

Unsupervised Learning for Pixel Mask Clustering and Cluster Tracking in LHCb's Velopix Sensor Calibration

M.W. MAJEWSKI*, P. RADOŃ AND T. SZUMLAK

*AGH University of Science and Technology in Krakow,
30 Mickiewicza Avenue, 30-059 Kraków, Poland*

Doi: [10.12693/APhysPolA.142.418](https://doi.org/10.12693/APhysPolA.142.418)

*e-mail: mwmajewsk@gmail.com

The silicon vertex detector is one of the core elements of the LHCb spectrometer. Its upgrade version features an innovative pixel sensor. The readout chip branches from the Medipix family of dedicated pixel ASICs. One of its operational challenges with the future data taking at the Large Hadron Collider will be the ability to detect faulty pixels and monitor them. In this work, we propose an innovative method for clustering faulty pixels and tracking their evolution in time. We compare the two methods of clustering (DBSCAN and OPTICS) and their influence on the proposed tracking method using a simulated dataset of masked pixels.

topics: unsupervised learning, VeloPix, tracking, particle detectors

1. Introduction

The LHCb detector is a single-arm forward spectrometer located at CERN's Large Hadron Collider (LHC). One of its main goals is the study of B-flavoured physics. LHCb differs from other main experiments at LHC in its design. It focuses on detecting particles at a low angle in relation to the beam [1]. High precision measurements are possible so far due to the state-of-the-art vertex detector VELO (Vertex Locator). After a technical stop at LHC, VELO was upgraded from a silicon strip detector to a silicon pixel detector Velopix [2]. Velopix belongs to the family of matrix detectors that originated from the Medipix Collaboration. This family of detectors has been used in other experiments [3] in particle physics as well as in medical imaging [4] and aboard the International Space Station [5].

The upgraded VELO modules will be located very close to the beam (up to 5.1 mm [1]). They will be susceptible to high amounts of ionising radiation for a prolonged periods of time. Due to this, we are preparing methods that will help to address the problems that may develop due to the radiation effects. Velopix is a digital hybrid detector, and the solutions that were applied to the previous generation of the detector are not directly transferable to the new one. The output from the sensors of Velopix is digitised. When the threshold of the electrical signal is exceeded at a group of pixels, the sensor chip will send information about the particle hit. If the threshold level for a pixel is set too low, the pixel will produce false information about a particle hit (due to naturally occurring noise) and flood

the readout channel with unnecessary information, which may cause a throttling problem. Because of that, noisy pixels are masked and their activity is ignored. Masking is one of the critical parts of the VELO calibration procedure. Just one bad channel can saturate the readout data link, rendering a given sensor useless^{†1}.

Masked pixels are less problematic when they are evenly distributed across the sensor, but when cluster together they can cause a problem for the particle track reconstruction algorithm. A simple counting of the masked pixels is obviously not enough to detect potential structures of bad pixels. The need for careful and robust monitoring of detector condition and the need for a solution for tracking the clusters of masks in the new Velopix motivates this work to use unsupervised learning algorithms in the proposed solution of this problem.

2. Clustering

The Velopix sensor (ASIC) is a pixel matrix 256×256 pixels. Typically, the masked pixels are uniformly distributed across the matrix, but in some cases, a cluster of masked pixels may occur. Masked pixels can emerge and disappear in the next calibration, but they can also persist. If such clusters persist permanently, this is a significant warning sign

^{†1}The VELO DAQ system operates in the so-called triggerless mode, which means that a noisy channel can send data many times within a single bunch crossing window of 25 ns.

indicating the need for additional action to mitigate the adverse effects of the inactive group of pixels. A group of masked pixels together in close proximity can be considered a cluster.

The DBSCAN algorithm [6] operates on a rule of clustering the neighbouring points within a certain radius. It starts by picking the core point if at least MinPts points are in the range ϵ . Then it proceeds to mark those points as reachable points and repeats the process on the newly discovered points until no new points are found. A group of points found this way is marked as a cluster and is excluded from the further search. The algorithm repeats with a unique core point if such exists. When all of the core points have been checked, all remaining points are marked as outliers.

The OPTICS [7] algorithm works in a very similar basis to DBSCAN. It was designed to address the problem of varying density in the dataset. It introduces a core-distance metric for which point o is the minimal range at which MinPts is reachable. The algorithm also defines reachability distance, which for points p , o is $\max(\text{core-distance}(o), \text{distance}(o, p))$. Based on those metrics, OPTICS builds a reachability graph and ordering list from which the assignment of a point to the cluster is calculated.

