Data Quality Monitoring for the CBM experiment

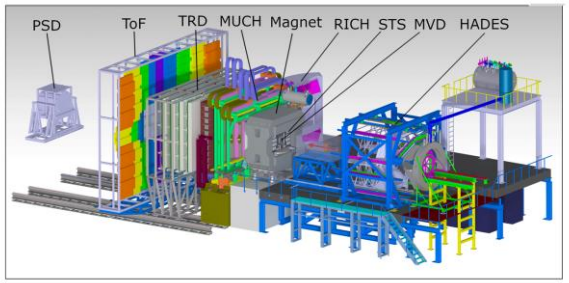
Sachin Gupta, PhD Researcher, s.gupta@gsi.de
Supervisors : Prof. James Ritman, Dr, Tobias Stockmannas

Outline

- What is Data Quality Monitoring (DQM) ?
- AI usage for DQM and HYDRA introduction.
- Expanding HYDRA capabilities with Autoencoders.
- Summary and Outlook.

Data Quality Monitoring (DQM)

- Experimental nuclear and particle physics data are costly to obtain.



Experiment

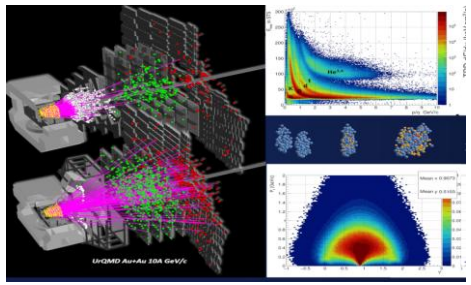
→

Raw Detector Data

→

Calibration + Reconstruction

→

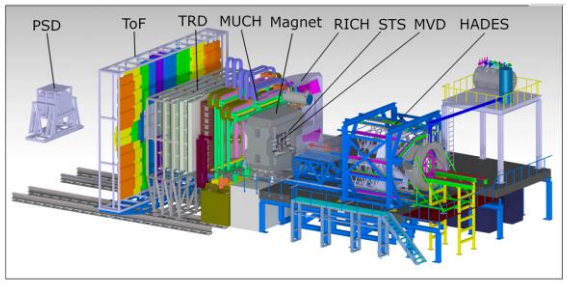


Physics Analysis

- During the run, hundreds of **QA plots (histograms)** are monitored by the shift crew.
- DQM ensures **safe operation** of sub detectors and **data integrity**.
- **Inconsistent task** due to human reaction time, different level of expertise during data taking.
- **Goal :** To automate QA plots monitoring for CBM with computer vision models.

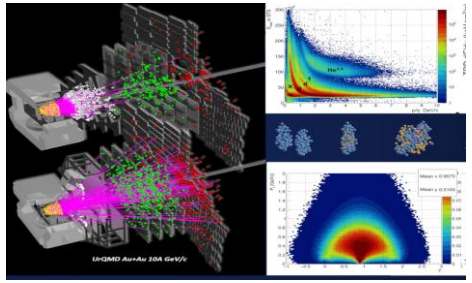
Data Quality Monitoring (DQM)

- Experimental nuclear and particle physics data are costly to obtain.



