

FPGA Implementation of an Artificial Neural Network for Subatomic Physics Experiment Particles Recognition

R. Zhao^{ab}, A. Besson^{ab}, C. Hu-Guo^a, A. Perez Perez^a, K. Jaaskelainen^a, M. Goffe^a, and Y. Hu^{ab}

^a IPHC

23 rue du loess, BP28, 67037 Strasbourg, France

^b Université de Strasbourg

4 rue Blaise Pascal, 67081 Strasbourg, France

E-mail: ruiguang.zhao@iphc.cnrs.fr

CMOS Pixel Sensors have been used in subatomic physics experiments for charged particles detection. In the International Linear Collider (ILC) vertex detector, the occupancy will be mainly driven by impacts coming from the beam background. This will have a huge impact to the data flow of the system. We propose a design of CMOS pixel sensors with on-chip Artificial Neural Network (ANN) to tag and remove these hits. It is based on different features of hits clusters. In this paper, we will describe the structure of an ANN implemented in an FPGA device. We will show and analyze the distribution of incident angles reconstructed by the ANN.

Topical Workshop on Electronics for Particle Physics (TWEPP2018)
17-21 September 2018
Antwerp, Belgium

*Speaker.

1. Introduction

CMOS pixel sensors are monolithic devices that integrate sensing diodes and readout electronics on the same silicon substrate. They have been used in the PiXeL detector (PXL) upgrade of the Solenoidal Tracker at STAR experiment at Relativistic Heavy Ion Collider (RHIC), since they offer an attractive balance among granularity, material budget, readout speed, radiation tolerance and power dissipation. It was the first time that a collider particle physics experiment was equipped with CMOS pixel sensors [1].

In the vertex detector of the International Linear Collider (ILC) experiment, a large amount of extra hits are generated by background particles coming from the beam background (mainly low momentum e^+/e^- leptons). These extra hits will increase drastically the data flow of the system. The idea is using CMOS pixel sensors with on-chip Artificial Neural Network (ANN) to tag and remove these hits. In this case, output data would contain only information of hits generated by particles coming from the physics processes [2].

In general, background particles have lower momentum (10-100 MeV/c) than particles coming from the physics processes. Under the effect of magnetic field, they have larger incident angles into the epitaxial layer of the CMOS pixels sensor and generate more hit pixels (elongated cluster shapes) than those of real physics particles (Figure 1). According to cluster shapes, one can use the ANN to reconstruct incident angles and then distinguish types of hits.

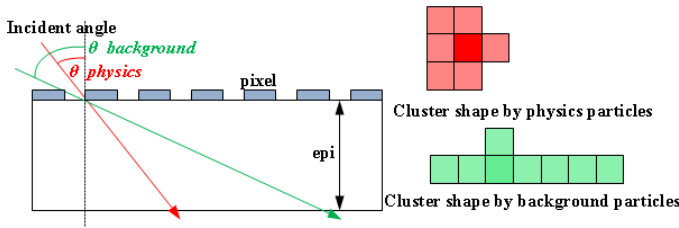


Figure 1: Cluster shapes generated by particles with different incident angles

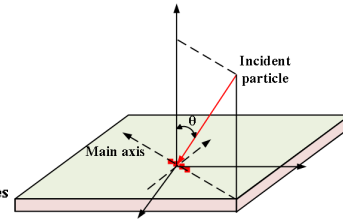


Figure 2: Main axis of a cluster

This paper aims to validate the feasibility study of reconstructing incident angles by the ANN implemented in hardware. In the section 2, a test system is presented in detail, including a CMOS pixel sensor to collect the raw data and a custom designed ANN implemented in an FPGA. The test procedure of the ANN is introduced in the section 3, including the training procedure processed in the Toolkit for MultiVariate Analysis (TMVA) [3] and the regression procedure executed in the FPGA device. The section 4 shows and analyzes the reconstructed result of incident angles.

2. Test system

2.1 System of the raw data collection

A system has been designed to collect the raw data from a CMOS pixel sensor named MIMOSA18 for training and testing the ANN structure. A platform supports MIMOSA18 with different rotations in order to vary incident angles with respect to a fixed ^{90}Sr source (Figure 2).