We empirically chose DBSCAN parameters to be set as $\epsilon = 5$ and $\text{MinPts} = 4$. We use the ‘‘xi-step’’ strategy for finding clusters for the OPTICS algorithm, which only requires the $\epsilon = 5$ parameter to be set.

3. Cluster features

Using the clusters found, we calculate the cluster’s features: position, size, and shape.

1. Position $p_k = (\bar{x}_k, \bar{y}_k)$, where $\bar{x}_k = \frac{1}{N_k} \sum_i^{N_k} x_k^i$, $\bar{y}_k = \frac{1}{N_k} \sum_i^{N_k} y_k^i$.
2. Size $s_k = \{n_k, d_k\}$ where n_k is the number of pixels in the cluster divided by the mean number of pixels in the given sensor’s cluster. The density of a cluster is $d_k = \frac{1}{N_k} \sqrt{x_k^i - \bar{x}_k)^2 + (y_k^i - \bar{y}_k)^2}$.
3. Shape $h_k = \{\alpha_k, c_k\}$, where α_k is the directional coefficient of the cluster measured by the fit to the line $y_k(x) = \alpha_k x + b_k$. Here, c_k is the roundness of the cluster calculated as its Pearson coefficient.

With those metrics, we can then define the spatial characteristic vector as $\mathbf{v}_k = [s_k; h_k]$. Then we define the cluster as a set of unique features $\text{cluster}_k = p_k, \mathbf{v}_k$.

4. Cluster tracking

As mentioned earlier, clusters can appear and disappear from calibration to calibration. They also can persist in between calibrations. An example of two calibrations with tracked clusters can be found

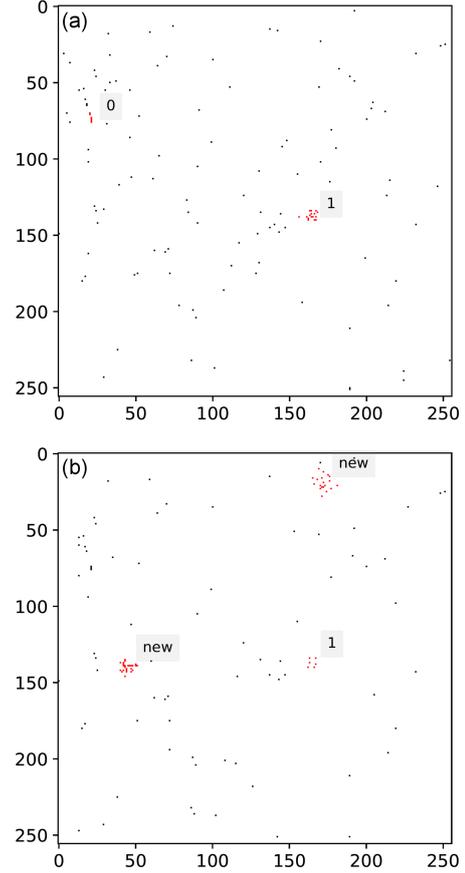


Fig. 1. Two consecutive calibrations with masks (from simulation), with labeled and tracked clusters. (a) Clusters labeled with the same integer are chosen by the algorithm as consecutive generations of the same cluster. (b) Clusters labeled as ‘new’ are clusters at the time step t_n that were absent at time step t_{n-1} .

in Fig. 1. To find if the clusters themselves have changed, we need a way of tracking them over time. For that purpose, we introduce the pair-wise similarity matrix M . Its i -th rows represent clusters at t_{n-1} and its j -th columns represent clusters at t_n . The values of M indicate the similarity between the clusters in successive sensor calibrations. The matrix M is constructed as a product of two matrices

$$M_{i,j} = \Phi_{i,j} V_{i,j}, \quad (1)$$

where $\Phi_{i,j}$ is define as

$$\Phi_{i,j} = \frac{1}{d_{\min}} \max(d_{\min} - D_{i,j}, 0). \quad (2)$$

Here

$$D_{i,j} = \|p_i - p_j\|. \quad (3)$$

The matrix V is defined using the cosine similarity between each pair of spacial characteristics vectors \mathbf{v}

$$V_{i,j} = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\|\mathbf{v}_i\| \|\mathbf{v}_j\|}. \quad (4)$$

