Fermilab U.S. DEPARTMENT OF ENERGY Office of Science

FERMILAB-PUB-23-180-PPD

Smart pixels with data reduction at source

MODE collaboration workshop – July 24, 2023

Jennet Dickinson

with Douglas Berry, Giuseppe Di Guglielmo, Karri DiPetrillo, Farah Fahim, Lindsey Gray, Jim Hirschauer, Rachel Kovach-Fuentes, Shruti Kulkarni, Ronald Lipton, Petar Maksimovic, Corrinne Mills, Benjamin Parpillon, Gauri Pradhan, Morris Swartz, Nhan Tran & Jieun Yoo

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.

Pixel detectors at the LHC

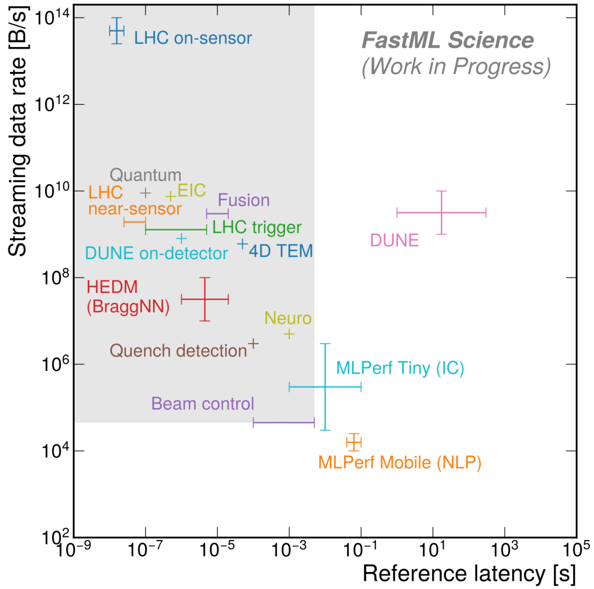
- Highest data rates in HEP!

  Current detectors only read out triggered events

- And getting higher…

  Next generation detectors promise better resolution (position & angle), precision timing

  More information, but also more data



2207.07958

What would we gain if we could analyze it all? Some aspirational targets:

- ***Higgs self-coupling*** *:* 5x increase in the low-m_{hh} spectrum from b-jet triggers.
- ***WIMP dark matter*** : 50x rate for low-p_T / disappearing tracks / long-lived particles.
- ***New capabilities for high-rate, soft objects*** : e.g. dark sector BSM, B-physics, and more!

