HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

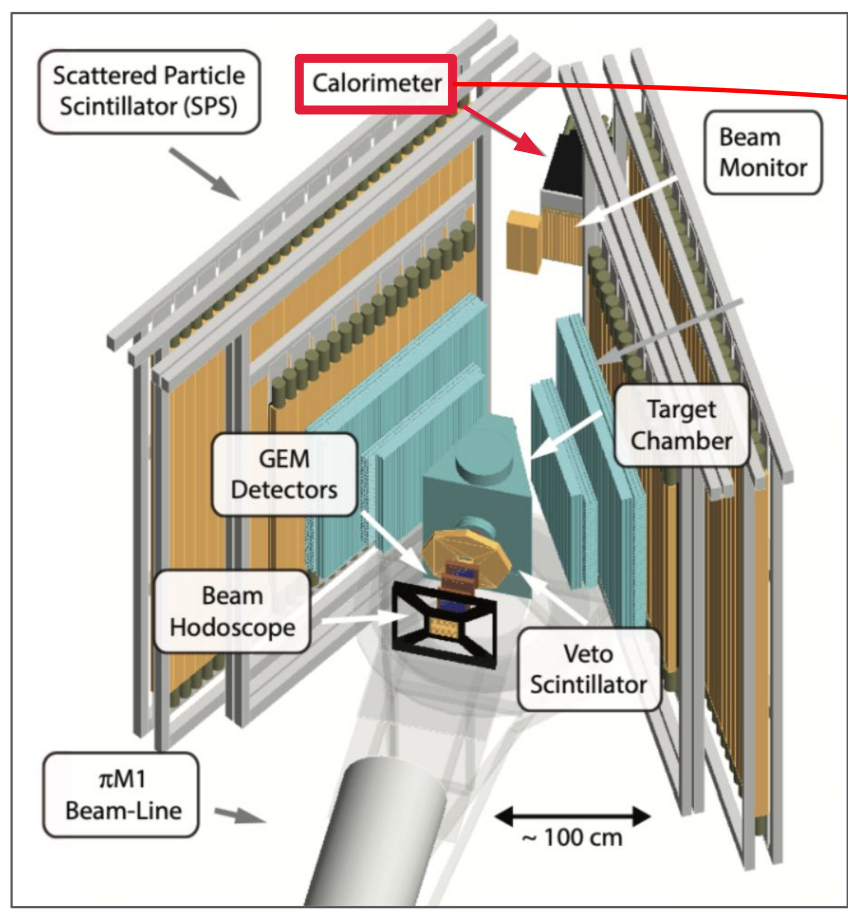
Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin
Department of Physics and Astronomy, Rutgers University, Piscataway NJ

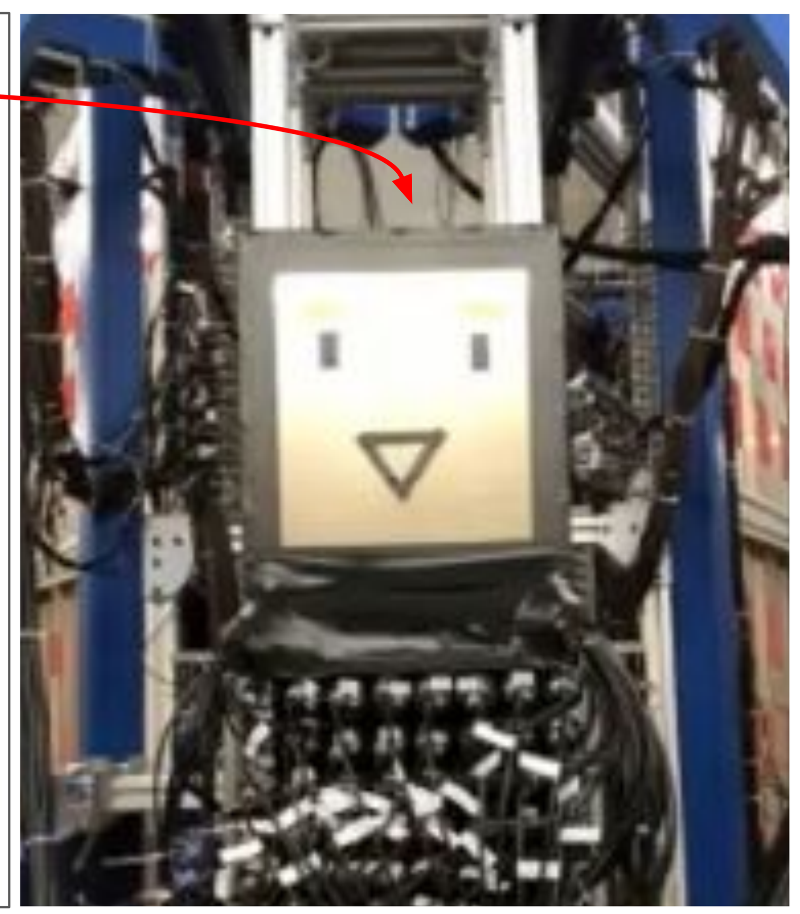
Rutgers Aresty Research Center for Undergraduates