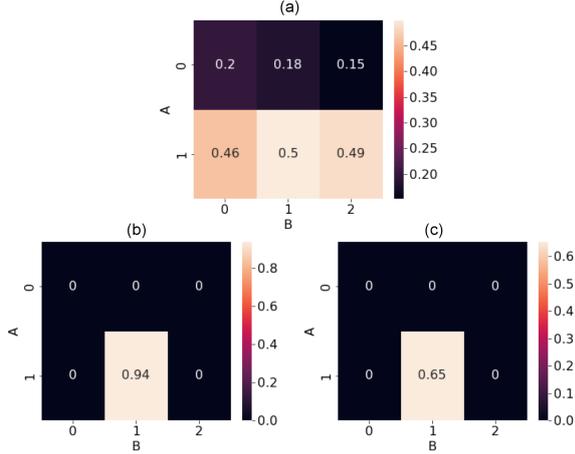


Fig. 2. Matrices used for the association of clusters between calibrations. Rows represent clusters in Fig. 1a, columns represent clusters in Fig. 1b. Values indicate the spacial characteristics similarity measure V (a) shape matrix, and positional similarity measure Φ (b) distance matrix. The M matrix is plot (c) (similarity score matrix).

An example of a similarity matrices shape and distance matrix can be found in Fig. 2. The matrix M is used to tell if the clusters on two consecutive calibrations are the same cluster. We choose the threshold τ_{\min} . If $M_{i,j} > \tau_{\min}$, then the clusters i and j are marked as the same cluster. Otherwise, if $M_{i,j} \leq \tau_{\min}$, the clusters are not connected. We empirically choose $\tau_{\min} = 0.3$. In Fig. 2 there are exemplary shape and distance similarity matrices as well as the calculated matrix M .

5. Results

Drawing from our experience with the VeloPix sensors during the testing phase, we have prepared a simulation of the VeloPix masks emerging in the calibrations. The simulation generates realistic clusters of masks as well as background noise masks. We tested both DBSCAN and OPTICS algorithms for the ability to correctly spot the clusters and their influence on cluster tracking during the time progression of a mask simulation of a single sensor [8]. We generated 3000 timesteps of single sensor calibration. The ground truth of a dataset is a nonzero pixel that belonged to a cluster generated less than 8 timesteps before the current one. You can see the overall confusion matrix and the accuracy of both methods in Table I. The DBSCAN accuracy for recognising generated clusters is much higher than OPTICS, and the false positive rate of OPTICS is more than three times higher than DBSCAN. Both algorithms (DBSCAN and OPTICS) were tested for tracking clusters in consecutive calibrations using the similarity matrix M . In Fig. 3, you can see a sample simulation of 300 timesteps and the number of recognised new and old clusters.

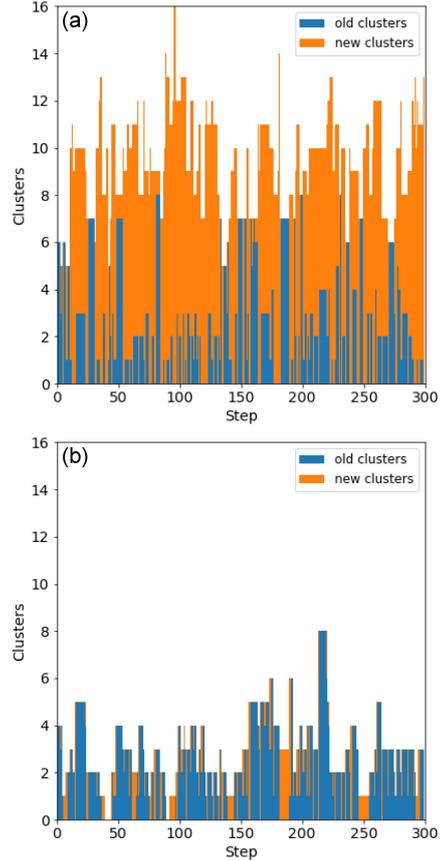


Fig. 3. Two progressions of the simulation, with number of identified clusters. (a) The plot was made with the OPTICS. (b) The plot was made with DBSCAN. The color differentiates clusters that are recognised as a continuation from the previous calibration (old clusters) and clusters that are not associated with clusters in the previous calibration (new clusters).

TABLE I

Confusion matrix values in 3000 consecutive simulation steps.

