AI-Enhanced Self-Triggering for Extensive Air Showers: Performance and FPGA Feasibility

Qader Dorosti

Center for Particle Physics Siegen,
Department für Physik, Universität Siegen,
Walter-Flex-Str. 3, 57072 Siegen, Germany

E-mail: dorosti@hep.physik.uni-siegen.de

ABSTRACT: Cosmic-ray detection with radio antennas has traditionally depended on external triggers from particle detectors, constraining sensitivity and increasing complexity. Previous attempts at fully standalone, radio-only triggers have often failed under intense radio frequency interference, making genuine air-shower signals difficult to isolate. We present a proof-of-principle artificial intelligence-based self-triggering system that overcomes these limitations. By training a deep learning model on both real noise data and injected cosmic-ray-like pulses, we achieve an exceptionally low false-positive rate alongside high detection efficiency. Configurable operating points can suppress false positives below 0.01% while retaining more than 88% of genuine signals, and can even eliminate false positives entirely at a modest reduction in signal efficiency. This flexibility makes single-station cosmic-ray detection feasible without requiring external trigger inputs. Applying our approach to real-world noise conditions reduces the initial false-positive event rate by several orders of magnitude, supporting large-scale deployments. Extrapolation to dedicated hardware implementations, such as FPGAs, indicates that sub-µs inference times are achievable, enabling real-time autonomous triggering. These results highlight the transformative potential of artificial intelligence for enhancing radio detection sensitivity and inaugurate a new generation of fully self-triggered cosmic-ray observatories.

Contents

1 Introduction

Ultra-high-energy cosmic rays (UHECRs) are primarily detected using large-scale ground-based observatories that employ a combination of particle detectors, fluorescence telescopes, and radio antennas [1]. Each technique provides complementary insights into extensive air showers (EAS), enabling detailed studies of cosmic-ray composition, energy spectrum, and arrival direction. Among these methods, radio detection has gained increasing attention due to its cost-effectiveness, full-sky coverage, and near-continuous operation [2]. By measuring the radio emission generated primarily by geomagnetic deflection [3] and the charge-excess effect [4] in the EAS, radio arrays can reconstruct key shower parameters, such as the atmospheric depth of the shower maximum (X_{max}) and primary energy, with high precision [5, 6].

A distinct advantage of radio detection is its sensitivity to inclined air showers, where traditional particle detector arrays struggle due to the attenuation of the electromagnetic shower component in the atmosphere [6]. However, despite these benefits, current radio detection techniques rely on

external triggers from other detector systems, such as surface or fluorescence detectors, limiting their autonomy and increasing operational complexity. The development of an effective self-triggering system is therefore crucial to fully exploit the potential of radio detection for cosmic-ray observations [7].

Previous attempts to implement self-triggering in radio arrays, such as those by CODALEMA and AERA, relied on conventional techniques like narrowband radio frequency interference (RFI) suppression and threshold-based triggering [8, 9]. However, these methods proved ineffective in distinguishing cosmic-ray-induced radio pulses from transient noise sources, leading to a high rate of false positives. As a result, these efforts faced significant challenges and did not produce established scientific results, underscoring the need for more advanced signal-processing strategies.

Recent developments in modelling EAS radio emissions, combined with experimental and simulation studies, have provided a deeper understanding of characteristic pulse shapes and ground radiation patterns [2]. This progress has paved the way for more sophisticated signal-processing techniques, including artificial intelligence (AI)-driven approaches for enhancing radio-based self-triggering. Recent studies show that deep learning models, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), can effectively distinguish cosmic-ray signals from background noise, significantly reducing detection thresholds. These techniques have been successfully applied to identify air-shower radio pulses in offline analyses [10], demonstrating their potential to enhance signal selection. Recent efforts, such as those in the GRAND experiment, continue to investigate self-triggering using AI-based methods, including CNNs. While preliminary results indicate that these methods can reject at least 40% of background noise at a 90% signal selection efficiency [11], achieving real-time deployment remains challenging. Self-triggering requires ultra-low latency processing to ensure the identification of cosmic-ray signals within microseconds of their arrival, while also maintaining a false-positive trigger rate at the sub-Hz level, a constraint that traditional software-based neural networks struggle to meet.

To address the challenge of self-triggering in cosmic-ray radio signal detection, AI-accelerated signal processing implemented on Field-Programmable Gate Arrays (FPGAs) offers a viable solution. FPGAs provide exceptional parallel processing capabilities and low-latency inference, making them ideal for real-time feature extraction and classification of radio pulses. Similar approaches have been explored in high-energy physics experiments, such as the ATLAS Tile Calorimeter, where deep learning algorithms implemented on FPGAs have demonstrated significant improvements in real-time signal reconstruction under high pileup conditions [12, 13]. These advancements highlight the potential of FPGA-based AI techniques in handling complex, high-rate signal environments, which are also crucial for enhancing the efficiency and accuracy of cosmic-ray self-triggering.

