Department of Physics and Astronomy
University of Heidelberg

Bachelor Thesis in Physics
submitted by

Konstantin Neureither

born in Mosbach (Germany)

2020

Towards an Online Reconstruction of Cosmic Muons for Mu3e using Hardware-Based Pattern Recognition

This Bachelor Thesis has been carried out by Konstantin Neureither at the
Physikalisches Institut Heidelberg
under the supervision of
Prof. Dr. André Schöning

Comment

This is an overworked version of the original document that was handed in. Corrections include minor spelling improvements and the following adaptions: Table 5.1 was supplemented with template counts, a mistake in Table 5.3 was corrected, a caption was added to Figure 5.6 and Figure 5.11 (*top*) was replaced as it contained the wrong plot.

Heidelberg, 08. January 2021,

Konstantin Neureither

Abstract

The *Mu3e Experiment* searches for the charged lepton-flavour violating decay $\mu^+ \to e^+e^-e^+$. The main detector is a multi-layer pixel tracker consisting of High-Voltage Monolithic Active Pixel Sensors (HV-MAPS). Due to vibrations, thermal fluctuations, and limited assembling accuracy, misalignment of the detector is a major precision bottleneck. An online alignment procedure based on particle tracks is foreseen to be integrated, to address this problem. In order to be suitable for online alignment, candidate particles should have a constant rate and a high momentum. With a detectable rate of about 10 Hz and momenta of a few GeV/c, cosmic muons meet these requirements. However, selecting cosmic muons from the decay particles produced at a beam rate of 10^8 Hz imposes great demands on data processing. To tackle this challenge, the utilization of hardware-based Pattern Recognition using associative memory chips has been proposed to filter potential cosmic muon tracks. This would reduce the heavy combinatorial effort and the data throughput to the reconstruction.

Within this thesis, the *Cosmic Trigger* simulation was developed, which allows for a feasibility study of this aforementioned approach. Furthermore, the optimisation of the concrete pattern design was a main objective. Based on insights gained from the simulation, recommendations towards a future implementation in hardware could be derived. In the scope of the study, it was possible to reach trigger selectivities such that only every *28th* frame must be reconstructed while detecting cosmic muons with an efficiency of 50 %.

Zusammenfassung

Das *Mu3e-Experiment* sucht nach dem leptonzahlverletzenden Zerfall von Anti-Myonen $\mu^+ \to e^+e^-e^+$. Die Hauptkomponente des Detektors bildet ein Pixeltracker, bestehend aus bis zu vier Lagen von hochspannungsbetriebenen monolitischen aktiven Pixelsensoren (HV-MAPS). Externe Einflüsse, wie Temperaturschwankungen, Vibrationen oder begrenzte Fertigungsgenauigkeit, können für Verformung und Fehlausrichtung sorgen, was die Messgenauigkeit des gesamten Systems beeinträchtigen kann. Um diesem Problem entgegenzutreten, ist eine geometrische Echtzeitkorrektur (*Alignment*) mithilfe von Teilchenspurdaten im Detektor vorgesehen. Geeignete Teilchen für *Alignment* haben eine konstante Rate und weisen einen hohen Impuls auf. Diese Anforderungen werden von kosmischen Myonen erfüllt, die mit ca. 10 Hz und mehreren GeV/c im Detektor eintreffen. Die Herausforderung besteht darin, die raren Myonen aus den bei einer Rate von 10^8 Hz produzierten Zerfallsteilchen herauszufiltern. Einen möglichen Lösungsansatz bietet die hardwarebasierte Mustererkennung (*Pattern Recognition*) mithilfe von assoziierten Speicherchips. Dadurch kann sowohl der Datendurchsatz als auch der kombinatorische Aufwand bei der Rekonstruktion deutlich reduziert werden.

Die vorliegende Arbeit bestand im Wesentlichen aus der Entwicklung einer Simulationssoftware, die eine Machbarkeitsstudie des *Cosmic-Trigger*-Konzepts ermöglicht. Desweiteren sollte das verwendete Musterdesign optimiert werden. Basierend auf den Erkenntnissen aus der Simulation konnten Empfehlungen hinsichtlich einer zukünftigen Realisierung gegeben werden. Bei einer Signaleffizienz von 50 % der kosmischen Myonen, war es möglich die Frame-Rate um einen Faktor von 28 zu reduzieren.

Contents

Part I

Introduction

Introduction

The field of experimental particle physics aims to answer some of the very fundamental questions of the universe by studying physical processes at the smallest spatial and largest energy scales possible. Since the precision of the detectors limits the observation of new discoveries, these are constantly improved. Mechanical misalignment is a major bottleneck for such high precision measurements. In order to achieve high measurement accuracy, detector alignment and calibration procedures are required.

One major alignment approach is the utilisation of external particles and their corresponding tracks through the detector from which systematic spatial deviations of individual sensor tiles can be calculated. Cosmic ray muons offer ideal alignment data due to their constant rate, high momentum and little scattering. Therefore, they are envisaged to be used for an online alignment of the *Mu3e experiment*.

Detecting cosmic muons with a rate in the order of 10 Hz, while the decays originated at the beam produce a event rate in the order of 10^8 Hz is a non-trivial task. It could be accomplished by reconstructing the complete data, which on the other hand leads to an enormous computational effort. Indeed, most particle physics experiments deal with event rates so high, that the produced data exceeds the handling capabilities of the associated data processing system if every event was supposed to be stored. Nevertheless, *trigger systems* offer a solution to this challenge and are a key component in most particle physics experiments. They are capable to make fast decisions which events to keep, while only a fraction of the total events can be stored.

At Mu3e, a trigger could be capable of narrowing down the rate of frames that potentially contain a cosmic muon track. In a second step, these frames are fully reconstructed. The selectivity of the trigger would reduce the total required data throughput to the reconstruction and thereby the hardware requirements.

The concrete trigger approach studied in this thesis is based on hardware pattern recognition using custom *Associative Memory* (AM) chips. Patterns of cosmic muon tracks are pre-computed using a Monte Carlo simulation data and subsequently loaded to the AMs. Reduced granularity hit information from the detector is used to quickly lookup if any pre-computed cosmic muon patterns can be found in the detector data within one frame of 50 ns. If this is the case, the frame is reconstructed.

This Thesis is dedicated to a detailed feasibility study of the presented approach, i.e. the *Cosmic Trigger*. Within its scope, a simulation framework that allows for the study of a pattern recognition on the Mu3e pixel detector was developed. The thesis takes the first step towards an evaluation and performance estimation of the Cosmic Trigger concept.

The Thesis is structured into three main parts. The *Introduction* (I) presents the physics theory and motivation for the Mu3e Experiment as well as some basics relevant for the further study (Chapter 1), and gives an introduction to the Mu3e experiment (Chapter 2). The *Methodology* part (II) focusses on pattern recognition in general, as well as in regard of the Mu3e experiment (Chapter 3). Furthermore, an introduction to the developed simulation software is given and important figures of merit are defined (Chapter 4). The final part, *Results* (III), walks through the evaluation process and shows the concrete simulation outcomes (Chapter 5). From these, a conclusion is drawn and practical recommendations for a future implementation are given (Chapter 6).

Chapter 1

Physics Motivation and Theory

This Chapter provides a short summary of the underlying physics of the Mu3e experiment and introduces to some physical effects of relevance for the *Cosmic Trigger*.

1.1 The Standard Model

The *Standard Model* (SM) of particle physics is supported by various experimental findings. So far, and especially since the discovery of the *Higgs boson* in 2012, which made the SM a self-complete theory, it describes the vast majority of observations in particle physics with a sublime accuracy. Nevertheless, the SM fails at offering a grand unifying theory. Fundamental short-comings are, for example, the missing description of the gravitational force and that it does not provide an explanation of neutrino masses or dark matter. The Mu3e experiment is one of the experimental approaches that search for physics beyond the Standard Model in order to find some missing but yet observable phenomena.

The SM describes the fundamental particles as well as their interactions. It incorporates two basic classes of particles: there are twelve matter particles, the *fermions* as well as particles that mediate interactions, the so-called *bosons*. These are depicted in Figure 1.1.

Fermions are particles with an elementary spin-1/2. They can be subdivided into quarks and leptons, both occurring in three generations respectively. There are charged leptons, the electrons (e), the muons (μ) and the tauons (τ). These experience the electroweak and the electromagnetic force. For each of them exists a corresponding neutrino, the electron-neutrino, the muon-neutrino and the tauon-neutrino, which only experience the electroweak force. *Quarks* experience the electromagnetic and the electroweak force and do carry a color charge, which implies that they experience the strong force as well. They are the building blocks of composite particles, such as the nuclear particles, the protons and the neutrons.

Vector bosons (or *gauge bosons*) are particles with spin-1. They are mediators of the different interactions or forces between particles. The *gluons* are the associated mediators of the *strong force*. There exist eight gluons that carry different color charge configurations. The electrically charged $W^{\pm}$ and the neutral Z *boson* mediate the electroweak force as the *photon* does for the electromagnetic force. There also exists one *scalar boson* within the SM, the *Higgs-Boson*. It has no spin and is neither electric nor colour charge.

1.2 Charged Lepton Flavour Violation

The *Mu3e* experiment searches for the charged lepton flavor violating (CLFV) decay $\mu \to eee$ which is strongly suppressed in the SM with branching ratio $\mathrm{B}(\mu \to eee) \ll 10^{-50}$ [2]. According to the SM, the lepton flavour is conserved at tree level. The lepton number L_ℓ is defined by the

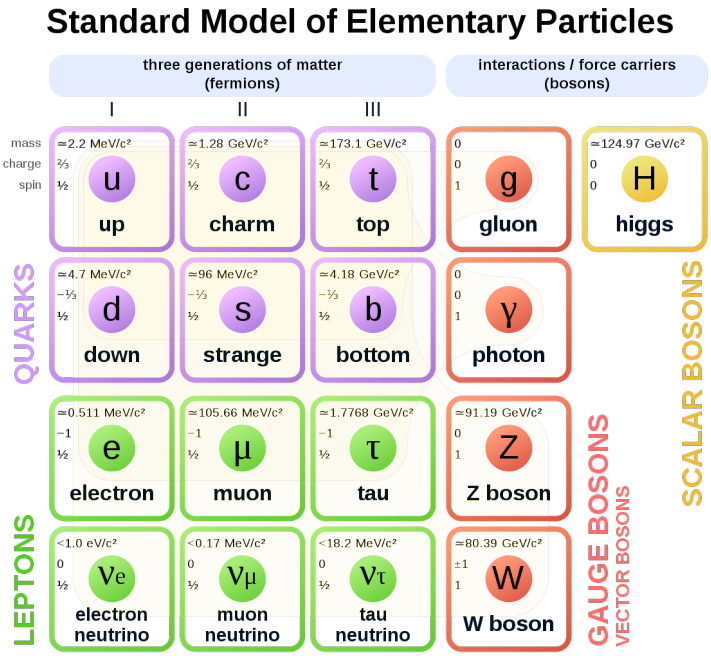

Figure 1.1: The *Standard Model* of particle physics [1].

difference of number of leptons n_ℓ and number of anti-leptons $\bar{n}_\ell$ as such:

$$L_\ell = n_\ell - \bar{n_\ell}, \tag{1.1}$$

whereby the index ℓ stands for the lepton flavor (e, μ or τ). L_ℓ is conserved within each family. However, within the SM, lepton flavour violation exists via neutrino mixing, which is strongly suppressed. The leading order Feynman-Diagram of this process is shown in Figure 1.2. The strong suppression is due to the branching ratio of the decay being proportional to the inverted fourth power of the mass of the *W boson* m_W^{-4}, which is $m_\mathrm{W} \approx 80\mathrm{GeV/c^2}$. Also the neutrino mass difference of the neutrino mixing influences the branching ratio, which reduces it even further.

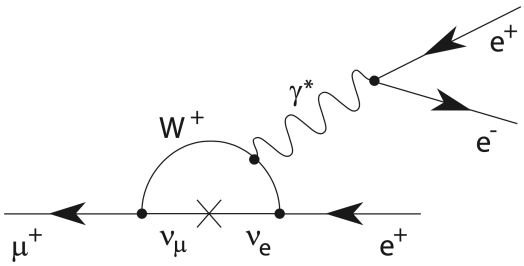

Figure 1.2: Feynman digramm of $\mu \to eee$ via neutrino-mixing (indicated by cross) [2]

In its final phase, *Mu3e* is supposed to be capable to search for the process $\mu \to eee$ with a sensitivity of 10^{-16}. If the decay could be observed underneath this sensitivity threshold, this would indicate the presence of other decay channels besides the ones allowed within the SM and would therefore give valid evidence on physics beyond the SM. One theoretical concept that allows for a decay with significantly higher branching ratio are *SUper-SYmmetric particles (SUSY)* as shown in Figure 1.3 (*left*). Additionally, lepton flavour violation on tree level could

be imaginable (Figure 1.3 (*right*)) which could point the existence of new, heavy mediating particles [2].

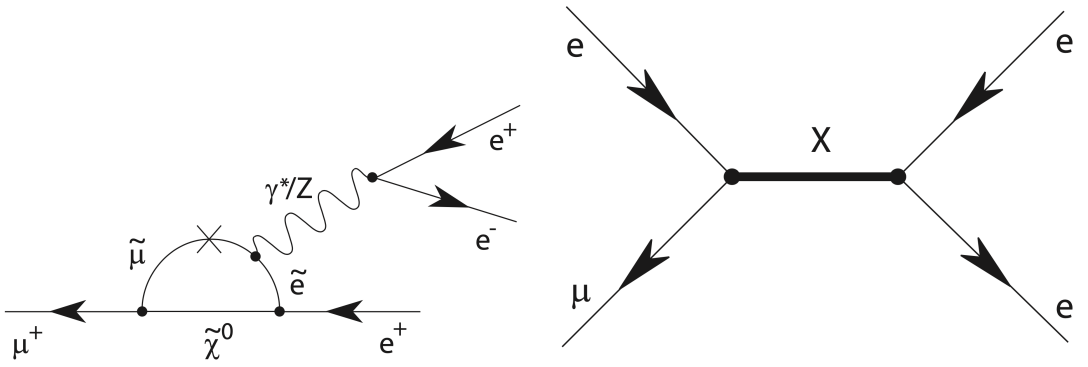

Figure 1.3: Feynman diagram of $\mu \to eee$ via super-symmetric particles (*left*) and a potential decay on tree level (*right*) [2].

1.3 Muon Decays and Background

The primary decay mode of muons at Mu3e is the *Michel decay* $\mu \to e\nu\bar{\nu}$. The Michel decay has a branching ratio of $\sim 100\,\%$ [3]. The resulting electron (positron) can emit another photon, leading to *radiative muon decay* which is described by $\mu \to e\nu\bar{\nu}\gamma$. Rarely, another e^+e^--pair is produced. According to [4], the branching ratio of this process is $B(\mu \to e\nu\bar{\nu}ee) = (3.4 \pm 0.4) \cdot 10^{-5}$.

From the perspective of the *Cosmic Trigger* that aims on collecting only cosmic muons, Michel decay events are the main source of *background*.

1.4 Particle Interaction with Matter

When particles traverse material, they can deposit some of their energy via ionisation. This is of great use in particle detectors, as this deposited energy can be measured and the particle thereby detected. At *Mu3e* the particles of interest are mostly electrons and also cosmic-ray muons (in case of the cosmic trigger). Both of them carry electric charge and are ionising particles. In general, the energy loss and scattering of a particle traversing a material is the summation of different contribution terms of different processes. It depends of the scattering characteristics of the particle as well as the material. The mass of the electron is a lot smaller than the mass of the muon ($m_\mu/m_e \sim 200$), therefore different processes dominate the scattering of these respectively.

Muons and heavy charged particles

Charged particles with a mass above $m \gg m_e$ are called heavy charged particles. This category includes muons, taus as well as protons or ions. Their scattering is dominated by inelastic collisions with electrons of the material, which lead to excitement or ionisation of the materials atoms. The energy loss E per travelled distance x can be described by the *Bethe-Bloch Formula* [5]. It consists of two main constituents, the first is derived from the energy transfer according to the *Bohr model* while the second (in brackets) adds a relativistic correction term:

$$-\frac{dE}{dx} = \frac{4\pi n z^2}{m_e c^2 \beta^2} \cdot \left(\frac{e^2}{4\pi\epsilon_0}\right)^2 + \left[\ln \frac{2m_e c^2 \beta^2}{I \cdot (1-\beta^2)} - \beta^2\right]. \tag{1.2}$$

c: speed of light | e: electron charge | $\beta = \frac{v}{c}$

ϵ_0: vacuum permittivity | n: electron number density | v: speed of particle

m_e: electron rest mass | I: mean excitation potential | z: charge of particle

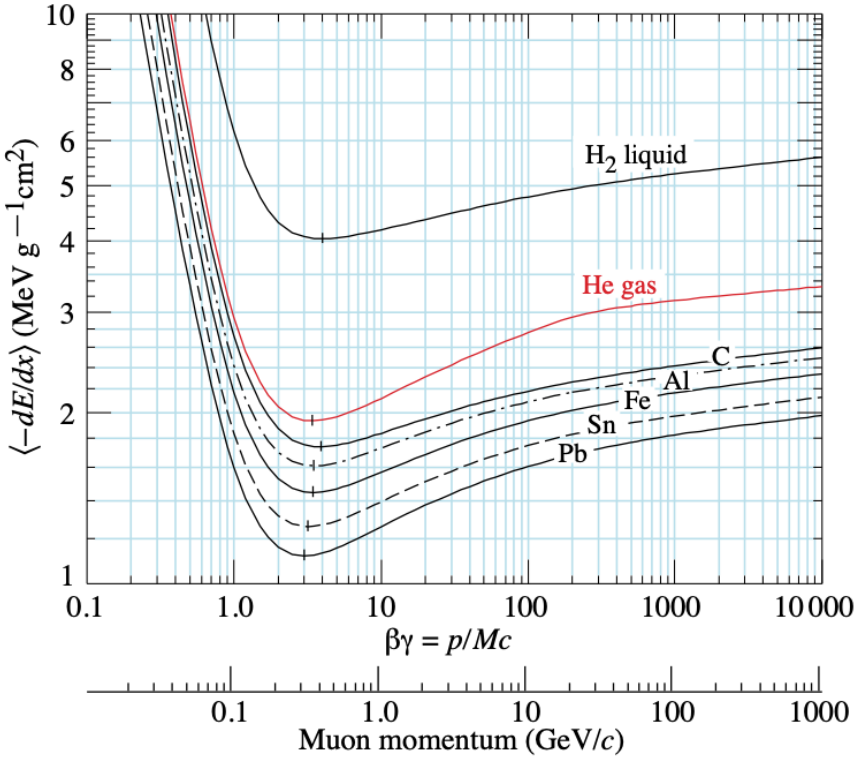

Figure 1.4: Mean energy loss for different particle energies and materials[6].

The energy loss depends on the energy of the incoming particle. The dependency is shown in Figure 1.4. The graph first decreases with increasing momentum of the particle. After the minimum at about $\beta\gamma \approx 2...3$ it increases logarithmically. The minimum corresponds to the energy of so called *minimum-ionising particles*. Particles with this momentum deposit the least energy in the material and therefore experience the least scattering. The increase for even higher momenta is due to relativistic effects on the electromagnetic fields.

Electrons and Positrons

Electrons posses only a fraction of the muon rest mass. For them, in addition to ionisation, the scattering process is dominated by *Bremsstrahlung* which means the deceleration of an electron by the electromagnetic field of another charged particle in the material it is currently traversing. For the contribution of ionisation the Bethe-Bloch formula must be modified. The ionisation of electrons therefore is described by the *Berger-Seltzer-Formula*. Its curvature looks similar to the one of the Bethe-Bloch formula; hence, there is an electron energy with minimal ionisation as well.

The contribution of the *Bremsstrahlung* is proportional to the particle energy and can be defined in relation to the radiation length X_0 according to [7] by

$$-\left.\frac{dE}{dx}\right|_{rad} = \frac{1}{X_0} \cdot E. \tag{1.3}$$

X_0 is the distance after which an electron has lost $\approx 37\,\%$ of its energy and is proportional to the one over the second power of the electron mass $X_0 \sim m_e^{-2}$.

Multiple Coulomb Scattering

When a charged particle traverses a medium, it is usually scattered multiple times by small scattering angles. The scattering is caused by Coulomb interactions of the particle with the nuclei. *Multiple Coulomb scattering* is a complex process because it consists of many single statistical events. A description is given by *Molière's Theory*. Furthermore it includes more parameters than single-scattering because in addition to the scatter-angle there is also an offset that can be added to the trajectory after traversing the medium. Figure 1.5 shows an overview of the process and the parameters that are used to describe it. The RMS width of the resulting Gaussian scattering distribution is approximately given by

$$\theta_{RMS} = \frac{13.6\,\mathrm{MeV}}{\beta cp} z \sqrt{\frac{x}{X_0}} \left[1 + 0.038 \ln \frac{xz^2}{X_0 \beta^2}\right], \tag{1.4}$$

where p denotes the momentum, βc and z the velocity and charge number of the particle. x/X_0 is the thickness of the medium in number of radiation lengths [6].

As shown in Equation 1.4, the multiple Coulomb scattering angle increases for less heavy and low momentum particles with small radiation lengths. Subsequently, multiple Coulomb scattering is predominantly present when electrons traverse "thick" media.

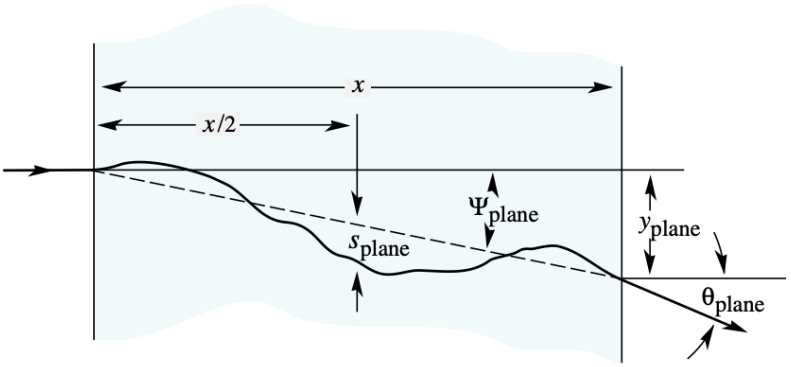

Figure 1.5: Schematic of multiple Coulomb scattering in one plane. [6]

1.5 Cosmic Ray Muons

Cosmic ray muons are produced when cosmic ray such as protons, alpha particles or heavier nuclei collide with particles in the earths atmosphere and subsequently decay. About 90 % of cosmic ray consists of protons, mostly originated at the sun and the milky way. If they collide with atmosphere molecules, they can decay into several secondary particles, e.g. pions or kaons. Some possible decay chains for protons are sketched in Figure 1.6.

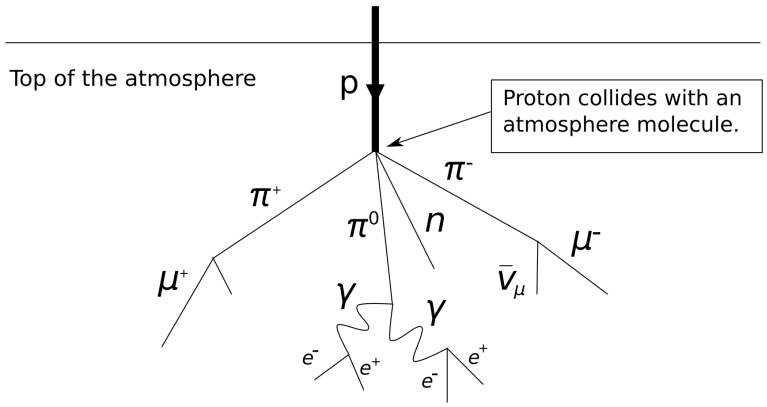

Figure 1.6: Primary cosmic particle collides with a molecule of atmosphere. [8]

Pions and kaons are not stable and therefore decay further into muons and muon-neutrinos by weak interaction as shown in the process 1.5 and 1.6.

$$\pi^- \to \mu^- + \bar{\nu_\mu} \qquad \pi^+ \to \mu^+ + \nu_\mu \tag{1.5}$$

$$K^- \to \mu^- + \bar{\nu_\mu} \qquad K^+ \to \mu^+ + \nu_\mu \tag{1.6}$$

These processes of muon production happen in the upper layers of the atmosphere at about $10 - 15\,\text{km}$ altitude. On their way to earths surface at sea level, they loose about $2\,\text{GeV}$ [9] of energy. Muons have a very short lifetime of $\tau_\mu = (2.1969811 \pm 0.0000022) \times 10^{-6}\mu\text{s}$ [10]. Despite that and because of relativistic effects of their high velocity, which leads to time dilation and length contraction from the earths perspective, they reach the ground. Their energy distribution at sea level lies in the GeV regime and has its maximum at $\approx 1\,\text{GeV}$ (see Figure 1.7).

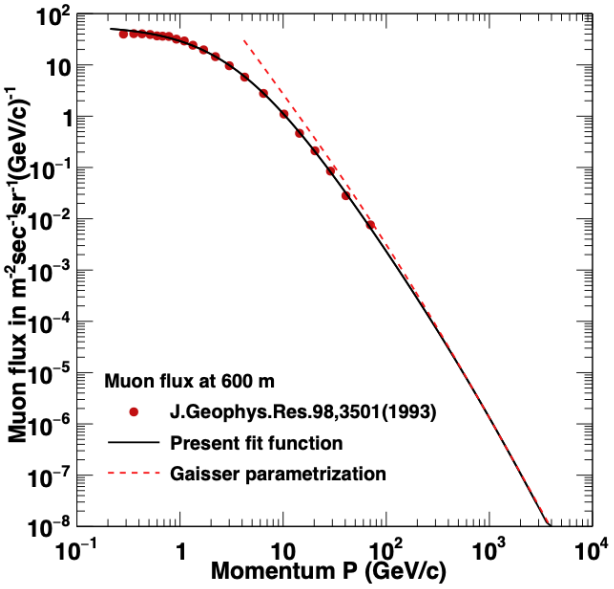

Figure 1.7: Muon momentum distribution at 0 deg zenith angle at 600 m altitude [9].

Figure 1.7 clearly shows that the rates decrease exponentially for higher momenta. In addition, structures like concrete or magnets lead to energy loss, because muons scatter when traversing them. In the detector of *Mu3e* mostly cosmic muons with energies at the lower end of the GeV spectrum are expected to be detected. According to Figure 1.4, their energy is close to

minimum-ionising particles, which makes them less prone to scattering in general and therefore suitable for alignment. Assuming a zenith angle of 0°, the approximate rate of cosmic muons at sea level with an energy $E < 0.5\,\mathrm{GeV}$ is given by [9]

$$I_{\mu,0^\circ} = 70\,\frac{\mathrm{muons}}{\mathrm{m}^2\,\mathrm{s}\,\mathrm{sr}}. \tag{1.7}$$

It decreases for steeper angles because the effective travelling distance increases and with it the probability of scattering.

Chapter 2

The Mu3e Experiment

2.1 Experimental Situation of Mu3e

The *Mu3e* experiment is planned to be installed at the *Paul-Scherrer-Insitut* (PSI) in Switzerland because it offers the worlds strongest anti-muon-beam, which will even be improved futher. In order to reach the final sensitivity of 10^{-16} for the observation of the CLFV decay $\mu^+ \to e^+e^-e^+$, the beam has to provide a muon rate of $2 \cdot 10^9\,\mathrm{Hz}$. The experiment was proposed in 2013 and is being under development since then.

2.1.1 The Search for Charged Lepton Flavour Violation

Charged lepton flavor violation has been studied for several decades. It is one promising candidate to find evidence on physics beyond the SM when searching for the branching ratios of decays like $\mu \to eee$. Different experiments have studied CLFV in detail and have also set different limits for certain decays as depicted in Figure 2.1. The two major ones are described in short.

SINDRUM Experiment

The *SINDRUM* experiment was operating from 1983-1986 at PSI and searched for the decay $\mu \to eee$. Limited by the muon beam intensity and operation time, it was able to set a branching ratio limit of $\mathrm{B}(\mu \to eee) < 10^{-12}$ with a confidence level of 90 % [11].