Value	OPTICS	DBSCAN
True Negative	7608	16207
False Positive	11680	3081
False Negative	1156	3167
True Positive	5665	3654
Accuracy	0.51	0.76
Precision	0.33	0.54

Here the old clusters are the clusters that prevail on the matrix, from one calibration to the next. The DBSCAN algorithm has proven to be more stable and able to associate the clusters together (more overall old clusters). On the other hand, the OPTICS algorithm was able to classify the same pixels as belonging to any clusters for a prolonged number of consecutive calibrations (Fig. 4). This means that

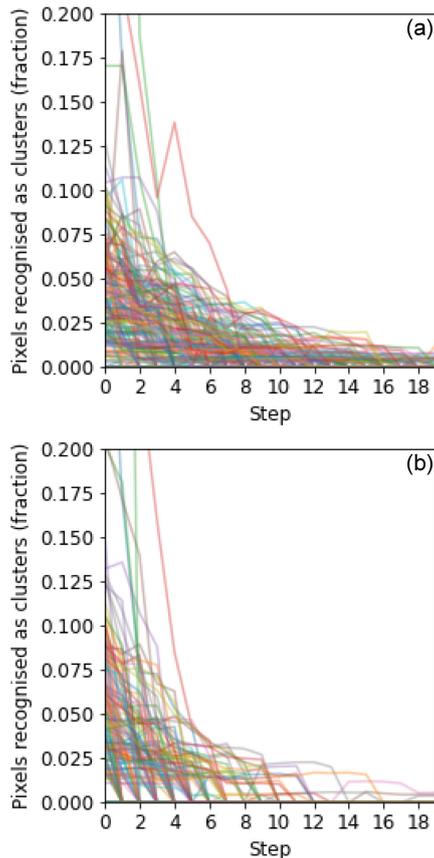


Fig. 4. The fraction of pixels of a cluster categorised as belonging to any clusters (Y-axis) in next consecutive calibrations (X-axis) since the cluster introduction to calibration (number of timesteps $n = 300$). Each line represents a different cluster and is differentiated with random color. The number of detected pixels slowly decreases with time. The OPTICS algorithm (a) recognises pixels of clusters as belonging to a cluster (not necessarily the same one) for a longer time. DBSCAN is more strict in distinguishing the pixels that belong to clusters.

when using OPTICS with the tracking algorithm, it classifies the pixels as belonging to a cluster more frequently than DBSCAN, but is less able to track the clusters correctly. From the standpoint of operating the detector, the OPTICS algorithm would more often falsely report more clusters.

6. Conclusions

In a very general way, an innovative approach has been proposed for monitoring the condition of pixel-based detectors using unsupervised machine learning. It can be applied to any system with long-term operation. The core innovation related to our approach is to use the calibration parameters to deduce the state of the detector rather than the data collected by the detector. Also, to our knowledge, it is one of the first applications of unsupervised learning techniques to monitor silicon pixel

detectors used in the field of experimental high-energy physics. Two different algorithms for density clustering were tested in the simulated dataset of Velopix calibrations masking. The DBSCAN algorithm was able to most consistently identify clusters with high accuracy and low number of false positives. The DBSCAN algorithm used with cluster tracking via the cosine similarity of unique features performed superior to the OPTICS algorithm. The proposed algorithm shows the capability of recognising and tracking clusters in the Velopix sensors masking simulation. The ultimate test of this work will take place after the upgrade of the LHCb detector and the upcoming new run of LHC.

This method may be used in other sensors belonging to the Medipix family. Other sensors that are descendants of Medipix might be using different modes of operation that do not use masking or may simply never need to mask pixels. However, when masking is used, the algorithm presented in this work can be used to detect damage in a pixel ASIC and track its evolution. Additionally, detectors that utilise multiple pixel sensors for tracking it may be used to detect blind spots created by clustered masks. The application of this algorithm will prevent the manifestation of adverse effects of the clusters of masks before the ASIC collects the data.

Acknowledgments

We acknowledge support from the National Science Centre (Poland), UMO-2017/27/N/ST2/01880.

References

- [1] LHCb Collaboration, *LHCb VELO Upgrade Technical Design Report*, CERN-LHCC-2013-021, LHCb-TDR-013, 2013.
- [2] T. Poikela, M. De Gaspari, J. Plosila et al., *J. Instrum.* **10**, C01057 (2015).
- [3] A. Sopczak, *Physics* **3**, 579 (2021).
- [4] R. Aamir, A. Chernoglazov, C.J. Bateman et al., *J. Instrum.* **9**, P02005 (2014).
- [5] N. Stoffle, L. Pinsky, M. Kroupa et al., *Nucl. Instrum. Methods Phys. Res. A* **782**, 143 (2015).
- [6] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, in: *Proc. of the Second Int. Conf. on Knowledge Discovery and Data Mining*, AAAI Press, 1996, p. 226.
- [7] M. Ankerst, M.M. Breunig, H.-P. Kriegel, J. Sander, in: *Proc. of the 1999 ACM SIGMOD Int. Conf. on Management of Data*, ACM Press, 1999, p. 49.
- [8] M.W. Majewski, P. Radoń, T. Szumlak, Github Repository, *Velopix Mask Studies for Clustering*, *velo-moni-ana* Ver. 1.0.0., 2022.