In this work, we investigate the feasibility of FPGA-based AI self-triggering for radio cosmic-ray detection. We begin by training and evaluating our deep neural network on Metal Performance Shaders (MPS) with data that includes *experimentally measured background noise*, then estimate their potential performance on FPGAs. While high-level synthesis (HLS) frameworks such as **HLS4ML** [14] and **Xilinx Vitis AI** [15] offer streamlined approaches to deploying AI models on FPGAs, we consider their integration a logical extension of this proof-of-concept study rather than part of our immediate scope.

Our method comprises a **fully convolutional autoencoder** that denoises radio traces by suppressing background noise while retaining cosmic-ray pulses, followed by a **fully convolutional**

classifier (FCN) that differentiates cosmic-ray signals from noise. By leveraging real data with measured background noise, we enhance the robustness of our approach in practical deployments. We also address critical issues associated with FPGA-based AI, including quantization strategies, resource utilization, and latency reduction, and propose a framework for real-time classification of cosmic-ray radio pulses. This proof-of-principle demonstration paves the way for future work incorporating more advanced HLS techniques to further optimize deployment on FPGA hardware.

The implementation of AI-accelerated FPGA-based self-triggering represents a significant advancement in radio detection technology. By reducing reliance on external detector triggers and increasing detection efficiency, this approach enables greater autonomy for radio observatories, facilitating the detection of rare and high-energy events, including ultra-high-energy neutrinos. Our results demonstrate the feasibility of deploying AI-driven self-triggering at the detector level, marking an important step toward fully autonomous, real-time radio detection systems for astroparticle physics.

2 Experimental Setup

The dataset used in this study consists of experimentally recorded noise traces collected in the physics campus of the University of Siegen. The measurements were performed using a butterfly antenna connected to a low-noise amplifier with 30 dB gain, which was further linked to a digital oscilloscope. These recordings capture real-world transient noise sources, including atmospheric and anthropogenic contributions.

To focus on the frequency range relevant for air shower radio detection, the recorded signals were digitally bandpass-filtered between 30 and 80 MHz. Furthermore, data was sampled at 250 MHz to ensure comparability with real-world cosmic-ray radio detectors, such as AERA and Auger RD, which operate at 180 MHz and 250 MHz, respectively. This adjustment reflects the practical constraints of large-scale detectors, where lower sampling rates are used due to budget and hardware limitations, ensuring the applicability of the method to future radio arrays.

The dataset used in this study was collected in an urban environment, where background noise levels are higher and more variable compared to remote cosmic-ray observatories. To capture a diverse range of real-world noise conditions, data was collected over approximately two hours, yielding 200 million samples. Data was continuously streamed, with traces read out at a rate of one per second. Given the high variability of the urban noise environment, the dataset presents a challenging test case for AI-driven signal detection methods. This setting introduces additional challenges, including increased transient interference and strong radio frequency interference (RFI). Two prominent narrowband RFI sources were identified at approximately 45 MHz and 71 MHz, along with other narrowband and transient RFI. The power spectrum of the recorded traces, shown in Figure 1, was obtained after applying a digital bandpass filter in the range of 30 to 80 MHz. For the initial analysis, these narrowband RFIs were not suppressed to assess the model's robustness in the presence of strong interference. The results presented in the following section reflect model performance when both training and testing were conducted on data sets containing these RFIs. Additionally, we examine how performance changes when these narrowband RFIs are mitigated prior to training and validation, providing insights into the impact of RFI suppression on the learning process, which we will report in the discussion section 7.
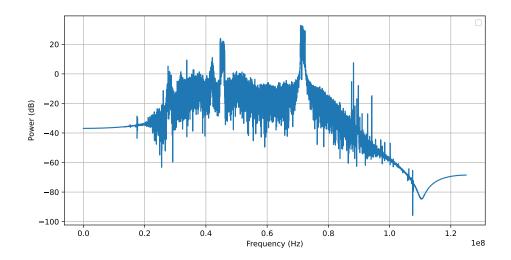
Figure 1. Power spectrum of the recorded signal after applying a bandpass filter in the range of 30–80 MHz. Two prominent narrowband RFI sources are observed at approximately 45 MHz and 71 MHz.