Experiment

→ Raw Detector Data → Calibration + Reconstruction →

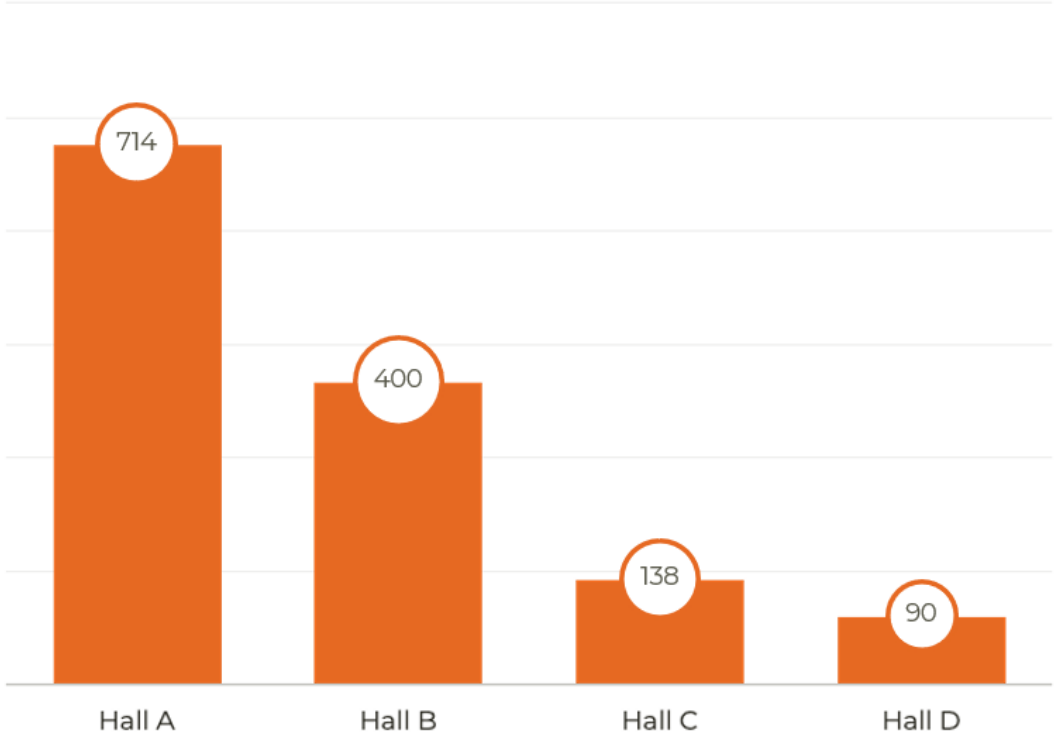


Physics Analysis

- During the run, hundreds of **QA plots (histograms)** are monitored by the shift crew.
- DQM ensures **safe operation** of sub detectors and **data integrity**.
- **Inconsistent task** due to human reaction time, different level of expertise during data taking.
- **Goal :** To automate QA plots monitoring for CBM with computer vision models.

HYDRA

- Hydra was first developed at Jefforson Lab in 2019 and is now operational across all experimental halls.
- It is a computer vision based experimental agnostic deployable system (docker,MySQL) that performs near real time image classification.
- Main features :
  - Training and deploying of multiple computer vision models.
  - Labelling images via web browser.
  - Online monitoring and prediction.



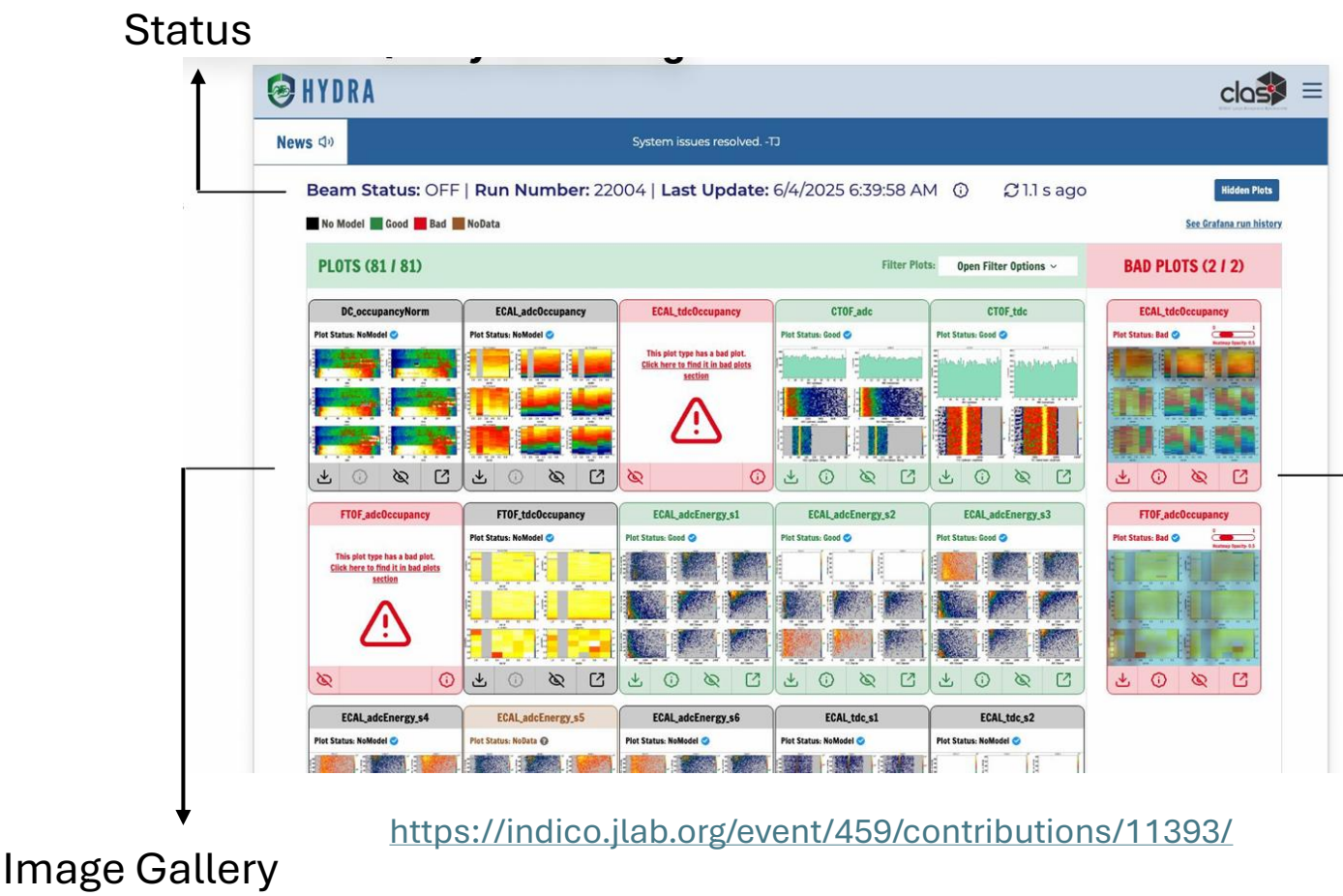
Approximate **number of individual histograms per experiment per run**, monitored by the shift crew for each experimental hall at JLab.