MIMOSA18 was fabricated in the AMS $0.35\ \mu\text{m}$ OPTO process with standard epitaxial layer ($14\ \mu\text{m}$, $\sim 10\text{-}15\ \Omega\cdot\text{cm}$). Charged particles generate the order of $70\text{-}80\ e^-h\ \text{pair}/\mu\text{m}$ in epitaxial

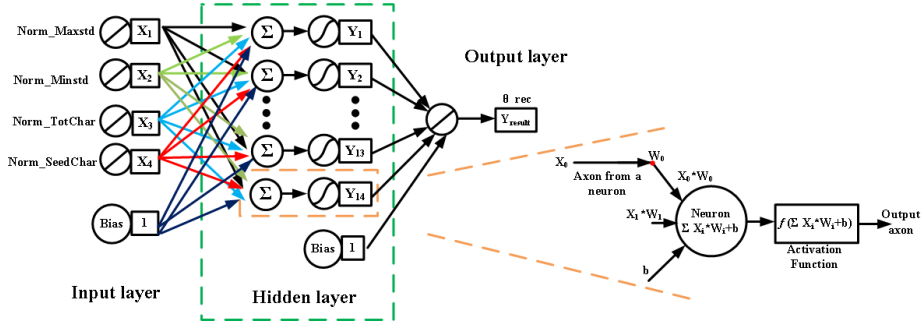


Figure 3: Artificial Neural Network structure

layer, resulting in typical total charge of $O(1000e^-)$. MIMOSA18 consists of four matrices of 256×256 pixels with a pitch of $10 \mu m$, and providing an active area of $5 \times 5 mm^2$. Analog output of MIMOSA 18 is encoded in 12 bits in the raw data.

2.2 Structure of the Artificial Neural Network

An FPGA board with a Xilinx Artix-7 chip has been used to implement the structure of the ANN and also the modules of data preprocessing including cluster search and feature extraction.

ANNs are inspired by biological neural networks that constitute animal brain. An ANN structure acquires knowledge through the training procedure based on a large amount of data. Knowledge is stored within each inter-neuron connection strength which is known as connection weight [4, 5]. As shown in the Figure 3, the structure of ANN consists of the input layer, the hidden layer and the output layer. The input layer contains 4 input neurons and 1 bias neuron. Input neurons take input vectors into the Artificial Neural Network. The hidden layer which is following the input layer has 14 calculation neurons and 1 bias neuron. The number of neurons in the hidden layer is defined after comparison and optimization in view of the balance between complexity and performance the ANN. Calculation neurons are composed of activation function (hyperbolic tangent) in equation (2.1) and sum units. There are 14×5 connection weights between the input layer and the hidden layer. The output layer presents the calculation result of the ANN.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.1)$$

Input neurons of the ANN are required to represent information of a cluster. We propose to use the following four inputs in this work:

- Maximum standard deviation (MaxStd) of the cluster (standard deviation on the main axis direction)
- Minimum standard deviation (MinStd) of the cluster (standard deviation on the perpendicular direction to the main axis)
- Charge from the seed pixel (SeedCharge) of the cluster
- Total charge (TotCharge) of the cluster

The Maximum and Minimum standard deviations are related to the incident angle and represent cluster shape (Figure 2). Charge from the seed pixel and the total charge are related to the incident angle and represent charge distribution.

3. Test procedure

3.1 Training procedure

The training procedure is processed by the TMVA to generate connection weights of the ANN. Reconstructed angles are regressed by the ANN based on initial connection weights and four input parameters of each cluster, then initial connection weights are adjusted by comparing reconstructed angles with incident angles. Connection weights could be settled until the error is acceptable, by calculating and comparing constantly.

3.2 Regression procedure

The regression procedure is processed in the FPGA device. It is used to reconstruct incident angles on the basis of four input parameters of each cluster and connection weights that are generated in the training procedure. It contains the modules of cluster search, main component analysis (MCA) and the ANN.

The module of cluster search is used to collect hit information from each frame of raw data. According to our experiences and taking into account hardware resources and the operation time, we assume the maximum size of each cluster is 7×7 pixels. The pixel containing the highest charge of a cluster is named “seed pixel”, and it is set at the center of the 7×7 pixels window. Steps of cluster search are as follows:

1. Searching the pixel with maximum charge in one frame of raw data as a seed pixel;
2. Checking neighbor pixels around the seed pixel in this 7×7 pixels window;
3. Erasing this hit information and seeking next seed pixel in this pixel charge matrix.

The MCA module is used to analyze charge distribution of a cluster and calculate two input parameters of MaxStd and MinStd for the ANN.

The ANN module is implemented in the FPGA by mathematic units and memory resources. The implementation of the activation function uses some IP cores such as exponential operations. Connection weights that are generated in the training procedure are fixed in registers.