To evaluate the AI model's performance, signal-like pulses were injected into measured noise traces at random positions, covering a broad range of amplitudes. The pulse power was scaled relative to the noise power, which was calculated using a pre-filtered version of each trace. This ensured that the injected pulses maintained a well-controlled signal-to-noise ratio (SNR) within a user-defined range while accurately reflecting real-world noise conditions. However, to prevent double filtering of the noise, which could distort its characteristics, the pulses were injected into the original (unfiltered) traces. This step also eliminated systematic biases by ensuring that the statistical properties of the background noise remained unaffected by the injection process. To maintain consistency in data processing, the digital bandpass filter was applied only after pulse injection, ensuring that both noise-only and signal-injected datasets underwent identical filtering and conditioning. This approach was chosen over COREAS simulations [16] to establish a proof of principle based purely on signal strength, avoiding dependencies on the absolute energy scale, antenna gain pattern, and shower inclination, which introduce additional uncertainties. By injecting short pulses into measured noise traces from a real-world, high-interference environment, we enable a focused evaluation of the AI model's ability to detect transient signal shapes under conditions where traditional threshold-based triggering systems struggle or fail entirely due to overwhelming radio frequency interference (RFI). This approach enables a systematic assessment of the model sensitivity in varying SNRs while ensuring robustness under challenging conditions. By directly testing and refining AI-based techniques on measured noise traces—without relying on an external event-reconstruction framework or requiring a particle detector—it provides a controlled and scalable test case. Although it does not fully replicate all details of cosmic-ray pulse shapes, much of which may already be washed out by bandpass filtering, it offers a robust estimate of the false positive rate of AI-based triggering methods in a realistic noise environment.

2.1 Training and Validation Datasets

To train and evaluate the model, multiple datasets were used, each designed to fulfil a specific role in ensuring robust performance. The datasets were categorized into training, cross-validation, and independent validation sets to assess the model's generalization and sensitivity. All noise traces are unique and are derived from experimental measurements, randomly selected from the full duration of the noise measurement campaign, whereas the signal-containing traces consist of injected pulses, as previously described. For the initial analysis, only a bandpass filter was applied, with no additional filtering. However, we later incorporate further filtering to suppress prominent RFI sources in the 30–80 MHz range for comparison, the results of which are discussed in section 7.
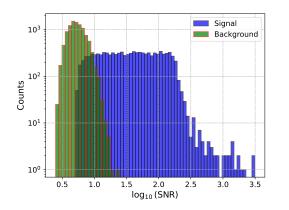
- **Training and Cross-Validation Datasets:** The dataset used for training and model tuning was randomly split into two equal parts:

  - **Training Dataset (50%):** Comprised **15,000 time series traces**, each containing 128 samples recorded at a sampling rate of 250 MHz, consistent with standard practices in radio-based air shower detection. It included an equal distribution of **5,000 pure noise traces** (background), **5,000 traces of isolated EAS-like pulses** (pure signal), and **5,000 traces with EAS-like pulses injected into noise** (signal). This setup ensured that the model effectively learned to distinguish genuine signals from background noise.

  - **Cross-Validation Dataset (50%):** Consisted of another **15,000 traces**, also evenly divided into **5,000 background traces**, **5,000 pure signal traces**, and **5,000 signal traces**. This dataset was used to fine-tune the model and monitor its generalization performance during training.

  These datasets were randomly derived from a larger pool of available traces, ensuring an unbiased and representative sample distribution.

- **Independent Validation Datasets:** Three independent validation datasets were used to assess the robustness of the model under different conditions.

  - **Validation Dataset 1:** Contained **20,000 traces**, equally split into **10,000 background** and **10,000 signal** traces. This dataset was applied to evaluate the performance of the model on unseen data with an SNR distribution similar to the training set.

  - **Validation Dataset 2:** Consisted of **20,000 traces**, equally divided into **10,000 background traces** and **10,000 signal traces with extremely low-amplitude pulses**. This dataset was specifically designed to test the sensitivity of the model in low-SNR scenarios.

  - **Validation Dataset 3:** Comprised **1 million pure noise traces** to assist in evaluating the **false positive triggering rate**. This dataset provides a large statistical sample to quantify how frequently the model incorrectly classifies noise as a signal event.

These independent validation data sets were crucial to assessing the model's ability to generalize to previously unseen data and to reliably detect signals under varying noise conditions, particularly at low SNR.

Figure 2 illustrates the SNR distributions for background and signal traces across Validation Dataset 1, which closely follows the distribution of the training dataset, and Validation Dataset 2, which consists of low-amplitude signals. This comparison provides insights into the differences in signal detectability across varying SNR conditions.
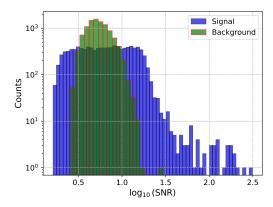

Figure 2. SNR distributions for background and signal traces in Validation Dataset 1 (left) and Validation Dataset 2 (right). Validation Dataset 1 follows the distribution of the training dataset, while Validation Dataset 2 contains low-amplitude signals.

