Generation of PXD background using Generative Adversarial Networks

Fabio Novissimo, Nikolai Hartmann, Thomas Kuhr

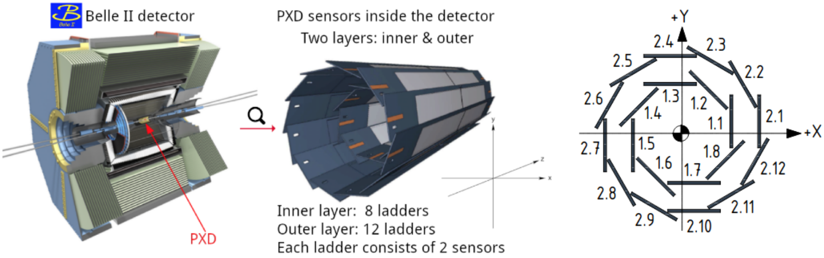
LMU Munich

KISS annual meeting, February 13th 2024

Bundesministerium
für Bildung
und Forschung

Introduction

- The **Pixel Vertex Detector (PXD)** is the innermost semi-conductor sub-detector of Belle II.
- The sensitive area of the PXD is made up by 40 modules. Each module consists of a 250×758 pixel matrix.
- **Inner layer**: 16 modules implemented into 8 ladders.
- **Outer layer**: 24 modules implemented into 12 ladders.



Backgrounds

PXD hits come from two sources:

- **Signal decays:** involve less then 1% hits per sensor.
- **Background hits:**
  - Beam-induced processes (Touschek scattering, Beam-gas scattering, Synchrotron radiation)
  - Luminosity dependent processes (BhaBha scattering, two photon process, ...)

Two ways to include the beam-background effects:

- Monte Carlo generation $\longrightarrow$ shows sizeable discrepancies with measurements.
- Taking random trigger events.

The problem

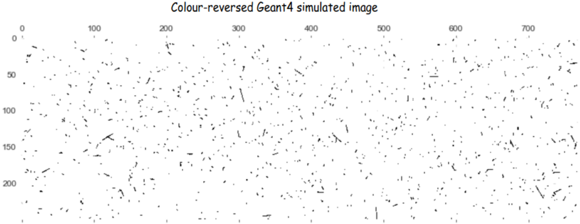
Large amount of resources required for the readout, storage and distribution of the background data:

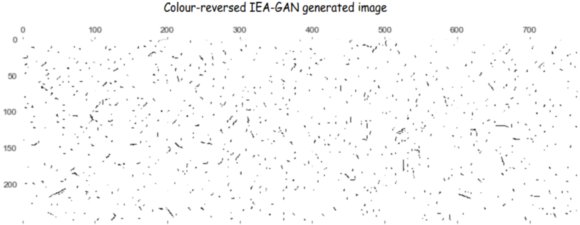
- Typical overlay event size is 200 kb ($\sim 20\times$ larger than the event used for analysis).
- Background files have to be distributed to all the sites where MC samples are produced

Solution: generate the background hits on the fly for each sensor.

IEA-GAN

- Generating background with a GAN (H. Hashemi et al. arXiv:2303.08046).
- Conditioning on the sensor number.
- Transformer-based relational reasoning module to reproduce correlations between sensors.



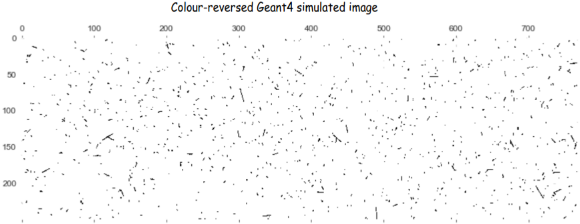


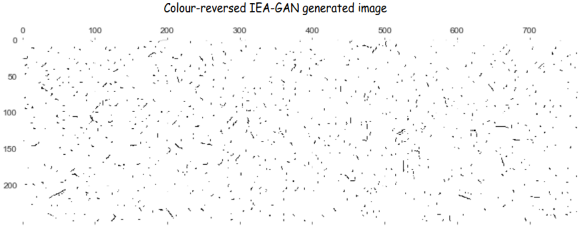
IEA-GAN

- Generating background with a GAN (H. Hashemi et al. arXiv:2303.08046).
- Conditioning on the sensor number.
- Transformer-based relational reasoning module to reproduce correlations between sensors.

Few problems:

- Correlation between sensors are not reproduced very well.
- The model is extremely complex.





