in the middle of the baseline architecture described in Section 2.3.2 and the current baseline TPG described in Section 7.1.

Next, we have seen in Chapter 6 that it is possible to build an efficient classification model to discriminate between EM-like (signal) and PU-like (background) input data. Section 7.2 describes the potential deployment of the selected ML algorithm in the L1 trigger chain, under the challenge of the strict latency constraints.

7.1 Current baseline architecture for the direct 3D algorithm

In order to exploit the full potential of the future HGCAL to provide a three-dimensional image of the shower, studies were made on how to enable a direct 3D clustering at the L1 trigger instead of the baseline approach (Section 4.3). The work is still ongoing, where the TDR of the Phase 2 L1 from March 2020 [162] foresees the architecture that is illustrated on Figure 7.1. Again, as when we were describing the baseline architecture, we will calculate the number of boards and the number of links when sending the data between TPG stages. There is one difference here compared to the baseline description from Section 4.3. Unlike in the baseline architecture, each board here has two FPGAs. They are from the Xilinx Kintex Ultrascale (KU15P) and Virtex Ultrascale family (VU7P) for the stage 1 and stage 2, respectively.

The HGCAL readout assumes in total 4632 optical links from the on-detector, i.e. 772 links per $60°$ sector in depth. When including 1008 links for the scintillator, this gives the total link number of 10272 for the full HGCAL. Each FPGA from the first stage has a sectorized view in depth, where it can read a part of the $120°$ sector (few layers), because it cannot read a full $120°$ sector in depth. The 12 boards are arranged to cover the full sector,

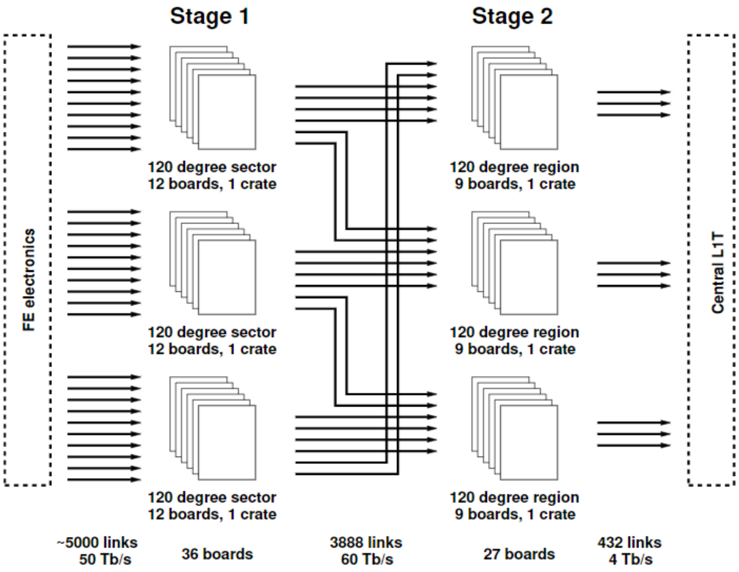


Figure 7.1: Current TPG architecture for the direct 3D clustering [162].

but each FPGA only cover one part, and the work on association of ECONs to stage 1 boards is still ongoing. According to the Figure 7.1, there are three identical sets of FPGAs in stage 1, where 24 FPGAs are in charge of each detector third. The number of FPGAs is calculated based on the FPGA type and their number of input links, where the number of on-detector links from one endcap for the calculation is around 5000. Hence, there are in total $\frac{5000}{72} = 70$ links (because the KU15P FPGA has 72 input links) or $\frac{70}{3} = 24$ FPGAs per each third.

The FPGAs in the second stage have also a sectorized $120°$ view in depth, but they are time multiplexed with a period $T_{mux} = 18$. It means that, on each third, 18 FPGAs receive data from 18 BXs (one per BX). There are $\frac{3888}{3} = 1296$ links between stages in each third data flow (parallel set of FPGAs on the figure). It corresponds to 18 stage-2 FPGAs that have 96 input links, where 72 input links used to receive data from all 24 stage-1 FPGAs on this third, and the remaining 24 links are used to receive the duplicated data. This is the data from the 24 stage-1 FPGAs covering a neighboring endcap region (neighboring third).The choice of data duplication reduces the communication between stage-2 FPGA boards, and the concept is described in a study from Section 4.3.

The other way around, 54 output links out of 72 are used from the each of the 24 stage-1 FPGAs to send the data to the 18 stage-2 FPGAs $(54 * 24 = 1296)$, while the remaining 18 links are used to send the duplicated data to stage-2 FPGAs of neighboring third. Since each out of 18 stage-2 FPGA receives data from all 24 stage-1 FPGAs, that is $\frac{72}{24} = 3$ links foreseen for sending the data from each BX from stage 1 to stage 2. The first stage implements the TC repacking before sending the data to the second stage on 16Gbps links. The second stage performs the actual 3D clustering in 2 steps [162]:

- The seeding algorithm - finds the seeds based on a 2D histogram projection of TCs, where a particle is coming from the center of the detector and following a straight line. The concept is described in a study of Section

5.1. The final choice for the histogram parameters is ($\frac{r}{z}, \phi$) and the number of bins is with 36($\frac{r}{z}$)x216(ϕ). Also, a smoothing filter is applied to the histogram and seeds are defined as local maximums above a threshold.

- The clustering around the identified seeds - the map between the TCs and the seeds is needed here, to attach the TCs to seeds (do a reverse procedure from projection, to see which TCs projected to the seed bin). Next, the seed positions or the positions of 3D clusters are calculated as the barycenter of the TC positions, weighted with their energy. It means that TCs are associated to seeds within a given distance in the ($\frac{x}{z}, \frac{y}{z}$) plane. There are clusters shapes used to discriminate EM, HAD or PU clusters, and the EM encodings are foreseen such to contain information on the longitudinal development of the shower and its transverse size.

To conclude, our studies from the thesis work have introduced some of the concepts used in the current baseline architecture design.

7.2 The hardware implementation potential of the selected ML model

The discriminative power of EM vs. PU-like showers was examined with the use of the ML in Section 6. We have seen that the NNet can be trained for a successful classification between data, and with high accuracy rates ($> 98\%$), which is very important for the L1 trigger decision. ML techniques are largely used in HEP processing, especially CNNs, but their application is always offline. The reason is that these techniques are usually resource-consuming and they need a larger latency available for the decision (a few hundred miliseconds). Since the real-time L1 trigger decision must be made with a maximal latency of 5 microseconds in the TPG algorithm, it would be very challenging to implement a CNN in real-time.

In order to overcome this, we have designed NNets without the convolutional layers, having only a few dense FC layers with a low number of neurons. Hence, the computational cost is minimized, while the most important parameter to handle is the memory demand, caused by the FC layers concept with an increased number of parameters. This is why in the performance evaluation of the models we have considered the model accuracy, but in the trade off with the total number of model parameters or model weights. This enables us to evaluate the hardware complexity when implementing ML solutions. The resource consumption is important for the trigger, because the memory requirements must be low enough to fit in the block random-access memory (BRAM) of the FPGA, and the latency must be satisfactory to fit the trigger demands.

Recently, Di Guglielmo et al. [163] have developed a tool called high level synthesis for machine learning (HLS4ML) that automatically converts the ML models to digital circuits with the FPGA firmware. It is shown that fast and efficient classification of particles is possible with the tool, and the network is optimized as much as possible to be less resource-consuming and to be able to bring an efficient classification decision. The model accuracy is reduced a little during the optimization process, but the compromise is reached such that the model accuracy remains satisfactory.

We have used the former tool to test the possibility of applying the selected ML model from our study in this rather early trigger level. We selected the model $M = 38$ that is the most accurate of all multi-channel ROI approaches (when the three-view model is omitted). Also, the version with the lowest input size is chosen, i.e. 3x3x38. We start from the model accuracy score, which is the "expected accuracy" gained by training the network with a 32-bit floating point precision. To enable the hardware implementation and fulfill the reduced memory requirements and simplified arithmetic operations, we investigate the fixed-point precision when coding the model inputs, outputs, weights and biases, and we test the effect on the model performance.