3 Methodology

We implemented a deep learning-based pipeline for **denoising and classifying radio time series** in the context of self-triggering for EAS events. The pipeline consists of two primary components:

1. A **CNN-based denoiser** to suppress noise in the input signals.

2. A **fully convolutional classifier (FCN)** to distinguish between **signal and background** in a **binary classification task**.

These models operate **sequentially**, where the denoiser first refines raw data before passing it to the classifier for event discrimination.

3.1 Model Architecture

3.1.1 CNN-Based Denoiser

The **denoiser** is a fully convolutional autoencoder designed to suppress noise and enhance cosmic ray pulses in noisy radio time series.. It follows an **encoder-bottleneck-decoder** structure with a skip connection to retain fine-grained features.

- **Encoder:** Two convolutional layers extract multi-scale features, followed by a single max pooling layer that reduces resolution by a factor of two.

- **Bottleneck:** A deeper convolutional block refines feature representations at the reduced resolution.

- **Decoder:** An upsampling layer restores the original resolution, followed by convolutional layers that refine the denoised signal. A skip connection concatenates high-resolution features from the encoder to enhance detail recovery.

The denoiser learns from a dataset where the inputs are **noisy traces containing EAS-like pulses (`signal`)**, and the targets are the corresponding **pure EAS-like pulses (pure signal)**.

The training process minimises the **Mean Squared Error (MSE) loss** between the clean target x_i and the denoised prediction $\hat{x}_i$:

$$\mathcal{L}_{\text{denoiser}} = \frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{x}_i)^2 \, . \tag{3.1}$$

This architecture ensures that the model preserves cosmic ray features while effectively filtering out background noise.

3.1.2 Fully Convolutional Classifier (FCN)

The **FCN classifier** classifies time series into **background** or **signal**, where **signal** refers to a noisy trace containing an EAS pulse. It includes:

- **Three convolutional blocks** with batch normalization, dropout, and max pooling.

- **Global average pooling (GAP)** to reduce spatial dimensions while retaining features.

- **A fully connected layer with a sigmoid activation function** for binary classification.

The classifier is trained using the **Binary Cross-Entropy (BCE) loss**:

$$\mathcal{L}_{\text{classifier}} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \right] \tag{3.2}$$

where y_i is the true label and $\hat{y}_i$ is the predicted probability.

3.2 Training and Optimization

Both models were trained using the **Adam optimizer**, with task-specific loss functions, following a **two-step training approach**:

1. **Denoiser Training:**

   - Trained on **traces with pulses injected into noise (`signal`) and `pure signal`**.
   - Optimised to minimize **Mean Squared Error (MSE) loss**, enhancing signal quality while suppressing noise.

2. **Classifier Training:**

   - Trained on **denoised `signal` traces** (output of the denoiser) alongside denoised **background traces**.

- Optimized to minimize **Binary Cross-Entropy (BCE) loss**, ensuring accurate discrimination between signal-containing and noise-only traces.

The model was trained for **15 epochs** with a **batch size of 64** and a learning rate of **0.001**, using the **Training Dataset**. Its generalization performance was monitored with the **Cross-Validation Dataset** described in Section 2.1. The model was implemented using **PyTorch** with support for MPS acceleration. Training was conducted using the Adam optimizer, and model performance was evaluated on the cross-validation dataset using **classification accuracy, precision, recall, and F1-score**.

4 Model Performance Analysis

The first two plots in Figure 3 (top row) illustrate the probability distributions of sigmoid prediction for validation datasets 1 and 2, which differ in their SNR distributions. In **Validation Dataset 1**, where the SNR distribution closely resembles that of the training set (see Section 2.1), the model exhibits a strong separation between background and signal traces. Most noise traces receive lower probabilities, while signal-containing traces are predominantly assigned higher values. In contrast, **Validation Dataset 2** contains traces with significantly lower SNRs, commonly below the noise floor. While the model still distinguishes between the two classes, the classification performance degrades: the distribution of signal-containing traces is more spread out, with a noticeable tail extending toward lower probabilities. This shift indicates a reduction in signal efficiency, as weaker signals are increasingly misclassified due to their low SNR.