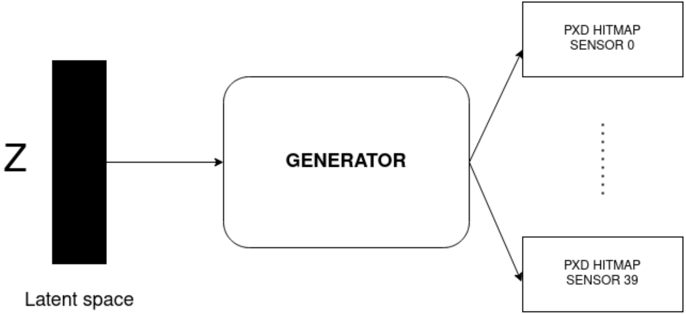
New approach

Generate the background using a GAN without conditioning on the sensor number.

- Generate instances of background for all sensors at once.
- Wasserstein GAN with CNN layers used in the Generator and Discriminator.

Main goals:

- Check if it is feasible to train the GAN without conditioning.
- Reproduce correctly the correlations between the sensors.



Generator architecture

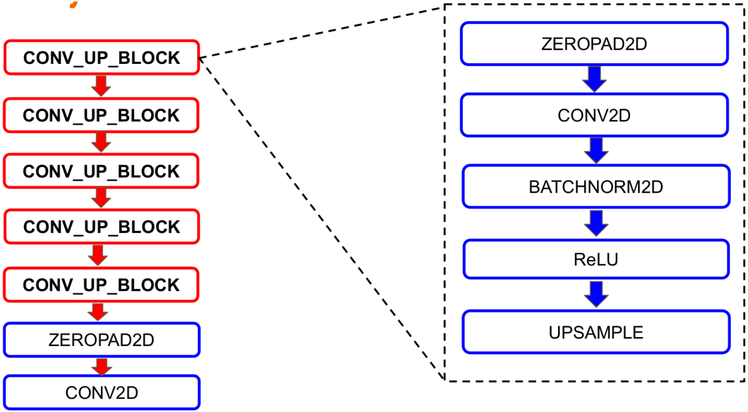


Figure: Generator architecture

Discriminator architecture

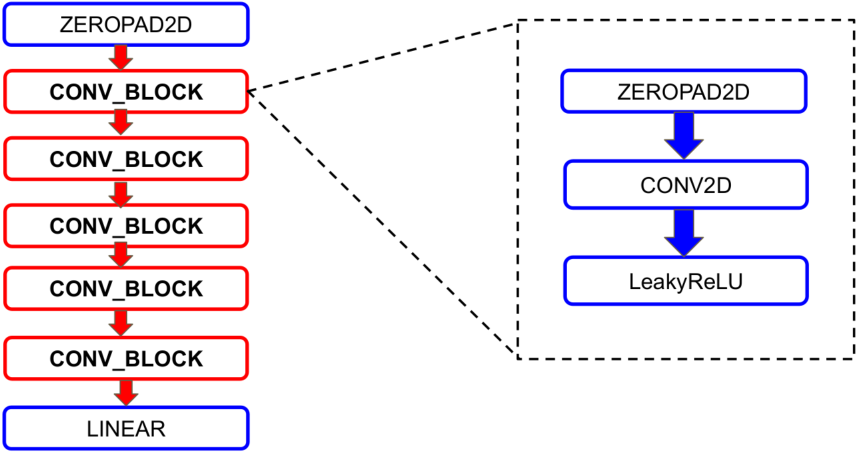
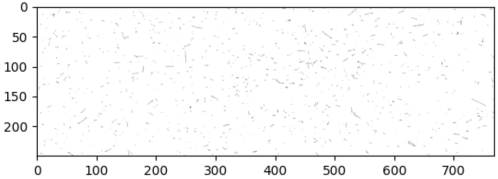


Figure: Discriminator architecture

Generated background

The generated images are visually very similar, but with some subtle differences.