The Proton Radius Puzzle

- In 2010, Pohl et al. measured the proton radius using *muonic* hydrogen spectroscopy and found it to be **5.6 σ smaller** than the accepted value determined through *electronic* hydrogen. It is still unclear how much of this difference results from interesting physics vs. experimental error.
- The **MUon Scattering Experiment (MUSE)** performs simultaneous muon and electron scattering off of a liquid hydrogen target with the goal of measuring the proton radius with high-precision. This will also provide a test of lepton universality and the possibility of two-photon exchange by switching between beam polarities.

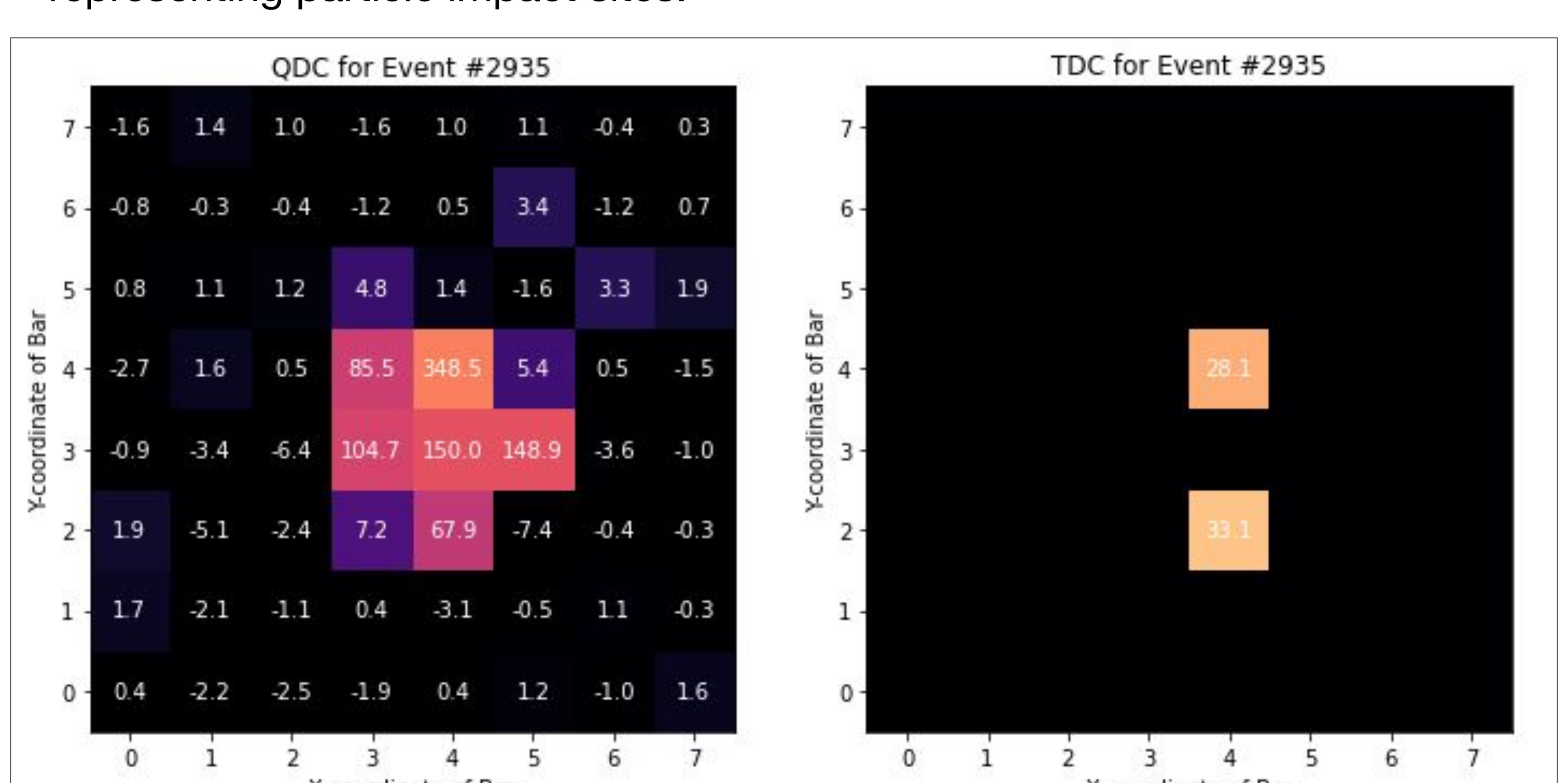
πM1 Calorimeter





The πM1 beam line at the Paul Scherrer Institute has an electromagnetic **calorimeter** (pictured above) which measures the ***energy and timing* of particles moving along the beam line**. When high-energy muons/pions/electrons enter the calorimeter, they initiate a "particle shower", a cascade of lower-energy secondary particles which deposit their energy into the calorimeter.

The calorimeter is an 8 by 8 array of independently read out crystals, which are monitored in 2 ways: QDC (Charge-to-Digital Converter) and TDC (Time-to-Digital Converter). We can visualize the data collected by the calorimeter for a single event in the colormaps below, with the "hotter" pixels representing particle impact sites.