Beyond assessing performance across different SNRs, it is also essential to evaluate the model's behaviour in environments dominated by overwhelming background noise, where false positives become a critical concern. To investigate this, we introduce **Validation Dataset 3**, which consists of one million background traces. The bottom row of Figure 3 explores the model's response to this dataset by comparing **background traces from Validation Dataset 3** with **signal traces from Validation Dataset 1**. The sigmoid probability distribution (bottom left) shows that while most noise traces receive low probabilities, the large dataset size results in a fraction extending into the signal region. This highlights the need to quantify the false trigger rate under realistic high-background conditions. The final plot (bottom right) presents the distributions of the model's **logits**, i.e. the raw scores before applying a probability function. These distributions reveal a clear separation between `background` traces from Validation Dataset 3 and `signal` traces from Validation Dataset 1. Using logits instead of sigmoid probabilities significantly improves false positive rejection, albeit at the cost of true positive efficiency. This trade-off is particularly relevant when designing self-triggering strategies in noisy environments.

4.1 Performance Metrics and Trigger Rates

Following the qualitative assessment in Figure 3, we systematically evaluate the model's performance across different validation datasets. This analysis focuses on three key aspects: **trigger efficiency**, **robustness to low-SNR signals**, and **false trigger rate in high-background conditions**. The results are presented as follows.
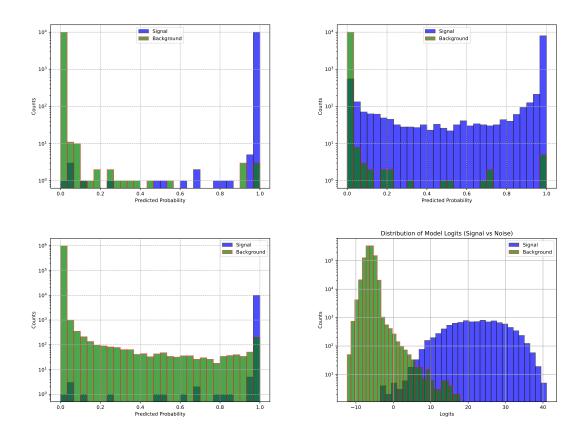
Figure 3. Probability distributions of sigmoid predictions for validation datasets with varying SNRs (top row) and in a high-background environment (bottom row). **Validation Dataset 1** has an SNR distribution similar to the training set, while **Validation Dataset 2** contains lower-SNR traces. The bottom row compares **background traces from Validation Dataset 3** with **signal traces from Validation Dataset 1**, highlighting false positives under high-background conditions. The logits distribution (bottom right) shows improved background rejection compared to sigmoid probabilities, though at the cost of efficiency.

4.1.1 Trigger Efficiency in Controlled Conditions (Validation Dataset 1)

- **True Positives:** 99.93%

- **False Positives:** 0.07%

A simple probability threshold is sufficient for efficient signal selection in this case.

4.1.2 Robustness to Weaker Pulses (Validation Dataset 2)

- **True Positives:** 87.96%

- **False Positives:** 0.08%

Despite the increased challenge, the model effectively distinguishes signal-containing traces, demonstrating its **robustness** to weaker pulses. The reduction in efficiency is expected as most of these signals fall outside the training range. Although further tuning could enhance performance,

the primary objective here is to assess the model's ability to generalise rather than optimise for specific low-SNR cases.

4.1.3 False Trigger Rate Estimation in a High-Noise Environment (Validation Dataset 3)

To assess the model's performance in the presence of overwhelming background noise, we evaluated it on **Validation Dataset 3**. This dataset simulates realistic high-RFI conditions, such as those found in urban environments (*e.g.,* LOFAR, LOPES) or certain AERA stations [7].

The primary objective of this evaluation is to quantify false trigger rates in such environments. The results are as follows:

- **True Positives (Initial Threshold):** 99.93%

- **False Positives:** 768 per 1M events (0.08%), which translates to approximately **800 Hz** first-level trigger (FLT) rate in a 1 MHz trigger scenario.

Applying confidence-based selection criteria (referred here as entropy cut) significantly reduces the false positive rate while maintaining high true positive efficiency.

- **Entropy Cut:** 99.35% True Positives, 79 False Positives per 1M events (0.008%), corresponding to approximately **80 Hz** false triggers in a 1 MHz trigger scenario.

The 80 Hz FLT rate is already suitable for a station-level trigger and provides a strong foundation for further refinement. A subsequent second-level trigger (SLT) stage, utilizing array-level coincidence searches, could further suppress false positives while preserving efficiency.

To explore the feasibility of detecting cosmic-ray events at a single station, we further applied logits-based cuts, which resulted in:

- **Logit Cut:** 72.4% True Positives, 0 False Positives, completely eliminating false triggers in this dataset.