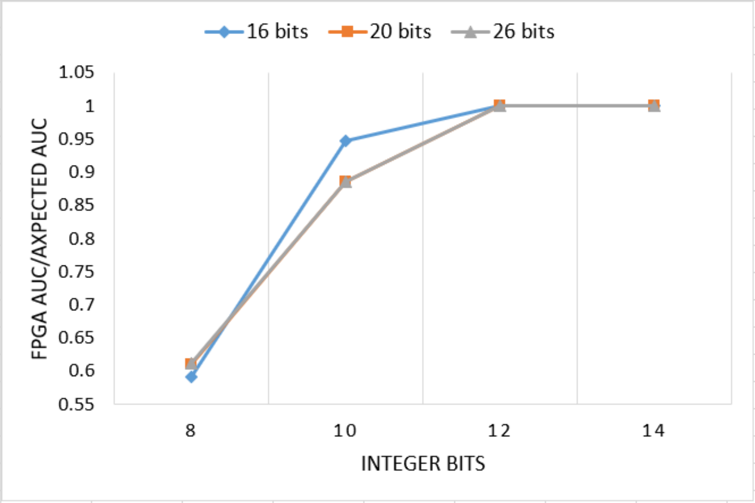


Figure 7.2: Model accuracy degradation compared to the "expected AUC" with the quantized low-integer precision.

This procedure, called quantization, reduces the number of bits used to code the NNet parameters. We define the precision by the parameters (X,Y), where X is the total number of bits and Y is the number of bits representing the signed number above the binary point (i.e. the integer part). The results presented in Figure 7.2 show that the number of integer bits has a significant impact on the model accuracy. It means that we can use a lower number of bits in the code word (lower parameter X) but increase the accuracy with the larger Y component. The results are motivating, and will drive the future work, where one should examine the latency of the synthesized FPGA hardware. Hence, we would see how these ML models can be included in the very early L1 trigger stages, where the reduced HGCAL latency is required (the total latency of 5 microseconds is an upper limit on timing). Afterwards, one could derive a compromise between the networks resource consumption and the numerical precision of the network.

Conclusion

In this thesis, we have described the research work done in the context of the HGCAL project, which is part of the upgrade of the CMS detector. Our work is motivated by the phase-2 upgrade during the third long shutdown of LHC, scheduled for the year 2025, when the experiment will be prepared for the demanding HL-LHC era. CMS detector upgrade will follow the two main directions, as described in the thesis work. First, the ECAL and HCAL calorimeters will be replaced by a new mechanical construction called HGCAL. Next, the upgrade is foreseen for the trigger system in the HL-LHC, where the HGCAL trigger electronics chain will be redesigned. We have shown the series of the conducted studies along the whole trigger path, from the generation of the trigger signals in the detector geometry, the selection of trigger cells in the FE electronics, and the studies devoted to the reconstruction of the trigger signals received at the BE stage.

HGCAL is a sampling calorimeter, whose structure will provide a new paradigm to 3D calorimetry, with the depth component included in the event reconstruction, such that a better separation between showers and the more efficient PU rejection is accomplished. Motivated by the mechanical upgrade of HGCAL, we have presented a set of geometry studies, where the hexagonal geometry is used for the sensor placement on the detector sensing layers. First, we explored the definition of the hexagonal sensor module design and how to efficiently pack the smaller hexagonal sensor cells inside. Our goal was to analyze the possibilities of hexagonal module construction with the maximized number of inner cells and the minimal number of different cells types at the borders. The module geometrical shape is moved from the regular to an irregular hexagon (but symmetric in shape), and we showed that this one can obtain the maximized number of inner packed items, as well as the natural voids at the module vertex obtained for the module fixation. Besides the efficient coverage of the circular detector region of interest, it is shown that the production of such modules is cost-effective, being very close to a regular hexagon.

Geometry of the module and its architecture defines the number of channels needed to send the sensor data. It is not possible to readout the data from all sensors, but a reduction is performed based on the geometry, by forming the trigger cells. They are diamond tetrahex structures, built from four hexagonal sensors clustered (TC4). We have analyzed the possibility of other symmetric structures, such as the TC3 or TC7, and it is shown that TC4 provides a good compromise between the maximized number of TCs packed inside the module, and a reduced number of border TC types. The most demanding requirement during the research of the detector geometry design is to keep all TCs packed inside the module and thus to avoid communication between two or three neighbouring modules sharing TCs. Even though the scientific contribution of the geometry studies is not that we have found the module for the new HGCAL, we contributed by a very detailed studies that were presented inside the CMS collaboration, inspiring other people from the group to contribute in the field. We have developed several module geometry solutions that represent a meaningful step towards the final module solution that is chosen. Our proposed module (called ROD1) had some advantages over the chosen one (called H(D)), such as for example the module cut which allowed for the lower number of partial cells at the module vertices. However, based on the presented research on the geometries, another schema was proposed by Gecse, and this one is finally accepted [67]. The final solution preserved all good characteristics of ROD1. However, in our research, we were constantly oriented towards keeping the TC plane uniform, and even with H(D), we have still explored the TC uniformity. Finally, we have shown that the inter-module communication can be minimized, but Gecse rotated the TCs on every third of the module by 120 degrees, and finally accomplished that the inter-module communication is completely avoided, since now all the TCs are fully packed inside the module borders. A compromise with the final architecture is that more sensor types are generated with the cuts performed at the module vertices.

Considering the scientific contribution of the geometry studies (Section 3), our work was part of the meaningful analysis that was needed before finding the final HGCAL sensor module solution (proposed by others). We have proposed our own module architecture, which satisfies the identified requirements and provides optimization of the module production cost. We have compared our solution towards the accepted one. During the research on the detector geometries, we have detected various patterns and regularities in the explored sensor cell and trigger cell plane when covering the detector layer. Also, many inner-packing hexagonal schemes are derived that enable the efficient forming of trigger cells inside the module. We have developed the mathematical formulations that bring novelty to the field of the optimal packing problem solutions. There are 3 scientific papers published from the research [35, 60, 66], and they directly reflect the contribution of the thesis work.

After the desired TCs are formed in Section 3, we have studied another form of data reduction that is the selection of TCs at the detector FE electronics (Section 4). We have concentrated on the strategies for selecting a fixed number of the highest energy trigger cells, and the problem was solved by the design of an efficient maximum-finder circuit (called BCT) that is synthesized in ASIC. We have shown that our proposed solution provides optimization in latency and area compared to the existing array-based topologies from the literature. The paper is published on this topic in [83], and it reflects the scientific contribution. We have simulated the functionality of other concurrent maximal-finder solutions, and we have verified their possibility of implementation in the trigger. It is shown that all circuits can be synchronized with the 40MHz clock cycle, and that it is possible to select a single maximum from the input dataset within the 2ns time frame. Also, all solutions were synthesized in ASIC and it is shown that the tree-based designs were more efficient in timing, while the array-based ones are more area-efficient. Since timing is critical for the trigger, it is shown that, for example if we perform a 4BX aggregation in the pipelined design, we can select more TCs with the tree-based maximum-finder circuits in 100ns than with BCT.

Also, in the meanwhile, many researches were done in paralel (by other people from the group). Their intention was to solve the same problem of finding the most energetic TCs, but using another strategy of sorting all the TCs before the selection. We have compared our BCT towards these other solutions. Results from the literature [94, 93] have shown that 48 TCs can be sorted in 25ns, while it takes around 6ns to select a single element with BCT. Then, we must go sequentially to select N TCs with BCT, such that roughly N*6ns is needed. It is concluded that it is better to sort and select than to select sequentially, because the most demanding part of the procedure is the sorting itself, and after it is done, we can select whatever number of TCs we want. Also, we have compared our BCT to a small sorting network implemented hardware [97]. It is shown that BCT is better in area, since sorting networks require larger number of logic gates, especially with larger number of comparators used. However, when timing is considered, BCT is better only when a single maximum or two maximums are selected, while in all other cases sorting is faster. We have discussed the advantages of both approaches and we emphasized that in sorting, TC addresses must be extracted together with the energy values, while BCT provides the coding of TC addresses with simple selection bits, so that more bits are left for coding the actual data values. Finally, sorting approach is accepted for the trigger, and the concept of address coding is dynamic. In detector regions with low occupancy, TC values and addresses are sent, while the selection bits are used in regions where more TCs are readout.