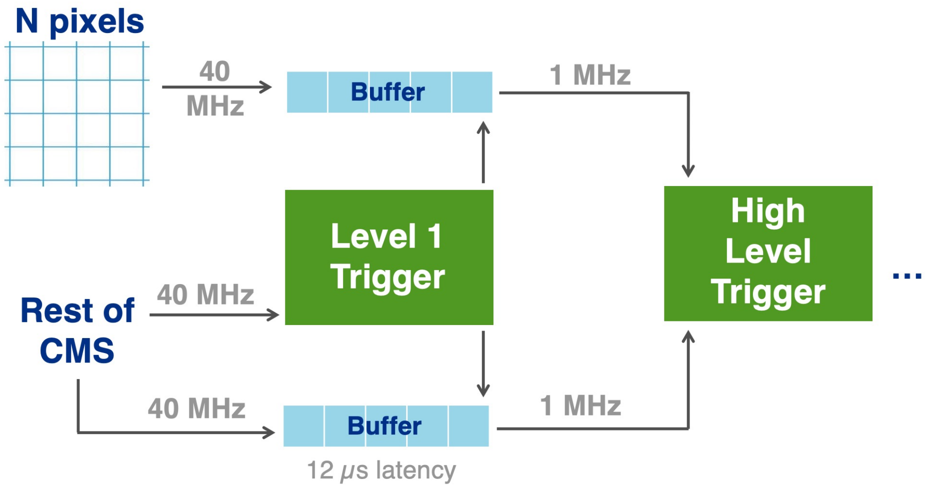
Pixel readout chain: CMS at HL-LHC

- Detector is an array of N pixels

  100 x 25 μm pitch

  100 μm thick sensor

- Pixel data sits in buffer until L1 decision is made
- Passed to HLT at 1 MHz



Pixel readout chain: our futuristic detector

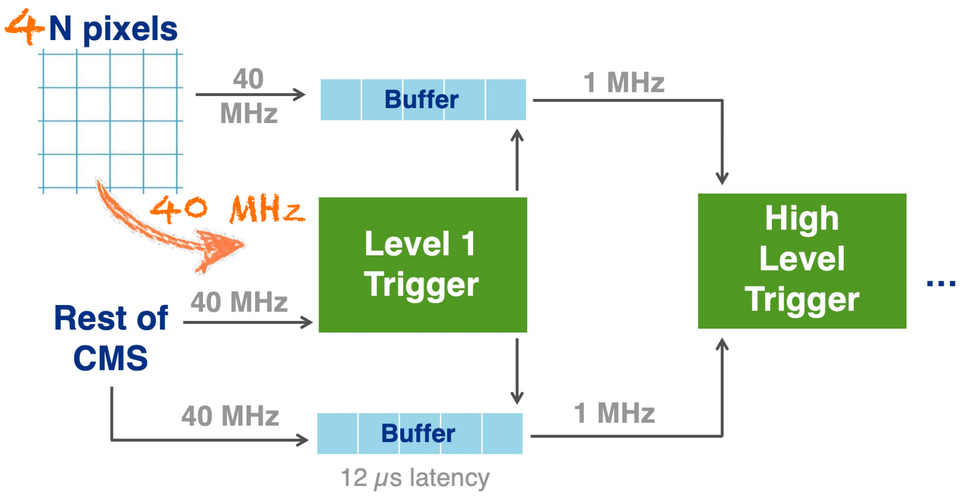
- Detector is an array of **4N** pixels

  50 x 12.5 μm pitch

  100 μm thick sensor

- **Pixel data is passed to L1 trigger at 40 MHz**
- Passed to HLT at 1 MHz

We have to transfer 4-160x more data



Pixel readout chain: our futuristic detector

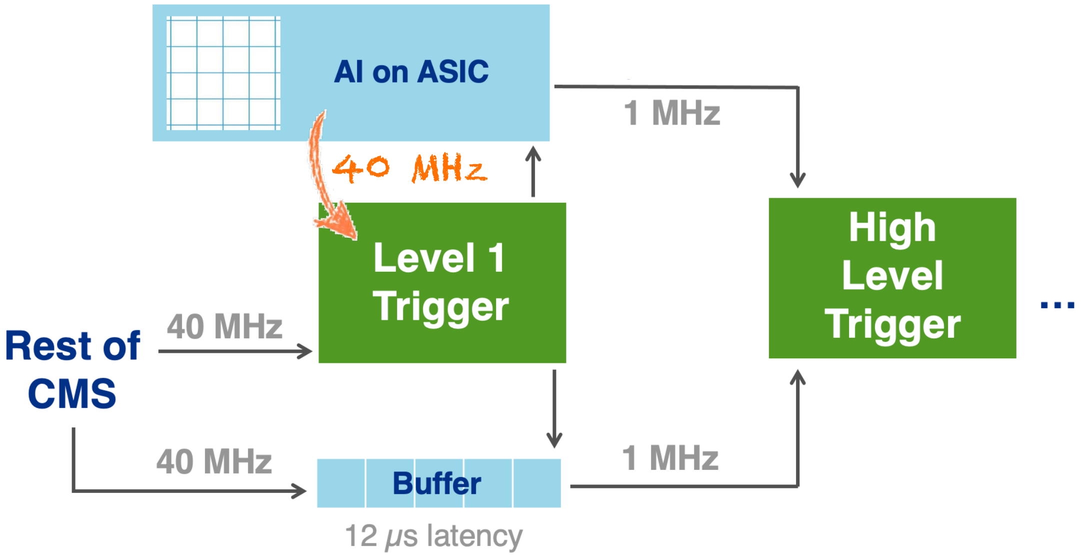
- Detector is an array of **4N** pixels

  50 x 12.5 μm pitch

  100 μm thick sensor

- **Pixel data is passed to L1 trigger at 40 MHz**
- Passed to HLT at 1 MHz

We have to transfer 4-160x more data



Use AI to perform physics-motivated data reduction on-ASIC

Fermilab

Charged particle signatures in our futuristic detector

- State-of-the-art dataset for developing algorithms for implementation on-ASIC (link)