MEG Experiment

The *MEG* experiment was also running at *PSI* during the years of 2008-2013. It observed the radiative LVF decay of a muon into an electron and photon without the production of neutrinos $\mu \to e\gamma$. Taking into account the complete data collected over the years, MEG was able to set the limit of $\mathrm{B}(\mu^+ \to e^+\gamma) < 4.2 \cdot 10^{-13}$ with a confidence level of 90 % [12].

2.1.2 Mu3e Road Map

From the very beginning, *Mu3e* was planned to be realised within two different phases, which mainly differ in the muon beam rate and some adaptions on the detector.

Phase I is subdivided into part A and B. Phase IA includes the overall development of all the Hardware (HW) components, such as the pixel sensors, the magnet and the simulation software, including testing of a prototype detector. The first data is then supposed to be collected with a minimal working detector from a muon beam with a rate of $2 \cdot 10^7\,\mathrm{Hz}$ in phase IA. In phase IB the detector is planned to be upgraded and equipped with additional recurl stations. The rate

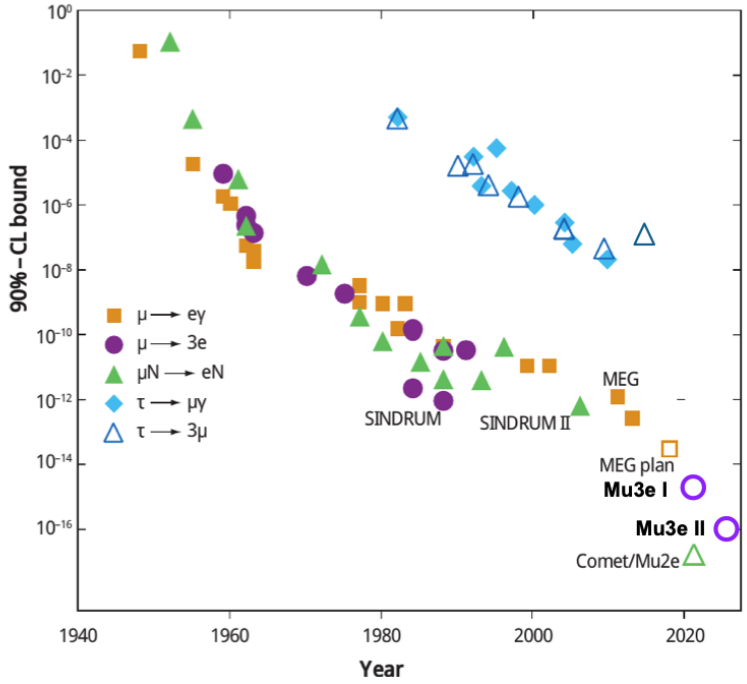

Figure 2.1: History of LFV decay experiments and their obtained results.[13]

is increased to $1 \cdot 10^8$ Hz, which would yield a sensitivity of $2 \cdot 10^{-15}$ at 90 % CL. As of 2020 the detector development is ongoing and the first actual measurements will start soon.

Phase II imposes new challenges to the different parts of the experiment. First, it depends on the new high intensity muon beam line that is currently under construction at PSI. Also, pixel sensor timing resolution as well as read-out and data acquisition requirements impose new challenges. However, with its planned sensitivity of 10^{-16} in Phase II, the *Mu3e* experiment would able to exceed the known limit for $\mu \to eee$ (*SINDRUM*) by about 4 orders of magnitude. Assuming that LFV dipole couplings are the dominant channel of this decay, the limit determined at *MEG* for $\mathrm{B}(\mu \to e\gamma) < 10^{-13}$ corresponds to $\mathrm{B}(\mu \to eee) < 10^{-15}$.

2.2 Detector Concept

A particle physics experiment usually puts very specific demands on the detector concept. At *Mu3e*, it is of high importance to be able to filter background events such as the $\mu \to eee\nu\nu$. Therefore, a high momentum resolution of below 1.0 MeV/c [14] is essential. This is dominated mainly by two factors, the first of which is the spatial hit information resolution (which corresponds to the pixel size on pixel detectors). Secondly, the momentum of the resulting decay particles is limited by half the muon energy, which means $p_e \lesssim 53\,\mathrm{MeV/c}$. At these magnitudes of momentum, electrons are highly susceptible to multiple Coulomb scattering. In order to reduce multiple Coulomb scattering, the detector must be constructed with as little material budget as possible. Another very important requirement is a precise time resolution of $\mathcal{O}(1\mathrm{ns})$ [2]. For higher beam rates, the timing of the hits is an important parameter to separate the particles according to their origin decay events. [15]

2.2.1 Geometry and Components

The *Mu3e* detector includes three main detecting components, the main component, a silicon pixel tracking detector and tile and scintillating fibre (*SciFi*) detectors for more precise timing

information. The detector is build cylindrically around the target and is subdivided into three "stations". The *central station* surrounds the target and two *recurl stations* are placed upstream and downstream along the beam line. The overall length of the active area of the detector is about 1.2 m, with a diameter of about 18 cm. A schematic view is presented in Figure 2.2. The complete detector setup including the double cone target is placed in a homogeneous magnetic field produced by a solenoid magnet of $B = 1\,\mathrm{T}$.

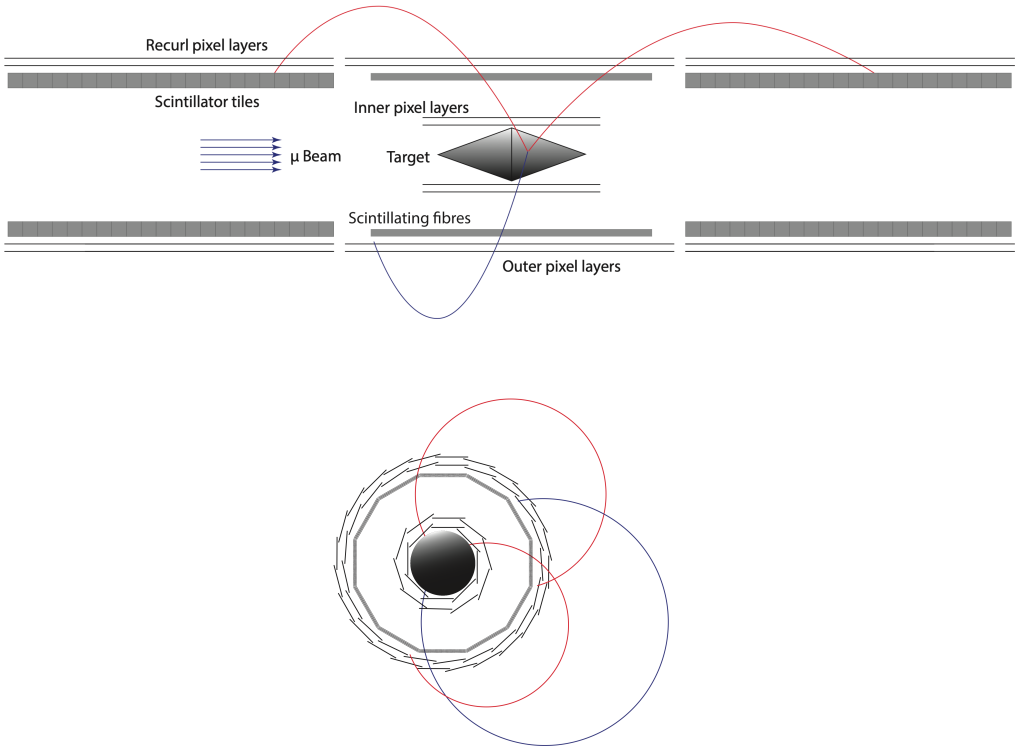

Figure 2.2: Schematic view of the experiment cut along the beam axis (*top*) and transverse the beam axis (central station) (*bottom*) in the phase I configuration [14].

Tracking Detector

The silicon pixel tracker is the main part of the detector. It consists of up to four layers of pixel sensors and provides accurate pixel hit information for ionising particles. From the hit information in each layer, particle tracks are reconstructed and their momenta are calculated. When a charged particle passes through the magnetic field, it can be described by a helical trajectory due to the Lorentz force. The momentum can then be obtained from the radius R_t transverse the B-field, the particle charge q and the strength of the magnetic field B via the formula:

$$p_t = R_t \cdot q \cdot B. \tag{2.1}$$

For single-charged particles in the case of *Mu3e* with $B = 1\,\mathrm{T}$ with units this simplifies to

$$p_t\,[\mathrm{GeV}] = 0.3 \cdot R_t\,[\mathrm{m}] \cdot 1\,[\mathrm{T}]. \tag{2.2}$$

The tracking detector consists of about 3000 High-Voltage Active Monolithic Pixel Sensors (*HV-MAPS*), called *MuPix*, each covering an active area of about $20 \times 20\,\mathrm{mm}$. Each *MuPix* chip has 256×250 pixels with a pixel pitch of $80 \times 80\,\mu\mathrm{m}$. The most recent prototype version is *MuPix10* which is currently under investigation.

The pixel tracker has 2 outer layers covering all three stations (see Figure 2.2). In the central station the inner tracker is placed, consisting of two additional layers of pixel sensors, surrounding the target closely. In order to build the cylindrical structure, the individual *MuPix* sensors are glued to ladders (linear arrays of sensor chips), which are then mounted to a support structure located in between, and at the end of the stations. At the bottom end each *MuPix* chip has another 3 mm of read-out electronics. Because of this and to eliminate blind spots, the ladders are mounted with a few millimeters of overlap. Table 2.1 gives an overview of the layer dimensions. Figure 2.2 (*bottom*) presents a cut through the x,y-plane).

layer (from innermost to outermost)	1	2	3	4
number of ladders	8	10	24	28
number of *MuPix* sensors per ladder	6	6	17	18
instrumented length [mm]	124.7	124.7	351.9	372.6
minimum radius [mm]	23.3	29.8	73.9	86.3

Table 2.1: Pixel layer geometry for central station. Adapted from [14].

Because multiple Coulomb scattering has a high influence on the decay particle's trajectory, the sensor chips are thinned to about 50 μm. Including the ladders this corresponds to a radiation length of about $x/X_0 = 0.115\,\%$ per layer.

2.2.2 Read-out, Data Acquisition and Online Reconstruction

Once the data is collected by the pixel, the tile and the fibre detectors it has to be processed further. Especially when reaching the upper end of desired muon beam rates, it is impossible to store the complete data collected by the detector. At a stopping rate of 10^8 Hz, a data throughput of about 60 GBit/s is expected [14]. However, all of this data needs to be filtered and a decision on what to keep and what to discard has to be made. This is done by the *Mu3e data acquisition* (DAQ) system.

The DAQ has three levels: the *front-end*, the *switching boards* and the *GPU filter farm*. The different sensors collect hit positions and timing information and send this data to their corresponding front-end FPGA. One front-end FPGA gathers the hit information of multiple sensors. All the information that arrives from them is brought into chronological order and sorted into frames of 50 ns. The sorted data is forwarded to the switching boards which merge the data of the different detector types, i.e. the pixels, the fibres, and the tiles. In a subsequent step, the data is analysed and filtered by the GPU filter farm. Several GPUs process individual time slices of the detector data and combine all hits to track candidates. Afterwards, a track fitting is performed and according to their quality, tracks are either accepted or rejected. The filter farm then stored frames with reconstructed tracks of interest, whereas other are dicarded. The complete DAQ chain is shown in Figure 2.3.

2.3 Alignment

As aforementioned, the *Mu3e* detector is a high precision and complex instrument. For the pixel detector, 2844 sensor chips are glued to ladders which are then mounted in a sophisticated and detailed mechanical process. The accuracy of this process is limited to a certain mechanical precision. On sensor-level, it is at the same magnitude as the pixel resolution of 80 μm. Globally, higher misalignment deviations of several hundred μm or in the worst case even at $\mathcal{O}$(1mm) are expected [15].

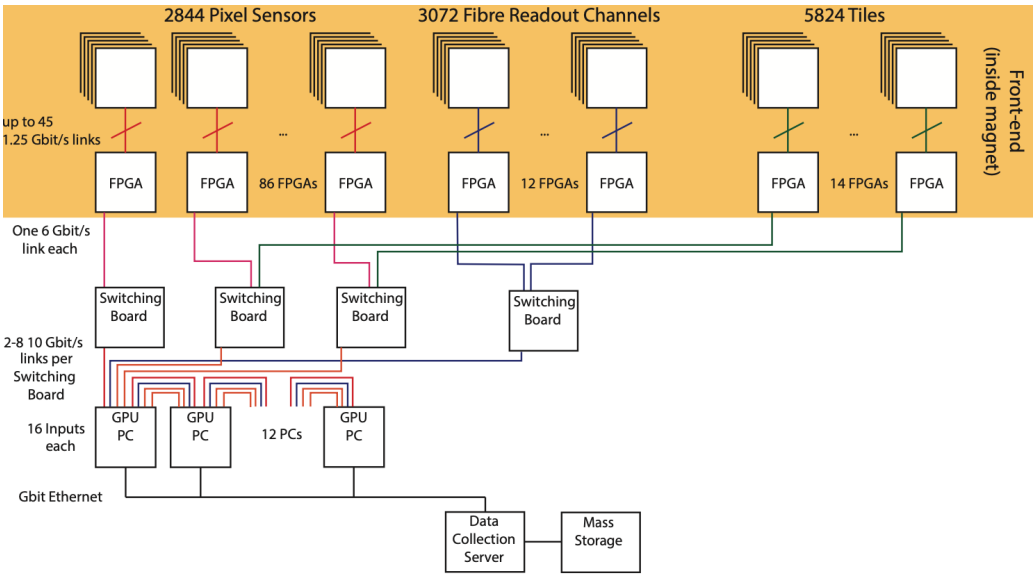

Figure 2.3: Overall Mu3e read-out scheme. [14]

In addition to the building accuracy and resulting constant misalignment, there is also non-constant effects. Everything inside the detector is build with the least material budget possible in order to reduce multiple Coulomb scattering. The ladders for example have a thickness of about 100 mm but freely span over distances of up to 400 μm in the recurl stations. This architecture makes the detector very fragile to any kind of environmental influence. First to be mentioned, the detector is located in a magnetic field. When switching it on, magnetic parts of the support structure can move slightly, which leads to misalignment of the detector. Even if this happens only in the area of $\mathcal{O}(\mu\text{m})$, it is enough to introduce invalidity into any pre-measured or pre-calculated corrections for construction inaccuracy and misalignment.

More misalignment can be caused by thermal influence of the experiments environment. The pixel sensors are cooled via a gaseous helium cooling atmosphere. This means that gas is circulating through the detector and can subsequently introduce movement and misalignment of sensors or of the ladders they are placed on.

In order to achieve the momentum resolution required for *Mu3e*, the detector has to provide sufficient spatial resolution and its data have to be systematically valid. Therefore, a precise detector alignment is a key ingredient for reliable tracking. "Alignment" in this case means to gather correction data and then to calculate a correcting transformation of the smallest sub-detector components in software. As the misalignment changes significantly over time, an online alignment could be performed so that correction can be directly taken into account for the track reconstruction on the GPU filter farm. Approaches to alignment and potential future concepts for alignment for *Mu3e* are discussed in the following sections.

2.3.1 Alignment Approaches at Mu3e

Two main approaches are envisaged at *Mu3e*. The first is a position monitoring system which relies on camera data taken from cameras inside the detector cage, focussing the three detector stations. Using these high resolution cameras (of presumably 2K or 4K) it is possible to measure the positions of the separate stations with respect to each other [2].

The second approach of alignment is based on actual detector data from measured particles, which has already been studied [15]. Using actual detector data allows for fine alignment on pixel resolution accuracy. The algorithm developed within this study uses reconstructed tracks

from the experiment and fits them using *General-Broken Lines* (GBL) fit, which increases the track precision. Then systematic deviations of individual sensor chips, so-called *residuals* are determined. These residuals are used to calculate pixel sensor specific alignment transforms, from which the sensor hit coordinates can be corrected to their actual aligned coordinates.

The track-based alignment works for trajectories originated at the stopping target, but also external particles could be used, yet might even allow for better accuracy. A good example for particles that offer this potential are cosmic muons.

2.3.2 Alignment using Cosmic Ray Muons

Cosmic ray muons (in the following referred to as *cosmic muons*) are produced when cosmic ray collides with molecules in the atmosphere, as described in Section 1.5. They approach earth's surface with momenta of a few GeV/c. Because of their high energy, they are very resistible to multiple Coulomb scattering. Also, their radius is very large in the magnetic field. According to Equation 2.2, a muon with 1 GeV/c describes a radius of $R_t = 3.3\,\mathrm{m}$. Inside the detector that has a diameter of $d_{\mathrm{detector}} \approx 18\,\mathrm{cm}$. Therefore, a cosmic muon track in the detector can almost be described as a straight line.

Another important advantage of cosmic muons over the decay particles from the target is, that they are originated in the atmosphere, which means that they arrive from above the detector while being evenly distributed over the complete detector area (projected in horizontal plane). A cosmic muon therefore hits every layer it traverses twice, the first time when entering the detector and the second time when leaving the detector again. While decay particles from the target mostly cause hits the central detector region, cosmics are found in the recurl stations at the same rate as in the central. Additionally, one cosmic muon can cause hits on central and recurl stations when approaching in a steep angle. Despite this happening seldomly, such a track offers the opportunity to align the stations relative to each other, which could only be done by an optical system (with less precision) otherwise.

With the advantages described above (including negligible multiple Coulomb scattering and straight line tracks), cosmic muons are ideal particles for (online) alignment.

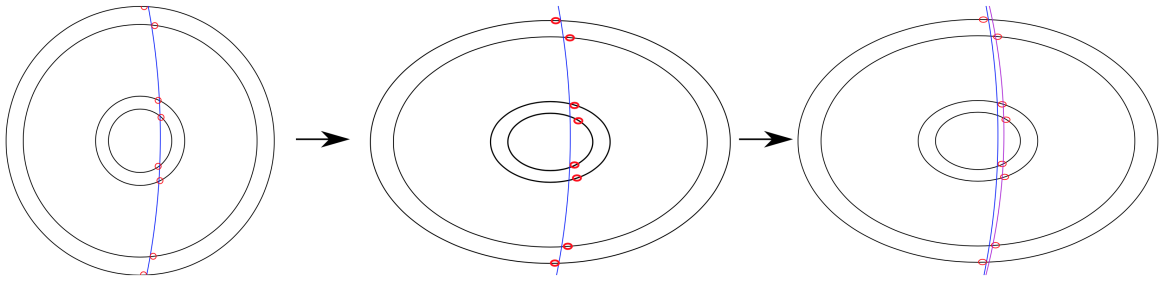

Figure 2.4: A cosmic traversing the central detector and a elliptical correction transform [15].

There is one notable difficulty when aligning with cosmic muons. As described in Section 1.5, their rate at sea level is about $I_{\mu,0^\circ} = 70\,\mathrm{muons\,m^{-2}\,s^{-1}\,sr^{-1}}$. The total surface area of the *Mu3e* pixel detector projected on the horizontal plane is approximately $A_{det} \approx 0.18\,\mathrm{m} \times 1.2\,\mathrm{m} \approx 0.22\,\mathrm{m}^2$. Taking into account further loss because of the experimental setup, the magnet, the ceiling, et cetera, a cosmic muon rate of about $\mathcal{O}(10\,\mathrm{Hz})$ is expected to be measurable. Compared to the rate of decay particles from the target, which is about $\mathcal{O}(10^7 - 10^8\,\mathrm{Hz})$, this is highly suppressed. Yet still, the track quality of cosmics for alignment compensates for the very low rate. This situation imposes two main challenges:

- For improved alignment an increased number of detected cosmic muons is favored.
- Each having a length of 50 ns, one second is split into $2 \cdot 10^7$ frames. Only every $\sim 10^6$th frame contains a cosmic muon.

Subsequently, a trigger or tagger system is required that recognises the frames that contain a cosmic muon. To tackle the above mentioned challenges, this system demands a high *signal* (or *cosmic*) *efficiency* to maximise the cosmic detection rate. Furthermore a sufficient *rejection rate* is crucial, so that the number of frames that are reconstructed is minimised. How these two requirements could be met by the *Cosmic Trigger*, a pattern recognition system, is to be studied within this thesis.

2.3.3 Cosmic Trigger Hardware Concept

In principle, different ways exist to find the frames where cosmics traversed the detector. One would be to just use the existing DAQ as described in 2.2.2. In the GPU filter farm one could not only reconstruct tracks according to the expected decay characteristics, but also try to find cosmic muons by performing the full combinatorial possibilities. This would mean that the track fitter block must be extended by several GPUs. Also, it has not been studied yet what the combinatorial effort would be to perform a full online reconstruction for cosmic muons. It might well be that it significantly exceeds the effort that is required for event reconstruction. Therefore, it is possible that a cosmic filter farm could require more PCs than the event filter farm does, in order to cope with the data rate from the detector. As shown in Figure 2.3, 12 GPU PCs are envisaged for Mu3e phase I. A full cosmic reconstruction filter farm could more than double this number. Assuming a cost of 7k-8k euros per filter farm PC (including main board, GPU and connectivity), the total cost of a cosmic filter farm could easily exceed 100k euros.

Another option is to use a dedicated trigger for cosmics. This system could be designed, such that it analyses the frames without computing the full combinatorial possibilities of every hit in each frame. By pre-filtering the detector data and thereby reducing the rate of frames that need to be reconstructed, this could significantly reduce the required GPU computation. *Pattern Recognition* (PR) is a method to do so, which is the reason why it was proposed be used to trigger cosmic muons in the *Mu3e* experiment. However, this approach requires different hardware components to be implemented. Eventually, a simple cost calculation will decide, if a PR system can sufficiently reduce the reconstruction data rate so that the savings in filter farm PCs outplay the cost for the PR hardware components.

Returning to 2.3, the read-out and data acquisition chain of Mu3e a separate trigger could be integrated next to the filter farm. It would then obtain the full pixel sensor data and could forward a reduced amount of data to a separate cosmic track fitting GPU block. This GPU block would do the fitting, recognise the frames containing an actual cosmic muon and forward them to the data collection server. The biggest challenge would be for this cosmic track fitting GPU block to also compute online alignment correction data, and to send this to the event track fitter block, so that it could be used in real time for beam event track reconstruction. Figure 2.5 shows, how the *Mu3e* read-out and DAQ system could be extended by a pattern recognition cosmic trigger relying on data from the pixel sensors.

2.4 The Mu3e Simulation Package

A simulation software was implemented to model the complete experiment, to study and optimise the detector design, to estimate the performance and to develop the online reconstruction code, which will later be run on the DAQ. The Mu3e simulation is based on *Geant4* [16], a Monte Carlo

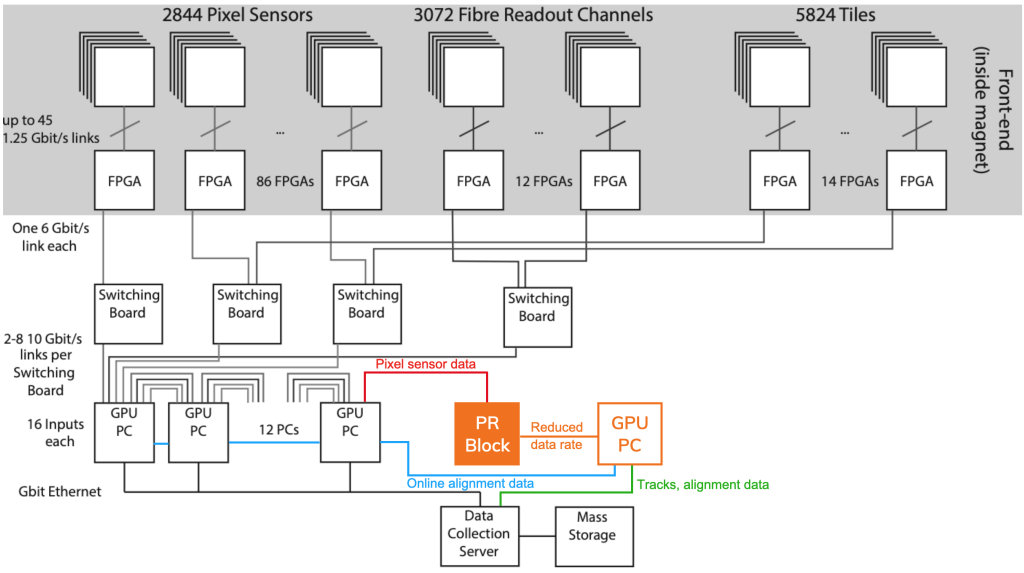

Figure 2.5: Conceptual Mu3e read-out scheme with cosmic trigger. The planned DAQ is depicted in grey scale and the proposed expansion in color. Potentially, more than one GPU could be required. Adapted from [14].

software framework developed at CERN for simulating particles traversing matter. The Mu3e software stack includes a detailed detector simulation and implements several different muon decay types. Also it delivers an accurate multiple Coulomb scattering simulation, which is of particular importance for *Mu3e*. Since 2020, also a cosmic muon generator is implemented, which is described in Section 2.4.4.

2.4.1 Motivation

In order to study pattern recognition, a key prerequisite are the patterns themselves as described in detail in Chapter 3. Those can be obtained from actual cosmic muon tracks. The *Mu3e* simulation implements a "cosmic mode", which uses a Monte Carlo (MC) algorithm to simulate cosmics muons traversing the detector. The detector data is then reconstructed using the reconstruction software. For the *Cosmic Trigger* study, a data set containing approximately $4 \cdot 10^7$ cosmic muon tracks was produced.

Furthermore, Mu3e simulation played a key role for the evaluation of the *Cosmic Trigger Concept*. It was used to create frames only containing hits of beam data decays, which in case of the *Cosmic Trigger* is referred to as *background*. This is due to the fact that the beam decays are the dominant source of background hits that must be suppressed compared to the hits of the cosmic muons, for which the *Cosmic Trigger* searches.

2.4.2 Reconstruction

In addition to its relevance for the simulation, the Mu3e Reconstruction Software is the algorithm that will later be implemented on the GPU filter farm. It will efficiently fit and evaluate track candidates in order to reconstruct all event particle tracks present in a simulation frame (and also tracks that extend over frame borders). In order to achieve this functionality online, fast and efficient methods have to be implemented.

Reconstruction of Beam Events

The first step of the processing of hit data is a very fast three-dimensional *Multiple Scattering Fit* [17]. Hits in the inner layers 1, 2, and 3 are combined to triplets and the χ^2 of each track is determined. If it is sufficiently low, the outermost layer 4 is searched for a hit that would extend the track with a triplet using the hits in layer 2 and 3. If this condition is met, a *short track* is found, defined by four hits. From here on, some more decisions are made, using e.g. graph partitioning algorithms and neural networks. After successfully reconstructing a frame, a decision is made on whether it is stored or discarded.

The stored frames are reconstructed again offline, using a more complex, the General Broken Lines fit. The GBL delivers the highest accuracy and allows for precise particle momentum and calculation and for energy loss determination.

Reconstruction of Cosmic Muons

To reconstruct and match the hits of cosmic muons, the reconstruction algorithm has to be modified. Cosmic muons come from the atmosphere (upper half of detector) and normally cause two hits on each layer they traverse. Thus, the algorithm first fits a triplet to three hits in the outer detector layer (h_0, h_1, and h_2 in Figure 2.6), then looks for a fourth hit (h_3) in the missing outer layer and then adds hits from inner layers – if present [18].

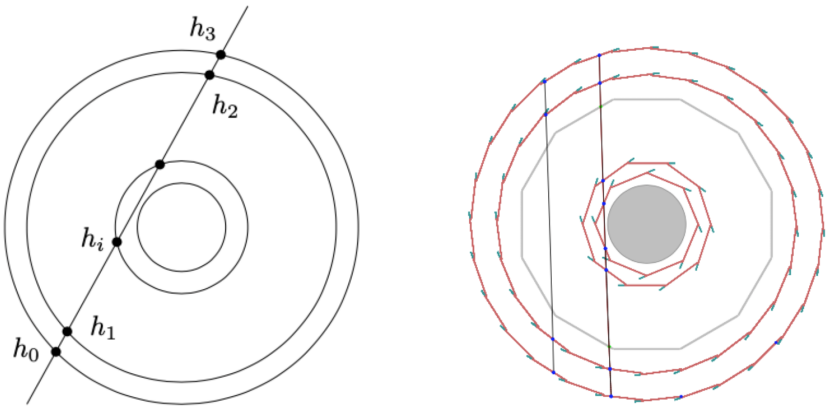

Figure 2.6: Reconstruction of cosmic muon hits. [18]

2.4.3 Coordinate Frame an Nomenclature

In the Simulation, the detector is placed in a cartesian reference coordinate system. The z-axis is oriented along the beam line, x- and y-axis horizontally and vertically respectively. This is shown in Figure 2.8 and 2.9. A 3D render of the whole detector with a cut at the target is shown in Figure 2.7. Note that the annotations concerning the super pixel binning will be introduced later, as they are important for the pattern design.