This analysis shows that entropy-based selection effectively reduces the trigger rate to an acceptable level for an FLT, while still maintaining high detection efficiency. Further refinement at the SLT stage, through array-level coincidence searches, could optimise false positive rejection even further. On the other hand, applying logits-based cuts enables the possibility of detecting cosmic-ray events at a single station, **drastically suppressing** false positives, with none observed in our dataset. While this comes with some reduction in signal efficiency, it remains within a reasonable range, making single-station detection feasible in certain scenarios.

4.1.4 Example of a Low-Amplitude Signal Classification

Figure 4 illustrates a **representative trace containing a faint signal pulse** correctly classified by the model. The pulse is indistinguishable by eye due to transient noise, yet the classifier confidently assigns it a high probability. This result highlights the model's ability to recover weak air shower signals that would otherwise remain undetectable using traditional threshold-based methods.
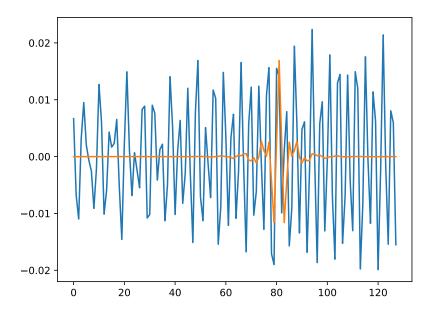
Figure 4. Representative trace with a faint signal pulse correctly classified by the model despite transient noise, demonstrating its ability to recover weak air shower signals undetectable by threshold-based methods.

5 Estimated Latency on FPGA Platforms

To evaluate the feasibility of real-time deployment, the model's inference latency was first measured on an Apple Silicon GPU using Metal Performance Shaders (MPS). Each inference took approximately 33 µs per trace, which implies a required buffer of around 8,250 samples when sampling at 250 MSPS. Since real-time streaming at this rate is continuous, reducing inference latency is essential to minimize buffering and avoid potential pipeline stalls.

Moving from a GPU to an FPGA requires a different set of assumptions compared to traditional CPU-to-FPGA speed-up estimates. In particular, FPGAs allow for specialized parallelism and hardware-friendly dataflow optimizations that can significantly reduce latency if designed properly. Table 1 summarizes conservative latency estimates for different FPGA classes, along with the corresponding buffer sizes at 250 MSPS. These values were derived by adjusting for GPU-to-FPGA acceleration, assuming an optimized hardware design that capitalizes on the inherent parallelism of FPGAs.

Here, the row labeled *High-end (INT8-Optimized)* represents a design flow that leverages 8-bit quantization on a device similar to other high-end FPGAs, rather than indicating a distinct family of boards. Narrower data paths in an INT8 pipeline can further reduce latency and resource utilization, at the expense of additional quantization efforts.

Even a low-end FPGA such as a Zynq device could reduce the baseline GPU latency from 33 µs down to approximately 11.17 µs, significantly lowering the buffering requirement. However, whether this meets "real-time constraints" depends on the total system requirements, including any overhead from I/O, external memory transfers, and other pipeline stages. In large-scale or

Table 1. Estimated inference latency and required buffer sizes for different FPGA platforms (sampling rate: 250 MSPS).

FPGA Type / Design Flow	Estimated Latency (per trace)	Required Buffer Size (samples)
Low-end FPGA (Zynq)	11.17 μs	2,792 samples
Mid-range FPGA (Arria 10)	4.78 μs	1,195 samples
High-end FPGA (Alveo U280)	2.23 μs	558 samples
High-end (INT8-Optimized)	1.12 μs	280 samples

power-sensitive deployments, factors such as cost, power consumption, and design complexity also guide the choice of FPGA platform and quantization scheme. The final selection must balance these considerations to ensure both feasibility and efficiency for the target application.

To fully enable real-time classification of cosmic-ray radio pulses, a dedicated FPGA-based framework must balance latency, resource utilization, and quantization precision while integrating seamlessly with experimental data streams. The results in Table 1 demonstrate that even low-end FPGA platforms can achieve substantial latency reductions, but practical deployment also requires optimizing DSP and memory utilization to handle high-rate data streams efficiently. Implementing 8-bit quantization (INT8) can further reduce both inference latency and resource consumption, although additional effort is needed to mitigate potential accuracy loss. A real-time classification system must also account for I/O bandwidth constraints, buffering mechanisms, and event prioritization to avoid pipeline stalls. Future work will focus on refining this framework by incorporating real-time data acquisition, firmware-level optimizations, and adaptive triggering strategies, ensuring a robust and scalable AI-driven self-triggering system for next-generation cosmic-ray experiments.