  Initial conditions = fitted track params from CMS Run 2 data, down to $p_T \sim 100$ MeV

  Simulation with time-sliced PixelAV, including E field and weighting field

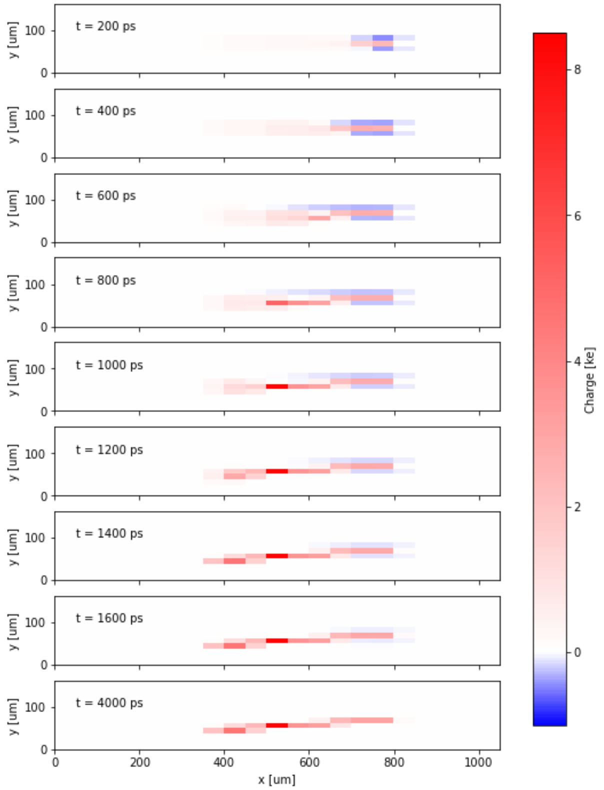
- Simulated MIP interactions in a 21x13 array of pixels

  50x12.5 μm pitch, 100 μm thickness

  Located at radius of 30 mm

  3.8 T magnetic field

  Time steps of 200 picoseconds



Fermilab

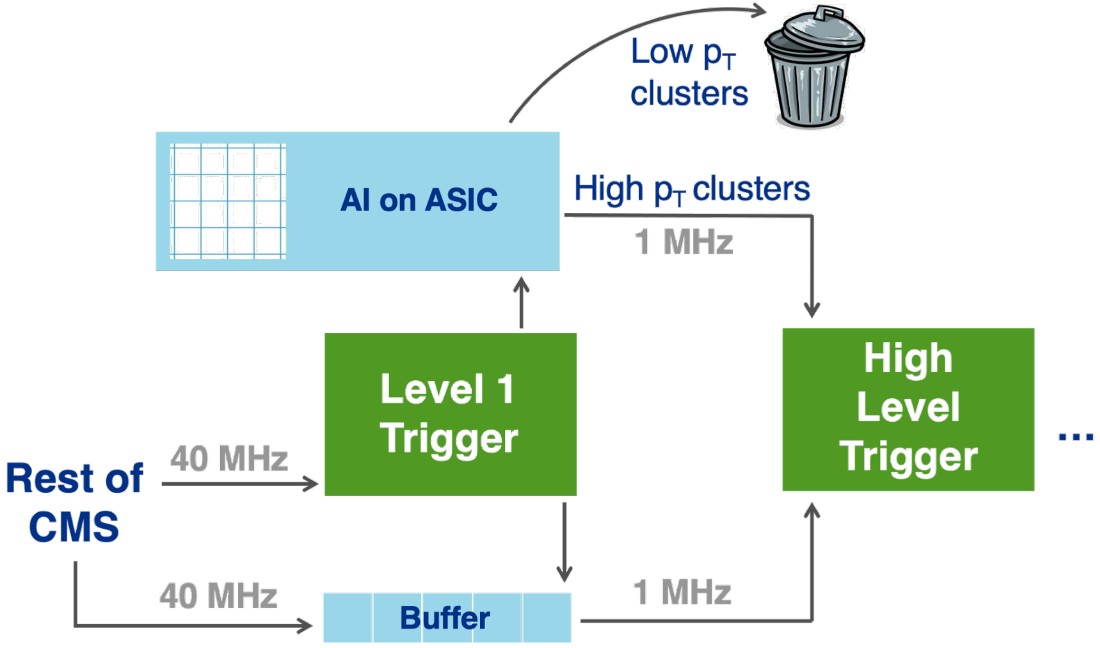
Applications for AI on-ASIC: data filtering