Abstract

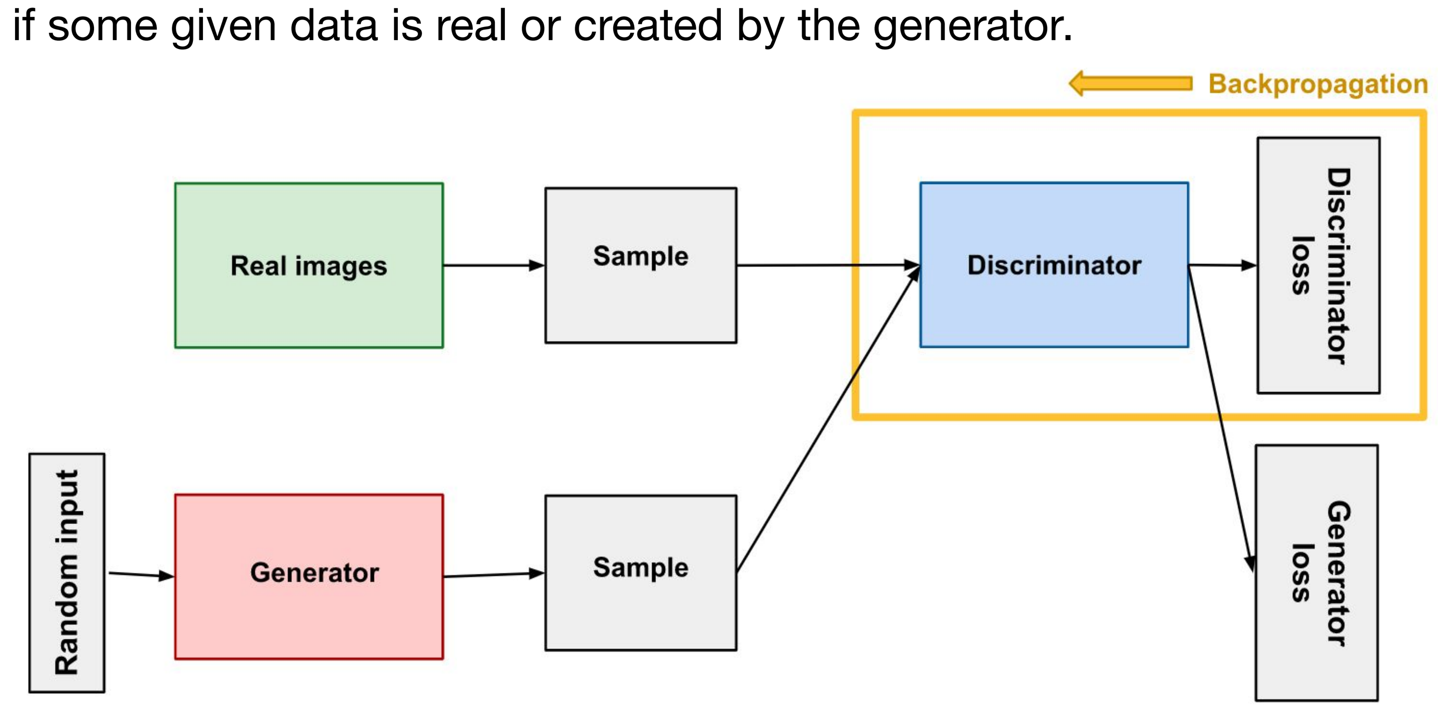
In high energy physics, detailed and accurate simulations of detectors (such as calorimeters) are the **single most expensive element of computational pipelines**. This is because most analyses require aggregated distributions of detector data as opposed to "realistic" data of individual events, creating heavy demand for fast simulation frameworks.

Traditionally ***GEANT4*** software is used to simulate detector responses, but this can take **minutes per individual scattering event** (of which millions are required) and is prone to systematic deviations from experimental data.

Thus we develop ***HapPyCal***, a fast detector simulation framework which can accurately reproduce calorimeter energy scans significantly faster than previous methods. We adapt a machine learning architecture called "Generative Adversarial Networks" and train it with data collected from experimental runs. Through analyses of energy scans from electron scattering events, we find that our model can **accurately reconstruct a larger distribution of data** with just **5%** of the original dataset.

Methodology

Generative Adversarial Networks (GANs) are a type of deep generative model characterized by two key components: a generator network and a discriminator network. The generator's goal is to create realistic data (in this case, calorimeter energy scans), while the discriminator's goal is to be able to tell if some given data is real or created by the generator.