Another contribution from the Section 4 besides the BCT design in the FE, is the work on defining the BE architectures that would enable the data reconstruction in the final step of the trigger before the trigger primitive generation. Our goal is to make use of the advantage of the upgraded HGCAL which will provide a 3D image of the particle showers for the first time. Therefore, the special interest will be devoted to direct 3D clustering of TCs, unlike in the baseline trigger strategy (2D followed by 3D). We contribute by deriving the main difficulties for the implementation of direct 3D clustering in the L1 trigger. Also, we propose architectures to solve the identified problems and we examine the critical points of the proposed solutions. We propose architecture that will bypass doing the direct 3D clustering in the whole detector at once, but it will first find regions of interest in the detector and afterwards apply data processing only on this reduced data volume. The proposed BE trigger data reconstruction architecture is divided into two stages: stage 1 FPGAs are used to perform the seeding algorithm and to define ROIs in the detector, and the stage 2 FPGAs receive the few rings of trigger cells around the seeds, and perform the 3D clustering. We have studied the critical parameter that is the number of links needed for sending the ROIs between stages. It is concluded that for the 98% signal efficiency and the 5 rings around the seed, there are are 2, 15 and 100 links needed to receive the data in the stage 2 FPGA for the unconverted photons, all photons and electrons, respectively. We have seen that we can reduce the signal efficiency to obtain the lower number of links needed to transfer the data. The system is not feasible, and the number of links is growing very fast, while the number of the input FPGA links is limited. Hence, it is better to perform seeding and clustering both in a single back-end stage, which is finally the case with the current baseline architecture.

In Section 5, we propose a tracking algorithm used for seeding and the ROI detection. We develop an oblique projection of TCs energies by using the Hough transform for mapping the detector Cartesian coordinates into our definition of (r,c) parameter space that is (eta,phi)-like but in centimeters. We perform the accumulation of the energies in the 2D histogram map by following a straight line coming from the center of the detector and we use a central local maximum filter to extract the seed bins. We have shown the efficiency of the proposed seeding approach, where more signal seeds and less background seeds can be extracted. It is described how the tracking algorithm helps to reduce the data, especially if a shower identification mechanism is added based on the known profile of the EM showers. With this knowledge included in the algorithm, we can reduce the number of tracks and select more efficiently the signal, reducing the required bandwidth if the two-stage back-end architecture is used. We have tuned the algorithm parameters, such as histogram bin size, bin space and the profile used for the EM identification. It is concluded that bins lower than 3x3cm^2 are the best, but we can even further increase the bin size, after we previously solve "the bump effect" that occurs in the background candidates distribution. There is another drawback of the too large bins, which is that the efficiency is dropped at the bin edge compared to the bin center (called "the bin edge effect"). We have seen that both effects can be solved with the proposed solutions, and this is important because the larger the bin, the smaller is the lookup table needed to map the TCs to bins in the FPGAs. Also, we have shown that using the maximum profile for identification (instead of the full one) is as efficient, so that we can obtain the same signal efficiency with less multiplications performed in hardware. The drawback of the proposed (r, c) parameter space is revealed during the studies, as it is noticed that the same event is reconstructed differently at the center of the detector and at the detector border. It is concluded that (eta, phi) space is better to use, but with lower bins in low eta, and larger bins used in high eta region.

To conclude the above contributions, our studies from the thesis work have introduced some of the concepts used in the current baseline TPG design. The stage 1 will prepare the data for the stage 2 FPGAs, as it will provide the projective regions so that 3D algorithm can be performed in seed 2 (both seeding and clustering). In Section 6, we have studied the application of machine learning for the discrimination between signal (EM-like) and background (PU-like) data. We have applied the designed tracking algorithm for the shower image generation from the detector ROI and we develop a few types of images: a multi-channel shower image with the full HGCAL layers (M=38), a reduced image with only ECAL layers used (M=14), a three-channel image with the summed energies from consecutive detector layers (M=3), a single-channel image (M=1), and the three sets of images providing separate views of the shower data in three independent directions. We generated a database of these images to be used by machine learning algorithms, and the contribution is the image generation procedure with the oblique trigger cells data projection. Since the goal is to implement the classification in the trigger, we have decided to omit the convolutional layers and use only the fully connected NNet with few dense layers. We have filled the revealed gaps in the state-of-the-art, and we examine the specific parameters for the image-based shower data representation such as the bin size or image resolution, image size and image depth. It is shown that each parameters variation provides the classification accuracy above 96%, so we can use any of the models which provides the lowest complexity.

The compromise is discussed between the accuracy of the five NNet models (classification performance) and the model complexity (expressed as the total number of parameters). The model with the three projected views of the shower data provides the best accuracy, but it is rather complex, while the complexity is the lowest for M=1 but in the compromise with the little bit lower accuracy. NNet model M=3 provides the best trade-off between the quality of the decision-making process and the complexity of the required hardware processing solution.

The ML study is motivating, and it has a great potential to be implemented in the trigger. First, classification accuracy result is very high, and second, we have shown that the ML model can be converted to the FPGA hardware by using HLS4ML tool. The quantization is performed on the input data in the network, such that lower number of bits can be used in the input code word, when the larger number of integer bits is used to remain the high accuracy score. This derives the future work, to synthesize the converted hardware model and to test whether the ML techniques can be used at the early trigger level. The timing conditions are very demanding, with less than 4 microseconds available to decide on the input data. In order to be fit in the current baseline architecture, it would be potentially performed in the stage 2 FPGAs. Once the data is received in the projective manner, the tracking algorithm would be performed to generate the shower image. Next, we do not need to do the 3D clustering, but we can decide right away on whether the data is signal or background. Also, another class should be included to have a multi-class EM-like, PU-like and HAD-like data. Once detected, the data can be sent directly as a trigger primitive towards the central HGCAL trigger for further processing and the higher-level triggering in the successive trigger stages.

Bibliography

[1] Ruder Boskovic Institute. Long-sought higgs boson decay finally measured. `https://www.irb.hr/eng/News/Long-Sought-Higgs-Boson-Decay-Finally-Measured`.

[2] Lyndon Evans and Philip Bryant. Lhc machine. *Journal of instrumentation*, 3(08):S08001, 2008.

[3] CERN Accellerating Science. The high luminosity lhc (hl-lhc) project. `https://hilumilhc.web.cern.ch/content/hl-lhc-project`.

[4] CERN Accellerating Science. Facts and figures about the lhc. `https://home.cern/resources/faqs/facts-and-figures-about-lhc`.

[5] Markus Zinser. The large hadron collider. In *Search for New Heavy Charged Bosons and Measurement of High-Mass Drell-Yan Production in Proton—Proton Collisions*, pages 47–51. Springer, 2018.

[6] CERN Accellerating Science. Cern's accelerator complex. `https://home.cern/science/accelerators/accelerator-complex`.

[7] CERN Accellerating Science. Pulling together: Superconducting electromagnets. `https://home.cern/science/engineering/pulling-together-superconducting-electromagnets`.

[8] Werner Herr and B Muratori. Concept of luminosity. 2006.

[9] Cid Vidal Xabier and Cid Ramon. Luminosity: Taking a closer look at lhc. `https://www.lhc-closer.es/taking_a_closer_look_at_lhc/0.luminosity`.

[10] Edda Gschwendtner and Massimo Placidi. Baseline and requirements for a luminosity monitoring at the lhc. 10 2020.

[11] S. Chatrchyan et al. The CMS Experiment at the CERN LHC. *JINST*, 3:S08004, 2008.

[12] G. Aad et al. The ATLAS Experiment at the CERN Large Hadron Collider. *JINST*, 3:S08003, 2008.

[13] K. Aamodt et al. The ALICE experiment at the CERN LHC. *JINST*, 3:S08002, 2008.

[14] A Augusto Alves Jr, LM Andrade Filho, AF Barbosa, I Bediaga, G Cernicchiaro, G Guerrer, HP Lima Jr, AA Machado, J Magnin, F Marujo, et al. The lhcb detector at the lhc. *Journal of instrumentation*, 3(08):S08005, 2008.

[15] Saranya Samik Ghosh. Highlights from the compact muon solenoid (cms) experiment. `https://www.groundai.com/project/highlights-from-the-compact-muon-solenoid-cms-experiment/1`.

[16] ATLAS. Atlas events. `http://opendata.atlas.cern/books/current/openatlasdatatools/_book/atlas_events.html`.

[17] Robert Francis Holub Hunter. *Development and Evaluation of Novel, Large Area, Radiation Hard Silicon Microstrip Sensors for the ATLAS ITk Experiment at the HL-LHC*. PhD thesis, Carleton University, 2017.

[18] CMS Collaboration. Cutaway diagrams of cms detector. `https://cds.cern.ch/record/2665537`, May 2019.