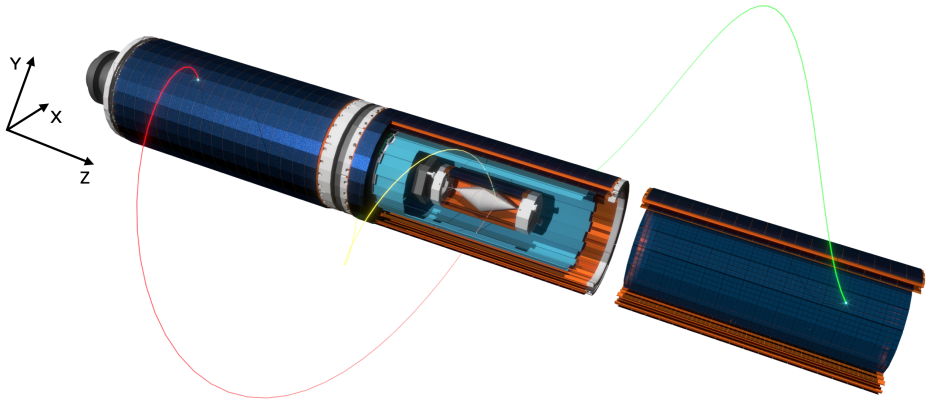

Figure 2.7: Three-dimensional render of the Mu3e detector [14].

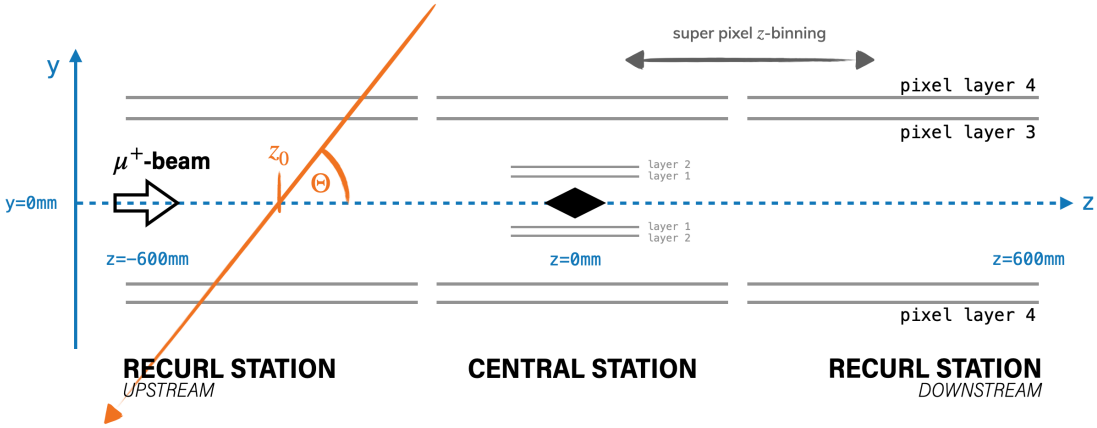

Figure 2.8: Schematic cut along the beam line, showing the y,z-plane.

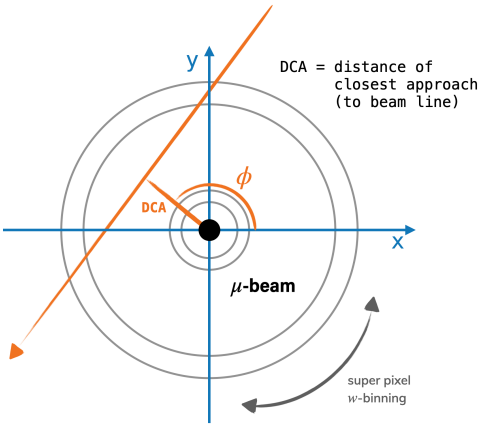

Figure 2.9: Schematic cut transverse the beam line, showing the x,y-plane.

2.4.4 Monte Carlo Cosmic Simulation

The cosmic muon simulation is based on the physical parametrisation described by [19], which was developed for the cosmic generator at the *CMS*. The Monte Carlo Simulation produces cosmics with a flat distribution of hits in the reference plane $y = 0$ in the area $z \in [-1\,\text{m}, 1\,\text{m}]$ and $x \in [-20\,\text{cm}, 20\,\text{cm}]$ (the coordinate system used in the simulation is described in Section 2.4.3). The cosmics are then moved 1 m back in the opposite direction of their momentum and are subsequently released.

Figures of Merit of Cosmic Simulation

The following plots show the distribution of track parameters of reconstructed cosmic tracks from the simulation. The parameters are described in Section 2.4.3.

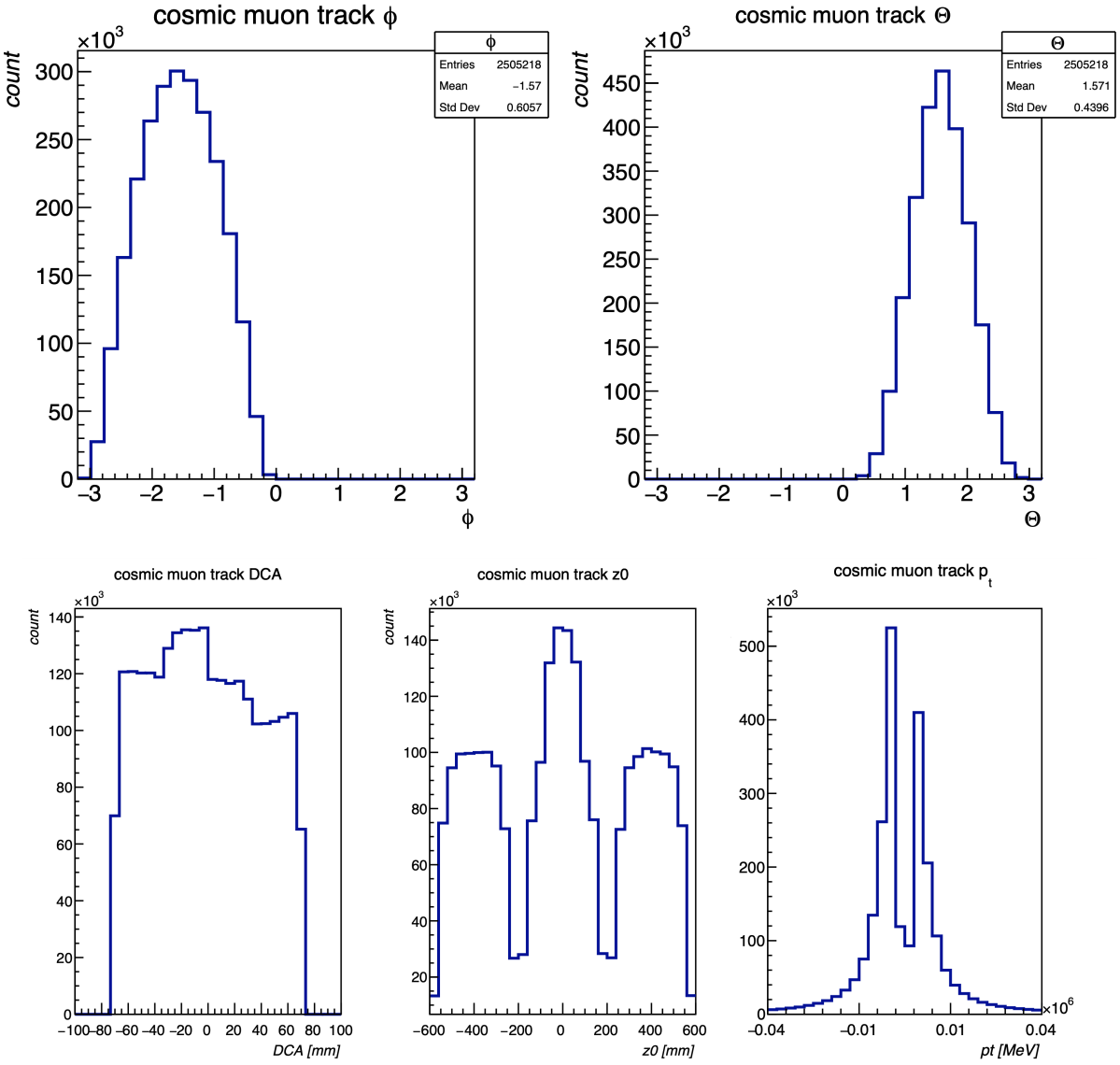

Figure 2.10: Track parameter distributions of reconstructed cosmic muons.

Part II

Methodology

Chapter 3

Pattern Recognition

In its essence, the term *Pattern Recognition* (PR) describes methods to find certain "structures" and regularities in data. An obvious application is in image analysis, when searching specific structures in an image. There are also less obvious use cases, for example in statistical data analysis. A modern technique that is connected to pattern recognition is machine learning, which in contrast to traditional pattern recognition, deduces features of the patterns on itself. The pattern recognition approach used for the *Cosmic Trigger* is rather conservative, as the format of the patterns will be designed manually in form of *Super Pixels* (regional clusters of pixels). The specific method of pattern recognition used for the Cosmic Trigger, uses specialised hardware, so called *Associative Memory (AM) chips*. These provide fast look up techniques for patterns stored in the memory and thereby enable high speed and low latency.

How pattern recognition will be used and studied at the *Cosmic Trigger* is described in the following sections. Additionally, the proposed hardware will be introduced briefly. In the following course of this thesis, the term *pattern recognition* will be used exclusively to refer to a hardware-focused approach using AMs if not denoted otherwise.

3.1 Applications of Pattern Recognition in High Energy Physics

In most experiments within particle physics, huge amounts of data arise at the detector and it is unfeasible to store all of it. Hardware pattern recognition can help to reduce the data rate in real time. Not only is pattern recognition capable of reducing the rate, but in particular of reducing the combinatorial effort for the track reconstruction (see Section 2.4.2). This is due to the detector hits being sorted into bunches in which they potentially belong to the same particle track. A reconstruction algorithm then only has to try all combinations within these bunches, instead of computing combinations of all the hits in the detector. As combinatorial tasks usually scale exponentially with the number of hits, this is a great save in computation effort.

The general approach of pattern recognition is used at several particle physics experiments at all sorts of detectors. The approach using hardware AMs, is rather special though. The first experiment that successfully implemented an PR trigger system based on AMs was CDF at Fermilab [20]. The first version of the *Silicon Vertex Trigger* was realised in the early 2000s and was later upgraded for higher luminosities which also included the development of new hardware. The most elaborate PR subsystem of the trigger at CDF featured a Monte Carlo generated pattern database containing 512K patterns. Eventually the system delivered the expected performance [21].

In the beginning of the 2010s the ATLAS experiment at CERN [22] also started to develop a AM pattern recognition system, the *Fast TracKer (FTK)* which was intended to become a key part of the *Trigger and Data Acquisition System (TDAQ)* for runs in 2020-2022. Its objective was to find and determine the momentum p_T of all tracks above 1 GeV [23] and to analyse

the data from the *ID inner tracker* which consists of silicon pixel sensors, similar to *Mu3e*. Generally, the concept at FTK is very similar to the one discussed for the *Cosmic Trigger*.

The high-luminosity upgrade at LHC (HL-LHC) which is planned for beyond 2026 will be a big leap for the ATLAS experiment with an estimated event pileup of 200 collisions per bunch crossing [24]. This imposes another level of demands on triggering and data filtering. Thus, an elaborate AM pattern recognition system is planned to be realised, the so-called *Hardware Track Trigger (HTT)*. In addition to reducing the band width from 3.2 Tb/s to 1 Tb/s, the system is also used to pre-fit the found particle patterns [24]. The *Cosmic Trigger* could make use of the hardware developed for HTT. It is described in section 3.3.2.

Concerning *Mu3e* there has been a study on using AM pattern recognition for efficient beam event reconstruction as well, which was carried out by Dohun Kim in 2018 [25]. The thesis examined how patterns could be designed on the *Mu3e* detector and if pre-fitting via such patterns would give a significant time and computation advantage, especially with phase II in mind. It was possible to reduce the combinatorial effort by a factor of 10. However, an implementation of pattern recognition for event reconstruction could not clearly be recommended, because of the cost-benefit ratio.

3.2 Pattern Recognition on a Pixel Detector

In order to perform pattern recognition, patterns need to be defined and specified. In the case of a pixel detector, a potential approach would be to just use pixels as pattern building blocks and pixel hit combinations, i.e. the tracks of particles themselves, as patterns. Problematically this fine-grained approach does not really reduce the combinatorial effort, as the whole combinatorics would just be calculated in advance. The calculation effort, as well as the required memory in the pattern recognition AMs would be gigantic. Also, the patterns are supposed to be determined by physical characteristics of particle trajectories and therefore must have a little bit of tolerance against multiple Coulomb scattering (MS) effects. Pixel patterns would not, because usually the pixel size is smaller than the effects of MS which is of magnitudes in the order of $\mathcal{O}(100\,\mu s)$. Therefore, a more coarse grained pattern design is required.

3.2.1 Concept of Super Pixels

By regionally grouping pixel to pixel clusters, so-called *Super Pixels* can be defined. This does not only reduce the amount of patterns needed (depending on the super pixel size), but is also more robust against multiple Coulomb scattering. A grid of of super pixels is defined to span over each layer of the detector. Every hit on a pixel can be associated to a super pixel by its (x,y,z) coordinates.

Super Pixel Mapping

One main subject to evaluate is the configuration of the super pixel mapping. It defines how the mapping is distributed over the detector, i.e. which pixels belong to each super pixel. This includes the size and form factor of the super pixels. The most basic parameters that define a mapping are the number of super pixel bins along the z-axis (beam direction) and the number of bins in the x,y-plane of the cylindrical pixel detector, the ϕ- or w-direction (see Section 2.4.3 for a sketch). Starting from these, more abstract parameters can also be studied, such as the *super pixel count (SPC)* which describes the granularity of a mapping and the *super pixel ratio (SPR)* which is defined as the number of w-bins divided by the number of z-bins.

$$\text{SPC} = N_{\text{wbins}} \cdot N_{\text{zbins}} \quad \text{and} \quad \text{SPR} = \frac{N_{\text{wbins}}}{N_{\text{zbins}}} \tag{3.1}$$

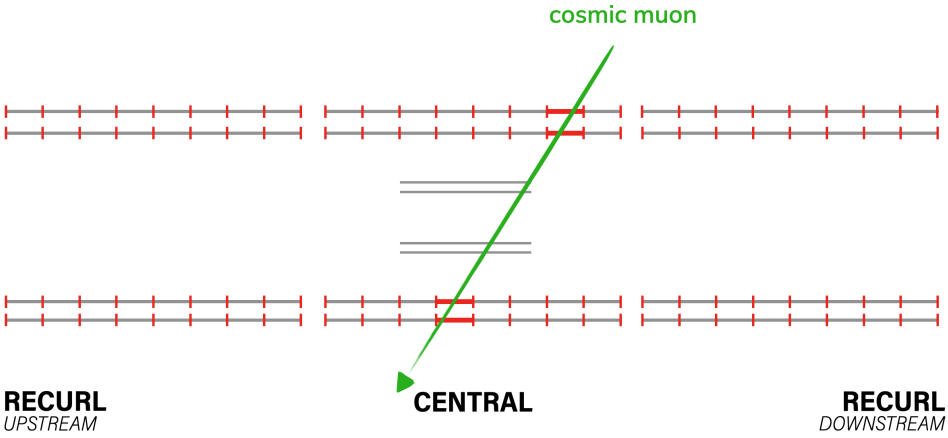

Figure 3.1: Schematic cut through the Mu3e detector. Red lines indicate super pixel boundaries. The hits of the cosmic muon traversing the detector result in a specific pattern of super pixels (thick red lines).

Where the number of bins N always refers to one layer of one detector stations. Subsequently, in order to obtain the total SPC, one must calculate

$$\mathrm{SPC}_{\mathrm{total}} = (N^{\mathrm{layer}}_{\mathrm{recurl\ down}} + N^{\mathrm{layer}}_{\mathrm{center}} + N^{\mathrm{layer}}_{\mathrm{recurl\ up}}) \cdot \mathrm{SPC} = (2+4+2) \cdot \mathrm{SPC}. \tag{3.2}$$

In this thesis a uniform mapping is chosen and the ϕ-coordinate of a hit is used to separate the w-bins. This means that all super pixels have the same length in z-direction and cover the same angle in ϕ-direction, thus being radially symmetric. However, modifications of this could be studied in the future.

3.2.2 Particle Roads and Super Pixel Templates

When a charged particle traverses the detector, it produces hits on the pixel sensors according to its trajectory. The individual pixel hits correspond to a super pixel respectively. The super pixels that were hit by a particle define a *road*, a coarse resolution track, through the detector. These roads, i.e. combinations of super pixels, are the patterns used for pattern recognition.

Usually, each *road* is accessible to more than one exact pixel hit combination (except when using single pixels as super pixels) (see Figure 3.2). A road therefore implicitly defines a small interval of momentum $\vec{p}$, and trajectory offset $\vec{x}_0$ that a particle taking this road can have. It is notable, that in terms of physics, a super pixel road defines a small subset of the 6 dimensional *phase space* (3 momentum and 3 spacial coordinates) a particle can be described in. The finer the super pixels are, the less potential particle tracks are allowed on each road, thus the smaller this subspace becomes.

What is tried at the *Cosmic Trigger* is to find a set of roads that describes a sufficiently high amount of possible cosmic muon tracks. These can then be used to search the detector data for their occurrence. At the same time it is intended to minimise the number and granularity of these roads, so that the chances of random combinations of non-cosmic hits populating one of the roads is reduced. Basically this means that the subset of phase space which can potentially be occupied by cosmic mouns is aimed to be described by cosmic roads, while keeping the total size of the described subset as small as possible.

In order to perform a feasibility simulation study, one must move from the rather abstract term of *super pixel roads* to a more technical implementation of it. At the *Cosmic Trigger*, super

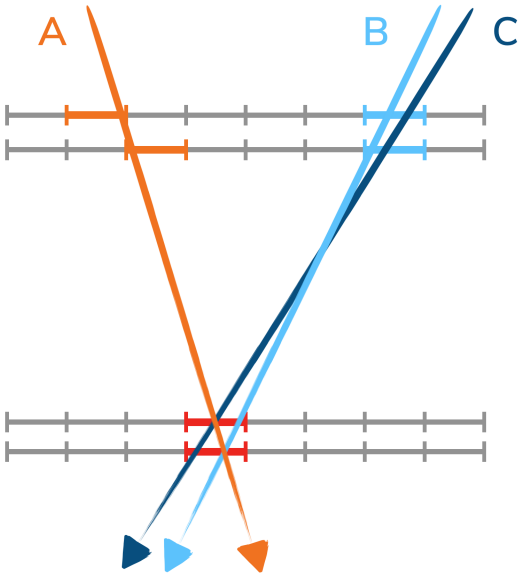

Figure 3.2: Three cosmic muon tracks. Two taking the same super pixel road (*blue*) and another one taking a different one (*orange*). The super pixel hits they all have in common are indicated in *red*, the individual ones in their corresponding color.

pixel roads are explicitly described as *Super Pixel Templates*. A super pixel template represents a pattern of multiple super pixel hits. According to the detector geometry of Mu3e, a template consists of 4 super pixels at the *Cosmic Trigger*. Within the study, the inner tracker layers were ignored and only hits in the outer layers were used for the creation of cosmic templates. In software, they are described with certain constrains. Cosmic muons approach from above the detector and have very large radia in the B-field. They traverse the detector in an almost straight line, leaving two hits on each layer they cross. Therefore a cosmic template is described by four super pixel hits ordered by layer following the chronological order of hits. The first super pixel that was hit is in upper layer 4, the second in upper layer 3, then lower layer 3 and 4 respectively.

3.3 Implementation in Hardware

3.3.1 Associative Memory

An *Associative Memory* (AM) (or Content-Addressable Memory) is a special category of computer memory, in contrast to a Random Access Memory (RAM) for example. Conventional computer memories, such as RAMs, work by receiving an address of a memory cell and returning the data stored at this address. An associative memory is content-addressable. The functionality of it shows similarities to a human brain, where data is stored by association, rather than by memory location. Content-addressable means that instead of receiving a memory address, it receives data and searches its entire memory for the cell it belongs to. The content itself serves as address and consists of a few "data words" that follows some format constrains. Usually, data words of a few bits in size are used for indexing a specific cell in the AM. Due to the search process being hard-wired, the architecture of an AM is more complex than of a RAM. One can imagine a AM functioning like this: When a call is put on it, a hardware circuitry switches on each data word that is present in the call. If a memory cell exists, whose word connections are all "fired" by the call, the corresponding memory cell is activated and returned. In this way the

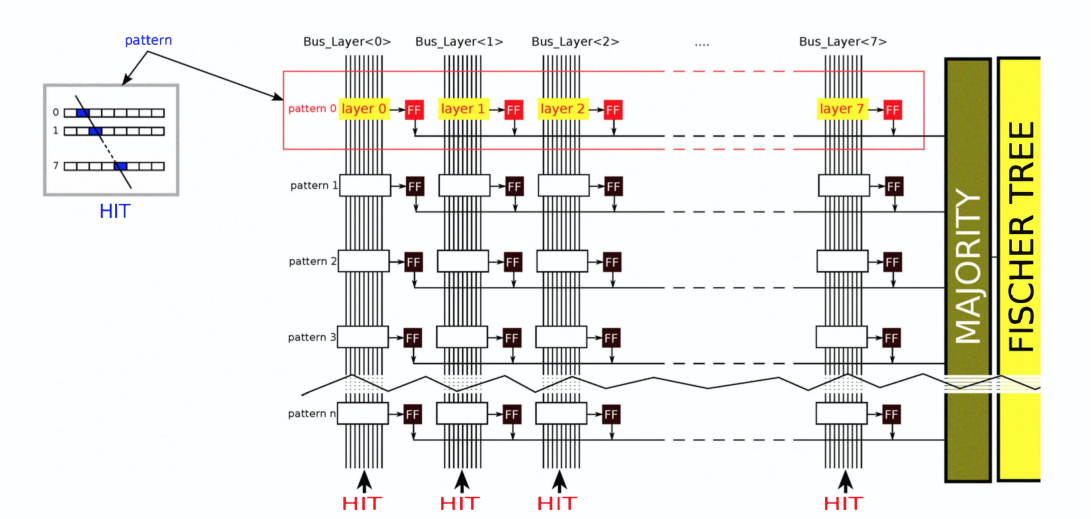

Figure 3.3: Architecture of an associative memory (AMchip06) developed for the PRM boards [27]. Each line represents one pattern, i.e. a cell in the memory. The circuitry hard-wires each super pixel *hit* input with the patterns it is part of. As soon as all super pixel hits of a pattern fire, then the whole pattern "fires". This diagram shows an implementation with patterns consisting of 8 hits each.

entire memory is searched in one single call which makes it highly superior to a RAM, when it comes to searching [26]. A circuitry scheme of an AM developed for the ATLAS experiment is depicted in Figure 3.3.

AMs are often used in network devices that quickly have to look up ports and addresses of other computers in the network. In almost no time, entries in routing tables can be found. In computer networks, this task has to be accomplished by every switch or router in between client and server, so reducing the computational effort of this specific task can distinctly improve the latency.

3.3.2 Pattern Recognition Mezzanine Board

The *Pattern Recognition Mezzanine (PRM)* board is developed for the ATLAS [22] trigger system for the HL-LHC upgrade forseen beyond 2026. It combines the core functionalities and components needed for pattern recognition hardware trigger systems. The board includes associative memory ASICs and a modern Field Programmable Gate Array (FPGA) [28]. The former provide the associative memory for the pattern database while the latter can handle the computations such as associating pixel hits to super pixels or performing basic fits [28].

The latest concept of the PRM board includes a recent Intel FPGA and four AM blocks, each containing five *AMchip09* ASICs. This yields enough memory space for about 7.5 million 8-hit patterns per PRM board which can be expanded to about 15 million 4-hit patterns [29].

Chapter 4

Cosmic Trigger Simulation

A main part of this work is the implementation of a software that allows to study the feasibility of the *Cosmic Trigger* concept by using Monte Carlo simulation data. Therefore, a software equivalent of an *Associative Memory*, which is the key ingredient for hardware-based pattern recognition, has to be designed. Also, the patterns themselves and their basic building blocks, the so-called *super pixels* must be defined. To cope with these tasks, two software core modules were implemented. Firstly, the *Pattern Engine* handles the super pixel assignment. Secondly, the *Template Bank* simulates an associative memory and builds up a template database. Around these modules, a small analysis framework was developed, mainly consisting of two chain processes. How this software evolved, how it was designed and how the emerging challenges were solved, is shortly presented in this Chapter. It first gives an overview of the overall concept and thereafter introduces the software core modules along with their functionalities.

4.1 Module Overview and Analysis Chains

The developed analysis software consists of two main analysis chains as well as two core modules, as depicted in Figure 4.1. Both analysis chains produce different figures of merit, which help to evaluate the performance of the cosmic trigger. The core modules simulate the associative memory and pattern creation.

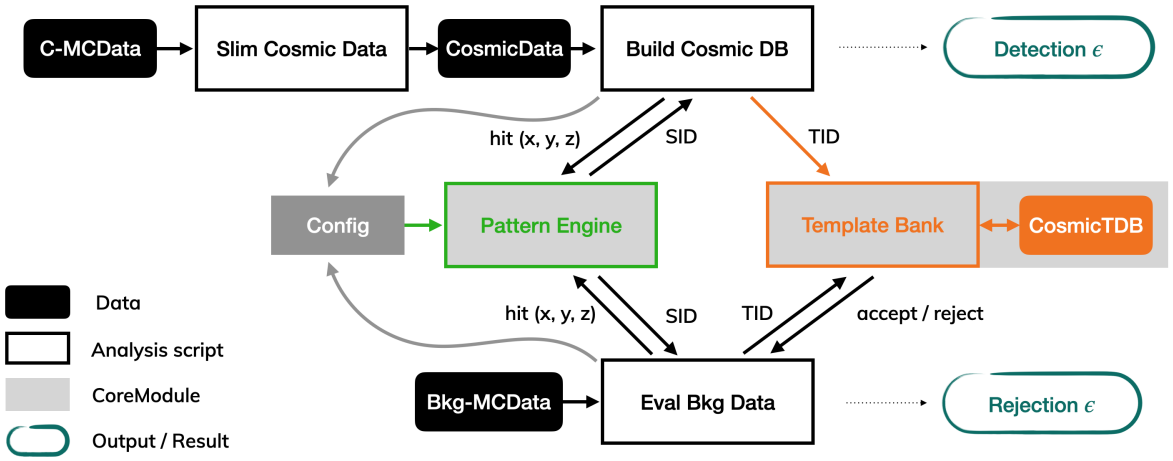

Figure 4.1: Schematic software component diagram of the two data analysis chains (*black* and *white*) and the core modules (*gray*) that simulate the pattern recognition hardware.

Nomenclature

SID refers to *Super Pixel ID* and represents a unique identifier for any super pixel on the detector.
TID means *Template ID*, which is the unique identifier of a cosmic road, i.e. the super pixels that correspond to the hits of cosmic traversing the detector.
CosmicTDB database file, containing the cosmic templates.

4.2 Building up a Template Database

The building chain (Figure 4.1 top row) uses Monte Carlo hit data of simulated cosmic muons on the Mu3e detector as input. The results of multiple runs of the simulation are slimmed in size and combined into one *CosmicData* file. Because a large amount ($\mathcal{O}(10^7)$) of cosmic tracks is needed in order to fill the database, this step reduces the data that must be stored long-term. For the second step, the *database training*, the *CosmicData* provides the hit information for the simulated cosmic muons. These hits are then assigned to their corresponding super pixel IDs, which is handled by the *Pattern Engine*. From the super pixel hits of a cosmic, a road (or template) is defined and added to the database by the *Template Bank* module.