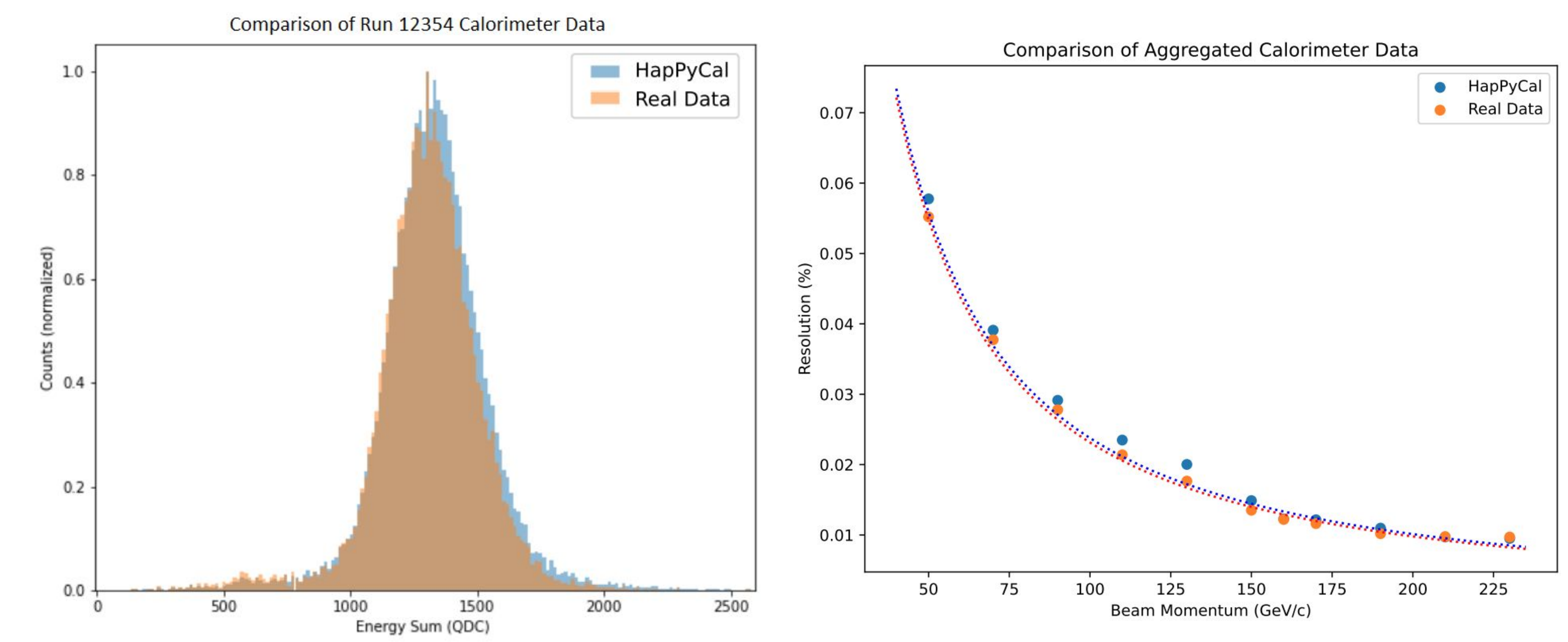


In the training process, **the generator repeatedly tries to "fool" the discriminator**. At first, the generator's outputs are recognized as fake every time, but by applying backpropagation algorithms, **the generator is able to "learn" from its mistakes** and gradually get better and better at producing "realistic" data. The discriminator is trained in tandem, creating a type of two-player minimax game.

On a more technical level, we use PyTorch to structure our generator as a **convolutional neural network** with three layers and a combination of ReLU and TanH activation functions. The discriminator is a standard **binary classification model** with a cross entropy loss function.

Results

After training our model on ~5% of the calorimeter data from each dataset over a target range of momenta, we **compare HapPyCal-generated data** (blue) with **real data** collected by the πM1 calorimeter (orange) below:



1. On the **left**, we focus on a *single* "run" (**fixed momentum**: 230 MeV/c) and show the distribution of total energy deposited by electron scattering events.
2. On the **right**, we look at *many* runs (**variable momentum:** 50-230 MeV/c) and show how the energy resolution (proportional to standard deviation) changes with momentum.

In both cases, the curve-fits for HapPyCal data and real data are incredibly close, suggesting that the generative modelling technique is **accurately "learning" the underlying distributions** of electron scattering data.

We note that in most of our analyses, HapPyCal's energy values tend to be greater than in real data. This can be attributed to the fact that GANs are usually susceptible to noisy outputs when the training data can't be normalized as in our case.

Conclusions / Further Work

The speed and accuracy of HapPyCal opens up some interesting opportunities for deeper analysis at MUSE. Notably, Monte Carlo analyses are more feasible, GEANT4 models can be evaluated much faster, and we can quantify how detector responses vary over small periods of time to identity any drift in the data.

GANs are a powerful tool, but have many well known problems such noisy outputs and unstable/non convergent training. Future work with more cutting-edge techniques from machine learning literature such as Normalizing Flows and WGAN-GPs could offer even greater speed and accuracy.

Acknowledgements

This material is based upon work supported by Aresty and the National Science Foundation under NSF grant PHY-1913653. The MUSE experiment is supported by the Department of Energy, NSF, PSI, and the US-Israel Binational Science Foundation.