DQM with HYDRA

Photo Generated by Google Gemini

Hydra in Operation

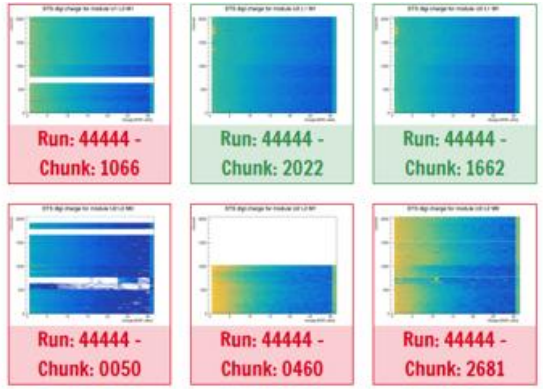
- Different models are trained for each subdetectors and their corresponding plots.
- The alarm is rung as soon as any bad image encounters during the run.



https://indico.jlab.org/event/459/contributions/11393/

HYDRA Web Interface

Labeller

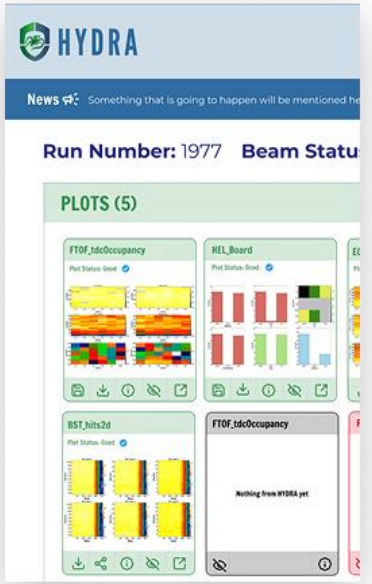


- Efficiently label images
- Task to be performed by detector experts

Library

Evaluate performances and manage models