In order to characterise the database coverage of possible cosmic muon tracks the *cosmic training efficiency* is continuously determined while adding templates to the database generated from MC data. $\epsilon_{\text{detect}}^{\text{cosmic}}$ is defined as the fraction of templates that were added, but already present in database (matched templates $N_{\text{matched}}^{\text{templates}}$) over total amount of generated templates. The latter can also be described as the sum of matched templates and new templates:

$$\epsilon_{\text{detect}}^{\text{cosmic}} = \frac{N_{\text{matched}}^{\text{templates}}}{N_{\text{matched}}^{\text{templates}} + N_{\text{new}}^{\text{templates}}} = \frac{N_{\text{matched}}^{\text{templates}}}{N_{\text{generated}}^{\text{templates}}} \tag{4.1}$$

Remark: Even if the label "training" indicates some machine learning affiliation, it only refers to profane filling of a database in this case.

In order to visualise the evolution of the *training efficiency* whilst adding more and more cosmic templates, this number is calculated for every bucket of 10^5 generated templates. This concurrent calculation also allows to stop the training process if a certain efficiency is reached.

4.3 Evaluation the Background Rejection

Once a template database is build, the crucial part of the analysis follows. First and foremost, this includes the observation of the background rejection capabilities of a certain super pixel and template configuration. As the term *background* in this case refers to everything that is of non-cosmic origin – including actual beam data in particular – Monte Carlo data of the "normal" experiment signal simulation (decays at the target) is used. For each hit in one of those "background frames", the super pixel is assigned. Then, every possible combination of super pixels is computed and associated to a template. The corresponding TID is sent to the *Template Bank* which will either accept or reject the TID, depending on whether or not already it exists in the database. The full combinatorics of super pixels are done within each 50 ns frame of the simulation. If none of the computed TIDs in a "background frame" match a cosmic template in the database, the whole frame is treated as rejected.

The most important figure of merit is the *background rejection efficiency* and is defined as the fraction of rejected frames over the tested frames.

$$\epsilon_{\text{reject}}^{\text{bkg}} = 1 - \frac{N_{\text{accepted}}^{\text{frames}}}{N_{\text{tested}}^{\text{frames}}} = \frac{N_{\text{rejected}}^{\text{frames}}}{N_{\text{tested}}^{\text{frames}}} \tag{4.2}$$

The background rejection efficiency implicitly defines the reduction factor of the frame rate that is passed on to the reconstruction to find potentially existing comic muon tracks. The higher $\epsilon^{\text{bkg}}_{\text{reject}}$, the less frames must be reconstructed. Therefore the frame rate reduction is defined as:

$$\mathcal{F}^{\text{rate}}_{\text{select}} = \frac{1}{1-\epsilon^{\text{bkg}}_{\text{reject}}} = \frac{1}{\text{false positive rate}} = \frac{N^{\text{frames}}_{\text{tested}}}{N^{\text{frames}}_{\text{accepted}}}. \tag{4.3}$$

Additional scripts

In the course of building and evaluating the database, different monitoring and benchmark plots are produced, which are mostly written to ROOT files for later review. There are several scripts that work up such plots. Additionally, some other scripts compare the results of different template databases with their different super pixel configurations and produce plots that visualise their quality.

4.4 Pattern Engine

The *Pattern Engine* handles the *Super Pixel Mapping* (SPM) on the detector. After initialising the mapping, its main functionality is to uniquely translate hits in the (x,y,z)-space into the *ID* of their corresponding super pixel. It was designed to be as flexible as possible in initialising the SPMs which facilitates the analysis of different mappings in the very end. The Pattern Engine is capable of using very different mappings with different resolutions and form factors. Furthermore, it is possible to use separate configurations for the three different *detector stations*. It was implemented in a way that even non-uniform super pixel mappings are feasible with some adaptions in the code.

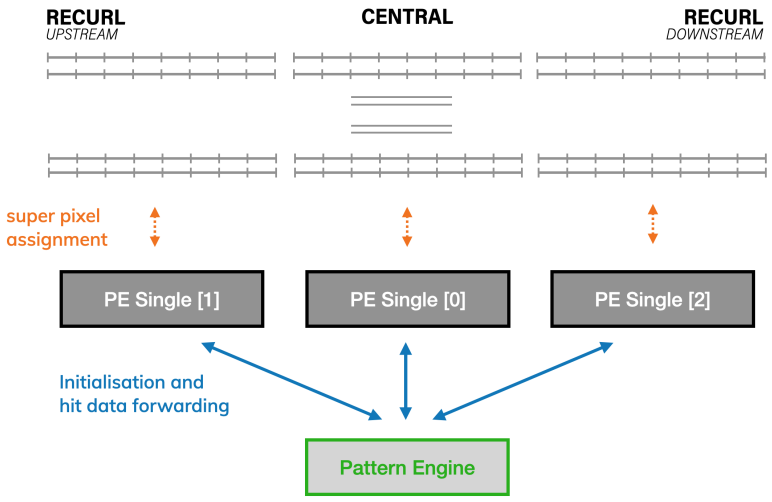

Figure 4.2: Components of the *Pattern Engine*.

The *Pattern Engine* module consists of three classes, namely *PatternEngineSingle*, *PatternEngine* and *SPCalculations*. The first two are derived from the latter, which contains some basic functions for super pixel calculations. Based on the detector architecture of Mu3e, the *PatternEngine* class has three *PatternEngineSingle* members, each handling one of the stations (center, recurl-upstream and recurl-downstream), as depicted in Figure 4.2. Theoretically, this could be adapted to other detector architectures. Also, for instance the center station could be excluded and only the *PatternEngineSingle* members for the recurl stations could be used.

4.4.1 Initialisation of Super Pixel Mapping

Characterisation and Building of a Mapping

A *Super Pixel Mapping* is created during initialisation of the *Pattern Engine*. Two parameters are needed for initialisation of a very basic uniform mapping, i.e. the number of super pixels in z- and in w-direction, where w stands for the generalised direction along x or ϕ. In the cosmic trigger, ϕ was used to define the binning along w-direction because of the radial symmetry of the pixel layers. However, x-binning could be studied in the future, since a binning planar projection would suit the cosmic distribution.

A mapping can be seen as a two dimensional grid overlaying each detector area. When unwinding the cylindrical detector, the binning can be shown on a plane as depicted in Figure 4.3. It can be characterised by the following set of numbers:

- `mode`
  Mode of Pattern Engine initialisation. In this thesis, only the default of `0` is used. This parameter could be used to switch to x-binning or to non-uniform super pixel bin distributions.

- `zBins`
  Number of bins along z-axis (beam line) of the detector.

- `wBins`
  Number of bins along ϕ-axis of the detector.

- `SPC` (Super Pixel Count)
  Number of super pixels on one detector layer of one area. The SPC is given by `SPC` = `zBins` · `wBins`.

- `SPR` (Super Pixel Ratio)
  The ratio of `wBins` to `zBins`. For example: w-bins = 40 and z-bins = 20 ⇒ `SPR`= 2 : 1 = 2
  Warning: This is not to be confused with a *super pixel aspect ratio*. This can only be defined for each layer individually because the size of w-bins scales with the radius, while the size of the z-bins does not. For non-uniform mapping, it would not even make sense to define a layer-specific *super pixel aspect ratio* as every super pixel could have its individual one. A super pixel aspect ratio is therefore not used to describe SPMs in this thesis.

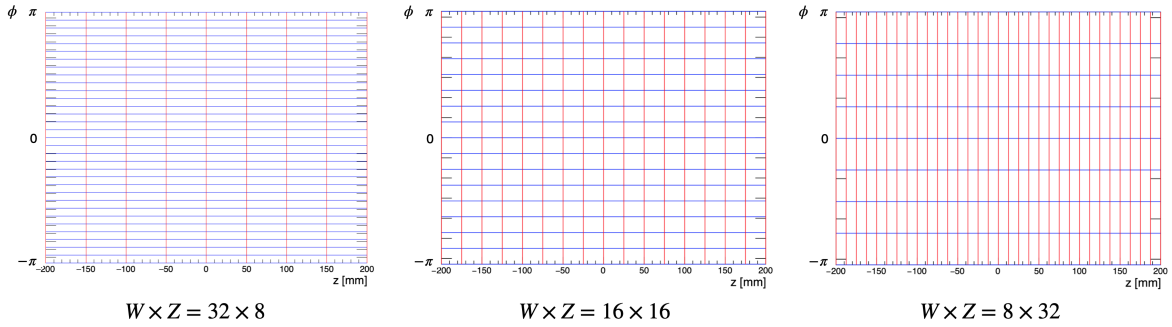

Figure 4.3: Different super pixel configurations for an SPC of 256. Three different SPRs of 4, 1, and 0.25 are depicted. The SPMs are $w \times z = 32 \times 8$ (stripe-like) (*left*), $w \times z = 16 \times 16$ (*center*) and $w \times z = 8 \times 32$ (ring-like) (*right*). Only the central detector station is shown.

4.4.2 Hit to Super Pixel Assignment

The *Super Pixel ID (SID)* is calculated by a member function of the Pattern Engine. It is made up by the parameters *area* (encodes the station), *layer*, *spWindex*, *spZindex* of a pixel hit. The last two are the 2D indices in the super pixel bin grid.

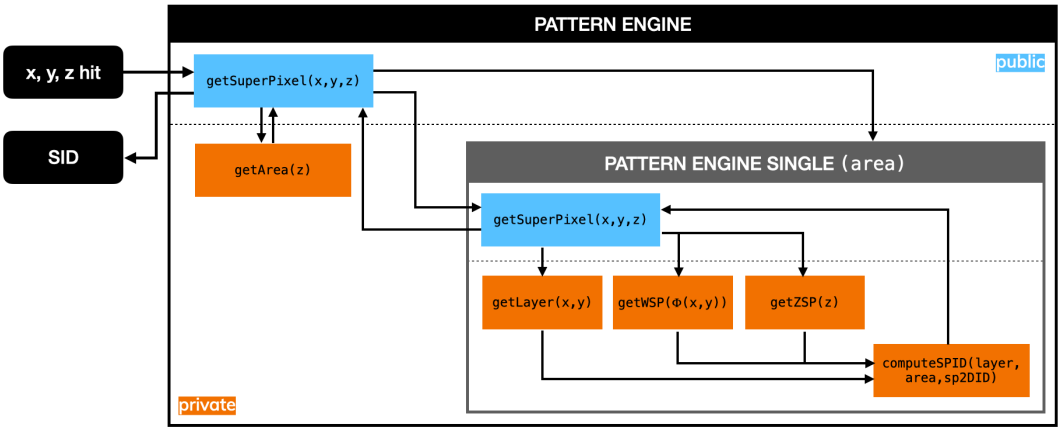

Figure 4.4: Calculation of the corresponding SID of a (x,y,z)-hit (some functions are combined to simplify the diagram).

A SID is encoded in a C++ `short` type, which has a length of 16 bits (or 4 hex words). The first 4 bits encode the area and layer information, which can be defined as *zone*

$$\text{zone}_{\text{hit}} = \text{area}_{\text{hit}} \cdot 4 + \text{layer}_{\text{hit}}. \tag{4.4}$$

Bits 5 to 16 encode a counting index in the two-dimensional super pixel bin grid, called *SP2Dindex*. As 12 bits can represent numbers up to 4096, this is the upper limit for the *Super Pixel Resolution*.

$$\text{SP2Dindex}_{\text{hit}} = \text{spZindex}_{\text{hit}} \cdot \text{wBinCount} + \text{spWindex}_{\text{hit}} \tag{4.5}$$

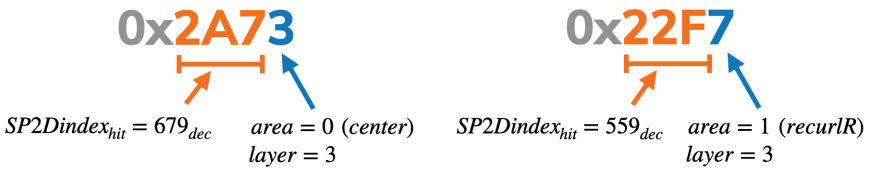

Figure 4.5: SID example in hexadecimal representation.

4.4.3 Monitoring Plots

Besides from SID assignment, an important feature of the Pattern Engine is keeping track of the *super pixel frequencies* (or super pixel weights). Each super pixel gets a specific entry in a counting vector. For each super pixel found from a pixel hit, its corresponding counter is incremented. The result can be visualised in a two dimensional histogram for each layer. Some of these are shown in Figure 4.6.

The super pixel weights could also be used to modify and improve the *super pixel mapping* in the future, for example, by splitting super pixels into two or four super pixels if they exceed a specific weight.

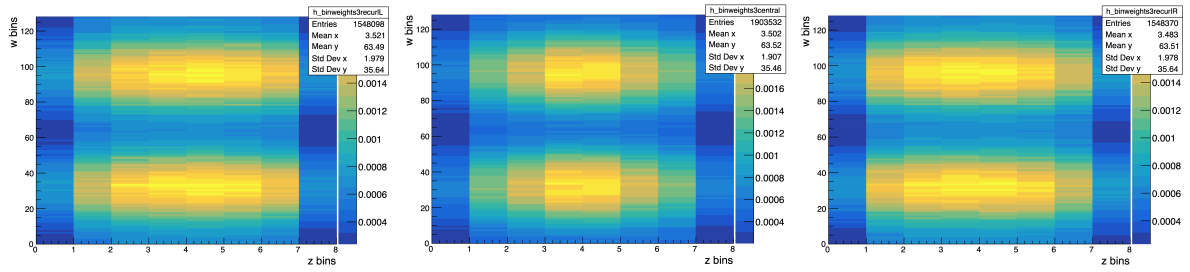

Figure 4.6: *Super pixel weights* for reconstructed cosmic tracks. The outer layer (4) of all three detector stations is shown for the super pixel mapping $w \times z = 128 \times 8$: the upstream recurl station (*left*), the central station (*middle*) and the downstream recurl station (*right*). Note, that more cosmics are reconstructed in the central station, therefore, the heat map does not have the same scaling for all three stations. The upper accumulation shows where most cosmics enter the detector and the lower where they leave. It can be observed that toward the station borders the super pixel weights decrease. This might be caused by the support structure in between, which allows cosmic muons to leave the detector without causing hits. Therefore these tracks are not reconstructable.

4.5 Template Bank

The main purpose of the *Template Bank* is to simulate an associative memory in order to allow for studying the feasibility in general as well as the required hardware capacity. Of course, simulating an associative memory basically removes its key advantages, namely the quick lookup technique. In software, this must be done on a "normal" CPU architecture, which includes sequentially searching for patterns in some data structure such as a list instead of looking it up in almost no computation time. However, the task of the Template Bank is by no means to beat an associative memory in time. The overhead of the computation is acceptable as the functionality itself is of major interest. How exactly the Template Bank was designed, which challenges needed to be tackled, and how it works, is described in this Section.

4.5.1 Simulation of an Associative Memory Chip

A detailed description of the associative memory along with its functionalities is provided in Chapter 3 of this thesis. As aforementioned, a data structure is required to simulate an associative memory as a database. As commonly known, searching is one of the computational tasks that can not be improved beyond a complexity of $\mathcal{O}(n \cdot \log(n))$ on average. For searching of a sorted list, the hardness is $\mathcal{O}(\log(n))$, whereby n denotes the number of elements in a container to be searched through. This adds a lot of computational effort on the simulation compared to the proper implementation in hardware of an associative memory whose complexity is $\mathcal{O}(1)$. A `std::map` is an associative container implemented in the C++ standard library. It uses a sorted self-balancing binary tree to ensure that each inquiry (adding and reading entries) has a logarithmic complexity of $\mathcal{O}(\log(n))$. This is quite some overhead, but still this data structure provides the fastest option to implement an *associative memory* on a CPU and therefore it is used in the *Cosmic Trigger Study*.

Secondly, an *associative memory* only needs SIDs as data input because the patterns (combinations of specific SIDs) would be implicitly activated if a certain combination corresponds to a valid address to a specific memory cell. An associative memory thereby does not need a specific order of SIDs as it just looks up *each and every* pattern on a bunch of data in parallel by its architecture.

For the simulation in software, a data format must be defined to characterise patterns in the template database, i.e. the list of all valid SID combinations that belong to a cosmic muon track. How this was done at the *Cosmic Trigger* is described in the following sections.

4.5.2 Template Data Format and Database Handling

As defined in Chapter 3, the elements of the *Template Banks* database are called *Super Pixel Templates*. They represent a road of a cosmic muon track, consisting of multiple Super Pixel IDs. The larger the super pixels, the more (slightly different) cosmic muon tracks can be represented by the same super pixel template.

When cosmic muons traverse the detector, they come from the atmosphere and thereby from above the experiment. Because of their high momentum of multiple GeV/c, their trajectories have large radia in the magnetic field. This implies that in most cases they hit every detector layer twice: the first time when they enter (positive y coordinate) and the second time when they leave (negative y coordinate). Therefore, the *Cosmic Trigger* uses super pixel templates that consist of four SIDs from the hits a cosmic caused in the outer tracker. As the inner tracker is very small and only detects a tiny fraction of cosmic muons compared to the outer tracker, it is ignored when building the super pixel templates. Also, the inner tracker has a much higher experiment signal rate, due to its installation close to the target, which leads to more background when observing cosmic muons.

The Super Pixel Templates can be identified by a unique *Template ID* or *TID* for short. The TIDs are implemented as a custom datatype that encapsulates the underlying data structure, a `short[4]` array, and some more functionality such as an ordering relation and the conversion of TIDs into a hexadecimal string. They are implemented in a way that allows to modify the *Template Bank* to deal with TIDs that contain 8 instead of 4 super pixels (or even 6). However, in the scope of this thesis only TIDs with four hits in the outer layers will be studied. Figure 4.7 shows how the TIDs are build.

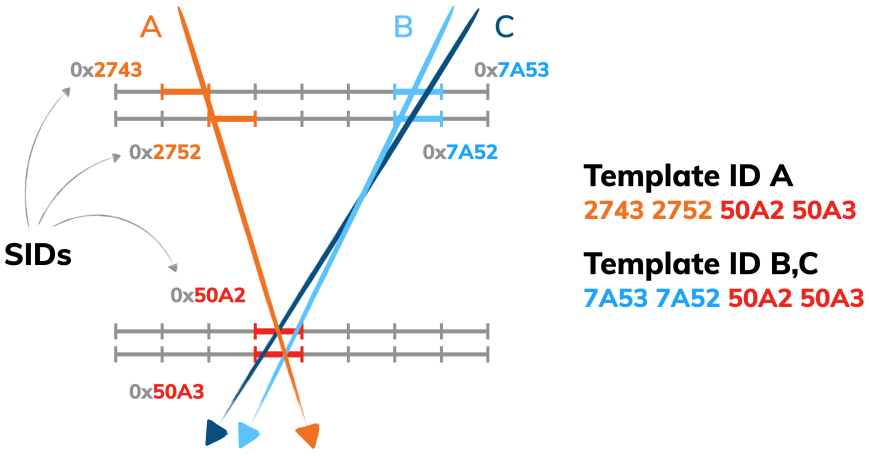

Figure 4.7: Different cosmic tracks and how they are represented as *Super Pixel Templates*.

The actual database is implemented as an associative container datatype. For each TID entry, the database contains further data. This covers the detailed track parameters, such as momentum p, radius r, Distance-of-Closest-Approach DCA of the beam line, z_0 along with the angles ϕ and θ. The track information of each cosmic that corresponds to one TID is appended when it is added to the database. Within the development of this thesis, this detailed meta data for each template was calculated and added, but never needed for the later analysis. In a future

stage of the study, the data could be used to determine the σ_p of each template, which could be used for some pre-fitting or categorisation or to cut on some templates to improve background discrimination.

For further studies, the *Cosmic Template Database* can be written to a file. Thereby, different template databases can be trained independently from the background data evaluation process. As the training of a database might well take several hours, this is an important functionality.

4.5.3 Efficiency Benchmarking and further Figures of Merit

Training efficiency

When filling the *Template Bank* with cosmic *TID*s, it keeps track of the *cosmic efficiency* defined in Formula 4.1. The $\epsilon^{\text{cosmic}}_{\text{detect}}$ is calculated over bunches of filled templates from *training events*. In the area between added training event 10^3 and 10^5, the step size is determined in a sort of decimal logarithmic way, such that $\epsilon^{\text{cosmic}}_{\text{detect}}$ will always be calculated every $10^3 th$ event between training events 10^3 and 10^4 and every $10^4 th$ event between training events 10^4 and 10^5. Beyond that, it will be calculated every $10^6 th$ event.

The values of $\epsilon^{\text{cosmic}}_{\text{detect}}$ versus the steps of training events are shown in one of the plots produced by the Template Bank. This plot illustrates how the fraction of recognised cosmic muons changes when adding more cosmic muon tracks. One example plot is shown in Figure 4.8. One can see that the graph grows faster in the beginning and subsequently saturates towards $\epsilon^{\text{cosmic}}_{\text{detect}} = 1$, which is the case if "every" generated template already exists in the template database.

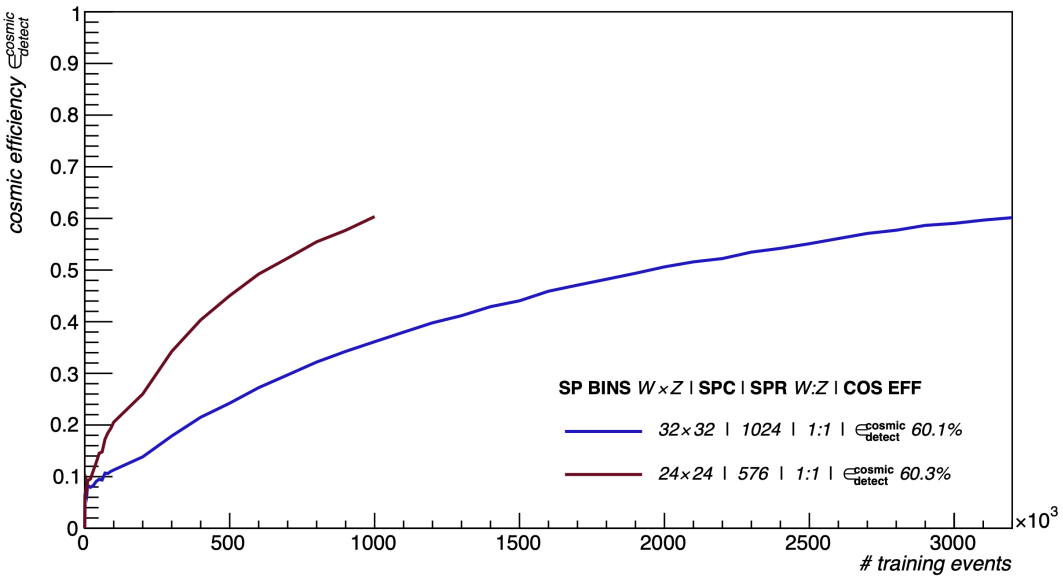

Figure 4.8: Combined plot for the training evolution of the cosmic efficiency for two different SP configurations. The training was terminated at $\epsilon^{\text{cosmic}}_{\text{detect}} = 60\,\%$. This plot is meant to illustrate the training process.

Template population histograms

A second important figure of merit is the determination of the most used templates and the *template frequency distribution*. Figure 4.9 shows a typical distribution, where the x-axis represents the template frequency of the templates (how often it occurred during training, also called "weight") and the y-axis how many templates exist with this frequency.

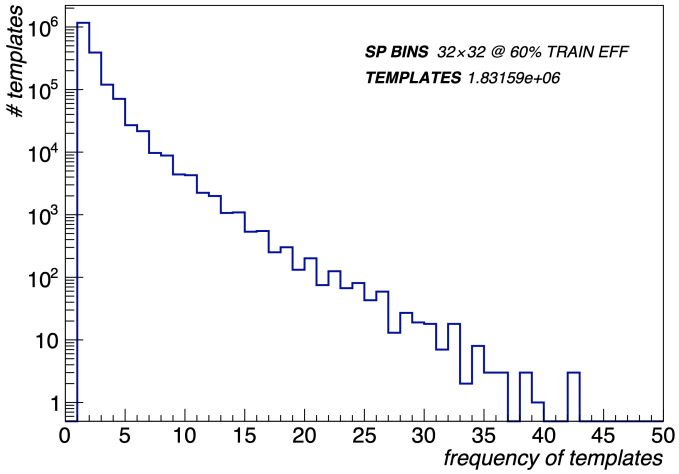

Figure 4.9: *Template frequency distribution* for w-bins = 32 and z-bins = 32 and a *stopping efficiency* of 60 %. The total number of templates in the database is about $1.8 \cdot 10^6$.

4.5.4 Geometry-based Template Categorisation

As it is the final goal of the *Cosmic Trigger* to filter out as many cosmics as possible while maximising the *background rejection*, it might be appropriate to think about further cuts and improvement methods for the cosmic template database. As aforementioned, not every cosmic track offers the same alignment quality and potential. One important difference between tracks is the region of the detector they traversed. Therefore, a *template categorisation* was implemented in the database. The List below and Figure 4.10 show the five defined regional TID categories:

- *Center - Center* (`CECE`)
  Cosmic muons that only traverse the central area of the detector. As the central area is where the target is located, this area has a very high background intensity. Additionally, there are different methods available to align the central detector, which reduces the importancy of *center - center* cosmics for alignment purposes.

- *Recurl Downstream - Recurl Downstream* and *Recurl Upstream - Recurl Upstream* (`RDRD` and `RURU`)
  Cosmic muons that only traverse either the downstream or upstream *recurl station*. Here, the background intensity is rather low. Those tracks are of high interest to use them for alignment.

- *Recurl Downstream - Center* and *Recurl Upstream - Center* (`RDCE` and `RUCE`)
  Cosmic muons that traverse the central area as well as one of the *recurl stations*. Those are the tracks of highest interest. Even if their rate is relatively low, they offer data to align *recurl* and *central stations* with respect to each other.

Now, every template can be associated to its specific template category. To make use of this categorisation, the whole template database can be filtered by different category settings, which exclude the corresponding TIDs from it. The *TIDLoadingFilter* can be activated when loading a template database from a file (during training the complete set of TIDs will be created). The possible settings are summarised in the following List:

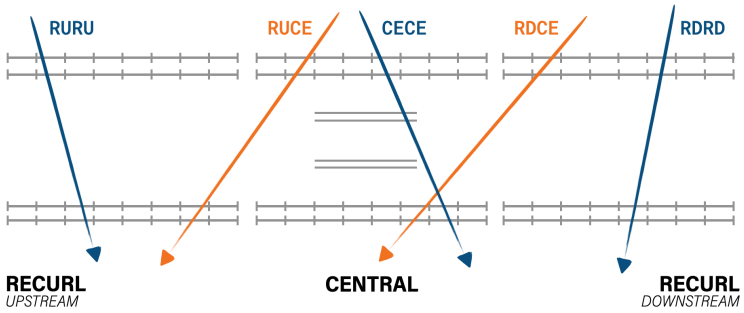

Figure 4.10: The five different cosmic Template categories. Mixed tracks are shown in *orange* and tracks only traversing one station are indicated in *blue*.

- `ALL`
  Every TID will be loaded. This is the default setting.
- `CENTER_ONLY`
  Only TIDs of the category `CECE` will be loaded.
- `RECURL_ONLY`
  Only TIDs of the categories `RDRD` and `RURU` will be loaded.
- `MIXED_ONLY`
  Only TIDs of the categories `RDCE` and `RUCE` will be loaded.
- `NO_CENTER`
  Only TIDs of the categories `RDRD`, `RDCE`, `RUCE`, and `RURU` will be loaded.
- `CUT_ON_FREQ`
  This filter setting loads TIDs of all categories, but only if their frequency is higher than 1. This is equivalent to excluding all templates that belong to the first bin in Figure 4.9.