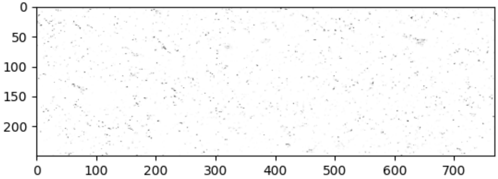
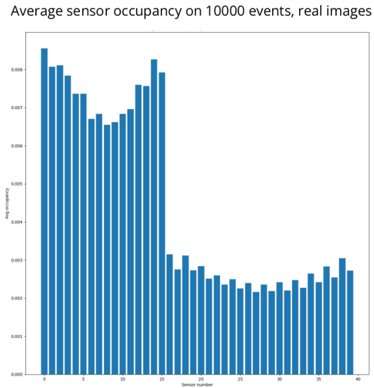
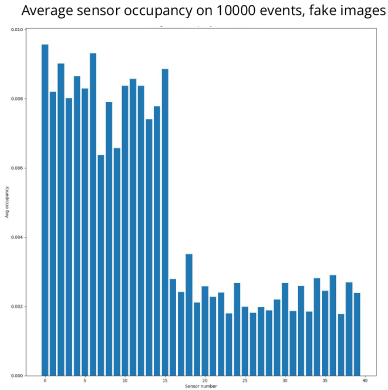


Figure: GEANT4 (left) and GAN simulated (right) background images for sensor 0.

Evaluation - Occupancy per sensor

The model seems to reproduce quite well the sensor occupancy, aside from some minor details probably due to some fluctuations in the weights of the model.





Evaluation - Correlation

The model does not reproduce correctly the correlation between the sensor occupancy.

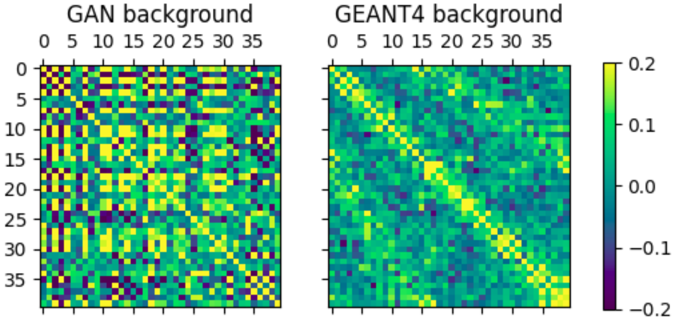
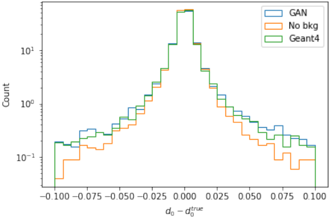
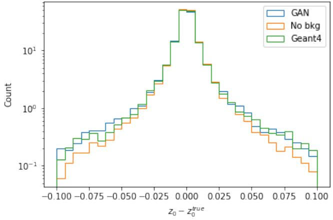


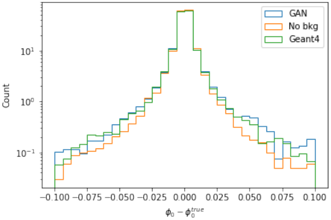
Figure: Correlation between sensors occupancy.

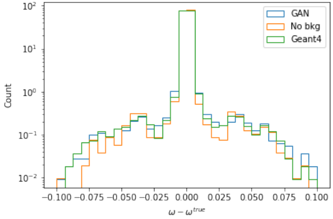
Evaluation - helix parameters resolution

GAN background can be used to reproduce resolution of the helix parameters.



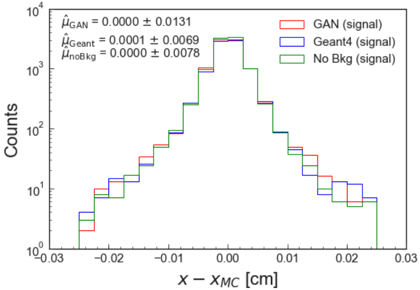


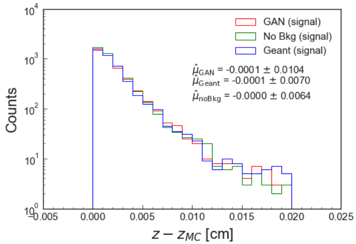


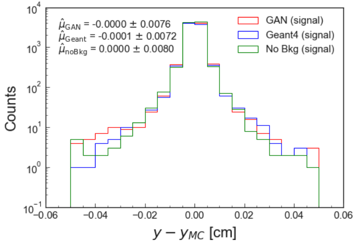


Vertex reconstruction

- Vertex resolution of D^0 in the decay $D^0 \to K^- \pi^+$
- Results suggests that there is no difference when including the background.