- Select and read out only those clusters created by particles with high transverse momentum (p_T)
- Particle p_T ~ radius of curvature, correlated with

  Incident angle in the bending plane of the magnetic field (β)

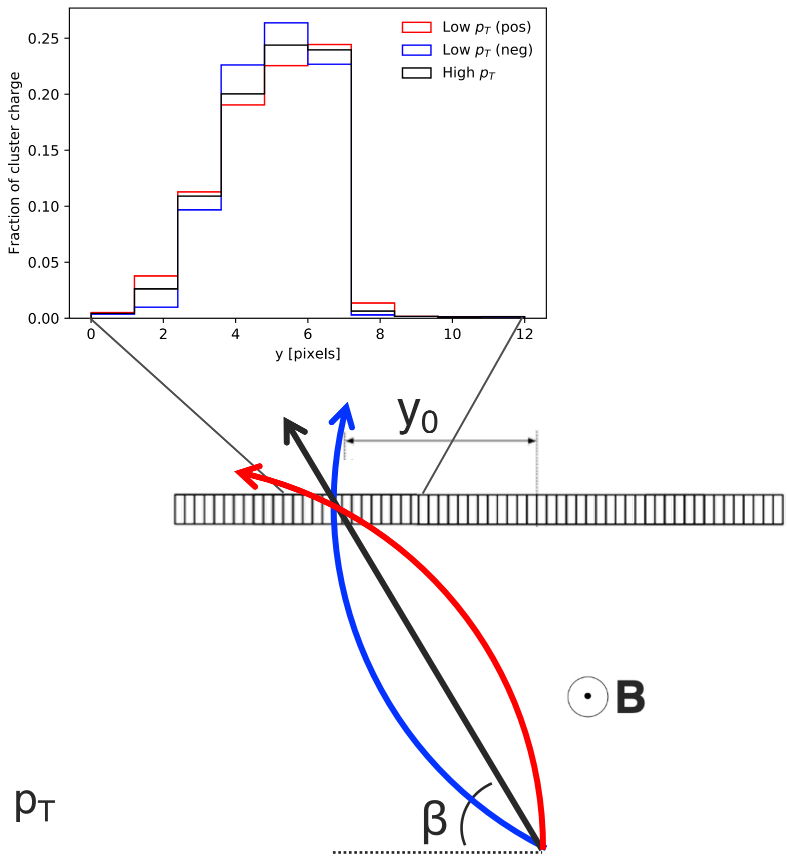
  Position of the hit in the bending direction (y_0)

  Sign of the charge



Classification based on particle p_T

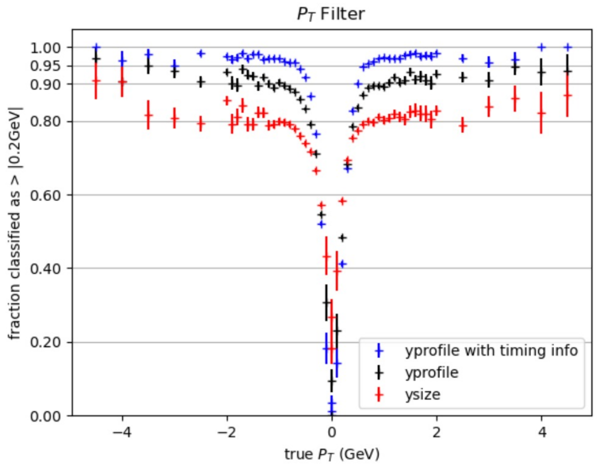
- p_T ~ radius of curvature, correlated with
  - Magnetic field strength (B)
  - Position of the hit in the bending direction (y_0)
  - Angle in the bending plane of B (β)
  - Sign of the charge
- Train a classifier to select clusters with $p_T > 200$ MeV
  - Input data: cluster image projected onto y-axis
- Three classes:
  - Low p_T negative charge, low p_T positive charge, high p_T