[19] Martin Lipinski. The Phase-1 Upgrade of the CMS Pixel Detector. Technical Report CMS-CR-2017-135. 06, CERN, Geneva, May 2017.

[20] Venkatesh S Kaushik. Electromagnetic showers and shower detectors. *Internal Documents, University of Texas at Arlington*, 2002.

[21] Andrea Benaglia. The cms ecal performance with examples. *Journal of Instrumentation*, 9(02):C02008, 2014.

[22] Simon Pekar. Reconstruction of electromagnetic showers in a pile-up environment, July 2017. Laboratoire Leprince-Ringuet, École polytechnique (private communication).

[23] Thomas Owen James. *A Hardware Track-Trigger for CMS: At the High Luminosity LHC*. Springer Nature, 2019.

[24] Siona Ruth Davis. Interactive Slice of the CMS detector. Aug 2016.

[25] D Barney. Calorimetry in particle physics, and the cms high-granularity calorimeter. *Journal of Instrumentation*, 15(07):C07018, 2020.

[26] Energy calibration and resolution of the cms electromagnetic calorimeter in pp collisions at sqrt(s) = 7 tev. *Journal of Instrumentation*, 8(09):P09009–P09009, Sep 2013.

[27] The Phase-2 Upgrade of the CMS Endcap Calorimeter. Technical Report CERN-LHCC-2017-023. CMS-TDR-019, CERN, Geneva, Nov 2017.

[28] Peter Paulitsch. The silicon sensors for the high granularity calorimeter of cms, 02 2020.

[29] Artur Lobanov. Electronics and triggering challenges for the cms high granularity calorimeter. *Journal of Instrumentation*, 13(02):C02056, 2018.

[30] Thorben Quast. Cms high-granularity calorimeter upgrade (hgcal). `https://twiki.cern.ch/twiki/pub/CLIC/TerascaleDW19/CMS_HGCal_TerascaleWS2019_final.pdf`, March 2019. (private communication).

[31] Artur Lobanov. High granularity hybrid timing and energy calorimetry. `https://indico.in2p3.fr/event/16567/contributions/56857/subcontributions/4149/attachments/45311/56370/HIGHTEC_P2IO_ALobanov.pdf`, Nov 2017. CERN internal document.

[32] David Dagenhart. Cmssw application framework. `https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookCMSSWFramework`. CERN internal document.

[33] Jean-Baptiste Sauvan. Hgcal trigger primitives: simulation and algorithms. `https://twiki.cern.ch/twiki/bin/viewauth/CMS/HGCALTriggerPrimitivesSimulation`, September 2020. CERN internal document.

[34] Andrea Giammanco. The fast simulation of the cms experiment. In *Journal of Physics: Conference Series*, volume 513, page 022012. IOP Publishing, 2014.

[35] Marina Prvan, Ozegovic Julije, and Burazin Misura Arijana. A review of embedding hexagonal cells in the circular and hexagonal region of interest. *International Journal of Advanced Computer Science and Applications*, 10(7):339–348, 2019.

[36] Taryn J Davis and Tuhin Sinha. Polygon die packaging, March 6 2018. US Patent 9,911,716.

[37] Sungun Kim, Hyunsoo Cheon, Sangbo Seo, Seungmi Song, and Seonyeong Park. A hexagon tessellation approach for the transmission energy efficiency in underwater wireless sensor networks. *Journal of Information Processing Systems*, 6(1):53–66, 2010.

[38] Colin PD Birch, Sander P Oom, and Jonathan A Beecham. Rectangular and hexagonal grids used for observation, experiment and simulation in ecology. *Ecological modelling*, 206(3-4):347–359, 2007.

[39] Kevin Sahr, Mark Dumas, and Neal Choudhuri. The planetrisk discrete global grid system. *Online], http://www.discreteglobalgrids. org/wpcontent/uploads/sites/33/2016/10/PlanetRiskDGGS. pdf*, 2015.

[40] Ivan Stojmenovic. Honeycomb networks: Topological properties and communication algorithms. *IEEE Transactions on parallel and distributed systems*, 8(10):1036–1042, 1997.

[41] Fabian Garcıa, Julio Solano, Ivan Stojmenovic, and Milos Stojmenovic. Higher dimensional hexagonal networks. *Journal of Parallel and Distributed Computing*, 63(11):1164–1172, 2003.

[42] M Chris Monica and S Santhakumar. Partition dimension of honeycomb derived networks. *International Journal of Pure and Applied Mathematics*, 108(4):809–818, 2016.

[43] Jin Ben, YaLu Li, ChengHu Zhou, Rui Wang, and LingYu Du. Algebraic encoding scheme for aperture 3 hexagonal discrete global grid system. *Science China Earth Sciences*, 61(2):215–227, 2018.

[44] Vishal Kumar, Vinay Kumar, DN Sandeep, Sadanand Yadav, Rabindra K Barik, Rajeev Tripathi, and Sudarshan Tiwari. Multi-hop communication based optimal clustering in hexagon and voronoi cell structured wsns. *AEU-International Journal of Electronics and Communications*, 93:305–316, 2018.

[45] Pedro Hokama, Flávio K Miyazawa, and Rafael CS Schouery. A bounded space algorithm for online circle packing. *Information Processing Letters*, 116(5):337–342, 2016.

[46] Dajin Wang, Liwei Lin, and Li Xu. A study of subdividing hexagon-clustered wsn for power saving: Analysis and simulation. *Ad Hoc Networks*, 9(7):1302–1311, 2011.

[47] Kevin Sahr, Denis White, and A Jon Kimerling. Geodesic discrete global grid systems. *Cartography and Geographic Information Science*, 30(2):121–134, 2003.

[48] Ali Mahdavi-Amiri and Faramarz Samavati. Atlas of connectivity maps. *Computers & Graphics*, 39:1–11, 2014.

[49] M-S Chen, Kang G Shin, and Dilip D. Kandlur. Addressing, routing, and broadcasting in hexagonal mesh multiprocessors. *IEEE Transactions on Computers*, 39(1):10–18, 1990.

[50] Paul Manuel, Rajan Bharati, Indra Rajasingh, et al. On minimum metric dimension of honeycomb networks. *Journal of Discrete algorithms*, 6(1):20–27, 2008.

[51] Přemysl Holub, Mirka Miller, Hebert Pérez-Rosés, and Joe Ryan. Degree diameter problem on honeycomb networks. *Discrete Applied Mathematics*, 179:139–151, 2014.

[52] Premysl Holub and Joe Ryan. Degree diameter problem on triangular networks. *Australas. J Comb.*, 63:333–345, 2015.

[53] Paul Manuel, Indra Rajasingh, Albert William, and Antony Kishore. Computational aspects of silicate networks. In *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)*, page 1. The Steering Committee of The World Congress in Computer Science, Computer . . . , 2011.

[54] Herman Haverkort. Recursive tilings and space-filling curves with little fragmentation. *arXiv preprint arXiv:1002.1843*, 2010.

[55] Xiaochong Tong, Jin Ben, and Ying Wang. A new effective hexagonal discrete global grid system: hexagonal quad balanced structure. In *2010 18th International Conference on Geoinformatics*, pages 1–6. IEEE, 2010.

[56] Vojtěch Uher, Petr Gajdoš, Václav Snášel, Yu-Chi Lai, and Michal Radeckỳ. Hierarchical hexagonal clustering and indexing. *Symmetry*, 11(6):731, 2019.

[57] CMS Collaboration et al. Technical proposal for the phase-ii upgrade of the compact muon solenoid, cms technical proposal cernlhcc-2015-010. Technical report, CMS-TDR-15-02, CERN, 2015. https://cds. cern. ch/record/2020886, 2015.

[58] Chen-Fu Chien, Chia-Yu Hsu, and Kuo-Hao Chang. Overall wafer effectiveness: A novel industry standard for semiconductor ecosystem as a whole. *Computers & Industrial Engineering*, 65(1):117–127, 2013.

[59] Péter Gábor Szabó, Mihaly Csaba Markót, Tibor Csendes, Eckard Specht, Leocadio G Casado, and Inmaculada García. *New approaches to circle packing in a square: with program codes*, volume 6. Springer Science & Business Media, 2007.

[60] Marina Prvan, Julije Ožegović, and Arijana Burazin Mišura. On calculating the packing efficiency for embedding hexagonal and dodecagonal sensors in a circular container. *Mat. Problems in Engineering*, 2019.