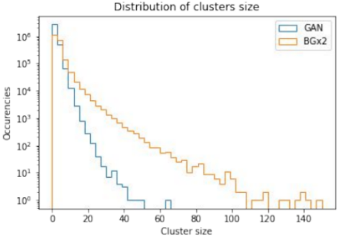


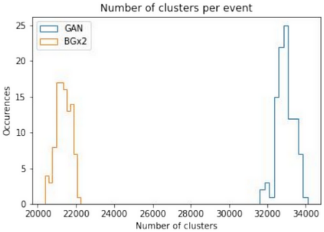


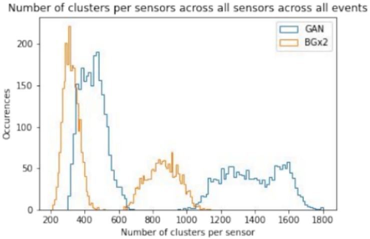


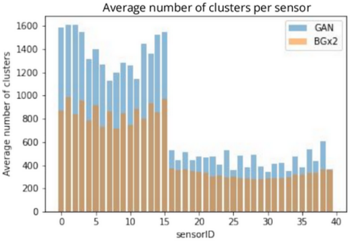
Evaluation: Clusters

The generated background images have different clusters distributions.









Cluster generation with GAN

- Train GAN to directly generate clusters instead of pixel hitmaps
- Trained using clusters of sizes from 1 to 30.
- Training dataset uniform in cluster size.

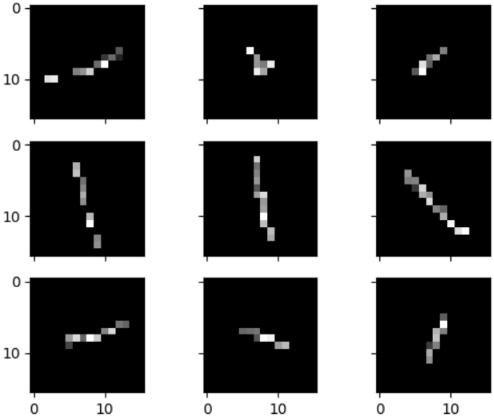
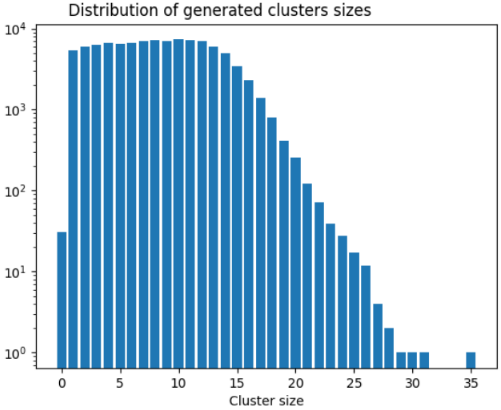
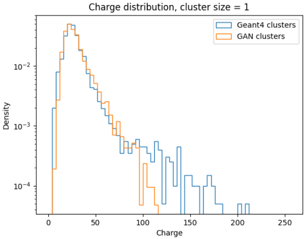


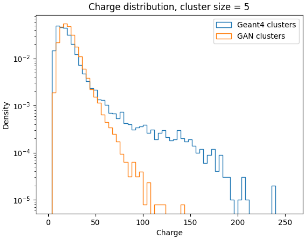
Figure: Example of generated clusters

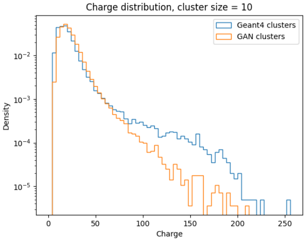
Evaluation: cluster size

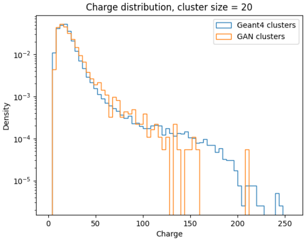


Evaluation: charge distribution









Conclusions and next steps

- Successfully trained a GAN to generate PXD hitmaps.
- Differences between simulated and generated images, especially regarding sensor occupancy correlation and clusters.
- Generated background reproduces helix parameters resolution well and does not have any effect on the vertex resolution for the decay $D^0 \to K^- \pi^+$.
- Successfully trained a GAN to generate clusters.

Next steps:

- Establish a sampling procedure to generate background data.
- Investigate possible correlations between cluster's length and sensor position.
- Reproduce the helix parameter resolutions using the generated clusters.
- Compare this method to other methods which do not use GAN (taking clusters directly from simulation).

Thank you for your attention!