Cosmic Detection Efficiency. Usually, $\epsilon_{\text{detect}}^{\text{cosmic}}$ is determined while building the database. This number is still of very high interest when using the *TIDLoadingFilter*; but, here it must be determined in a different way, because the post-filter efficiency will be different, than the one obtained during the training. $\epsilon_{\text{detect}}^{\text{cosmic}}$ is therefore determined by using a separate cosmic data set that was not used during training. Each cosmic muon TID in this data set is checked by the filtered Template Bank and either matched or rejected. The cosmic efficiency is then defined the same way as before in Equation 4.1. However, when using a filter, it is also of interest to observe the efficiency relative to the condition of the filter setting. This number is called the *Cosmic Acceptance*, which is defined by the fraction of number of accepted TIDs given by $N_{\text{accepted}}^{\text{cosmic TID}}$ over the number of tested TIDs that fulfill the *TIDLoadingFilter* condition $N_{\text{tested, allowed}}^{\text{cosmic TID}}$:

$$\mathcal{A}_{\text{filter}}^{\text{cosmic}} = \frac{N_{\text{accepted}}^{\text{cosmic TID}}}{N_{\text{tested, allowed}}^{\text{cosmic TID}}}. \tag{4.6}$$

$\epsilon_{\text{detect}}^{\text{cosmic}}$ and $\mathcal{A}_{\text{filter}}^{\text{cosmic}}$ can be read out from the Template Bank after checking a sufficient number of cosmics.

Part III

Results

The last and final part of this thesis describes the concrete outcomes and results of the *Cosmic Trigger* simulation. Its main goal is to provide an implementation recommendation. Summing up what was introduced in the past chapters, it is almost certain that cosmic muons are required for a precise alignment for the Mu3e experiment. In order to detect and reconstruct the muon tracks, two main approaches exist. On the one hand, it is possible to expand the existing GPU filter farm, which could then perform a triggerless reconstruction, as it is done for the beam decays. For the second approach, as studied in this thesis, a separate cosmic trigger can be used to pre-filter the frames from the detector and thereby reduce the data rate for the cosmic reconstruction. Eventually, with or without trigger, the cosmic muons must be reconstructed by a filter farm. When using a trigger, it reduces the rate of frames to be reconstructed which subsequently reduces the amount of required filter farm PCs. In practise that implies, that when achieving a frame rate reduction, i.e. a selectivity of e.g. 10, this could also reduce the required filter farm PCs for cosmic reconstruction by factor 10 (or a factor of the same magnitude).

In the very end, a simple cost calculation will determine which solution is the optimal one. A PRM board as well as one filter farm PC cost about 7k - 8k euros (using a pessimistic cost estimation based on [30] (PRM) and [31] (GPU)). If a trigger implementation using PRM boards is capable of reducing the reconstruction data rate for the filter farm by a sufficient factor, leading to a reduction of the overall cost, then this concept will be used. As an ambitious, but specific goal, a frame rate reduction factor of 10^2 to 10^3 is set. If and how this could potentially be reached will be investigated.

Chapter 5

Cosmic Trigger Performance Evaluation

The effort that went into the development of the *Cosmic Trigger* study software, as described in the last Chapters, was meant for one reason, namely to provide a framework to comprehend the actual quintessence of this thesis: the simulation and feasibility analysis of the *Cosmic Trigger* and furthermore to give practical recommendations towards the future implementation. The following Chapter is dedicated to presenting the conclusions that can be drawn and to derive a trigger performance from them. The challenge of this chapter is to deduce the dependencies in a multi-parameter system such as by mainly looking at individual two-parameter correlations.

The evaluation process itself is subdivided into three parts. First, the template databases are build up by using cosmic muon Monte Carlo data. This part is also called the *training phase*. The cosmic detection efficiency $\epsilon_{\text{detect}}^{\text{cosmic}}$ (cosmic efficiency in short) is analysed using different super pixel mappings. The main focus in this phase lays on observing the number of templates that are necessary to achieve certain benchmarks.

The second part focuses on the background rejection capabilities, rather than on cosmic detection efficiencies. The pattern recognition (PR) system will be analysed in dependency of different super pixel mappings and template bank configurations.

In the third and last part, some further improvement strategies and cuts that can be used to enhance the trigger performance will be examined.

5.1 Building up a Template Bank

The first part of the evaluation concerns the effect of different Super Pixel Mappings (SPM) on the cosmic muon detection rate, i.e. the size and shape of the super pixels. The main parameters to be studied are the super pixel binning in z-, as well as in w-direction. In this study w defines the binning along ϕ. Further relevant parameters for this analysis are the*Super Pixel Ratio (SPR)*, i.e. the ratio of w- to z-bins, and the *Super Pixel Count (SPC)*, i.e. the total number of super pixels on one layer in one station as defined in chapter 3. Therefore a certain maximum value for the cosmic detection efficiency ($\epsilon_{\text{detect}}^{\text{cosmic}}$) is set and different settings of these parameters are used for the training of a database. The training is terminated as soon as the maximum $\epsilon_{\text{detect}}^{\text{cosmic}}$ is reached.

5.1.1 Training for different Super Pixel Ratios

Figure 5.1 shows the evolution of $\epsilon_{\text{detect}}^{\text{cosmic}}$ during a training with Monte Carlo cosmic muon data, using different SPRs with a fixed SPC of 400. The training was stopped at a cosmic efficiency of $\approx 90\,\%$. The settings used are summarised in Table 5.1. Both extremes, meaning one-fold

super pixels stripes in z-direction, as well as super pixel rings (one w-bin), are included as well. Note that the super pixel z-size of 400 mm is the length in software. The actual active sensor area is slightly shorter, also depending on the layer (see Table 2.1).

Setting	w-bins	z-bins	w-size [rad]	z-size [mm]	sp area [rad·mm]	$\epsilon_{\text{detect}}^{\text{cosmic}}$	# templates	shape
1	400	1	0.0158	400	2π	90.6 %	$5.1 \cdot 10^5$	one-fold stripes
2	100	4	0.0628	100	2π	90.3 %	$6.2 \cdot 10^5$	four-fold stripes
3	20	20	0.3141	20	2π	90.0 %	$8.1 \cdot 10^5$	rectangles
4	4	100	0.5π	4	2π	90.1 %	$1.7 \cdot 10^6$	quarter rings
5	1	400	2π	1	2π	85.8 %	$6.7 \cdot 10^6$	full rings

Table 5.1: Super Pixel Mappings used in Figure 5.1.

In principle, those SPMs share the same granularity such that so every super pixel covers the same area, and one would expect them to perform similarly in terms of number of stored templates and $\epsilon_{\text{detect}}^{\text{cosmic}}$. However, Figure 5.1 clearly shows that they do not. What can be learned from this plot is that super pixels that have a stripe shape (longer in z-direction than in ϕ-direction) are significantly advantageous to those having a ring shape. Stripe super pixels need less templates to achieve the same $\epsilon_{\text{detect}}^{\text{cosmic}}$ than ring super pixels. Note that the x-axis in Figure 5.1 has a logarithmic scale. This behaviour continues even for the extreme cases of one-fold stripes (400×1 bins) when looking at higher $\epsilon_{\text{detect}}^{\text{cosmic}} > 75\,\%$. For training efficiencies below this threshold, the four-fold stripes (100×4 bins) perform slightly better. This is going to be discussed later.

Super Pixel Stripes vs. Super Pixel Rings To find a potential explanation for the phenomenon that stripes perform better than rings, the super pixel geometry has to be studied. First, the extreme case of one-fold super pixel stripes is considered, where each stripe covers the full z-length of one station. Two cosmics, whose trajectories only differ in the longitudinal angle Θ populate the same template as this template configuration is totally independent of Θ. The full super pixel rings (2π) work similarly for cosmics whose trajectories differ by the azimutal angle ϕ. A coordinate frame is given in Figure 2.8 and 2.9 or one can have a look at the angles in 5.2.

Given these considerations, one could assume that it is the distribution of the angles ϕ_{cosmic} and Θ_{cosmic} of Monte Carlo generated muons that biases the number of templates needed. If the distribution of Θ is significantly broader than in ϕ, super pixel stripes would have a higher "tracks-per-template" ratio and would therefore need less templates. When turning back to Figure 2.10, one can observe that the angular distribution of MC cosmics actually is broader in ϕ than in Θ. Subsequently, if this argument was true, the opposite effect should be observed. Therefore this argument can not be the explanation.

Another potential explanation is depicted in Figure 5.2, which shows a bias of the super pixels templates caused by the detector geometry. For super pixel rings (Figure 5.2 (*top*)), not every template is accessible to the same "amount" of cosmic muons, because the theta interval $\Delta\Theta_{\text{cosmic}} \equiv \gamma_L$ that is covered by one template depends on the track angle Θ itself. For rather flat Θ only the projection of the super pixel z-size in the particles' direction contributes to the template. Subsequently, templates that represent steeper tracks in Θ have a higher "tracks-per-template" ratio than planar templates. The size of the subset of phase space $\Delta\Pi$ that is represented by a template (see Section 3.2.2) depends on Θ for super pixel rings. On the other hand, super pixel stripes make use of the radial symmetry of the detector. The "tracks-per-template" ratio does not depend on ϕ as shown in Figure 5.2 (*bottom*). This means that the fraction of phase space a template describes is constant for different ϕ. In summary, more

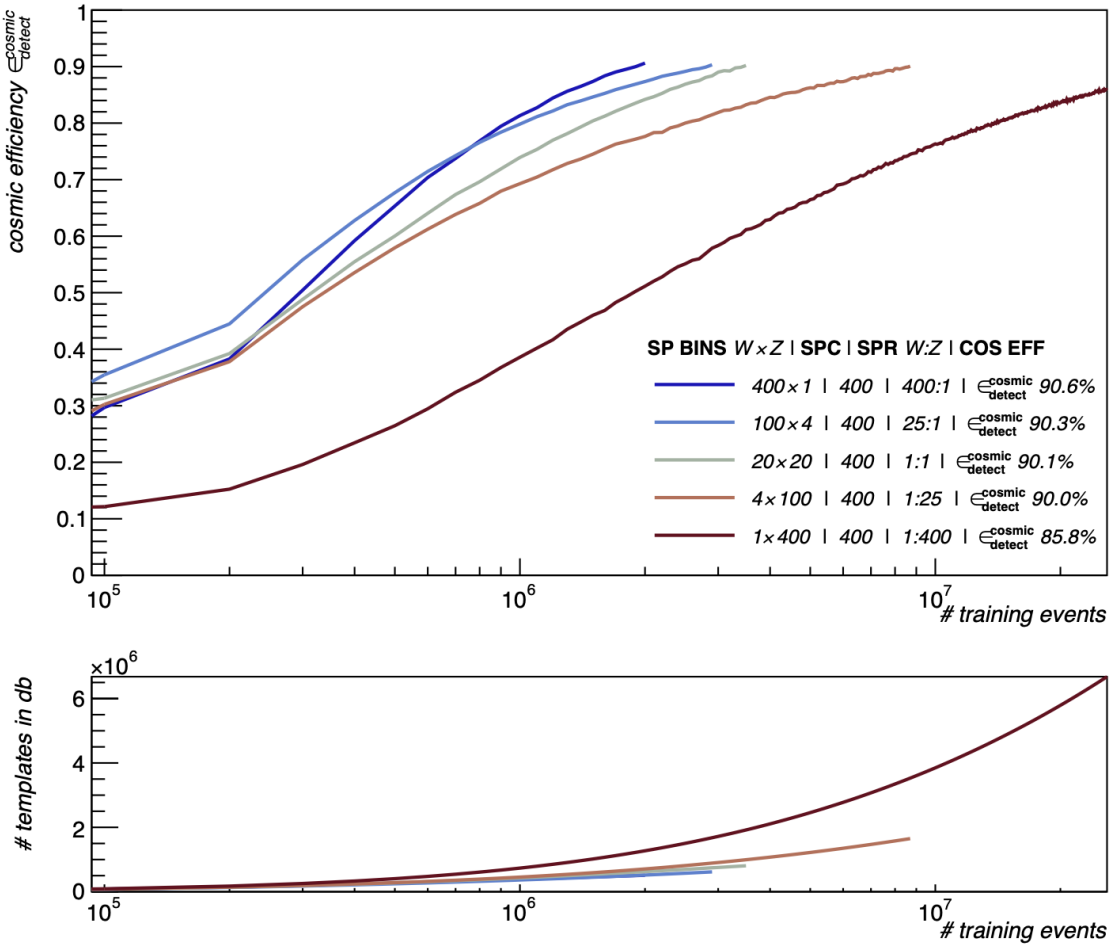

Figure 5.1: Cosmic efficiency evolution for different super pixel ratios plotted over the number of Monte Carlo cosmic tracks that were added to the database during training. The curves end at the desired cosmic efficiency, which in this case is 90 %.

templates are needed for more planar tracks when using super pixel rings. Super pixel stripes meanwhile describe the tracks of cosmic muons with a higher efficiency. It is assumed that this effect contributes significantly and is therefore capable to compensate for the ϕ and Θ distribution bias of the Monte Carlo simulation.

Limitation of Super Pixel Sizes Assuming SPC = 400 as in Figure 5.1, one could conclude that one-fold stripes offer the best performance in the training phase, when looking at $\epsilon^{\text{cosmic}}_{\text{detect}} > 75\,\%$. There is however a problem with this conclusion, as it might lead to unfeasible concept propositions. When designing SPMs, it is important to take into account that the detector potentially suffers from misalignment of a magnitude of several hundred micrometers to a millimeter as described in Section 2.3. The super pixel dimensions should be robust against misalignment effects, because otherwise the trigger performance could heavily suffer from it. Therefore a super pixel should not be smaller than a few millimeters in either direction. It is beyond the scope of this thesis to give a validated lower size limit for the super pixels. Therefore, a lower size limit for the super pixels of 2 mm in either direction is estimated. In z-direction, one detector station is about 400 mm long. This leads to a maximum of 400 mm/2 mm = 200 z-bins. In ϕ (w) direction the actual SP size depends on the layer radius. As four-hit templates are used, taking into account layer 4 and 3, their average radius of about 75 mm. This leads to a maximum of $2\pi \cdot 75\,\text{mm}/2\,\text{mm} = 251$ w-bins. For practical reasons, a feasibility maximum of

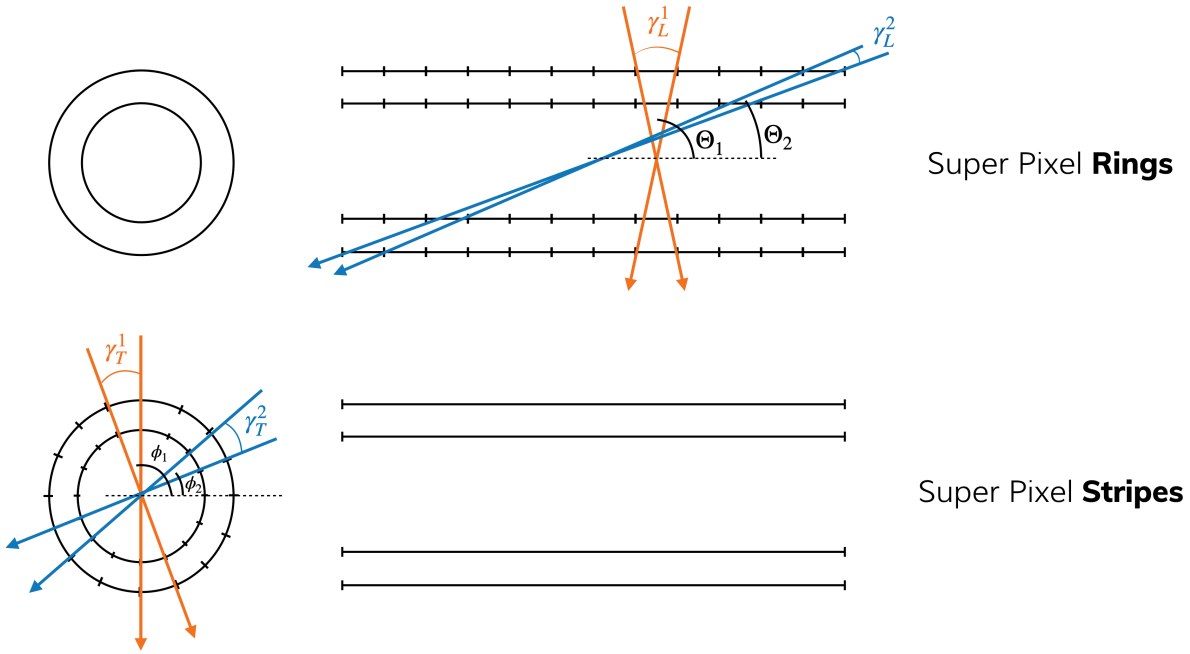

Figure 5.2: Geometrical effects on the fraction of phase space represented by a template. For super pixels rings (*top*) the size of this fraction depends on the azimutal track angle Θ, while for super pixel stripes (*bottom*) it is constant for different ϕ, assuming a fixed SPC. A potential explanation why ring architectures require more cosmic templates to realise the same $\epsilon_{\text{detect}}^{\text{cosmic}}$.

256 w-bins is assumed in the following study. In order to study some correlations it might still be interesting to expand this margin.

The exact effects of misalignment on feasible super pixel size should however be studied in the future. It might well be that the limit lies differently, also depending on the final effects on online misalignment. Previously known and determined misalignment could be taken into account when defining the pixel-super pixel dependencies.

Conclusion From Figure 5.1 a clear advantage of super pixel stripes compared to super pixel rings can be concluded for the extreme cases. However, for stripes in the upper SPR region of $SPR = 25:1$ and $SPR = 400:1$ a clear superior configuration can not be determined, also because the behaviour of the crossing of curves might look slightly different for other SPCs. In the following course of this thesis, SPCs larger than 400 will be studied. Subsequently, also higher ratios are accessible. Within the training phase, the slight deviations between SPRs in the area at about $SPR = 25$ will therefore be assumed to be negligible. A simulation result that supports this assumption is given in the Appendix. Generally, from now on the focus of the analysis will lie at super pixel stripe implementations.

5.1.2 Training for different Super Pixel Counts

The super pixel count (SPC) implicitly defines the size of super pixels and the granularity of the mapping. It is straight-forward to predict the impact of a higher super pixel count on the numbers of templates that are necessary to reach a certain $\epsilon_{\text{detect}}^{\text{cosmic}}$: a higher super pixel count leads to finer super pixels, which decreases the size of the phase space fraction a template represents. Therefore more templates are needed. How this correlates precisely is studied in this section. Furthermore, referring to the last section it is assumed that the effect of varying the SPR geometry is negligible for sufficiently high SPRs (see also in appendix). For this reason,

the super pixel z-binning is fixed at 4, so that each super pixel has a z-length of 100 mm. With this configuration, the SPC can be studied as a function of the w-binning, which seems to be the more relevant of the two binnings.

Database Training Figure 5.3 and Table 5.2 shows the training progress of $\epsilon_{\text{detect}}^{\text{cosmic}}$ over the number of training events for four different SPM configurations. The curvature of all four configurations tends to grow steeply at the beginning and then saturate to a value close to 100 %. The behaviour of $\epsilon_{\text{detect}}^{\text{cosmic}}$ and template count will be discussed in Section 5.1.3. In this Section, the focus is on the number of templates that are necessary to realise a certain cosmic efficiency with using a specific SPM.

Setting	w-bins	z-bins	w-size [rad]	z-size [mm]	SP area [rad·mm]	$\epsilon_{\text{detect}}^{\text{cosmic}}$	remark
1	512	4	0.0123	100	1.23	80.0 %	$w < 2$ mm
2	384	4	0.0164	100	1.64	80.2 %	$w < 2$ mm
3	256	4	0.0245	100	2.45	80.4 %	
4	128	4	0.0491	100	4.91	80.8 %	

Table 5.2: Super Pixel Mappings used in Figure 5.3

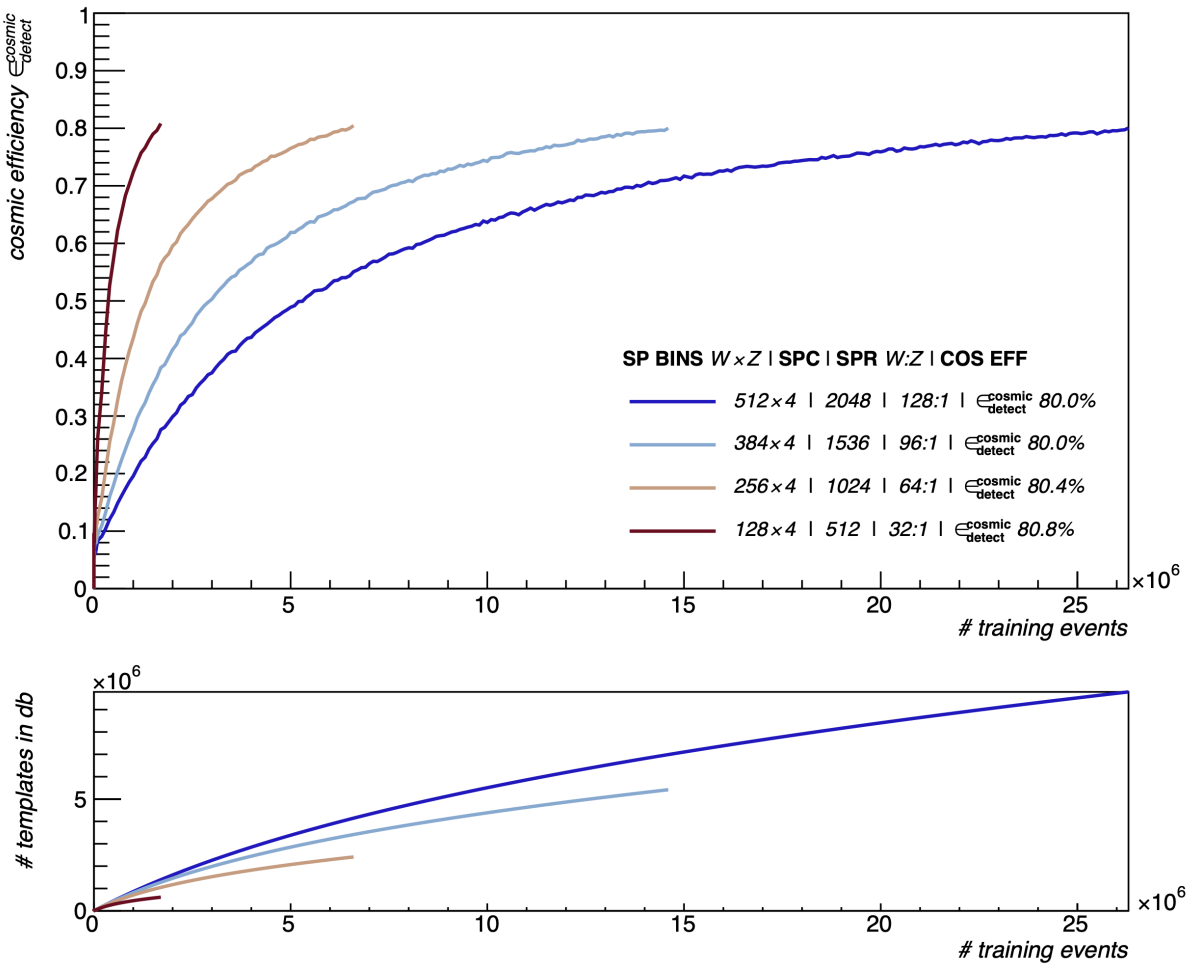

Figure 5.3: Training of database for different super pixel count (SPC) configurations using a stripe mapping with fixed z-size at 100 mm.

Number of Templates In order to evaluate the feasibility of the *Cosmic Trigger* and the associated hardware requirements, the number of required templates must be accurately estimated. Figure 5.4 shows the template count in relation to SPC for the configurations shown in Table 5.2 for a $\epsilon_{\text{detect}}^{\text{cosmic}}$ of 60 % and 80 % respectively. It is notable that the two highest SPCs simulated for $\epsilon_{\text{detect}}^{\text{cosmic}} = 60\,\%$ lie beyond the limit of 2 mm set for the minimal super pixel size of in either direction.

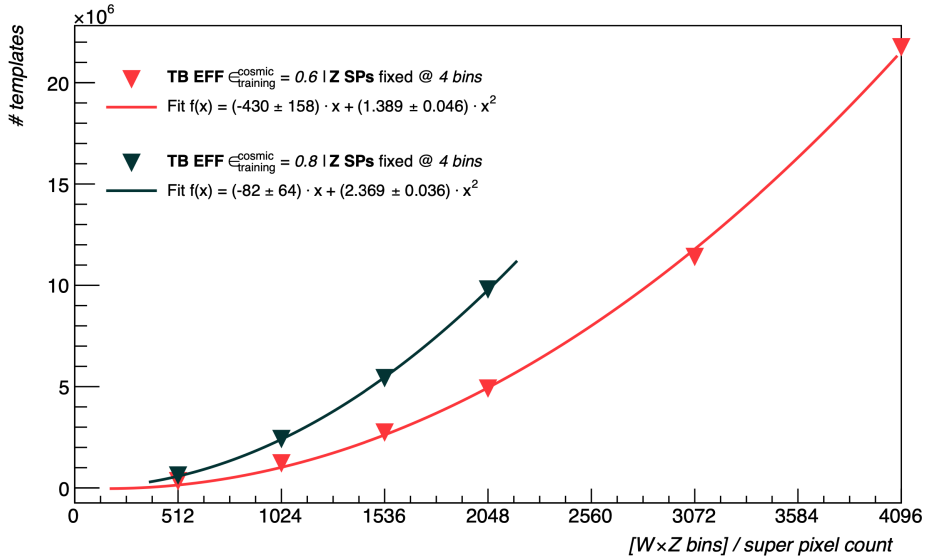

Figure 5.4: Number of templates that are required for different super pixel counts at fixed sp z-size to reach $\epsilon_{\text{detect}}^{\text{cosmic}}$ of 60 % and 80 % and their fitted functions.

In order to characterise the correlation, the plot was fitted with a second-order polynomial using an offset of zero, described by the formula

$$f(x) = ax + bx^2. \tag{5.1}$$

Higher-order polynomials were also studied, but the contributions of the higher order terms were negligible. Table 5.3 lists the parameters used for the fits.

Cosmic efficiency	a	b
60 %	-430 ± 158	1.389 ± 0.046
80 %	-82 ± 64	2.369 ± 0.036

Table 5.3: Fit parameter for Equation 5.1 in Figure 5.4.

It is interesting that a quadratic polynomial suits the data very well. A potential explanation can again be given by the geometry. Given the small scattering, the cosmic tracks can be approximately considered as straight lines. With this assumption, the direction of their linear trajectory has two degrees of freedom and is parameterised by ϕ and Θ. In the detector, these parameters are already determined by the first two of the four super pixels the muon hit. Hits 3 and 4 must occur in a closely restricted region on other layers, confined by the super pixels of hits 1 and 2. The combination possibilities on each layer scale linearly with the SPC. Because the number of templates mainly scales with the hits in two layers it is quadratically correlated to the SPC. The upper limit of the increase in templates when increasing the SPC is thereby given by the second order term and its coefficient because the first order term is negative.

It should emphasised that this observation was made by increasing the SPC by only modifying the w-bins, which are symmetric in their represented partition of phase space. For increasing

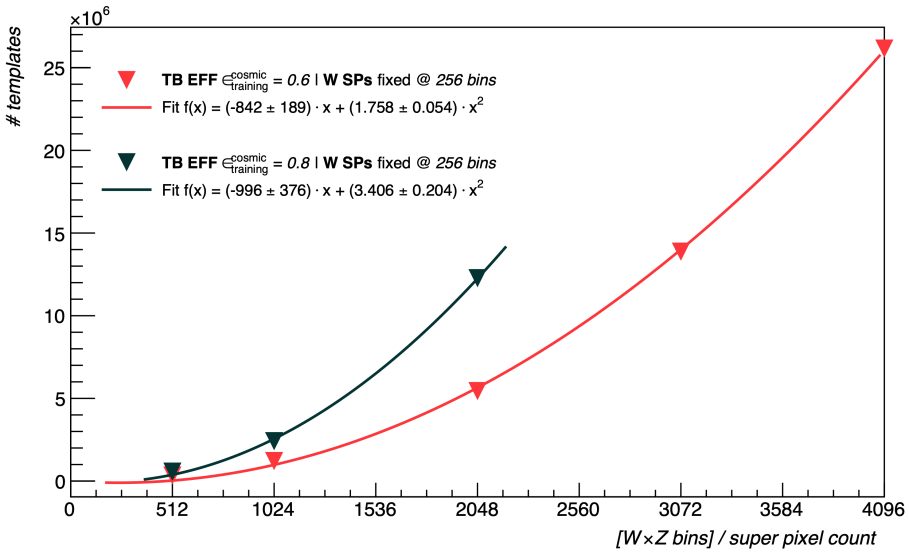

Figure 5.5: Number of templates that are required for different super pixel counts at fixed sp w-size to reach$\epsilon^{\text{cosmic}}_{\text{detect}}$ of 60 % and 80 % and fitted second order polynomials.

the SPC by changing the z-bins and keeping the w-bins fixed, it is expected that N_{tmpl} increases faster than for fixed z-bins in SPC, because the effects described in Figure 5.2 also play a role here. Figure 5.5 shows this correlation for a fixed w-binning of 256. Also, a second order polynomial is fitted and one can notice that the coefficients are significantly larger than for the fit for fixed z-bins, which supports the expectation. A higher order polynomial was not fitted because of the small number of data points and the missing explanation for higher terms.