[61] Andreas Alexander Maier. Sensors for the cms high granularity calorimeter. *Journal of Instrumentation*, 12(06):C06030, 2017.

[62] Roger Rusack. *Challenges and synergies of silicon sensors for a high granularity calorimetry*, 2014.

[63] Jean-Baptiste Sauvan. Concepts and design of the cms high granularity calorimeter level-1 trigger. In *J. Phys.: Conf. Ser.*, volume 928, page 012026, 2016.

[64] Rizuwana Parween, Yuyao Shi, Karthikeyan Parasuraman, Ayyalusami Vengadesh, Vinu Sivanantham, Sriharsha Ghanta, and Rajesh Elara Mohan. Modeling and analysis of hhoneycomb—a polyhex inspired reconfigurable tiling robot. *Energies*, 12(13):2517, 2019.

[65] Chris Seez. Hgcal simulation and performance. `https://indico.cern.ch/event/544718/contributions/2210658/attachments/1295182/1930748/CRTalk9-v1.pdf#search=chris%20seez%20june%202016`, June 2016. CERN internal document.

[66] Marina Prvan, Arijana Burazin Mišura, Zoltan Gecse, and Julije Ožegović. A vertex-aligned model for packing 4-hexagonal clusters in a regular hexagonal container. *Symmetry*, 12(5):700, 2020.

[67] Zoltan Gecse. Optimal sensor cell geometry proposal, November 2016. (private communication).

[68] Marina Prvan and Julije Ozegovic. Trigger cell regularity. `https://indico.cern.ch/event/580951/contributions/2356336/attachments/1364959/2067284/Trigger_Cell_Regularity_v9.pdf#search=Trigger%20Cell%20Regularity`, November 2016. CERN internal document.

[69] Dimitri Van De Ville, Thierry Blu, Michael Unser, Wilfried Philips, Ignace Lemahieu, and Rik Van de Walle. Hex-splines: A novel spline family for hexagonal lattices. *IEEE Transactions on Image Processing*, 13(6):758–772, 2004.

[70] Julije Ozegovic and Puljak Ivica. Sensor and trigger cells geometry studies 192 64 architecture properties. `https://indico.cern.ch/event/518757/contributions/2033319/`, April 2016. Sensors, Modules and Cassettes meeting (cern internal document).

[71] V Di Lecce and E Di Sciascio. Evaluation of a bit-serial asic chip for sar processing. *Microprocessing and microprogramming*, 33(2):71–78, 1991.

[72] Mei Yang, SQ Zheng, B Bhagyavati, and S Kurkovskyt. Programmable weighted arbiters for constructing switch schedulers. In *2004 Workshop on High Performance Switching and Routing, 2004. HPSR.*, pages 203–206. IEEE, 2004.

[73] Giorgos Dimitrakopoulos, Emmanouil Kalligeros, and Kostas Galanopoulos. Merged switch allocation and traversal in network-on-chip switches. *IEEE Transactions on Computers*, 62(10):2001–2012, 2012.

[74] Yu-Wen Huang, Shao-Yi Chien, Bing-Yu Hsieh, and Liang-Gee Chen. Global elimination algorithm and architecture design for fast block matching motion estimation. *IEEE Transactions on circuits and systems for Video technology*, 14(6):898–907, 2004.

[75] Bilgiday Yuce, H Fatih Ugurdag, Sezer Gören, and Günhan Dündar. Fast and efficient circuit topologies forfinding the maximum of n k-bit numbers. *IEEE Transactions on Computers*, 63(8):1868–1881, 2014.

[76] Bilgiday Yuce, H Fatih Ugurdag, Sezer Gören, and Gunhan Dundar. A fast circuit topology for finding the maximum of n k-bit numbers. In *IEEE 21st Symposium on Computer Arithmetic*, pages 59–66. IEEE, 2013.

[77] Swaminathan Kathirvel, Rajkumar Jangre, and Seokbum Ko. Design of a novel energy efficient topology for maximum magnitude generator. *IET Computers & Digital Techniques*, 10(3):93–101, 2016.

[78] Xiao-Ping Huang, Xiao-Ya Fan, Sheng-Bing Zhang, and Fan Zhang. An optimized tag sorting circuit in wfq scheduler based on leading zero counting. In *2010 10th IEEE International Conference on Solid-State and Integrated Circuit Technology*, pages 533–535. IEEE, 2010.

[79] Chin-Long Wey, Ming-Der Shieh, and Shin-Yo Lin. Algorithms of finding the first two minimum values and their hardware implementation. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 55(11):3430–3437, 2008.

[80] Luca G Amarù, Maurizio Martina, and Guido Masera. High speed architectures for finding the first two maximum/minimum values. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 20(12):2342–2346, 2011.

[81] Guoping Xiao, Maurizio Martina, Guido Masera, and Gianluca Piccinini. A parallel radix-sort-based vlsi architecture for finding the first w maximum/minimum values. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 61(11):890–894, 2014.

[82] Andrea Dario Giancarlo Biroli and Juan Chi Wang. A fast architecture for finding maximum (or minimum) values in a set. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7565–7569. IEEE, 2014.

[83] Marina Prvan, Julije Ožegovic, Ivan Soco, and Duje Coko. Best-choice topology: An optimized array-based maximum finder. *International Journal of Advanced Computer Science and Applications*, 10(12), 2019.

[84] I. Sočo. Comparison of circuit topologies for choosing the best input data. Master's thesis, 2018.

[85] Xilinx. Xilinx ise design suite. `https://www.xilinx.com/products/design-tools/ise-design-suite.html`.

[86] Amin Farmahini-Farahani, Anthony Gregerson, Michael Schulte, and Katherine Compton. Modular high-throughput and low-latency sorting units for fpgas in the large hadron collider. In *2011 IEEE 9th Symposium on Application Specific Processors (SASP)*, pages 38–45. IEEE, 2011.

[87] Tu Nguyen Van, Vu Tang Thien, Son Nguyen Kim, Nam Pham Ngoc, and Thanh Nguyen Huu. A high throughput pipelined hardware architecture for tag sorting in packet fair queuing schedulers. In *International Conference on Communications, Management and Telecommunications (ComManTel)*, pages 41–45. IEEE, 2015.

[88] Julije Ozegovic and Puljak Ivica. Status of the work on firmware. `https://indico.cern.ch/event/462420/`, October 2015. Trigger Primitives and Back-End Electronics meeting (CERN internal document).

[89] Cadence Design Systems Inc. Cadence genus synthesis solution. `https://www.cadence.com/en_US/home/tools/digital-design-and-signoff/synthesis/genus-synthesis-solution.html`, 2020.

[90] Raj Kumar Kante and V Thrimurthulu. Efficient sorting mechanism for finding first w maxima/minima values. *i-Manager's Journal on Embedded Systems*, 3(3):39, 2014.

[91] Changhui Hu, Mengjun Ye, Yijun Du, and Xiaobo Lu. Vector projection for face recognition. *Computers & Electrical Engineering*, 40(8):51–65, 2014.

[92] Rene Mueller, Jens Teubner, and Gustavo Alonso. Sorting networks on fpgas. *The VLDB Journal*, 21(1):1–23, 2012.

[93] L. Pacheco Rodriguez, T. Romanteau, F. Thiant, and J.B. Sauvan. Sorting architectures for the concentrator trigger cell selection. `https://indico.cern.ch/event/742880/`, July 2018. HGCAL Backend TDAQ meeting (CERN internal document).

[94] Duje Coko, L. Pacheco Rodriguez, T. Romanteau, F. Thiant, J.B. Sauvan, A. Kristic, J. Music, J. Ozegovic, and I. Puljak. Sorting network ip synthesis, May 2020. Fermilab LLR and Split meeting (private communication).

[95] Kenneth E Batcher. Sorting networks and their applications. In *Proceedings of the April 30–May 2, 1968, spring joint computer conference*, pages 307–314, 1968.

[96] Vinod K Valsalam and Risto Miikkulainen. Using symmetry and evolutionary search to minimize sorting networks. *Journal of Machine Learning Research*, 14(Feb):303–331, 2013.

[97] Marina Prvan, Vinka Mimica, Ivan Nizic, and Julije Ozegovic. 2d data processing with a sorting network and a rank order filter, September 2018. Projecting digital systems design reports (private communication).