4. Results

The raw data is collected under 10 different incident angles to train the ANN. 500 continuously frames at each respective incident angle are taken out to test the ANN.

The distribution of reconstructed angles θ_{rec} by the ANN is shown in the Figure 4. In the left graph, the mean value of reconstructed angles has slow growth with increasing of θ_{inc} (0-73 degrees). It is basically the same with the mean value of reconstructed angles regressed by the ANN in TMVA. The variation of reconstructed angles has the same trend with that of incident angles, while in the ideal case, reconstructed angles should have the same values as incident angles. It could be found in the right hand of the Figure 4, as the incident angle θ_{inc} increases, the percentage of reconstructed angles between 60 to 70 degrees is growing. Just few or even no reconstructed angles are located within 0 to 20 degrees or 70 to 90 degrees. The range of effective reconstructed angles that we concern is from 50 to 75 degrees. Particles with incident angles below 50 degrees

could be directly reconstructed from hits information of a double-sided ladder [6]. Due to the limitation of the 7×7 pixels window, the ANN is not sensitive to clusters with incident angles above 78 degrees in this work.

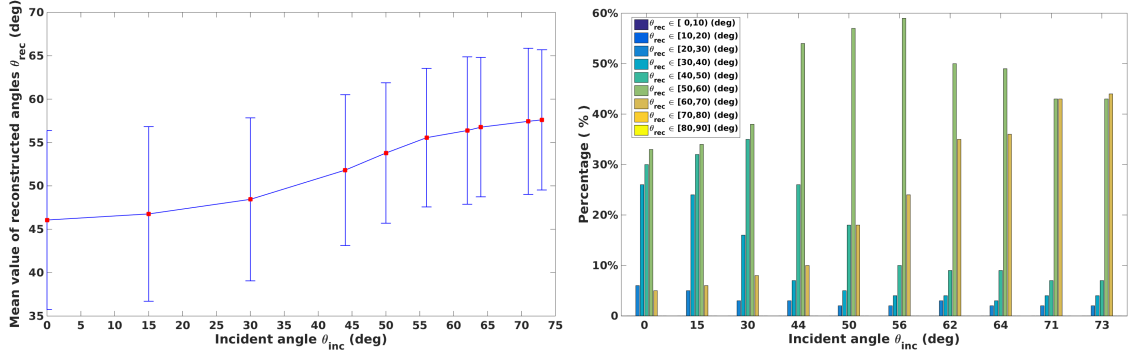


Figure 4: Distribution of reconstructed angles θ_{rec} versus incident angles θ_{inc}

5. Conclusions

Artificial Neural Network implemented in an FPGA device has the same structure and test results as those of the ANN in the TMVA. It is proved that the ANN structure could be implemented in the FPGA. However, it is difficult to translate the structure directly into the sensor ASIC. For example, an entire frame of raw data is stored for algorithm of cluster search implemented in the FPGA, which means that a large amount of memory resources and delay are required. Considering the balance among reconstructed result, the power consumption and the timing requirement, the structure of ANN and other modules could be optimized. However, CMOS pixel sensors with on-chip ANN could be implemented with the improvement of fabrication processes and the optimization of the design.

References

- [1] Hu-Guo, C., et al, (2009). *CMOS pixel sensor development: a fast read-out architecture with integrated zero suppression*, *Journal of Instrumentation* 4(04): P04012.
- [2] Besson, A., *CMOS pixel sensors with on-chip Neural Network : A new horizon for embedded systems*, *ATTRACT TWD Symposium : Trends, Wishes and Dreams in Detection and Imaging Technologies*, Strasbourg, 4-5 November 2016.
- [3] Hoecker, A., Speckmayer, P., Stelzer, J., Therhaag, J., von Toerne, E., Voss, H., ... and Dannheim, D. (2007). *TMVA-Toolkit for multivariate data analysis*. *arXiv preprint physics/0703039*.
- [4] Gomperts, A., Ukil, A., and Zurfluh, F. (2010). *Implementation of Neural Network on Parameterized FPGA*, *In AAI Spring Symposium: Embedded Reasoning (pp. 45-51)*.
- [5] Jain, A. K., et al, (1996). *Artificial neural networks: A tutorial*. *Computer*, 29(3), 31-44.
- [6] Nomerotski, A., et al, (2011). *PLUME collaboration: Ultra-light ladders for linear collider vertex detector*. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 650(1), 208-212.