Cosmic efficiency	a	b
60 %	-842 ± 189	1.758 ± 0.054
80 %	-996 ± 376	3.406 ± 0.204

Table 5.4: Fit parameter for Equation 5.1 in Figure 5.5

5.1.3 Cosmic Efficiency and Template Count

In the next step, the behaviour of the template count in relation to the cosmic efficiency will be studied. In the last section the cosmic efficiency was fixed to certain values to allow the studying of other parameters. Now, these parameters are fixed.

Again, the configurations from Table 5.2 are used and $\epsilon^{\text{cosmic}}_{\text{detect}}$ vs. the number of templates is studied. The resulting correlation is shown in Figure 5.6. The following exponential function was fit to the curves

$$\epsilon^{\text{cosmic}}_{\text{detect}}(N_{\text{tmpl}}) = 1 - a \cdot \exp(-b \cdot N_{\text{tmpl}}) \tag{5.2}$$

where N_{tmpl} denotes the template count. Provided that the fit shows a high accuracy, the inverse function can be used to estimate the template count that is necessary to reach a desired $\epsilon^{cosmic}_{detect}$:

$$N_{\text{tmpl}}(\epsilon^{\text{cosmic}}_{\text{detect}}) = -\frac{1}{b} \cdot \ln\left(\frac{1 - \epsilon^{\text{cosmic}}_{\text{detect}}}{a}\right) \tag{5.3}$$

It can be observed that the fit only roughly describe the data, as the points deviate from the exponential curve. The actual data shows a trend that is steeper in the beginning, but then approaches the 100 % slower than the basic exponential course of Equation 5.2 does.

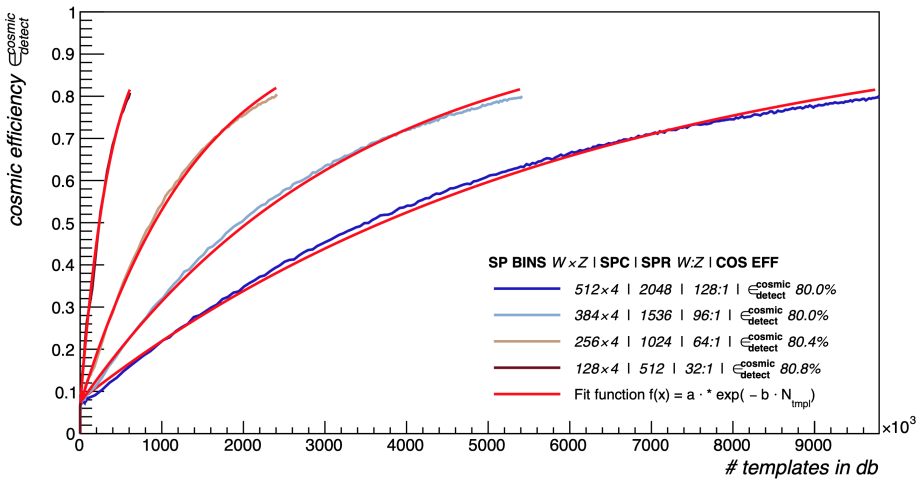

Figure 5.6: Cosmic efficiency vs. template count for the configurations shown in Table 5.2 with a fixed z-size of 100 mm. Fitted with Equation 5.2.

SP Bins	SPC	a	b
512×4	2048	0.9208 ± 0.0021	$(0.1650 \pm 0.0006) \cdot 10^{-6}$
384×4	1536	0.9273 ± 0.0023	$(0.3006 \pm 0.0013) \cdot 10^{-6}$
256×4	1024	0.9318 ± 0.0027	$(0.6844 \pm 0.0041) \cdot 10^{-6}$
128×4	512	0.9346 ± 0.0036	$(2.6700 \pm 0.0327) \cdot 10^{-6}$

Table 5.5: Fit parameter for Equation 5.2 in Figure 5.6

To provide a potential explanation one must look at the effects that play a role when creating the templates. In the most simple case, multiple Coulomb scattering is completely ignored. This would mean that the number of templates has an upper limit. At a certain point, the complete phase space that is accessible to muons will be described by the templates, thus leading the template data base to reach 100 % cosmic efficiency. It is assumed that this effect can be quite accurately described by Equation 5.2. In practice, however, another effect must also be taken into account, which is caused by Multiple Coulomb Scattering. Heavy MS has a significant impact on the accessible templates, as it changes the trajectories while passing the detector. Because it is a statistically distributed process, this suspends the finite template limit. To accurately model the curves of $\epsilon^{\text{cosmic}}_{\text{detect}}$ vs. N_{tmpl} the formula should actually contain two additive contributions, one describing the correlation without MS, which eventually saturates, and one modelling the MS dominated part. The latter would prevent the formula to eventually saturate, as there can always be one more template created by MS. However, for cosmic muons, the MS dominated contribution is expected to be almost negligible (compared to electrons for example). The small but noticeable deviation from the exponential fit in Figure 5.2 can yet be the result of MS, which would explain, why the fit slightly overestimates the data. Also, the higher $\epsilon^{\text{cosmic}}_{\text{detect}}$, the higher the proportion of templates caused by scattered muons that is added to the database. As for alignment, scattered muons are not very valuable, it could even be useful to stop at roughly $\epsilon^{\text{cosmic}}_{\text{detect}} \approx 90\,\%$.

Behaviour for Cosmic Efficiencies beyond 90 % In Figure 5.2 training efficiencies of up to 80 % were studied. It is worth to also look at what is happening beyond this limit. Depending on the SPC, the training process of a database often takes hours. Therefore looking beyond this limit will only be done for one of the configurations above. Anticipating some conclusion that will be found in the following section, the configuration from Figure 5.2 with $SPC = 1024$ is

chosen. It is trained up to the maximum $\epsilon_{\text{detect}}^{\text{cosmic}}$ that is achievable with the $\approx 4 \cdot 10^7$ Monte Carlo generated cosmics that are available. The fitted correlation is shown in Figure 5.7.

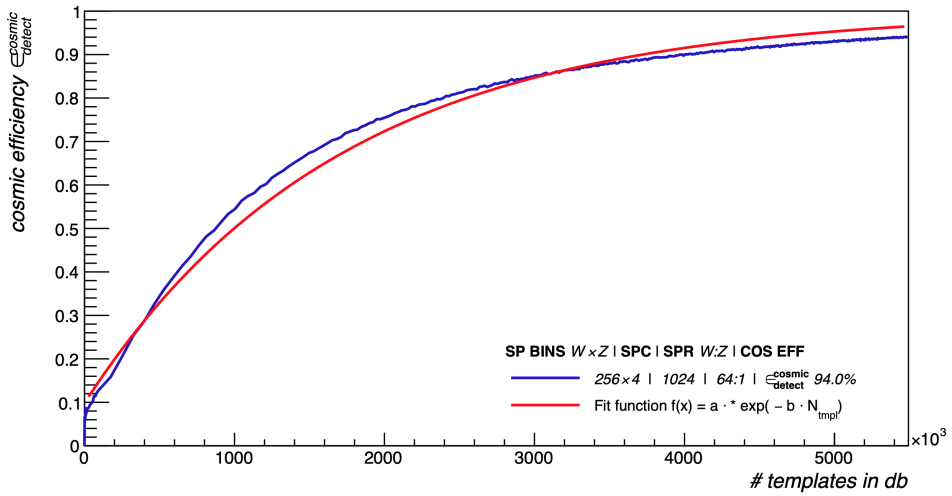

Figure 5.7: Cosmic efficiency vs. template count, fitted with Equation 5.2. Fit parameters are given by $a = 0.9016 \pm 0.0047$ and $b = (0.5909 \pm 0.0041) \cdot 10^{-6}$.

The observations from the fits in Figure 5.6 can be confirmed, when looking at Figure 5.7. Towards high template counts, the increase in $\epsilon_{\text{detect}}^{\text{cosmic}}$ is reduced, potentially by MS contributions and the partition of phase space covered by a template being dependent on the azimutal angle Θ as described in Section 5.1.1.

Conclusion The dependency of the template count on the desired cosmic efficiency can not be described with very high accuracy by an exponential limited growth. However, as a rough estimate, this relation delivers meaningful results. When using Equation 5.3 for estimating hardware capacities, one should keep a sufficiently high safety margin in mind.

5.2 Background Rejection Evaluation

A powerful trigger must be capable of detecting as many of the particles it searches for as possible. However, if the false-positive rate it too high, almost every frame would be stored anyway. A sufficient reduction of the data rate is therefore desired. In the second phase of the study we will have a look at the background discrimination capabilities of a template database, because this will describe the false-positive hit rate and the probability of triggering frames that are actually not of interest.

In the end, the background rejection will determine the trigger rate. In order to reach desired trigger rates that reduce the frame rate by a factor of at least 10^2 or 10^3, also referred to as *selectivity*, a background rejection efficiency of 99 % or 99.9 % is required respectively.

The background of the cosmic trigger are the hits caused by decays on the target. This means that the false-positive rate can be evaluated by using simulated frame data from the beam decay simulation. From the hits in each frame, templates can be created by computing the full combinatorics as described in Chapter 4. These templates can then be checked against the template database. The figure that will be studied is the background rejection efficiency $\epsilon_{\text{reject}}^{\text{bkg}}$, which is defined by the number of rejected frames divided by the number of tested frames, see Equation 4.2. The background rejection also satisfies $\epsilon_{\text{reject}}^{\text{bkg}} = 1 - \text{rate}_{\text{false positive}}$.

Remark on Background Data For the beginning, $\epsilon_{\text{reject}}^{\text{bkg}}$ will be studied using a simulation run that contains frames with one michel-decay each. One of these frames contains approximately 15 hits. By using this method, pattern configurations can be evaluated without a lot of pile-up in a toy study. Later, also beam rates up to 10^8 Hz, the planned final beam rate for Mu3e's phase-I, will be studied. (For reference: one decay in each 50 ns frame corresponds to a beam rate of $2 \cdot 10^7$ Hz.

5.2.1 Background Rejection for different Super Pixel Ratios

Figure 5.8 shows the background rejection efficiencies for different simulations using the same super pixel counts but different super pixel ratios. The different simulation settings used to create the data points are summed up in Table 5.6.

SPR	bins w	SPC	$\epsilon_{\text{reject}}^{\text{bkg}}$ (60 %)	$\epsilon_{\text{reject}}^{\text{bkg}}$ (80 %)
64	256×4	1024	98.3 %	97.8 %
16	128×8	1024	97.2 %	96.2 %
4	64×16	1024	92.0 %	89.7 %
1	32×32	1024	86.4 %	83.9 %
0.25	16×64	1024	85.2 %	83.3 %
0.0625	8×128	1024	85.5 %	83.7 %

Table 5.6: Simulated configurations used in Figure 5.8 and their background rejection efficiencies.

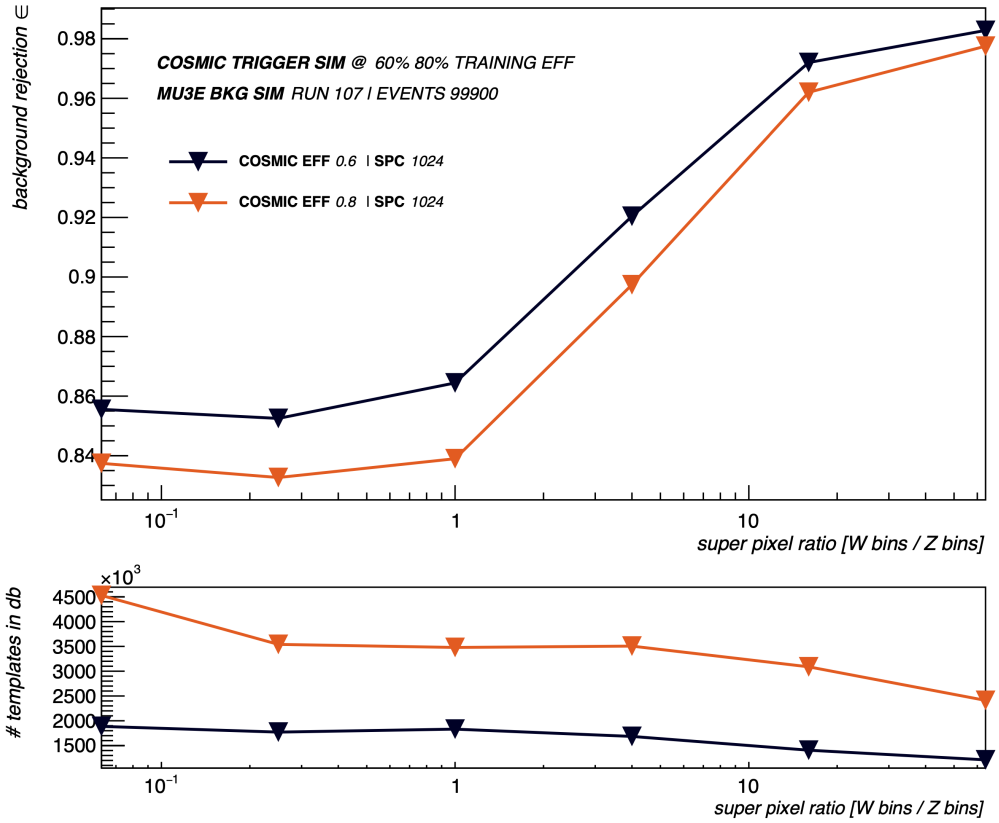

Figure 5.8: Background rejection for SPC=1024 and different SPRs.

The first conclusion that can be drawn from Figure 5.8 is that SPMs with higher SPRs, such as 128×8 or 256×4 (super pixel stripes) are superior in their background rejection compared

to SPMs with lower SPRs (rings). Secondly, it can be observed that above values of SPR = 16, the improvement with higher SPRs becomes less prominent.

The fact that higher SPRs provide a better background rejection is traceable to several factors. Generally it is due to the false-positive probability being influenced by different factors. The first is of statistical nature, namely that the number of templates has an impact on the false-positive rate. The more templates exist in the database, the to more possibilities there are, to hit one of them by randomly combining hits in one background data frame. These template matches occur "by chance" because some hits of (mostly different) particles coincidentally trigger one template. As studied in Section 5.1.1, a stripe SPM requires less templates to reach a certain $\epsilon_{\text{detect}}^{\text{cosmic}}$ than a ring SPM. Therefore a high SPR template bank has comparably less templates which reduces the chance of matching templates by chance. Subsequently, this improves $\epsilon_{\text{reject}}^{\text{bkg}}$.

However, one has to take into account that the data presented in Table 5.6 and Figure 5.8 was simulated with background frames that contain only one Michel Decay, which leads to one ionising particle in the final state producing hits int the detector layers. There are chances that decay or scattering processes produce more particles, but it can be assumed that the resulting template matches by combining hit of multiple particles are rather negligible. The dominating effect can again be explained geometrically and gives a very good argument for using super pixel stripes. The energies of particles produced at the target is limited to $\approx 53\,\text{MeV/c}^2$. In the B-field, these particles have radia of $\mathcal{O}(10cm)$, while cosmic muons have radia of multiple meters (using Equation 2.2). As the B-field is oriented in z-direction, the particles describe a curvature in the x,y-plane, along the w-bins. How broad the particle radius distribution for one template ist, is therefore given by the w-size of super pixels. Looking at super pixel rings, covering 360°, a electron (or positron) and a cosmic muon have the same template, as long as their azimutal angles and spatial offsets correspond. The discrimination power by the particles momentum is completely lost for super pixel ring templates. On the contrary, a fine w-binning can almost eliminate false-positive matching caused by a single decay particle. Here, one single decay particle can almost never use the same super pixel template as a cosmic muon, just because their momenta are so different.

Conclusion As a form of preliminary conclusion it can be stated that super pixel stripes offer the best cosmic efficiency per template count ratio as well as background rejection and are therefore suitable to be used for the *Cosmic Trigger*. As the finest w-binning feasible 256 was assumed, because for higher numbers the super pixel width would significantly fall below the limit of 2 mm in layer 3. For one-decay frames, the background rejection goal of 99 % could almost be reached with a $\epsilon_{\text{reject}}^{\text{bkg}}$ of 98.3 % at $\epsilon_{\text{detect}}^{\text{cosmic}} = 60\,\%$ without using any further cuts. This result corresponds to a frame selectivity of $0.6 \cdot 10^2$. How this develops for higher beam rates, other cosmic training efficiencies and SPCs are about to be studied in the following sections.

5.2.2 Background Rejection for different Super Pixel Counts

The next parameter that is subject to investigation is the super pixel count and its effect on $\epsilon_{\text{reject}}^{\text{bkg}}$. Again, the number of z-bins will be fixed and SPC studied by varying the number of w-bins. Two different z-sizes will be observed, namely 100 mm and 50 mm. The results are shown in Figure 5.9. Table 5.7 shows the values of the simulation outcomes. The maximum SPC configurations that are feasible ($w \leq 256$) with each z-bins are underlined. For a cosmic efficiency of $\epsilon_{\text{detect}}^{\text{cosmic}}$ =60 %, the simulation was performed up to SPC = 4096 in order to get an idea of the curvature that can be expected for other configurations.

Several findings can be derived from this plot. First, higher SPCs obviously yield a higher background rejection because of the higher resolution of each individual template. When doubling the SPC, the frame suppression rate also doubles, which is especially true for $\epsilon_{\text{reject}}^{\text{bkg}} < 98\,\%$

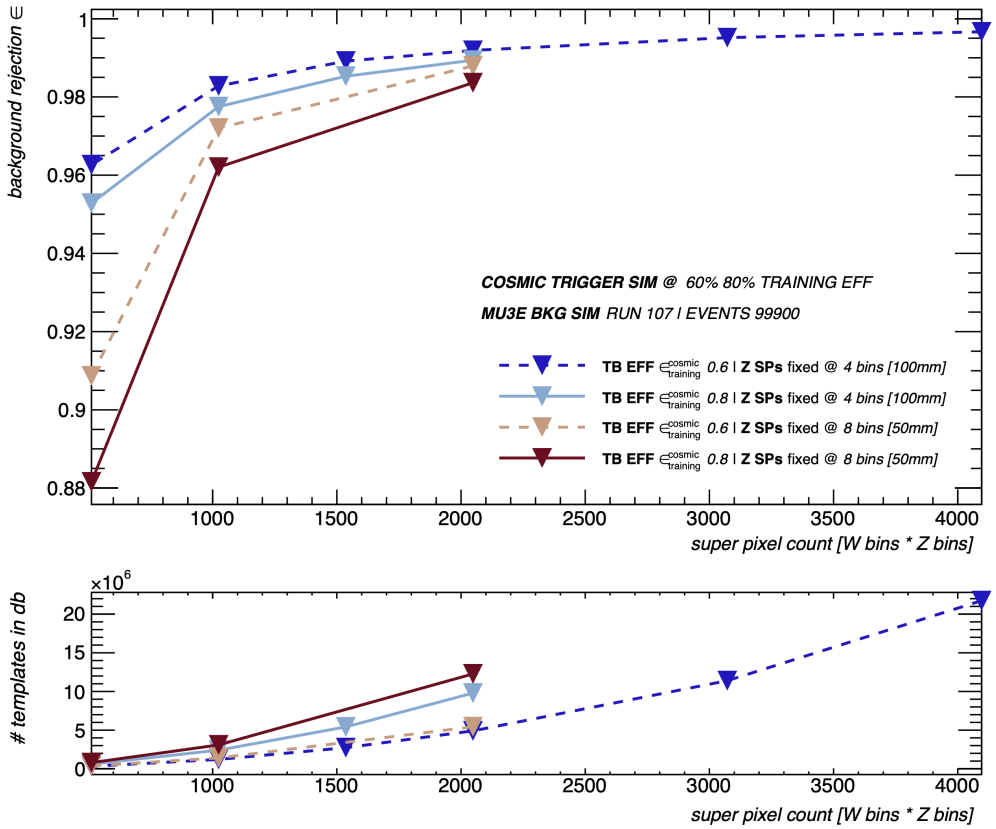

Figure 5.9: Background rejection and template count for a fixed number of z-bins and different SPCs.

SPC	bins WxZ	$\epsilon^{\text{bkg}}_{\text{reject}}$ (60 %)	$\mathcal{F}^{\text{rate}}_{\text{select}}$	$\epsilon^{\text{bkg}}_{\text{reject}}$ (80 %)	$\mathcal{F}^{\text{rate}}_{\text{select}}$
w-size 100 mm					
512	128×4	96.27 %	$0.3 \cdot 10^2$	95.28 %	$0.2 \cdot 10^2$
1024	256×4	98.28 %	$0.6 \cdot 10^2$	97.75 %	$0.5 \cdot 10^2$
1536	384×4	98.92 %	$0.9 \cdot 10^2$	98.53 %	$0.7 \cdot 10^2$
2048	512×4	99.19 %	$1.3 \cdot 10^2$	98.94 %	$0.9 \cdot 10^2$
3072	768×4	99.51 %	$2.0 \cdot 10^2$	-	
4096	1024×4	99.66 %	$2.5 \cdot 10^2$	-	
w-size 50 mm					
512	64×8	90.86 %	$0.1 \cdot 10^2$	88.16 %	$0.1 \cdot 10^2$
1024	128×8	97.20 %	$0.4 \cdot 10^2$	96.20 %	$0.3 \cdot 10^2$
2048	256×8	98.79 %	$0.8 \cdot 10^2$	98.36 %	$0.6 \cdot 10^2$
w-size 33 mm (only shown in 5.10)					
3072	256×12	99.03 %	$1.0 \cdot 10^2$	-	
w-size 25 mm (only shown in 5.10)					
4096	256×16	99.20 %	$1.25 \cdot 10^2$	-	

Table 5.7: Simulated configurations used in Figure 5.9 and 5.10 their background rejection efficiencies. The maximum SPC feasible $w \leq 256$ for each z-size is marked.

and the first configuration (z-size 100 mm, $\epsilon_{\text{detect}}^{\text{cosmic}} = 60\,\%$). Nevertheless, it must be said that for $\epsilon_{\text{reject}}^{\text{bkg}} > 99,5\,\%$ more background events would be needed to simulate this results with enough statistical evidence. The data in Figure 5.9 was determined with $\approx 10^5$ training events, which means that for $\epsilon_{\text{reject}}^{\text{bkg}} = 99.66\,\%$, 34 frames were matched. Within the scope of this thesis, this statistical uncertainty is acceptable. The error can be determined by $\sigma_{99.66\,\%} = \sqrt{N_{\text{matched frames}}} \approx 0.1\,\%$. In order to improve this significantly, the training events must be increased by an order of magnitude, which was too much computational effort for the scope of this thesis. In a future study it could be done for some few specific configurations or by using a cluster.

Generally, increasing the $\epsilon_{\text{detect}}^{\text{cosmic}}$ means that more templates are required, as discussed earlier. This leads to lower values of $\epsilon_{\text{reject}}^{\text{bkg}}$. It can be deduced from Figure 5.9 that increasing the SPC can be a means to counteract this effect. It can be clearly seen that $\epsilon_{\text{detect}}^{\text{cosmic}} = 80\,\%$ implementations perform systematically worse, than those at 60 %. When doubling the SPC, the $\epsilon_{\text{reject}}^{\text{bkg}}$ of the configuration at 80 % lies above the one with 60 % at the original SPC (for the shown simulation results).

When talking about doubling the SPC, one should keep in mind that this at least quadruples the template count. When adding more z-bins it grows even faster than that (see Section 5.1.2).

Fixed w-bins So far, mostly fixed z-bins were used to realise different SPCs. In Figure 5.10 the w-bins are fixed at the finest feasible size, leading to 256 bins. The simulation results that were used in this plot are marked in Table 5.7 (SPC = 512 not shown). $\epsilon_{\text{detect}}^{\text{cosmic}} = 80\,\%$ could only be reached for SPC ≤ 2048, because the amount of available Monte Carlo data.

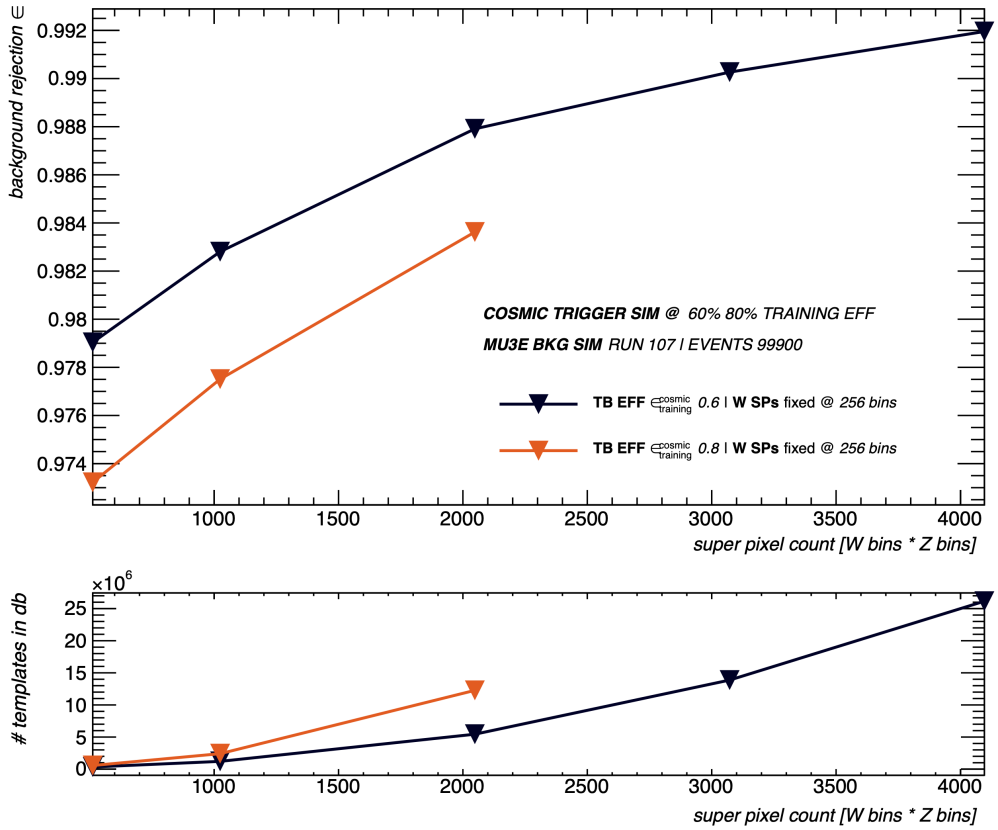

Figure 5.10: Background rejection and template count for a fixed number of w-bins and different SPCs.

What strikes first, is that SPC = 3072 with 256×12 bins is the first feasible configuration to reach the 99 % background rejection at $\epsilon_{\text{detect}}^{\text{cosmic}} = 60\,\%$, which corresponds to a frame selectivity of 10^2. Roughly extrapolated, the same configuration at 80 % could potentially reach $\epsilon_{\text{reject}}^{\text{bkg}} = 98.6\,\%$, which corresponds to a selectivity of $0.7 \cdot 10^2$. When observing how the frame rate suppression develops for higher SPCs for fixed w-bins, one can observe in Table 5.7 that they grow slower than for fixed z-bins. This observation is in accordance with the conclusions drawn in Section 5.1.2, which stated that the template count increases faster with z-bins than with w-bins.

5.2.3 Background Rejection vs. Cosmic Efficiency

One further correlation that was only implicitly studied in the previous sections is how the background rejection efficiency is related to the cosmic efficiency. When increasing $\epsilon_{\text{detect}}^{\text{cosmic}}$, more templates are required, which increases the false-positive matching chances and therefore decreased the $\epsilon_{\text{reject}}^{\text{bkg}}$. How large this effect actually is, is shown in the Receiver Operating Characteristic (ROC) curve in Figure 5.11 and Table 5.8 for three different SPC and SPR configurations that seemed promising in the previous analysis.