[98] Jim Hirschauer. Econ asics. `https://indico.cern.ch/event/862245/contributions/3632336/attachments/1943107/3222874/hirschauer_ECON_ASICs_HGCALweek_12nov2019.pdf`, November 2019. HGCAL Week (CERN internal document).

[99] Luca Mastrolorenzo. Hgc - status of 2d clustering in cmssw. `https://indico.cern.ch/event/624663/contributions/2524699/attachments/1431954/2203199/HGC_TPG_BE_22_03_17.pdf`, March 2017. CERN internal document.

[100] Rudolf K Bock, H Grote, and D Notz. *Data analysis techniques for high-energy physics*, volume 11. Cambridge University Press, 2000.

[101] A Abba, G Punzi, F Spinella, P Marino, D Tonelli, S Stracka, F Lionetto, D Ninci, M Petruzzo, A Cusimano, et al. A specialized track processor for the lhcb upgrade. Technical report, 2014.

[102] A Abba, F Bedeschi, M Citterio, F Caponio, A Cusimano, A Geraci, P Marino, MJ Morello, N Neri, Giovanni Punzi, et al. Simulation and performance of an artificial retina for 40 mhz track reconstruction. *Journal of Instrumentation*, 10(03):C03008, 2015.

[103] ANDREA Abba, F Bedeschi, M Citterio, FRANCESCO Caponio, ALBERTO Cusimano, ANGELO Geraci, P Marino, MJ Morello, N Neri, Giovanni Punzi, et al. The artificial retina processor for track reconstruction at the lhc crossing rate. *Journal of Instrumentation*, 10(03):C03018, 2015.

[104] Zi-Xuan Song, Wen-Di Deng, Gilles De Lentdecker, Guang-Ming Huang, Hua Pei, Yi-Fan Yang, Dong Wang, and Frédéric Robert. Study of the retina algorithm on fpga for fast tracking. *Nuclear Science and Techniques*, 30(8):127, 2019.

[105] N Neri, A Cardini, Roberto Calabrese, Massimiliano Fiorini, Eleonora Luppi, U Marconi, and M Petruzzo. 4d fast tracking for experiments at high luminosity lhc. *Journal of Instrumentation*, 11(11):C11040, 2016.

[106] Maxim Borisyak, Mikhail Belous, Denis Derkach, and Andrey Ustyuzhanin. arxiv: Numerical optimization for artificial retina algorithm. In *J. Phys.: Conf. Ser.*, volume 898, page 032046, 2017.

[107] R Aggleton, LE Ardila-Perez, FA Ball, MN Balzer, G Boudoul, J Brooke, M Caselle, L Calligaris, D Cieri, E Clement, et al. An fpga based track finder for the l1 trigger of the cms experiment at the high luminosity lhc. *Journal of Instrumentation*, 12(12):P12019, 2017.

[108] Zhengcheng Tao. Level-1 track finding with an all-fpga system at cms for the hl-lhc. *arXiv preprint arXiv:1901.03745*, 2019.

[109] E Bartz, G Boudoul, R Bucci, J Chaves, E Clement, D Cranshaw, S Dutta, Y Gershtein, R Glein, K Hahn, et al. Fpga-based tracking for the cms level-1 trigger using the tracklet algorithm. *arXiv preprint arXiv:1910.09970*, 2019.

[110] Fabrizio Palla, M Pesaresi, and A Ryd. Track finding in cms for the level-1 trigger at the hl-lhc. *Journal of Instrumentation*, 11(03):C03011, 2016.

[111] Richard O. Duda and Peter E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, January 1972.

[112] Nicola Pozzobon, Fabio Montecassiano, and Pierluigi Zotto. A novel approach to hough transform for implementation in fast triggers. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 834:81–97, 2016.

[113] scikit learn. Scikit-learn: Ensemble methods. `https://scikit-learn.org/stable/modules/ensemble.html`.

[114] Simon Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, USA, 2nd edition, 1998.

[115] Arash Ardakani, Carlo Condo, and Warren J Gross. Sparsely-connected neural networks: towards efficient vlsi implementation of deep neural networks. *arXiv preprint arXiv:1611.01427*, 2016.

[116] Mário P Véstias. A survey of convolutional neural networks on edge with reconfigurable computing. *Algorithms*, 12(8):154, 2019.

[117] Arash Ardakani, Carlo Condo, Mehdi Ahmadi, and Warren J Gross. An architecture to accelerate convolution in deep neural networks. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(4):1349–1362, 2017.

[118] Dan Guest, Kyle Cranmer, and Daniel Whiteson. Deep Learning and its Application to LHC Physics. *Annual Review of Nuclear and Particle Science*, 68(1):161–181, October 2018. arXiv: 1806.11484.

[119] Daniel Guest, Julian Collado, Pierre Baldi, Shih-Chieh Hsu, Gregor Urban, and Daniel Whiteson. Jet Flavor Classification in High-Energy Physics with Deep Neural Networks. *Physical Review D*, 94, July 2016.

[120] Jannicke Pearkes, Wojciech Fedorko, Alison Lister, and Colin Gay. Jet Constituents for Deep Neural Network Based Top Quark Tagging. *arXiv:1704.02124 [hep-ex, physics:hep-ph, stat]*, August 2017. arXiv: 1704.02124.

[121] Benjamin Nachman. Deep Learning usage by Large Experiments. *Journal of Physics: Conference Series*, 1085:022002, September 2018.

[122] Taoli Cheng. Interpretability Study on Deep Learning for Jet Physics at the Large Hadron Collider. *arXiv:1911.01872 [hep-ph, stat]*, November 2019. arXiv: 1911.01872.

[123] Murat Abdughani, Jie Ren, Lei Wu, Jin Min Yang, and Jun Zhao. Supervised deep learning in high energy phenomenology: a mini review. *Communications in Theoretical Physics*, 71(8):955, August 2019. arXiv: 1905.06047.

[124] Hui Luo, Ming-xing Luo, Kai Wang, Tao Xu, and Guohuai Zhu. Quark jet versus gluon jet: fully-connected neural networks with high-level features. *arXiv:1712.03634 [hep-ex, physics:hep-ph]*, May 2019. arXiv: 1712.03634.

[125] Biplob Bhattacherjee, Swagata Mukherjee, and Rhitaja Sengupta. Study of energy deposition patterns in hadron calorimeter for prompt and displaced jets using convolutional neural network. *Journal of High Energy Physics*, 2019(11):156, November 2019. arXiv: 1904.04811.

[126] Luke de Oliveira, Michael Kagan, Lester Mackey, Benjamin Nachman, and Ariel Schwartzman. Jet-Images – Deep Learning Edition. *Journal of High Energy Physics*, 2016(7):69, July 2016. arXiv: 1511.05190.

[127] Pierre Baldi, Kevin Bauer, Clara Eng, Peter Sadowski, and Daniel Whiteson. Jet Substructure Classification in High-Energy Physics with Deep Neural Networks. *Physical Review D*, 93(9):094034, May 2016. arXiv: 1603.09349.

[128] Wahid Bhimji, Steven Andrew Farrell, Thorsten Kurth, Michela Paganini, Prabhat, and Evan Racah. Deep Neural Networks for Physics Analysis on low-level whole-detector data at the LHC. *arXiv:1711.03573 [hep-ex, physics:physics]*, November 2017. arXiv: 1711.03573.

[129] Celia Fernández Madrazo, Ignacio Heredia Cacha, Lara Lloret Iglesias, and Jesús Marco de Lucas. Application of a Convolutional Neural Network for image classification to the analysis of collisions in High Energy Physics. *arXiv:1708.07034 [hep-ex]*, August 2017. arXiv: 1708.07034.

[130] Gregor Kasieczka and David Shih. *DisCo Fever: Robust Networks Through Distance Correlation*. 2020.

[131] John Alison, Sitong An, Michael Andrews, Patrick Bryant, Bjorn Burkle, Sergei Gleyzer, Ulrich Heintz, Meenakshi Narain, Manfred Paulini, Barnabas Poczos, and Emanuele Usai. *End-to-end particle and event identification at the Large Hadron Collider with CMS Open Data*. October 2019.