Future work will extend beyond optimizing the model for FPGA deployment while preserving classification performance. A key focus will be on benchmarking against alternative AI architectures, refining network optimization techniques, and expanding the training dataset to enhance generalization. Incorporating extensive air shower (EAS) simulations with state-of-the-art tools such as **CoREAS** will provide a more comprehensive and physically motivated dataset. Additionally, integrating experimental data from particle detector-triggered events will create an unprecedentedly realistic training set. Our preliminary results indicate that training on high-amplitude datasets enables the detection of even the faintest radio pulses—signals that have remained undetected by traditional methods to date. By leveraging these advancements, we aim to push the boundaries of real-time cosmic ray detection, making AI-driven self-triggering a transformative technology for next-generation astroparticle physics experiments.

6 Conclusion

In this work, we have demonstrated that artificial intelligence can dramatically enhance the self-triggering capability of radio detectors for extensive air showers (EAS). By training on large volumes of measured noise data and injected cosmic-ray-like pulses, our deep learning pipeline—incorporating both an autoencoder-based denoiser and a fully convolutional classifier—achieves excellent separation of genuine signals from background. Over a wide range of

signal-to-noise ratios, the approach retains efficiencies in excess of **99%** while simultaneously suppressing false positives down to sub-percent levels. Even in an urban-like noise environment, with RFI sources and transient interference, the model maintains robust performance, indicating strong potential for deployment under realistic conditions. Crucially, the flexibility of our inference stage enables an operating point where false positives can be entirely eliminated (as observed in our large-scale noise dataset), albeit at a modest reduction in true-positive rate to roughly **72%**. This result opens the door to single-station cosmic-ray detection purely by radio triggers, removing the need to rely on any particle detectors for trigger generation. Although multiple-station coincidences remain a valuable method for additional false-positive suppression, these findings show that a single antenna unit could be made sufficiently selective to capture genuine air-shower events in high-noise environments. A key outcome of this study is the successful demonstration of latency estimates that make real-time triggering feasible. Our analysis of FPGA implementations shows that when using optimized designs and dataflow pipelines, inference times can reach the microsecond regime. This aligns well with the strict timing requirements of large-scale radio observatories, where trigger decisions must be made rapidly to record short-lived air shower signals. The ability to process data in-stream, directly on reconfigurable hardware, underscores the practical viability of these AI-driven methods for next-generation detectors. Overall, these results address one of the longstanding challenges in radio-based cosmic ray detection: reducing or eliminating the need for external triggers from particle or fluorescence detectors. An autonomous trigger operating purely on radio data lowers complexity and costs, while also unlocking sensitivity to high zenith angles and potentially exotic UHE particle interactions. Moving forward, further integration with simulation tools (e.g. CoREAS) and additional real shower data will refine the models' accuracy and increase robustness to a wider spectrum of signal morphologies. The high efficiency and low false trigger rates shown here pave the way for a new class of AI-based, fully self-triggered radio arrays, promising both improved coverage and enhanced scientific reach in ultra-high-energy cosmic ray research.

7 Discussion

The results presented in this study highlight several key insights into the performance and robustness of our AI-driven self-triggering framework for radio detectors. Beyond achieving high detection efficiencies and minimal false-positive rates, a deeper examination of the model's behaviour reveals its ability to generalise well across different noise conditions, preserve signal fidelity through denoising, and maintain robustness in realistic deployment scenarios.

A notable finding is the classifier's ability to maintain high purity on non-denoised background traces, even though it was never trained on them. Because the classifier was trained exclusively on denoised signal and background data, one might have expected it to misinterpret raw background as containing signals. However, testing showed that it continued to distinguish signal from background with high accuracy, suggesting it was not misled by potential artifacts introduced by the denoiser on background samples. This indicates that the combined denoiser–classifier pipeline does not merely adapt to denoiser-induced modifications but instead learns robust, physically meaningful pulse features, demonstrating strong generalization across different noise conditions. Furthermore, these results suggest that mixing raw and denoised data in the training set may be an intriguing avenue for future exploration, as it could further improve robustness to a wider range of conditions.

We also investigated the effect of narrow-band RFI suppression. For the datasets discussed in this work, we did not initially remove strong RFI components. However, we produced an alternate noise dataset in which these narrow-band RFI components were suppressed prior to pulse injection. Because suppressing RFI lowers the noise floor by about 17 dB, we scaled down the injected pulses by a matching factor to preserve the same relative signal-to-noise ratio (SNR) distribution that our model was originally trained on. As expected, the classifier's performance remained virtually unchanged, since the effective classification task difficulty was not altered. Nevertheless, eliminating RFI could simplify the denoiser and classifier in a real-world deployment, as it reduces background complexity. A simpler network architecture may lead to faster inference times and lower computational costs, paving the way for real-time triggering with reduced latency.