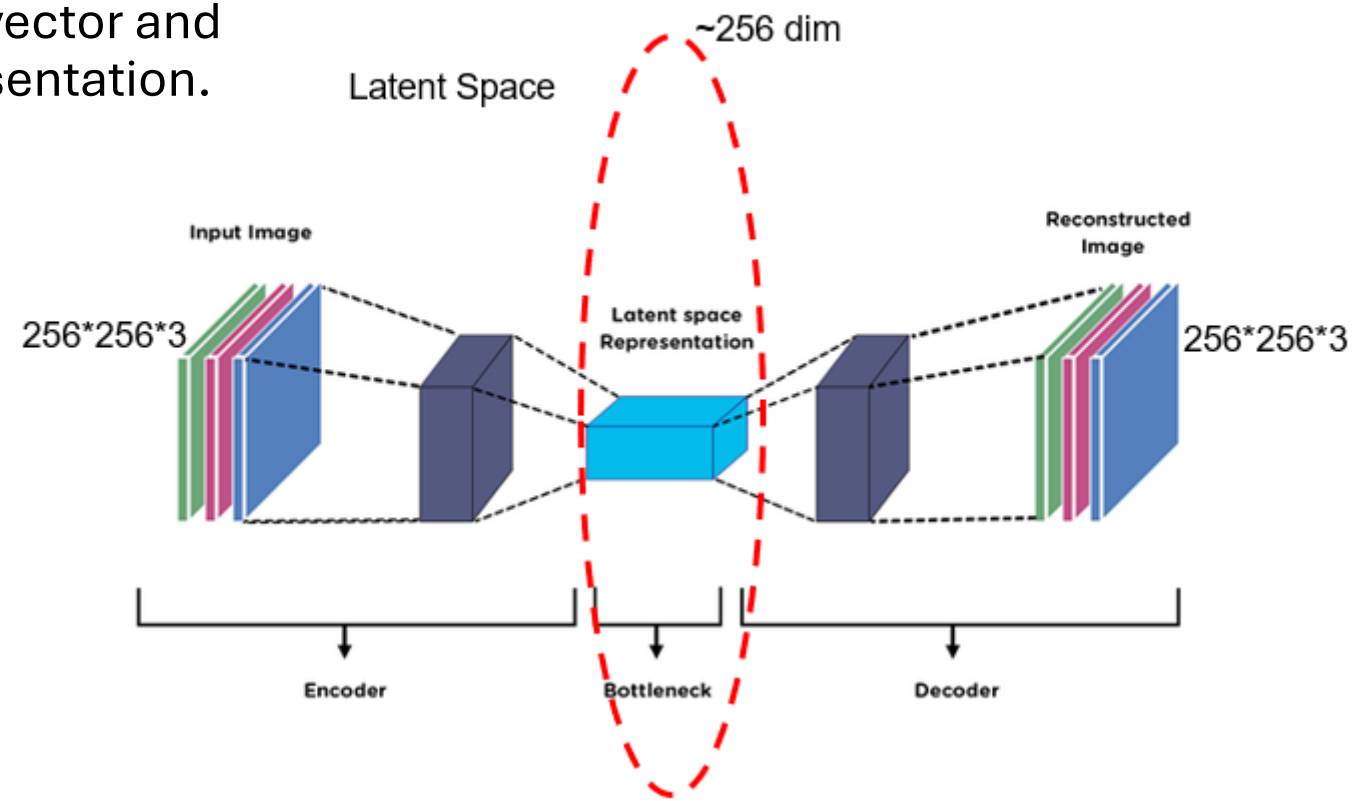
Run



See predictions in real time

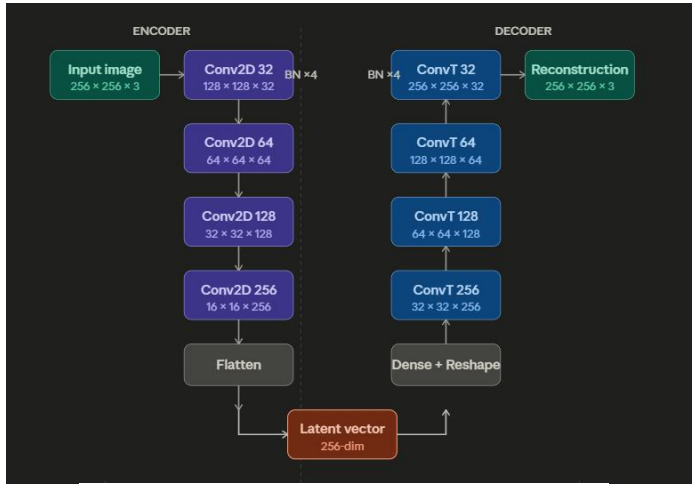
Autoencoders

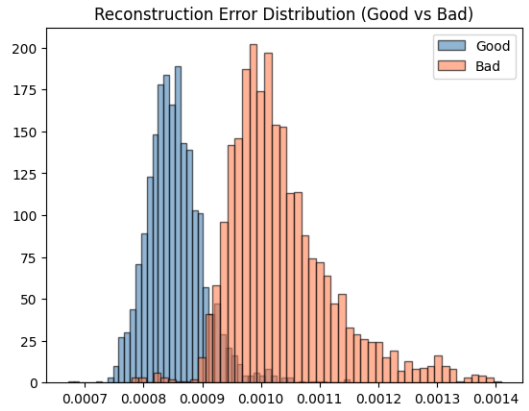
- Neural network architecture that compresses the input data into lower dimensional vector and reconstructs the input from that representation.
- Encoder :
  - $g(x) = E(x)$, g is a latent vector
- Decoder :
  - $x' = D(g(x))$
- Autoencoder :
  - $x' = D(E(x))$
- Goal : Try to replicate the input.
- DQM Usage :
  - 1. Anomaly Detection 2. Clustering