Fermilab

Performance of the DNN p_T filter

- Full precision network:
  1. Projected cluster size only
  2. Projected cluster shape (selected for implementation)
  3. Timing information promises 5-10% efficiency gain



	How much of what we keep is $p_T > 2$ GeV?	How much of what we discard is $p_T < 2$ GeV?	How much do we discard overall?
Model	**Sig. efficiency**	**Bkg. rejection**	**Data reduction**
Model 1	84.7 %	41.1%	26.3 %
Model 2	93.2 %	48.4%	24.5 %
Model 3	97.6 %	51.6%	21.1 %

Model 4: Spiking neural network is a work in progress

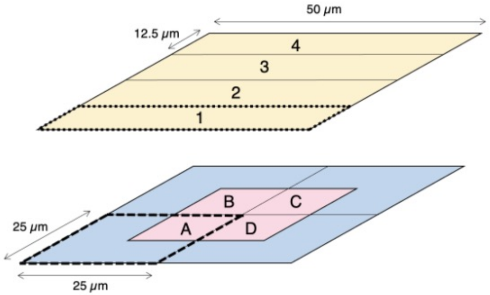
Fermilab

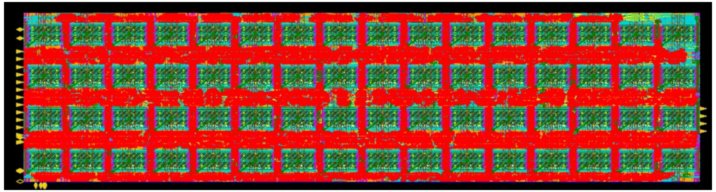
Implementation on-ASIC

- Following quantization aware training with qKeras and further optimization with hls4ml, the algorithm has 1,163 parameters

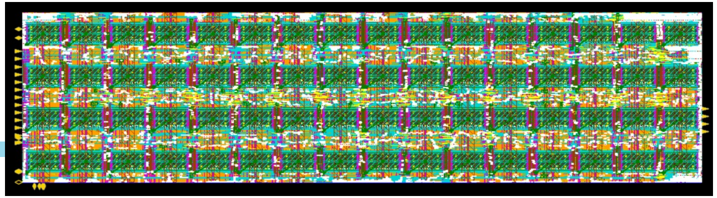
  Operates at < 300 μW, area of less than 0.2 mm^2

- Each 2x2 array of readout pixels maps to a 1x4 array of sensor pixels





Red:
classifier algorithm

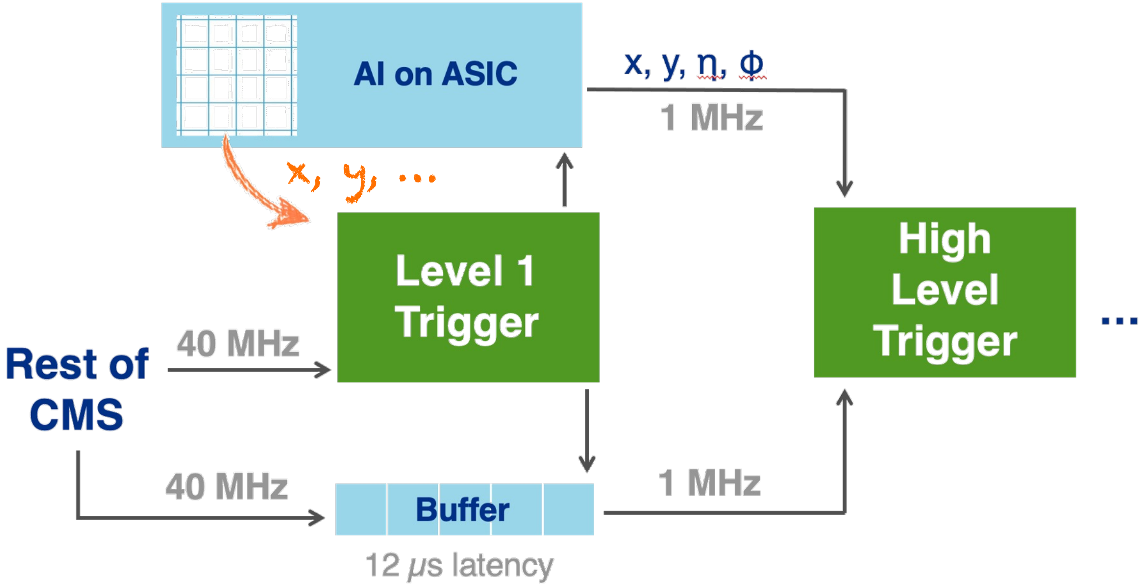


White:
network weights

Applications for AI on-ASIC: featurization

- Train an algorithm to extract properties of the incident particle. Read this out instead of raw data