Another key consideration for real-world applications is the resilience of our pipeline to variations in noise characteristics. Our tests in an urban-like interference environment suggest that the denoiser–classifier chain is learning fundamental signal-versus-noise traits rather than overfitting to specific noise profiles. Further studies could systematically probe this robustness using augmented training datasets (e.g., artificially injecting varied noise profiles) or by testing on independent, geographically diverse datasets.

Beyond classification accuracy, latency remains critical for real-time self-triggering applications. Preliminary estimates indicate that microsecond inference is feasible on FPGA platforms, but hardware-specific optimisations are needed to confirm these figures in practice. Refining the neural network architectures to strike an optimal balance between complexity and speed could ensure that real-time decision-making constraints are met without compromising accuracy.

Looking ahead, our framework could be extended by incorporating CoREAS-simulated pulses covering a wide range of air shower scenarios, including variations in energy, zenith angle, and primary composition. Training on these simulated signals would allow the model to learn a more comprehensive representation of shower-induced radio emission, likely improving its ability to detect real cosmic-ray pulses in challenging regimes. This approach would also provide a clear path for systematically evaluating the pipeline's performance under diverse air shower conditions.

Finally, although our results demonstrate the feasibility of single-antenna triggering, integrating AI-driven self-triggering with multi-station coincidence algorithms is an exciting possibility. Cross-station coincidence can lower false-positive rates while preserving sensitivity to rare events, including highly inclined cosmic rays that might elude traditional triggers.

In summary, our findings underscore the transformative potential of deep learning for self-triggered radio detection of extensive air showers. By preserving robust generalisation, demonstrating strong performance even with various noise conditions, and operating at latencies amenable to real-time applications, AI-based triggering stands poised to play an integral role in next-generation cosmic-ray observatories. Future work will focus on refining model architectures, evaluating performance across different noise environments, integrating large-scale radio arrays, and exploring how single-station AI triggers may combine with multi-station coincidence for enhanced detection confidence.

Acknowledgments

References

[1] F.G. Schröder, *Radio detection of cosmic-ray air showers and high-energy neutrinos*, *Prog. Part. Nucl. Phys.* **93** (2017) 1.

[2] T. Huege, *Radio detection of cosmic ray air showers in the digital era*, *Phys. Rep.* **620** (2016) 1.

[3] F. D. Kahn and I. Lerche, *Radiation from cosmic ray air showers*, *Proc. R. Soc. Lond. A* **289** (1966) 206.

[4] E. Conti and G. Sartori, *On the coherent emission of radio frequency radiation from high energy particle showers*, *Int. J. Mod. Phys. D* **26** (2017) 1750083.

[5] Pierre Auger collaboration, *Radio measurements of the depth of air-shower maximum at the pierre auger observatory*, *Phys. Rev. D.* **109** (2024) .

[6] Pierre Auger collaboration, *Energy estimation of cosmic rays with the engineering radio array of the pierre auger observatory*, *Phys. Rev. D.* **93** (2016) .

[7] A. Schmidt, *Realization of a self-triggered detector for the radio emission of cosmic rays*, Ph.D. thesis, Karlsruher Institut für Technologie (KIT), 2011.

[8] CODALEMA collaboration, *Latest results of the CODALEMA experiment: cosmic rays radio detection in a self trigger mode*, *J. Phys. Conf. Ser.* **409** (2013) 012074.

[9] J. Kelley, *Data acquisition, triggering, and filtering at the auger engineering radio array*, *Nucl. Instrum. Methods Phys. Res. A''* **725** (2013) 133.

[10] IceCube collaboration, *Application of machine learning to identify radio pulses of air showers at the south pole*, *Proc. PoS(ARENA2024)* (2024) 034.

[11] GRAND collaboration, *Development of an autonomous detection-unit self-trigger for GRAND*, *Proc. PoS(ARENA2024)* (2024) 060.

[12] J.O. Arciniega, F. Carrió and A. Valero, *Fpga implementation of a deep learning algorithm for real-time signal reconstruction in particle detectors under high pile-up conditions*, *JINST* **14** (2019) Article No. P09002.

[13] N. Chiedde and on behalf of the ATLAS liquid argon calorimeter group, *Machine learning for real-time processing of atlas liquid argon calorimeter signals with fpgas*, *JINST* **17** (2022) Article No. C04010.

[14] J. Duarte, S. Han, P. Harris, S. Jindariani, E. Kreinar, B. Kreis et al., *Fast inference of deep neural networks in fpgas for particle physics*, *JINST* **13** (2018) Article No. P07027.

[15] AMD Xilinx, *Vitis AI Development Environment*, 2025.

[16] T. Huege, M. Ludwig and C.W. James, *Simulating radio emission from air showers with CoREAS*, *Proc. PoS(ARENA2012)* (2013) .