The simulation shows that $\epsilon_{\text{reject}}^{\text{bkg}}$ approximately scales linearly with the cosmic efficiency in the interval from $\epsilon_{\text{detect}}^{\text{cosmic}} = 40\,\%$ to $\epsilon_{\text{detect}}^{\text{cosmic}} = 80\,\%$. While increasing the cosmic efficiency by a factor of 1.6 (50 % → 80 %), the suppression rate is only reduced by about one third ($0.66 \cdot 10^2 \rightarrow 0.44 \cdot 10^2$ for SPC = 1024, other SPC show similar results). This is a key finding, as it means that it is possible to detect 60 % more cosmic muons, by storing 30 % more frames, when making the step from 50 % to 80 % cosmic efficiency. However, this behaviour changes for training efficiencies above 80 %. In particular this can be seen in the SPC = 512 curve, which decreases faster for higher training efficiencies. Figure 5.11 (*top*) indicates that the higher the SPC, the longer the correlation persists linear for higher training efficiencies.

	SPC	512	SPC	1024	SPC	2048
$\epsilon_{\text{detect}}^{\text{cosmic}}$	$\epsilon_{\text{reject}}^{\text{bkg}}$	$\mathcal{F}_{\text{select}}^{\text{rate}}$	$\epsilon_{\text{reject}}^{\text{bkg}}$	$\mathcal{F}_{\text{select}}^{\text{rate}}$	$\epsilon_{\text{reject}}^{\text{bkg}}$	$\mathcal{F}_{\text{select}}^{\text{rate}}$
50 %	96.68 %	$0.30 \cdot 10^2$	98.49 %	$0.66 \cdot 10^2$	89.99 %	$0.99 \cdot 10^2$
60 %	96.27 %	$0.27 \cdot 10^2$	98.28 %	$0.58 \cdot 10^2$	98.79 %	$0.83 \cdot 10^2$
70 %	95.88 %	$0.24 \cdot 10^2$	98.07 %	$0.52 \cdot 10^2$	98.61 %	$0.72 \cdot 10^2$
80 %	95.28 %	$0.21 \cdot 10^2$	97.75 %	$0.44 \cdot 10^2$	98.36 %	$0.61 \cdot 10^2$
90 %	94.39 %	$0.18 \cdot 10^2$	97.35 %	$0.38 \cdot 10^2$	-	-

Table 5.8: Background rejection efficiencies and corresponding selectivities for configurations shown in Figure 5.11.

It should be emphasised that these two Figures (5.11) are main results within this analysis as they show that the cosmic efficiency can be increased up to values of 90 %, while still providing acceptable frame selectivities. They provide the data that is required to decide for a working point based on the desired signal efficiency and selectivity and give an overview on performances that are accessible.

By inspecting Figure 5.11 is clear that even when using low values for the cosmic detection of $\epsilon_{\text{detect}}^{\text{cosmic}} \leq 20\,\%$, a frame selectivity of 10^3 is beyond what is feasible by simply increasing the SPC. One can observe that for very low training efficiencies, $0.5 \cdot 10^3$ can be reached, but this would also imply that only every *20th* cosmic muon would be detected. For performing online alignment, detecting at least every second cosmic muon would be desirable. Another fact to consider is that these simulations still were computed based one one-decay frames. About five decays per frame will be expected in Mu3e phase I with a beam rate of 10^8 Hz. Subsequently,

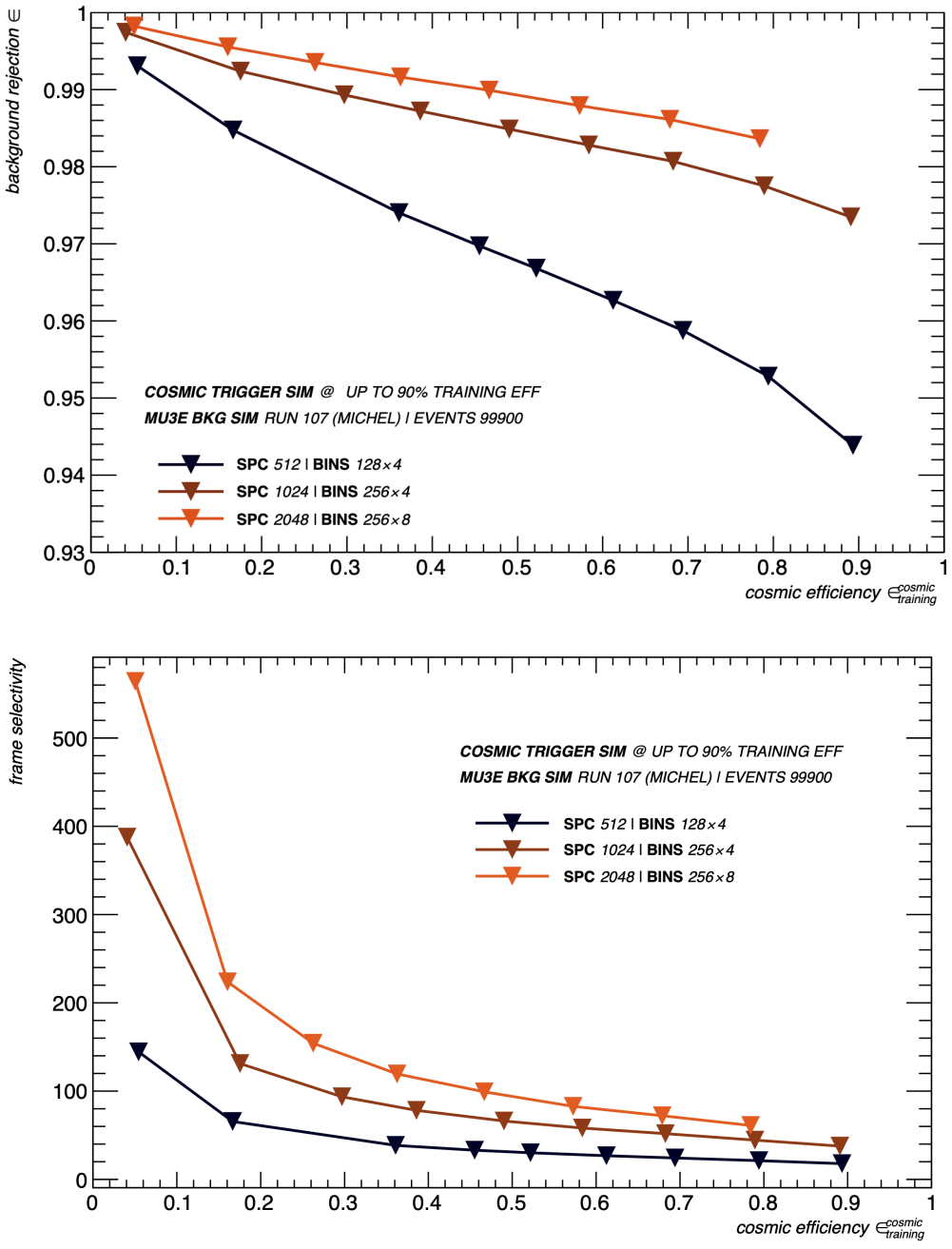

Figure 5.11: Receiver operating characteristics for background rejection versus cosmic signal efficiency (*top*) and frame rate suppression factor (*bottom*) for three different SPCs. $\epsilon_{\text{detect}}^{\text{cosmic}}$ was determined by probing the database with TIDs computed from a separate cosmic muon track simulation that was not used for training.

further methods to improve the background rejection must be taken into account in order to achieve the ambitious goal of a frame selectivity of $\mathcal{O}(10^3)$. This leads to the next Section.

5.3 Improvement Strategies and Cuts

Several methods to modify and to improve the pattern recognition configuration were discussed in the last Section. Pattern design parameters were studied such as the SPC and the SPR as well as the effects of the super pixel binning in general. It was observable that a reduction of the number of templates (also by limiting the cosmic efficiency at a certain point) led to improvements in the background rejection. In this Section, further methods will be evaluated that also reduce the number of templates. This will be done by either imposing geometrical selection rules to the template bank or by cutting off templates by the frequency they are "fired".

Template Types In order to classify templates, five categories were introduced, which are described in detail in Section 4.5.4. Figure 5.12 shows the distribution of these types within a template bank that was used in the previous analysis with a super pixel mapping of 256×4 and $\epsilon_{\text{detect}}^{\text{cosmic}} = 80\,\%$.

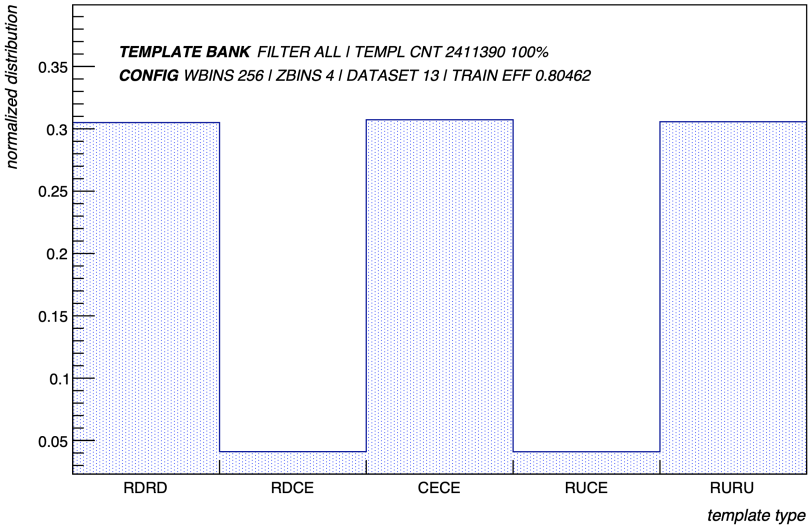

Figure 5.12: Distribution of template types for a template bank with $\epsilon_{\text{detect}}^{\text{cosmic}} = 80\,\%$. The categories stand for the detector stations a template includes, namely Recurl Downstream (RD), Recurl Upstream (RU) and Central (CE). Besides from including one station, templates can also represent tracks that traverse two different stations, so-called mixed tracks, represented by $RDCE$ and $RUCE$.

It can be observed that the templates are equally distributed over the three different detector stations. Per station, about 30.7 % of the templates are dedicated to tracks that caused all their hits in this station. Additionally, 8 % of templates represent cosmic muons that traversed the central and one of the recurl stations, 4 % on either side. The cosmic muon tracks corresponding to these 8 % of templates can be very useful for alignment, as they offer the opportunity to align the stations with respect to each other.

Template Bank Filter After introducing template types, these can be used to restrict the templates in the template bank, for example by excluding one of the categories shown in Figure

5.12. In order to study the effect of such exclusions, a filter functionality was introduced in the template bank. It has several settings that are described in Section 4.5.4.

In addition to the exclusion of templates by their type, also the frequency of a template can be used as a discrimination. It is expected that the template frequency is not equally distributed, but that there are some templates that collect a large amount of muons, while others are only triggered rarely. By cutting off the templates that were only populated once during training, it might be possible to reduce the templates in the template bank significantly while only loosing a few percent of cosmic detection efficiency.

Figure 5.13 shows the six different filter settings. For these filtered template banks it is not possible anymore, to refer to the efficiency that was measured during the training. The cosmic detection efficiency was therefore calculated by using a separate cosmic track Monte Carlo data set with approximately $5 \cdot 10^5$ cosmic muon tracks. $\epsilon^{\text{cosmic}}_{\text{detection}}$ is calculated by using the same equation as for the cosmic efficiency (see Equation 4.1). Furthermore the cosmic acceptance is defined in Equation 4.6. It is similar to the detection efficiency, but while the efficiency takes into account every tested muon no matter the type, the acceptance is restricted to the template types that are allowed by the filter.

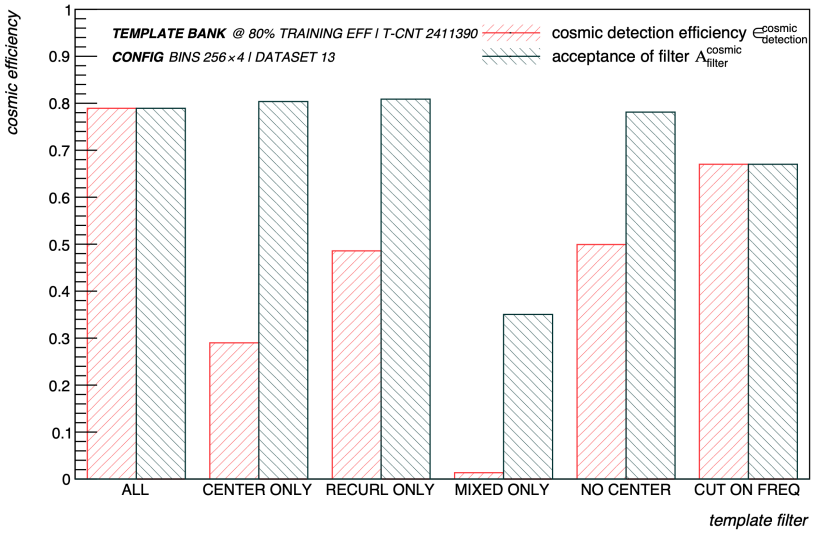

Figure 5.13: Efficiency and acceptance for different filter settings.

According to Figure 5.13, the cosmic efficiency for different filter settings is roughly determined by the percentage of templates that the filter incorporates. For example, the "center only" filter corresponds to 31 % of templates in the template bank and reduces the cosmic efficiency to about 30 % (which are 38 % of the former 80 %). The filter "recurl only" corresponds to 62 % of the templates in the template bank and reduced the cosmic efficiency to 48 %, which is 60 % of the former 80 %. The fact that these post-filter efficiencies do not exactly correspond to their fraction of templates is caused by a slight bias in the simulation, which leads to more cosmics being reconstructed in the central area than in the recurl stations.

While the acceptances for the filters by template types approximately stay at $\approx 80\,\%$ for the central and recurl stations it is reduced to about 35 % for mixed templates. This indicates that during training, the mixed station template categories saturate slower. This is due to the fact that these tracks occur very rarely. For a future implementation of the Cosmic Trigger, one must therefore keep in mind that it might be useful to train these templates in a separate process, so that this category can independently reach the same $\epsilon^{\text{cosmic}}_{\text{detect}}$ as the other ones.

For the last filter, "cut on freq", which excludes all templates that were only found once during training, the acceptance can not be defined. The cosmic efficiency is reduced to about

70 %. The number of templates that are required is reduced by about 50 %, from $\approx 2.41 \cdot 10^6$ to $\approx 1.21 \cdot 10^6$. Assuming that this behaviour persists similar for mappings with higher SPCs, this filter can significantly reduce the template number and with it the cost of required hardware. However, it comes with one downside, namely that most templates in the mixed category have a frequency of one. When using this filter the percentage of mixed templates in the template bank is reduced from 8 % to 1.6 %. A possible solution would be to use the frequency cut only for "recurl only" and "center only" templates, while keeping all templates in the "mixed only" category.

To motivate the template filtering and to choose the best suitable category, some more aspects can be taken into account. For example, as discussed in the introduction, the main source of background for the Cosmic Trigger are particles from the muon decays originated at the target. Therefore, the station with the highest amount of background hits that can cause false-positive template matches is the central station. By excluding templates that represent tracks that only go through the central detector, it is expected to also reduce the background significantly. Additionally, excluding central-central tracks is not even of great disadvantage, because the central detector can also be aligned with beam data (see Section 2.3). For the recurl stations this data might not be sufficient, therefore it is in particular the cosmic muons with mixed tracks and tracks in the recurl stations, which are important to be detected.

Background Rejection with Template Filter The performance of the different filters can be evaluated by looking at their respective ROC-curves. First, the two most promising filters, "no center" and "cut on freq" are compared with the unfiltered configuration for SPC = 1024 (solid line in Figure 5.11). The result is depicted in Figure 5.14.

It should be noted that the values of the cosmic efficiency were simulated with the activated filter. This is the reason why the data points for "no center" are notably shifted towards the left. The upper limit of $\epsilon_{\text{detect}}^{\text{cosmic}}$ for a template bank with this filter lies at about $60 - 70\,\%$ due to the exclusion of one third of the detector. As aforementioned, a data set of 10^5 background frames was used to calculate these data points. Therefore very high values for the frame selectivity should be treated with caution because of the little statistical evidence.

The results show that the "no center" filter heavily outperforms the other two settings. It is capable of increasing the frame selectivity by one magnitude. To reach a cosmic efficiency of 50 % for example, ten times less frames are triggered when using "no center" compared to using "cut on freq" or no filter at all. Interestingly, "cut on freq" performs slightly worse than the unfiltered configuration. The reduction of cosmic efficiency by about 10 % (data points shifted towards left) thus outweighs the gain in background rejection caused by a reduced number of templates.

Since it appears that excluding the center station brings the desired goal of a frame selectivity of 10^3 back within reach, excluding the "no center" filter will be examined more closely. Figure 5.15 shows the same SPMs as Figure 5.11, using "no center" this time.

Figure 5.15 shows that combining the previously found best performing settings for the parameters, such as high SPC, high SPR and "no center" filter increase the rejection even beyond the desired limit of 99 % or $99,9\,\%$. The curvatures indicate an exponential decay (with exception of some discrepancies for SPC = 1024, most likely caused by statistical fluctuations in the background simulation data) for all three configurations, steadily improving for higher SPCs.

Conclusion As aforementioned, from the pool of simulated feasible configurations, an SPM of 256×8 offers the best performance, i.e. a selectivity of $\approx 10^2$ at $\epsilon_{\text{detect}}^{\text{cosmic}} = 50\,\%$. By excluding cosmics that only traverse the central detector it was possible to further improve this to $\approx 1.4 \cdot 10^3$ at $\epsilon_{\text{detect}}^{\text{cosmic}} = 50\,\%$. The performance of the "no center" filter is summed up in Table 5.9.

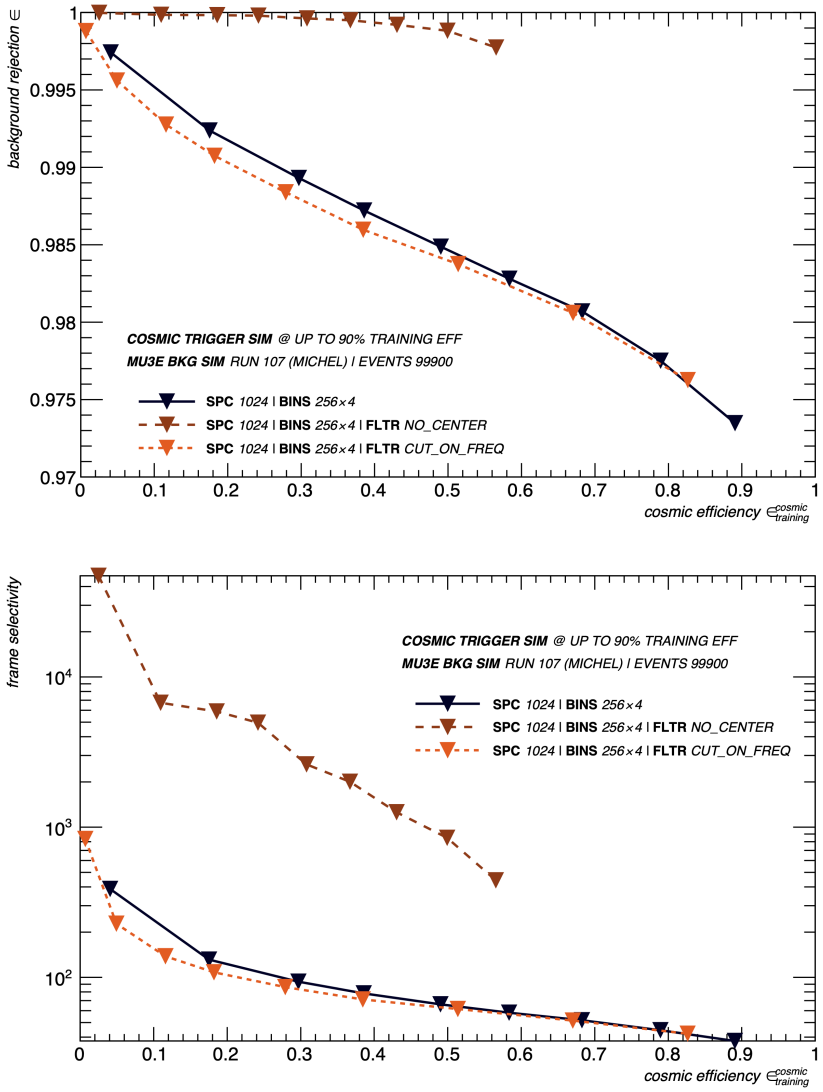

Figure 5.14: ROC curves (*top*) and frame selectivities (*bottom*) for SPC = 1024 with no filter (*solid*) and two different template bank filters: "no center" (*long dashes*) and "cut on freq" (*short dashes*). The "no center" filter improves the selectivity approximately by one order of magnitude.

Filter	"none"				"no center"			
SPC	$\epsilon_{\text{detect}}^{\text{cosmic}}$	$\mathcal{A}_{\text{filter}}^{\text{cosmic}}$	$\mathcal{F}_{\text{select}}^{\text{rate}}$	N_{tmpl}	$\epsilon_{\text{detect}}^{\text{cosmic}}$	$\mathcal{A}_{\text{filter}}^{\text{cosmic}}$	$\mathcal{F}_{\text{select}}^{\text{rate}}$	N_{tmpl}
512	50 %	50 %	$0.30 \cdot 10^2$	$0.25 \cdot 10^6$	–			
1024	50 %	50 %	$0.66 \cdot 10^2$	$0.91 \cdot 10^6$	–			
2048	50 %	50 %	$0.99 \cdot 10^2$	$3.8 \cdot 10^6$	–			
512	80 %	80 %	$0.21 \cdot 10^2$	$0.61 \cdot 10^6$	50 %	80 %	$0.21 \cdot 10^3$	$0.42 \cdot 10^6$
1024	80 %	80 %	$0.44 \cdot 10^2$	$2.4 \cdot 10^6$	50 %	80 %	$0.85 \cdot 10^3$	$1.6 \cdot 10^6$
2048	80 %	80 %	$0.61 \cdot 10^2$	$12.2 \cdot 10^6$	50 %	80 %	$1.4 \cdot 10^3$	$8.5 \cdot 10^6$

Table 5.9: Benchmark comparison of a template bank using no filter for $\epsilon_{\text{detect}}^{\text{cosmic}} = 50\,\%$ and $\epsilon_{\text{detect}}^{\text{cosmic}} = 80\,\%$. The latter was also filtered with "no center".

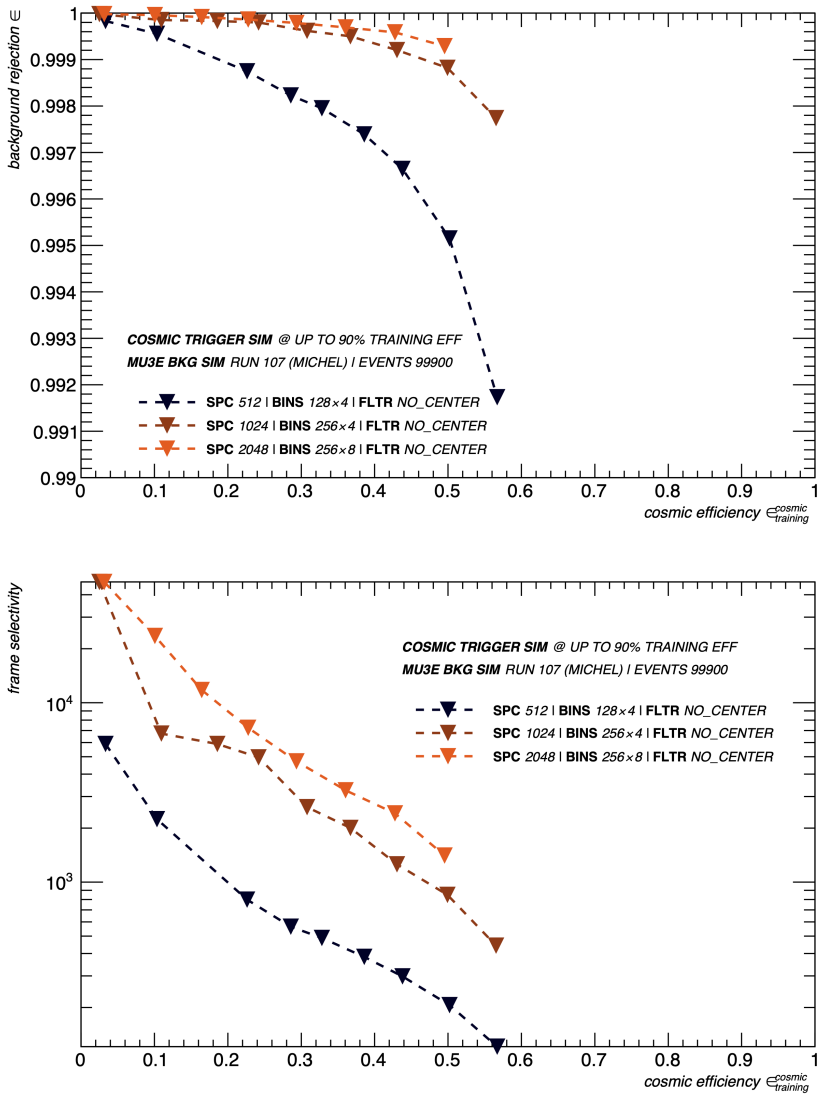

Figure 5.15: Receiver operating characteristic curve (*top*) and frame selectivity (*bottom*) with "no center" filter for SPMs used in Figure 5.11.

5.4 Performance at higher Beam Rates

In the previous steps of the analysis, background simulations with frames containing one Michel-decay were used to examine the behaviour of the cosmic trigger in a "toy study". It was thereby possible to identify the impact of different values for SPC and SPR as well as examine the effect of some systematic cuts and restrictions on the template bank. In the following, the beam rate of the background simulation is increased towards the rate that the *Cosmic Trigger* would encounter in the actual phase-I experiment. Figure 5.16 shows the background rejection for a stopping rate of up to 10^8 Hz.

An SPM of 256×8 and a cosmic efficiency of 80 % were used, without, and with "no center" filter. Additionally, a cut was implemented that excludes approximately the 15 % of frames that contain the most hits. It is expected that these frames have a higher false-positive match probability. Therefore, it could be a major improvement, to exclude them from the pattern recognition. Also, these frames are of higher interest in general, thus it is reasonable to reconstruct them anyway.

As predicted in the study for single-event decays, the "no center"-filter leads to a significant performance improvement by about one magnitude. For both configurations, the 15 % cut does only slightly improve the performance. However, unless the frames excluded by the cut are reconstructed anyway, increases the overall computational effort of the cosmic filter farm and should therefore not be used.

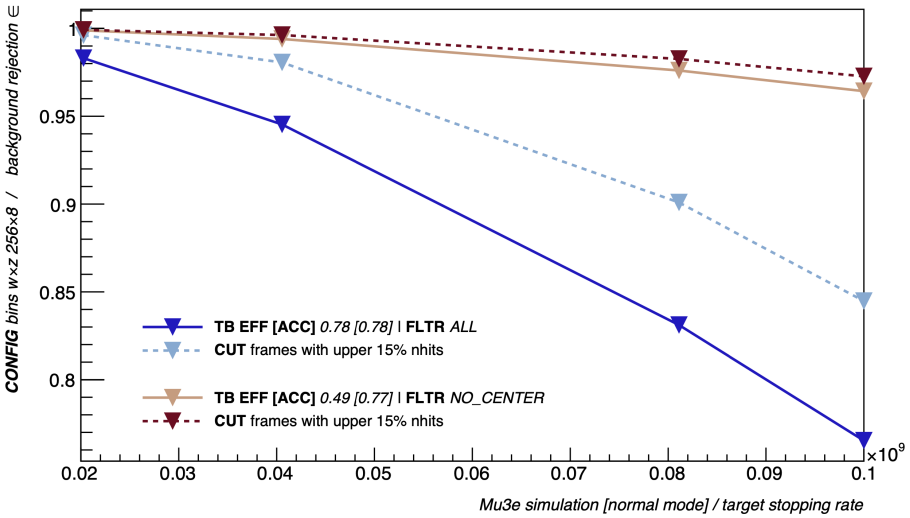

Figure 5.16: Background rejection for different stopping rates, using $\approx 10^5$ frames. In the simulation, the stopping rate can be obtained by $f_{\text{stopping}} = f_{\text{beam}}/2.466$. A stopping rate of $2 \cdot 10^7$ corresponds to one Michel decay per frame, as used previously.

		no cut		15 % cut	
Filter	$\epsilon_{\text{detect}}^{\text{cosmic}}$ [%]	$\epsilon_{\text{reject}}^{\text{bkg}}$ [%]	$\mathcal{F}_{\text{select}}^{\text{rate}}$	$\epsilon_{\text{reject}}^{\text{bkg}}$ [%]	$\mathcal{F}_{\text{select}}^{\text{rate}}$
–	47	85.5	6.9	91.9	12.3
–	78	76.5	4.2	84.5	6.4
"no center"	49	96.5	28	97.3	37

Table 5.10: Background rejection and frame selectivity for the final phase I stopping rate of 10^8 Hz.