Anomaly Detection

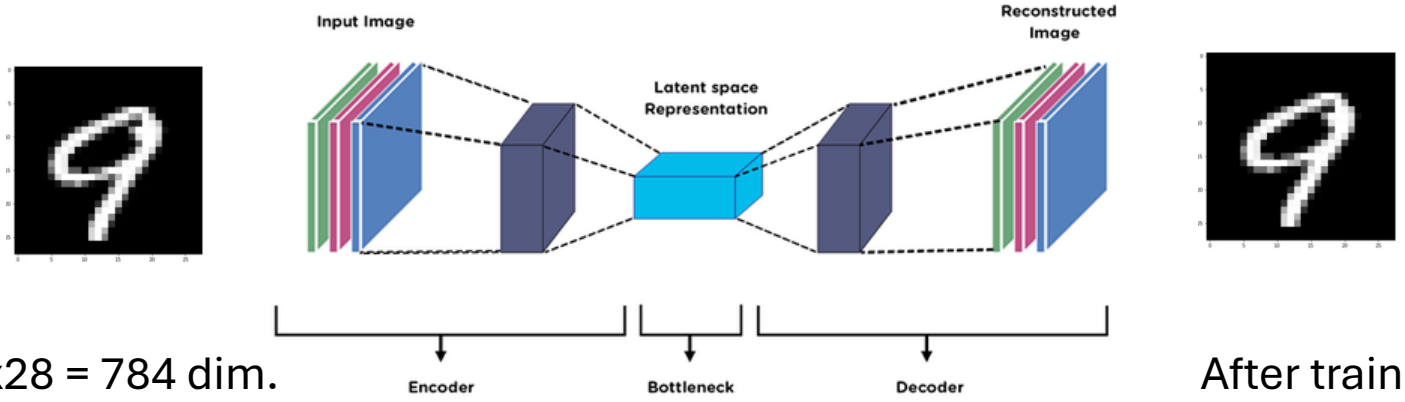
- Only one label of images are used to train the network by minimizing the MSE.
- $MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - x_i')^2$
- Once trained, the network achieves low MSE on the images it was trained on.
- Classification is later performed based on MSE values.





Clustering

- The low-dimensional latent space can be used to cluster similar images.
- Toy example of low-dimensional representation:

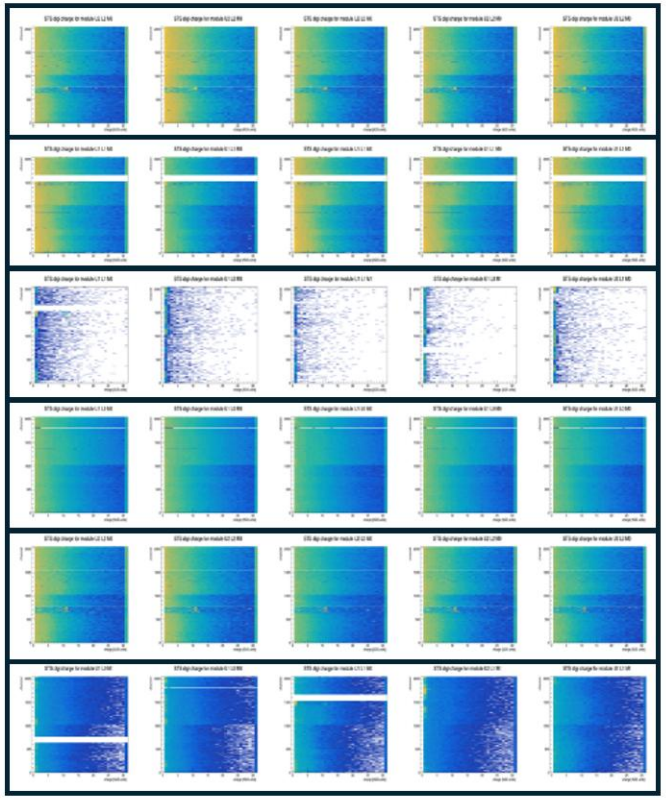


28x28 = 784 dim.

7 dim.

After training the AE, all images are mapped to their latent representations, where clustering algorithms are then applied.

Clustering example



STS Images clusters by Autoencoders

Summary

- Monitoring histograms can be done via computer vision models.
- HYDRA tool for DQM task.
- The HYDRA tool allows users (detector experts) to label images, train networks and deploy them for real time classification.
- Anomaly detection with autoencoders has been successfully added within the HYDRA.
- Clustering functionality is still underdevelopment.

- BACKUP

PhD Project : HYDRA for CBM

- Understanding the HYDRA tool and inherent database structure (Podman, MySQL, Tensorflow).
- Deployment at GSI computing farm. (DCS node and gsi GPU farm)
- Tested with STS QA plots (mCBM 2024), and Vertical test simulation.
- Expanding its capabilities with unsupervised learning (Autoencoders).