[132] Thorsten Kurth, Jian Zhang, Nadathur Satish, Ioannis Mitliagkas, Evan Racah, Mostofa Ali Patwary, Tareq Malas, Narayanan Sundaram, Wahid Bhimji, Mikhail Smorkalov, Jack Deslippe, Mikhail Shiryaev, Srinivas Sridharan, Prabhat, and Pradeep Dubey. Deep Learning at 15PF: Supervised and Semi-Supervised Classification for Scientific Data. *arXiv:1708.05256 [cs]*, August 2017. arXiv: 1708.05256.

[133] F. Psihas, E. Niner, M. Groh, R. Murphy, A. Aurisano, A. Himmel, K. Lang, M. D. Messier, A. Radovic, and A. Sousa. Context-enriched identification of particles with a convolutional network for neutrino events. *Physical Review D*, 100(7):073005, October 2019. Publisher: American Physical Society.

[134] Ruben Vera-Rodriguez, Julian Fierrez, and Aythami Morales, editors. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 23rd Iberoamerican Congress, CIARP 2018, Madrid, Spain, November 19-22, 2018, Proceedings*. Image Processing, Computer Vision, Pattern Recognition, and Graphics. Springer International Publishing, 2019.

[135] Dawit Belayneh, Federico Carminati, Amir Farbin, Benjamin Hooberman, Gulrukh Khattak, Miaoyuan Liu, Junze Liu, Dominick Olivito, Vitória Pacela, Maurizio Pierini, Alexander Schwing, Maria Spiropulu, Sofia Vallecorsa, Jean-Roch Vlimant, Wei Wei, and Matt Zhang. *Calorimetry with Deep Learning: Particle Simulation and Reconstruction for Collider Physics*. December 2019.

[136] Jason Sang Hun Lee, Inkyu Park, Ian James Watson, and Seungjin Yang. Quark-Gluon Jet Discrimination Using Convolutional Neural Networks. *Journal of the Korean Physical Society*, 74(3):219–223, February 2019.

[137] J. Renner et al. Background rejection in NEXT using deep neural networks. *Journal of Instrumentation*, 12(01):T01004–T01004, January 2017. Publisher: IOP Publishing.

[138] Michael Andrews, Manfred Paulini, Sergei Gleyzer, and Barnabas Poczos. End-to-End Physics Event Classification with CMS Open Data: Applying Image-Based Deep Learning to Detector Data for the Direct Classification of Collision Events at the LHC. *arXiv:1807.11916 [hep-ex, physics:physics]*, July 2019. arXiv: 1807.11916.

[139] Thong Q. Nguyen, Daniel Weitekamp III, Dustin Anderson, Roberto Castello, Olmo Cerri, Maurizio Pierini, Maria Spiropulu, and Jean-Roch Vlimant. Topology classification with deep learning to improve real-time event selection at the LHC. *Computing and Software for Big Science*, 3(1):12, December 2019. arXiv: 1807.00083.

[140] Adrian Alan Pol, Gianluca Cerminara, Cecile Germain, Maurizio Pierini, and Agrima Seth. Detector monitoring with artificial neural networks at the CMS experiment at the CERN Large Hadron Collider. *arXiv:1808.00911 [physics, stat]*, July 2018. arXiv: 1808.00911.

[141] Josh Cogan, Michael Kagan, Emanuel Strauss, and Ariel Schwarztman. Jet-images: computer vision inspired techniques for jet tagging. *Journal of High Energy Physics*, 2015(2):118, February 2015.

[142] Leandro G. Almeida, Mihailo Backović, Mathieu Cliche, Seung J. Lee, and Maxim Perelstein. Playing tag with ANN: boosted top identification with pattern recognition. *Journal of High Energy Physics*, 2015(7):86, July 2015.

[143] James Barnard, Edmund Noel Dawe, Matthew J. Dolan, and Nina Rajcic. Parton Shower Uncertainties in Jet Substructure Analyses with Deep Neural Networks. *Physical Review D*, 95(1):014018, January 2017. arXiv: 1609.00607.

[144] Patrick T. Komiske, Eric M. Metodiev, and Matthew D. Schwartz. Deep learning in color: towards automated quark/gluon jet discrimination. *Journal of High Energy Physics*, 2017(1):110, January 2017.

[145] A. Aurisano, A. Radovic, D. Rocco, A. Himmel, M. D. Messier, E. Niner, G. Pawloski, F. Psihas, A. Sousa, and P. Vahle. A Convolutional Neural Network Neutrino Event Classifier. *Journal of Instrumentation*, 11(09):P09001–P09001, September 2016. arXiv: 1604.01444.

[146] Suyong Choi, Seung J. Lee, and Maxim Perelstein. Infrared Safety of a Neural-Net Top Tagging Algorithm. *Journal of High Energy Physics*, 2019(2):132, February 2019. arXiv: 1806.01263.

[147] Gregor Kasieczka, Tilman Plehn, Michael Russell, and Torben Schell. Deep-learning Top Taggers or The End of QCD? *Journal of High Energy Physics*, 2017(5):6, May 2017. arXiv: 1701.08784.

[148] Sebastian Macaluso and David Shih. Pulling out all the tops with computer vision and deep learning. *Journal of High Energy Physics*, 2018(10):121, October 2018.

[149] Patrick T. Komiske, Eric M. Metodiev, and Jesse Thaler. Energy flow networks: deep sets for particle jets. *Journal of High Energy Physics*, 2019(1):121, January 2019.

[150] Miles Winter, James Bourbeau, Silvia Bravo, Felipe Campos, Matthew Meehan, Jeffrey Peacock, Tyler Ruggles, Cassidy Schneider, Ariel Levi Simons, and Justin Vandenbroucke. Particle identification in camera image sensors using computer vision. *Astroparticle Physics*, 104:42–53, January 2019.

[151] Luke de Oliveira, Benjamin Nachman, and Michela Paganini. Electromagnetic Showers Beyond Shower Shapes. *arXiv:1806.05667 [hep-ex]*, June 2018. arXiv: 1806.05667.

[152] The ATLAS collaboration. Quark versus Gluon Jet Tagging Using Jet Images with the ATLAS Detector, July 2017. ATL-COM-PHYS-2017-1000 Library Catalog: cds.cern.ch Number: ATL-PHYS-PUB-2017-017.

[153] Juliette Alimena, Yutaro Iiyama, and Jan Kieseler. Fast convolutional neural networks for identifying long-lived particles in a high-granularity calorimeter. *arXiv:2004.10744 [hep-ex]*, April 2020. arXiv: 2004.10744.

[154] R. Acciarri et al. Convolutional Neural Networks Applied to Neutrino Events in a Liquid Argon Time Projection Chamber. *Journal of Instrumentation*, 12(03):P03011–P03011, March 2017. arXiv: 1611.05531.

[155] Michela Paganini, Luke de Oliveira, and Benjamin Nachman. CaloGAN: Simulating 3D High Energy Particle Showers in Multi-Layer Electromagnetic Calorimeters with Generative Adversarial Networks. *Physical Review D*, 97(1):014021, January 2018. arXiv: 1712.10321.

[156] Michela Paganini, Luke de Oliveira, and Benjamin Nachman. hep-lbdl/CaloGAN: CaloGAN generation, training, and analysis code, May 2017.

[157] Jonghwa Yim and Kyung-Ah Sohn. Enhancing the Performance of Convolutional Neural Networks on Quality Degraded Datasets. *arXiv:1710.06805 [cs]*, October 2017. arXiv: 1710.06805.

[158] Jinlong Hu, Yuezhen Kuang, Bin Liao, Lijie Cao, Shoubin Dong, and Ping Li. A multichannel 2d convolutional neural network model for task-evoked fmri data classification. *Comp. intelligence and neuroscience*, 2019.

[159] Flávio HD Araújo, Romuere RV Silva, Daniela M Ushizima, Mariana T Rezende, Cláudia M Carneiro, Andrea G Campos Bianchi, and Fátima NS Medeiros. Deep learning for cell image segmentation and ranking. *Computerized Medical Imaging and Graphics*, 72:13–21, 2019.

[160] Badder Marzocchi, CMS Collaboration, et al. Prospects for a precision timing upgrade of the cms pbwo4 crystal electromagnetic calorimeter for the hl-lhc. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 962:160181, 2020.

[161] Christopher McCabe. New constraints and discovery potential of sub-gev dark matter with xenon detectors. *Phys. Rev. D*, 96:043010, Aug 2017.

[162] The Phase-2 Upgrade of the CMS Level-1 Trigger. Technical Report CERN-LHCC-2020-004. CMS-TDR-021, CERN, Geneva, Apr 2020. Final version.