Technically lossy, but preserves information useful for physics



Features to predict

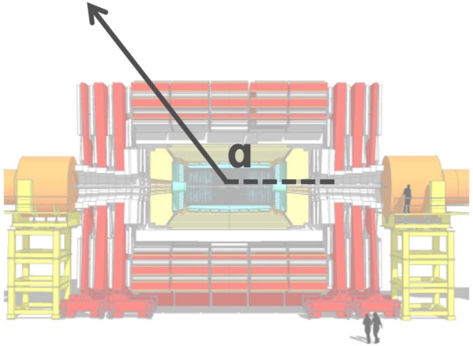
- Hit position (x, y) and incident angle (cot α, cot β)
- Mixture density network can give us a prediction for each feature, plus a **meaningful uncertainty**
- For each cluster, assume the likelihood is described by a multivariate Gaussian in (x, y, cot α, cot β)

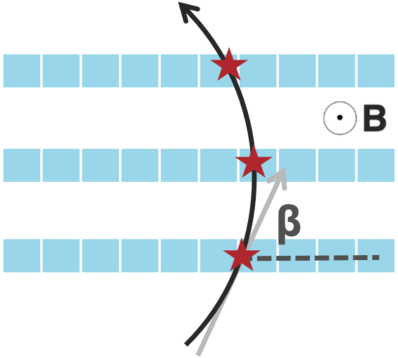
  Training minimizes loss = negative log likelihood

- Build a model that predicts all parameters of the likelihood

  Mean of x, y, cot α, cot β and full covariance matrix!

  14 features in total





Angles & their uncertainties

- More complex final states ⟶ more hits ⟶ more hit combinations for track seeding

  Computationally very expensive and slow 😕

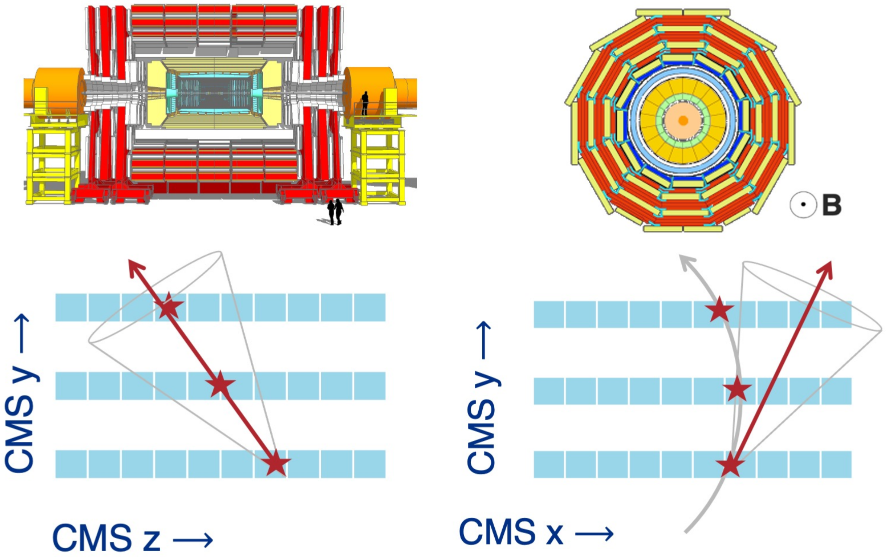
- Predicted angle + uncertainty gives a cone where you can expect a hit in the next layer, **reducing combinatorics**

  Small uncertainty ⟶ small cone

- Fast tracking and vertexing

  Very valuable for hh, e+e- and $\mu\mu$!

  At HL-LHC: makes L1 pixel trigger feasible?