Chapter 6

Discussion

6.1 Conclusion and Summary

This thesis studied the performance of the *Cosmic Trigger*, i.e. a pattern recognition cosmic muon trigger for the Mu3e experiment. A simulation software was implemented in order to perform the feasibility study. Several aspects were investigated. The first concern belonged to the super pixel mapping, which described the layout, size and shape of the super pixels. From the results, it can be concluded that super pixel stripes, with the long side in parallel to the beam line, are superior to ring-like structures for a cosmic muon PR trigger. Studied were stripes with dimensions of a few millimeters in ϕ direction and up to 100 mm in z-direction.

In the first part of the analysis, one-decay beam simulation frames were used to examine the false-positive trigger rate. It was thereby possible to reach frame selectivities of up to a magnitude of 10^3 by using additional geometrical selection constraints. For example, a significant improvement could be achieved by ignoring all tracks that only traverse the central detector. It was then studied how the cosmic efficiency for a fixed configuration correlates with the selectivity. A linear relation was found for cosmic efficiencies between 30 % and 90 %. Therefore, there is no systematic optimum, but one can choose the performances based on trigger requirements and hardware availability. In a second step, the stopping rate at the target was increased up to 10^8 Hz, which corresponds to final rate planned for phase I of the Mu3e experiment. The maximum selectivity that could be achieved via simulation was roughly 28 (excluding tracks that only traverse the central station).

All the results were produced with a software that was developed within the course of this thesis. It provides a framework to simulate hardware pattern recognition on a cylindrical pixel tracking detector and could be used for further studies.

As a conclusion, it can be stated that the *Cosmic Trigger* study produced promising results towards a future implementation in the Mu3e experiment. Also for even higher beam rates, a pattern recognition system is capable of achieving remarkable selectivities and thereby reduces the rate of frames that need to be reconstructed. As a major conclusion of this project, the results found are used to give a practical recommendation on how the *Cosmic Trigger* could be implemented, as it is shortly described in the following.

Recommendations for Implementation

Before giving recommendations, it must be said that there is no "ideal" configuration that performs better than all others in every regard. However, depending on certain constrains, it is possible to propose a solution and also estimate its performance and hardware requirements. Constrains that may narrow down the realm of suitable options are:

- The desired *cosmic detection efficiency* $\epsilon_{\text{detect}}^{\text{cosmic}}$,
- The desired *number of templates* restricted by the amount of available PRM boards,
- The desired *frame selectivity* $\mathcal{F}_{\text{select}}^{\text{rate}}$,
- Whether or not to use template filters.

So far it has not been studied, how exactly the frame selectivity is related to the amount of required filter farm PCs. In a simple model, one could assume that a selectivity of 10 also reduces the number of filter farm PCs by a factor of 10. Also assuming that the cosmic reconstruction would require at least 12 additional filter farm PCs (the same as for the beam event reconstruction), a selectivity of 12 or higher would be sufficient to shrink down the cosmic filter farm to one PC. Nevertheless, some points can be raised against this argument. One of them is that the computational effort of reconstructing cosmics is expected to be significantly higher than for beam events, lacking the additional constraint that they are originated at the target. However, the simplified estimation proposed above gives a scale for how to relate the sensitivity to the of hardware cost.

In the following, two different example implementations are described, one using a template filter and one that does not. For the cosmic efficiency, a working point of 50 % is fixed, which implies that an average detection rate of 5 Hz can be expected. For the implementation without template filter, the working point at $\epsilon_{\text{detect}}^{\text{cosmic}} = 80\,\%$ is taken into account aswell.

Implementation using Template Filters

If it is acceptable to exclude cosmics only traversing the central area, the "no center" filter definitely yields the best performance in selectivity. As simulated, a SPM of 256×8 super pixel bins is proposed, meaning that super pixels have a $w \times z$ size of $2.13 \times 50\,\text{mm}^2$ in the outer layer (4) and $1.81 \times 50\,\text{mm}^2$ in the second-to-outer layer (3). In order to reach a cosmic efficiency of 50 %, the database must be trained up to 80 % (which also corresponds to a cosmic acceptance of $\mathcal{A}_{\text{filter}}^{\text{cosmic}} = 80\,\%$) before applying the template filter. Using this configuration at $\epsilon_{\text{detect}}^{\text{cosmic}} = 50\,\%$, it is possible to reach a selectivity of approximately 28 at a beam rate of 10^8 Hz. About $8.5 \cdot 10^6$ templates are required with this configuration. Therefore, one PRM board with its memory for about $15 \cdot 10^6$ patterns is sufficient.

Implementation without Template Filters

If cosmic muons are supposed to be used for the alignment of every detector station, no filter can be used. In this case, as a starting point, an SPM of 256×8 is chosen. At a cosmic efficiency of $\epsilon_{\text{detect}}^{\text{cosmic}} \approx 80\,\%$ a selectivity of about 4.2 can be reached, while for $\epsilon_{\text{detect}}^{\text{cosmic}} \approx 50\,\%$ a $\mathcal{F}_{\text{select}}^{\text{rate}}$ of about 10 can be obtained.

In order to further improve this result, the SPC could be increased by increasing the number of z-bins and thereby decreasing their size. As discussed in Section 5.1.2 it can be expected that doubling the SPC quadruples the number of templates. Using the fit results from Table 5.4 the resulting number of templates is given in Table 6.1.

Subsequently, it can be stated that without a template filter, it is possible to reach a selectivity of ≥ 10 with a cosmic efficiency of 50 %. For a cosmic efficiency of 80 %, the selectivity

	cosmic efficiency 50 %			cosmic efficiency 80 %		
$w \times z$ bins	$\mathcal{F}^{\text{rate}}_{\text{select}}$	N_{tmpl}	# boards	$\mathcal{F}^{\text{rate}}_{\text{select}}$	N_{tmpl}	# boards
256×8	6.9	$3.8 \cdot 10^6$	1	4.2	$12.2 \cdot 10^6$	1
256×16	8.6*	$< 26 \cdot 10^{6**}$	2	5.3*	$53 \cdot 10^6$	4
256×32	10.8*	$< 111 \cdot 10^{6**}$	6-8	6.6*	$220 \cdot 10^6$	14-16

Table 6.1: Specifications for 256×8 bins and estimation for increased z-binning by doubling ($z = 16$) and quadrupling ($z = 32$). (*) Extrapolation with a factor of 1.25 per SPC doubling, see Table 5.7. (**) Using $\epsilon^{\text{cosmic}}_{\text{detect}} = 60\,\%$ fit from Figure 5.5 as an upper limit.

is reduced to $\gtrsim 5$. However, compared to the previous implementation using the "no center" filter, this comes at a higher cost because more PRM boards are required. On the other hand, a filterless implementation can yield the highest cosmic detection rates.

If the minimisation of the template count has the highest priority, it is also possible to use the "cut on frequency" filter, which excludes all templates that were only triggered once during training. This filter reduces the $\epsilon^{\text{cosmic}}_{\text{detect}}$ by about 10 % while reducing the number of templates to its half. It is recommended to only use this filter for cosmic muons that traversed exactly one of the stations. Mixed tracks (tracks that go through the central and a recurl station) are important for alignment, therefore they should not be excluded. Also, these tracks only account for $\approx 8\,\%$ of the total templates, so using a cut would not yield a huge improvement in template count anyway.

6.2 Outlook and Questions for Future Studies

The research that was done within this thesis was a first step towards the development of an efficient cosmic muon reconstruction system. The result of this work provided the first answers on if and how a pattern recognition trigger would perform, yet there are questions left open, waiting to be answered in future studies. The following list gives a short summary of the questions that evolved:

- **Super Pixel Size Limit**
  The Mu3e pixel detector suffers from misalignment of up to several $\mathcal{O}(100\mu m)$. In order for the pattern recognition system to be able to detect cosmic muons without being significantly affected by misaligment the super pixels can not be designed infinitely small. Within this thesis, a lower limit of $2 \times 2\,\mathrm{mm}^2$ was assumed, but the actual limit might differ.

- **Non-uniform Super Pixel Mapping**
  The *Cosmic Trigger Simulation* only used a uniform super pixel binning. However, refering to Section 5.1.1 it could be useful to switch to a non-uniform mapping, especially in z-direction. This might be capable of balancing the phase space effects.

- **Hardware Requirements for Cosmic Muon Reconstruction**
  At this point, the computational effort, needed to reconstruct cosmic muon tracks is not exactly known. A further investigation could give an estimation on the required number of filter farm PCs and also of the required frame selectivity.

- **Cosmic Trigger Hardware Concept**
  In terms of a hardware concept, several aspects have to be examined. Next to data flow and readout integration, the super pixel assignment should later be realised by pixel chips and pixel IDs, not by (x,y,z)-coordinates. Solving this problem in an efficient way, probably is a non-trivial task.

- **Hit Selection based on Timing**
  In particular for higher beam rates (also with Mu3e phase II in mind), it can be useful to introduce further discrimination variables. One candidate could be the timing information of hits with an expected accuracy of $< 10\,\mathrm{ns}$. The timestamp of hits could be taken into account when sending the patterns to the AM chips, meanwhile presorting the frame into smaller bunches.

- **Cosmic Muon Event Rate**
  How many cosmic muon tracks that can used for alignment are actually measurable in the detector? In this thesis, a rate of $\mathcal{O}(10\,\mathrm{Hz})$ was assumed, but the actual rate is not yet known, as concrete effects of scattering in the magnet, the ceiling et cetera were not precisely taken into account in this estimation.

Bibliography

[1] *Standard Model illustration, PBS NOVA, Fermilab, Office of Science, United States Department of Energy, Particle Data Group, Public Domain*. Sept. 2019. URL: https://commons.wikimedia.org/w/index.php?curid=4286964. (accessed: 13.10.2020).

[2] A. Blondel et al. *Research Proposal for an Experiment to Search for the Decay $\mu \to eee$*. 2013. arXiv: 1301.6113 [physics.ins-det].

[3] Yoshitaka Kuno and Yasuhiro Okada. "Muon decay and physics beyond the standard model". In: *Rev. Mod. Phys.* 73 (1 Jan. 2001), pp. 151–202. DOI: 10.1103/RevModPhys.73.151. URL: https://link.aps.org/doi/10.1103/RevModPhys.73.151.

[4] W. Bertl et al. "Search for the decay $\mu^+ \to e^+e^+e^-$". In: *Nuclear Physics B* 260.1 (1985), pp. 1–31. ISSN: 0550-3213. DOI: https://doi.org/10.1016/0550-3213(85)90308-6. URL: http://www.sciencedirect.com/science/article/pii/0550321385903086.

[5] *Bethe Bloch*. URL: https://en.wikipedia.org/wiki/Bethe_formula.

[6] Particle Data Group. *33. Passage of particles through matter*. 2018. URL: https://pdg.lbl.gov/2018/reviews/rpp2018-rev-passage-particles-matter.pdf. (accessed: 19.10.2020).

[7] *Strahlungsslänge*. URL: https://de.wikipedia.org/wiki/Strahlungsl%C3%A4nge.

[8] *Atmospheric Collision*. URL: https://commons.wikimedia.org/wiki/File:Atmospheric_Collision.svg.

[9] Prashant Shukla and Sundaresh Sankrith. *Energy and angular distributions of atmospheric muons at the Earth*. 2018. arXiv: 1606.06907 [hep-ph].

[10] Masaharu Tanabashi et al. "Review of particle physics". In: *Physical Review D* 98.3 (2018), p. 030001.

[11] U. Bellgardt et al. "Search for the Decay $\mu^+ \to e^+e^+e^-$". In: *Nucl. Phys. B* 299 (1988), pp. 1–6. DOI: 10.1016/0550-3213(88)90462-2.

[12] The MEG Collaboration. *Search for the Lepton Flavour Violating Decay $\mu^+ \to e^+\gamma$ with the Full Dataset of the MEG Experiment*. 2016. arXiv: 1605.05081 [hep-ex].

[13] William J. Marciano, Toshinori Mori, and J. Michael Roney. "Charged Lepton Flavor Violation Experiments". In: *Annual Review of Nuclear and Particle Science* 58.1 (2008), pp. 315–341. DOI: 10.1146/annurev.nucl.58.110707.171126. eprint: https://doi.org/10.1146/annurev.nucl.58.110707.171126. URL: https://doi.org/10.1146/annurev.nucl.58.110707.171126.

[14] K. Arndt et al. *Technical design of the phase I Mu3e experiment*. 2020. arXiv: 2009.11690 [physics.ins-det].

[15] Ulrich Hartenstein. *Track based alignment for the Mu3e pixel detector*. 2019.

[16] Conseil européen pour la recherche nucléaire (CERN). *Geant4 Simulation Framework*. URL: https://geant4.web.cern.ch.

[17] A Schöning. "Three-dimensional triplet tracking for LHC and future high rate experiments". In: *Journal of Instrumentation* 9.10 (Oct. 2014), pp. C10025–C10025. ISSN: 1748-0221. DOI: 10.1088/1748-0221/9/10/c10025. URL: http://dx.doi.org/10.1088/1748-0221/9/10/C10025.

[18] Alexandr Kozlinsky. *Internal presentation*. 2020.

[19] Philipp Biallass and Thomas Hebbeker. *Parametrization of the Cosmic Muon Flux for the Generator CMSCGEN*. 2009. arXiv: 0907.5514 [astro-ph.IM].

[20] Bill Ashmanskas et al. "The CDF Silicon Vertex Trigger". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 518.1-2 (Feb. 2004), pp. 532–536. ISSN: 0168-9002. DOI: 10.1016/j.nima.2003.11.078. URL: http://dx.doi.org/10.1016/j.nima.2003.11.078.

[21] Jahred A. Adelman et al. "The Silicon Vertex Trigger upgrade at CDF". In: *Nucl. Instrum. Meth. A* 572 (2007). Ed. by Franco Cervelli et al., pp. 361–364. DOI: 10.1016/j.nima.2006.10.383.

[22] Georges Aad et al. "The ATLAS experiment at the CERN large hadron collider". In: *Jinst* 3 (2008), S08003.

[23] N Kimura and. "ATLAS FTK a - very complex - custom super computer". In: *Journal of Physics: Conference Series* 762 (Oct. 2016), p. 012005. DOI: 10.1088/1742-6596/762/1/012005. URL: https://doi.org/10.1088%2F1742-6596%2F762%2F1%2F012005.

[24] S. Dittmeier. "The ATLAS Hardware Track Trigger design towards first prototypes". In: Feb. 2020, p. 049. DOI: 10.22323/1.373.0049.

[25] Dohun Kim. *Acelerated track reconstruction for Mu3e using fast pattern lookups*. 2018.

[26] *Content-addressable memory*. URL: https://en.wikipedia.org/wiki/Content-addressable_memory.

[27] A. Annovi et al. "Associative memory design for the fast track processor (FTK) at ATLAS". In: *2011 IEEE Nuclear Science Symposium Conference Record*. 2011, pp. 141–146. DOI: 10.1109/NSSMIC.2011.6154467.

[28] G. M. Bilei et al. “A pattern recognition mezzanine based on associative memory and FPGA technology for Level-1 track triggers for the HL-LHC upgrade”. In: *2015 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*. 2015, pp. 1–2. DOI: 10.1109/NSSMIC.2015.7581863.

[29] Alberto Stabile et al. *Phase-II Associative Memory ASIC Specifications*. Tech. rep. 2018.

[30] *Technical Design Report for the Phase-II Upgrade of the ATLAS Tile Calorimeter*. Tech. rep. CERN-LHCC-2017-019. ATLAS-TDR-028. Geneva: CERN, Sept. 2017. URL: https://cds.cern.ch/record/2285583.

[31] Prof. Dr. Andre Schöning. *private communication*. Nov. 2020.

[32] Conseil européen pour la recherche nucléaire (CERN). *ROOT Framework*. URL: https://root.cern.ch.

Danksagung

Diese Arbeit wäre ohne die Unterstützung einiger Menschen nicht möglich gewesen, für deren Beitrag ich im folgenden meine Wertschätzung ausdrücken möchte.

In erster Linie möchte ich mich bei *Prof. Dr. André Schöning* für das in mich gesetzte Vertrauen, dieses Projekt bearbeiten zu dürfen, sowie die verlässliche und inspirierende Betreuung bedanken. Außerdem danke ich *Prof. Dr. Norbert Herrmann* für seine Bereitschaft das Zweitgutachten zu übernehmen.

Mein besonderer Dank gilt allen Mitgliedern der *Mu3e*- und *ATLAS*-Gruppe am Physikalischen Institut für die herzliche Atmosphäre und Gemeinschaft. Erwähnen möchte ich vor allem *Sebastian Dittmeier*, *David Immig*, *Thomas Rudzki* und *Luigi Vigani* für ihre sorgfältigen Rückmeldungen und Verbesserungsvorschläge zur Arbeit. *Nik Berger* und *Alexandr Kozlinskiy* danke ich für die schnelle Hilfe bei Fragen zur Mu3e Simulation. Besonders hervorheben möchte ich *Christof Sauer* für die zuverlässige fachliche Unterstützung über die gesamte Dauer der Arbeit hinweg, für den geduldigen Software Support, lange Brainstorming Sessions, unzählige Diskussionen, seine Korrekturvorschläge, sowie vor allem für die mentale Unterstützung mit Rat und Tat.

Desweiteren danke ich *Madeleine Weber*, *Johannes Grabenstein*, *Daniel Satterly* und *Jonathan Schmalzridt* für die besonderen Plätze, die sie jeweils in meinem Leben einnehmen.

Zu guter Letzt gilt mein größter Dank meinen Eltern *Matthias und Renate Neureither* sowie meiner Schwester *Magdalena* für den bedingungslosen Rückhalt, den sie mir seit Beginn meines Lebens geben und für die Gewissheit, dass ich immer auf sie zählen kann.

Soli deo gloria.

Erklärung

Ich versichere, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, den 9.11.2020,

Appendix A

Software Remarks

A.1 Comments on Settings and Versions

The software developed for the *Cosmic Trigger* can be found on Bitbucket (`https://bitbucket.org/kneureither/cosmictrigger/src/master/`). For the development of this software, ROOT [32] in version 6.20/04 was used. The cosmic Monte Carlo data was simulated using the Mu3e software package on branch `v4.4_dev`.

A.1.1 Selected Simulation Settings for Cosmics Muons

- *Mu3eSim* was used in the "cosmic mode". A beam rate of 0 generates one cosmic per simulation frame.
- *Mu3eTriRec* was used with the output from *Mu3eSim* in order to create tracks from detector hits.
- Hits can be obtained from the *TTree segs* in the TriRec output ROOT File.

The following snippet shows some important paramters from the *digi.json* file. Everything else was left at default.

```
"frameLength" : 50.0,
"rate" : //values from 0 up to 1e8
"mode" : 30,
...
"tracker" : {
    "efficiency" : 1.0,
    "noiseRate" {
        "inner" : 0.0,
        "outer" : 0.0
    }
}
```

A.1.2 Selected Simulation Settings for Background

- Usually, 10^5 frames were simulated in *Mu3eSim*
- MC hit position truth information was included.
- Either the Michel decay mode or the normal mode were used.

digi.json snippet for Michel-Decays:

```
"frameLength" : 50.0,
"rate" : 0,
"overlapMode" : "m",
"mode" : 2,
...
"tracker" : {
    "efficiency" : 1.0,
    "noiseRate" {
        "inner" : 0.0,
        "outer" : 0.0
    }
    ...
},
"write" : {
"truth" : 2,
...
}
```

digi.json snippet for full simulation ("normal mode"):

```
"frameLength" : 50.0,
"rate" : 1e8,
"overlapMode" : "s",
"mode" : 0,
...
"tracker" : {
    "efficiency" : 1.0,
    "noiseRate" {
        "inner" : 0.0,
        "outer" : 0.0
    }
    ...
},
"write" : {
"truth" : 2,
...
}
```

A.2 Pattern Engine

A.2.1 SPM Initialisation

The Pattern Engine can either be initialised with the same binning at all areas or with different binnings for center area and recurl areas. The constructor takes 3 or 5 parameters respectively.

```
PatternEngine(int spWBinsCenter, int spZBinsCenter,
              std::string plottingpath)

PatternEngine(int spWBinsCenter, int spZBinsCenter,
              int spWBinsRecurl, int spZBinsRecurl,
              std::string plottingpath)
```

During initialisation two *bin boundary vectors* are created for z-bins and w-bins which hold the z and ϕ (or x) coordinates of borders between two super pixels. If `mode` is set to 0, this happens in a linear and equidistant way for each bin boundary. If one would like to study non-linear binning, the only thing that must be done is to fill these vectors with custom non-linear boundaries or to add another constructor that does so.

The main member function, that the pattern engine provides is:

```
unsigned int PatternEngine::getSuperPixel(float x, float y, float
    z)
```

It takes the coordinates in (x,y,z)-space as argument and returns the `unsigned int` value of the corresponding SID. An SID is calculated from the w- and z-bin in the boundary vectors, where a pixel hit lies in. These two indices are determined by searching the *bin boundary vectors*. For better speed, a recursive *bin search* was implemented that has $O(log(n))$ complexity (instead of $O(n)$ for a *sequential search*). The reason it was implemented in that way is to allow as much flexibility as possible for custom *super pixel binning*.

A.3 Template Bank

A.3.1 Components and Main Functionalities

The main functionality of the *Template Bank* is building up a database which can either be done by adding many cosmic templates or by loading it from a file. Additionally it can handle operations on the database in order to benchmark it. This also includes testing if some given templates occur in the database and furthermore keeping track of some figures of merit. The main functions that deliver these functionalities are listed below:

- `TemplateBank(std::string plottingpath, float stopping_efficiency, int dataset , int mode, int wBins, int zBins)`
  Constructor of *TemplateBank* class. As parameters it uses a path where plots are to be saved, some metadata of the SPM, and the *training dataset* that is used (important when saving and loading the database to and from a *ROOT File*). The *stopping efficiency* tells, up to which $\epsilon_{detect}^{cosmic}$ the database should be trained.

- `bool fillTemplate(unsigned int *SPIDs, int hitcount, float p, float dca, float phi, float theta)`
  Adds a TID based on SIDs in the `SPIDs` array into the database with the corresponding trajectory data. If the TID is already present in the database, the trajectory data is appended to its entry. The function calculates the cosmic efficiency for every 10^5 entries that were added. It returns *false*, as soon as the *stopping efficiency* is reached, else *true*.

- `bool checkTemplate(TemplateID &TID)`
  Checks if the given TID exists in the database and returns true if it does. It also keeps track of the tested TIDs that existed in the database in a separate `std::map<> CheckedMemory`.

- `bool checkCosmicTemplate(TemplateID &TID)`
  Basically this function does the same as the previous one, but it has some specialities. It was designed to measure the $\epsilon_{\text{detect}}^{\text{cosmic}}$ after the training, using additional Monte Carlo cosmic data. The functionality can be seen as an external evaluation of a pre-trained database. Also, it can take into account a *TID filter*, that ignores some of the templates based on a (geometric) categorisation. This is described in more detail in Section 4.5.4.

- `void writeAMtoFile(std::string path, int *zBins, int *wBins, ...)`
  Stores the database of TIDs in a *ROOT File* and adds some meta data, which includes the *zBins*, the *wBins*, the *dataset* and the *cosmic efficiency*.

- `bool readAMfromFile(std::string path, float stopping_efficiency, TIDLoadingFilter filter)`
  Loads a database of TIDs from a ROOT File into the template database.

A.3.2 Database File Handling

A *Cosmic Template Database* can be written to a file with the function `writeAMtoFile(...)` and loaded with `readAMfromFile(...)`. A template database File contains two *TTree*s. The first consists of one entry which contains all the configuration meta data, such as the super pixel mapping parameters, the cosmic efficiency, the training events, and so on. The second *TTree* holds the TIDs as `short[4]` arrays, a `std::string` representation of the *TID*, and could also store the track parameters of the particles corresponding to this TID. The two functions mentioned basically implement a `std::map<>` $\Longleftrightarrow$ *TTree* conversion.

The *ROOT File* data representation with writing and reading is implemented in a separate class, from which reading and writing classes are derived, the `TemplateDatabaseFile`.

Appendix B

Additional Simulation Results

B.1 Differences for higher SPRs

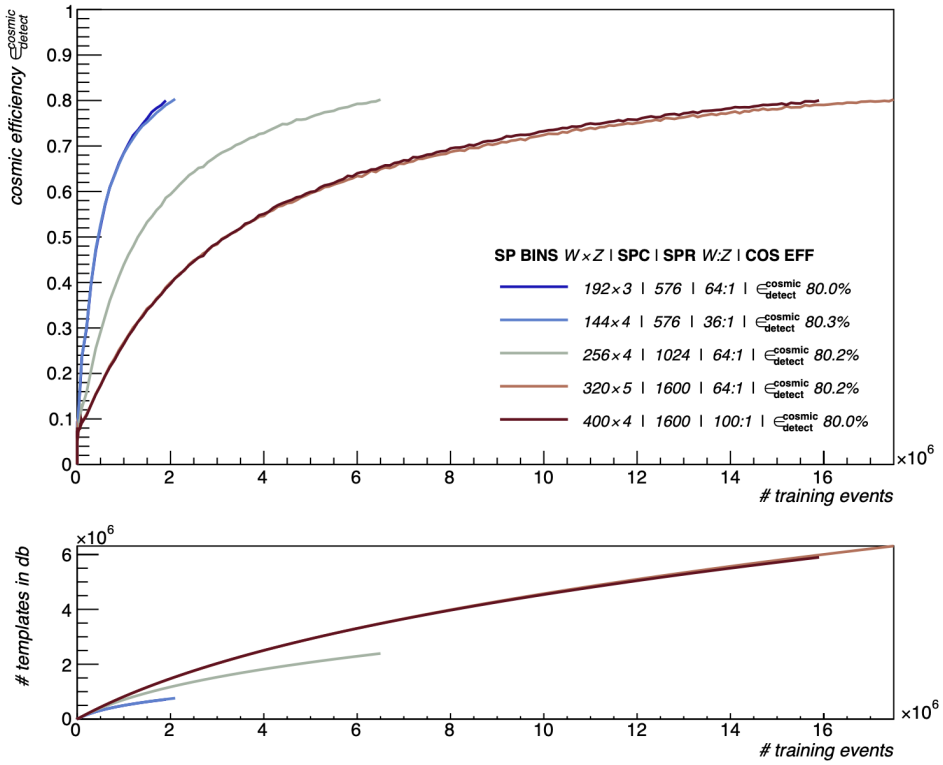

Figure B.1: Training processes for same SPC with slightly different SPR in the super pixel stripe regime. This plot is included, because it shows that for higher SPRs, their improvement effect can be seen as almost negligible

Setting	w-bins	z-bins	w-size [rad]	z-size [mm]	sp area [rad·mm]	$\epsilon^{\text{cosmic}}_{\text{detect}}$	# templates
1	192	3	0.0327	133	4.36	80.0 %	$7.1 \cdot 10^5$
2	256	4	0.0245	100	2.45	80.2 %	$2.4 \cdot 10^6$
3	320	5	0.0196	80	1.57	80.2 %	$6.3 \cdot 10^6$

Table B.1: Super Pixel Mappings used in Figure B.1