[163] Giuseppe Di Guglielmo, Javier Mauricio Duarte, Philip Harris, Duc Hoang, Sergo Jindariani, Edward Kreinar, Mia Liu, Vladimir Loncar, Jennifer Ngadiuba, Kevin Pedro, and et al. Compressing deep neural networks on fpgas to binary and ternary precision with hls4ml. *Machine Learning: Science and Technology*, Jun 2020.

Curriculum Vitae

Marina Prvan

Marina Prvan was born on May 30, 1989. in Split. After graduating from the General High School "Marko Marulić", she finished the study of Computing at the Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture (FESB) and the graduate study of Informatics at the Faculty of Science (PMF) in Split. Marina has worked as an external associate before she was employed as a teaching assistant at the Department of Digital Systems and Networks at FESB. She enrolled in the Postgraduate Study of Electrical Engineering and Information Technology, and she enrolled in an international dual doctorate („cottutelle“) in cooperation between Ecole Polytechnique (Institute Polytechnique de Paris) and FESB (University of Split) in 2016. Marina has been working on the CERN project (CMS HGCAL) ever since. She has published few scientific and technical papers in the journals and in the conference proceedings. Marina received a scholarship from the French Government in 2016 and the Dean's Award of the Faculty of Science in Split in 2016. She was also awarded with a recognition by expression of gratitude from the University of Split for cooperation with CERN in 2019.

Marina Prvan rođena je 30.05.1989. godine u Splitu. Nakon završene Opće gimnazije ”Marko Marulić”, završila je studij računarstva na Fakultetu elektrotehnike, strojarstva i brodogradnje (FESB) te diplomski studij informatike na Prirodoslovno-matematičkom fakultetu (PMF) u Splitu. Nakon završenog studija, Marina je radila kao vanjski suradnik prije nego što je zaposlena na FESB-u kao asistentica na katedri za Digitalne sustave i mreže. Upisala je Poslijediplomski studij elektrotehnike i informacijske tehnologije te međunarodni dvojni doktorat u suradnji fakulteta Ecole Polytechnique u Parizu (Institut Polytechnique de Paris) i FESB-a (Sveučilišta u Splitu) 2016. godine. Od tada, Marina radi na projektu CERN-a (CMS HGCAL). Napisala je nekoliko znanstvenih i stručnih radova u časopisima i zbornicima s međunarodnih skupova. Marina je dobitnica stipendije Francuske Vlade (2016) i Dekanove nagrade Prirodoslovno-Matematičkog Fakulteta u Splitu (2016). Dobitnica je i zahvalnice Sveučilista u Splitu za suradnju sa CERN-om (2019).

ECOLE
DOCTORALE

Titre: Algorithmes pour le déclenchement de niveau 1 pour le calorimètre HGCAL du détecteur CMS au HL-LHC

Mots clés: CMS, HGCAL, LHC

Résumé: L'instrumentation moderne en physique des particules (HEP) fait face à une augmentation rapide de la segmentation des détecteurs. Cela conduit à une augmentation du volume de données, qui requière une mise à niveau des détecteurs. Aussi, l'évolution des détecteurs est liée à la nécessité de suivre les évolutions technologiques, ainsi qu'à la nécessité de remplacer des parties du détecteur endommagées par les radiations. En particulier les détecteurs auprès du Large Hadron Collider (LHC) devront être mis à niveau pour la phase de haute luminosité (HL-LHC). Cette thèse décrit le travail de recherche effectué dans le contexte du calorimètre de haute granularité (HGCAL) envisagé pour la mise à niveau du détecteur CMS. Dans l'environnement difficile du LHC, avec des volumes de données plus élevés, plus de radiation, et plus d'empilement (PU), et où le nombre d'événements intéressants est faible, il est essentiel de fournir une décision de qualité en vue de garder ou non les données de l'événement. Ce processus, appelé déclenchement, doit opérer en temps réel, en prenant en compte les contraintes de communication et de capacité de calcul des processeurs disponibles. Les conditions d'opération du système de déclenchement sont difficiles car les algorithmes doivent être exécutés en un temps limité, sans possibilité de revoir la décision à postériori puisque les événements non sélectionnés sont définitivement perdus. Cette thèse présente les études réalisées pour la conception du nouveau système de déclenchement avec le HGCAL. Les études présentées concernent les aspects essentiels de la chaine de déclenchement, depuis la lecture des éléments de détecteurs à pixels et l'électronique de sélection frontale (FE), jusqu'au flot de données en sortie d'électronique dorsale (BE). Tout d'abord, la conception des modules du HGCAL est revue de façon à former des cellules de déclenchement à partir des cellules hexagonales, afin de réduire le volume de données par un regroupement des cellules de lecture. Lorsque le module est défini, une part important du travail est consacré aux stratégies en vue de réduire les données au niveau du FE et du BE. Des architectures sont étudiées en vue d'une génération de primitives de déclenchement pour laquelle une approche en deux étapes pour une agrégation en 3D est proposée. La première étape consiste en la recherche de régions d'intérêts (ROIs) dans le détecteur, et est basée sur un algorithme de reconstruction des traces (TA), qui permet l'identification des gerbes électromagnétiques (EM) et la sélection d'un germe pour le signal. Il est montré que plus de germes de signaux peuvent être sélectionnés lorsqu'une paramétrisation des gerbes EM est utilisée dans le TA. Finalement, le TA est utilisé dans un algorithme d'apprentissage pour la génération des ROIs. Cela conduit à une image de la gerbe, et un réseau de neurones (NNet) est appliqué pour effectuer la classification (gerbes EM ou PU). Nous avons comparé plusieurs modèles de NNet et leur performances (précision de la classification) sont mesurées en fonction de la complexité du modèle (nombre total de paramètres). Le meilleur compromis est ainsi obtenu entre la qualité de la décision et les contraintes sur le processeur.

Title: Algorithms for the Level-1 trigger with the HGCAL calorimeter for the CMS HL-LHC upgrade

Keywords: CMS, HGCAL, LHC

Abstract: Modern instrumentation in high energy physics (HEP) is facing the exponential growth of amount of data from the sensor arrays. This results in an enormous increase of output data volume, which requires in-time upgrades of detectors in HEP experiments. Also, the detector evolution is driven by the need to follow the newest technological trends as well as to replace parts of the mechanical construction that are damaged by radiation. In particular, the detectors at the Large Hadron Collider (LHC) will have to be upgraded before entering the high luminosity (HL) operational phase. This thesis describes the research work done in context of the High Granularity Calorimeter (HGCAL) project, which is part of the upgrade of the Compact Muon Solenoid (CMS) detector. In the challenging environment of HL-LHC, with higher data rates, harder radiation, and high pile-up (PU), and where the number of „interesting" events is low, it is essential to provide a quality decision on whether to read-out the event data or not. This process, called the trigger, should operate in real-time, under constrains of communications and processing limits from the available hardware. The working conditions of the trigger are challenging since the algorithm must be executed in a very limited time, without the possibility to revisit the decision of keeping the event for further processing or not. To cope with HL-LHC requirements, the current trigger system must be upgraded. The thesis presents the related studies that were necessary for the design of such trigger. The presented studies relates to key aspects that were necessary along the whole trigger path from the detector sensors read-out and front-end (FE) selection to the back-end (BE) electronics output data flow. First, a re-design of the mechanical HGCAL construction is studied on forming hexagonal sensor cells as well as larger polyhex structures of trigger cells (TCs) used to reduce the amount of data by using grouping of cells. Once the sensor module of the future HGCAL detector is defined, a large amount of work is devoted to the strategies for further FE data reduction and BE reconstruction studies. Architectures are explored for a possible trigger primitive generation from which a two-step approach for a direct 3D clustering is proposed. The first step consists of finding the regions of interest (ROIs) in the detector and is based on a designed tracking algorithm (TA), which enables the identification of electromagnetic (EM) shower tracks and of a signal seed selection. Also, it is shown that more signal seeds can be selected when an EM shower parametrization is used in the TA. Finally, the TA is used in the machine learning study for the ROI generation procedure. It results in an image of the physical shower, and a neural network (NNet) is applied to perform the data classification (EM-like or PU-like). We have compared several NNet models and the performance (classification accuracy) is measured against the model complexity (the total number of model parameters). The best trade-off is obtained between the quality of the decision-making process and the requirements on the hardware processing power.

Institut Polytechnique de Paris
91120 Palaiseau, France