Featurization network

- Deep 2D **convolutional neural network**

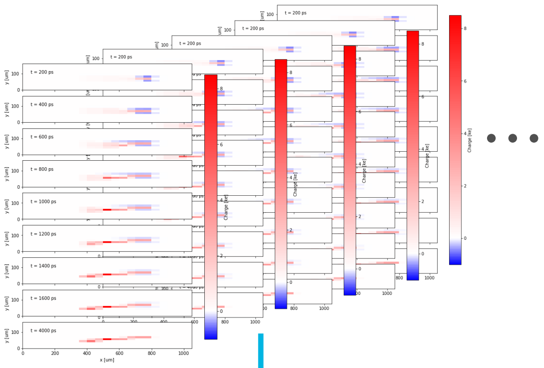
  Treat charge deposited in pixel array as 2D image

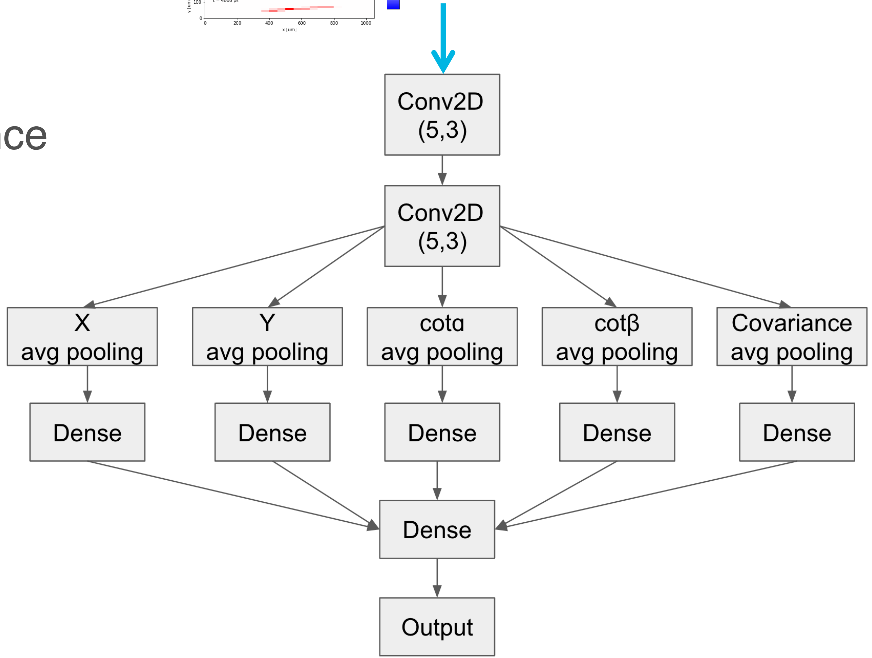
  Treat each 200 ps time slice as a channel

- 5 branches with pooling layers

  Corresponding to x, y, cotα, cotβ, and covariance

- 2,181 trainable parameters in total





Featurization network

- Deep 2D **convolutional neural network**

  Treat charge deposited in pixel array as 2D image

  Treat each 200 ps time slice as a channel

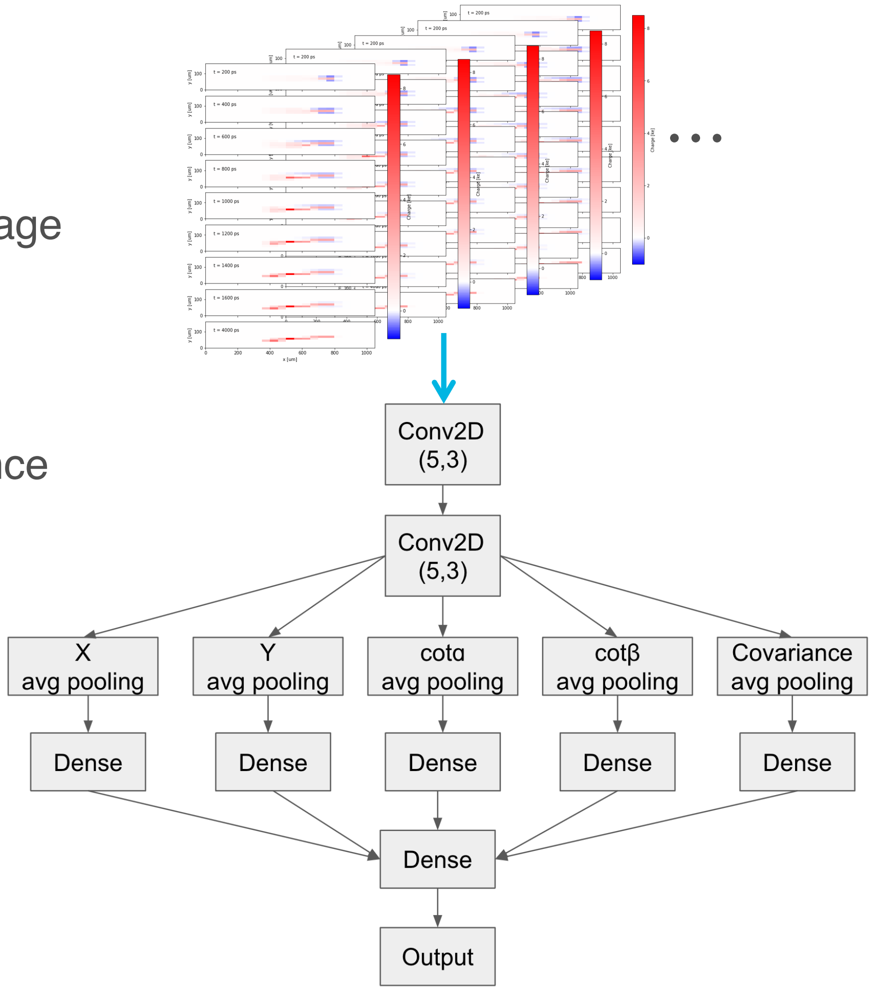
- 5 branches with pooling layers

  Corresponding to x, y, cotα, cotβ, and covariance

- 2,181 trainable parameters in total

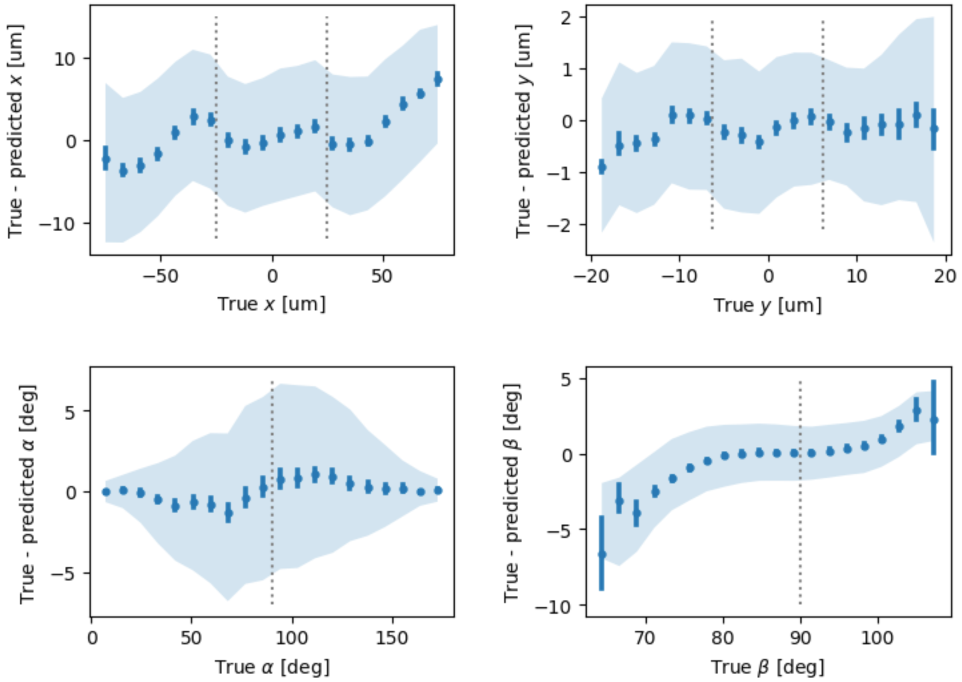
Do we have (low power) detectors that can sample every 200 ps?

Opportunity to incorporate fast timing detectors or spiking neural networks



Performance of the featurization network

- Residuals vs. truth, with band showing mean predicted uncertainty



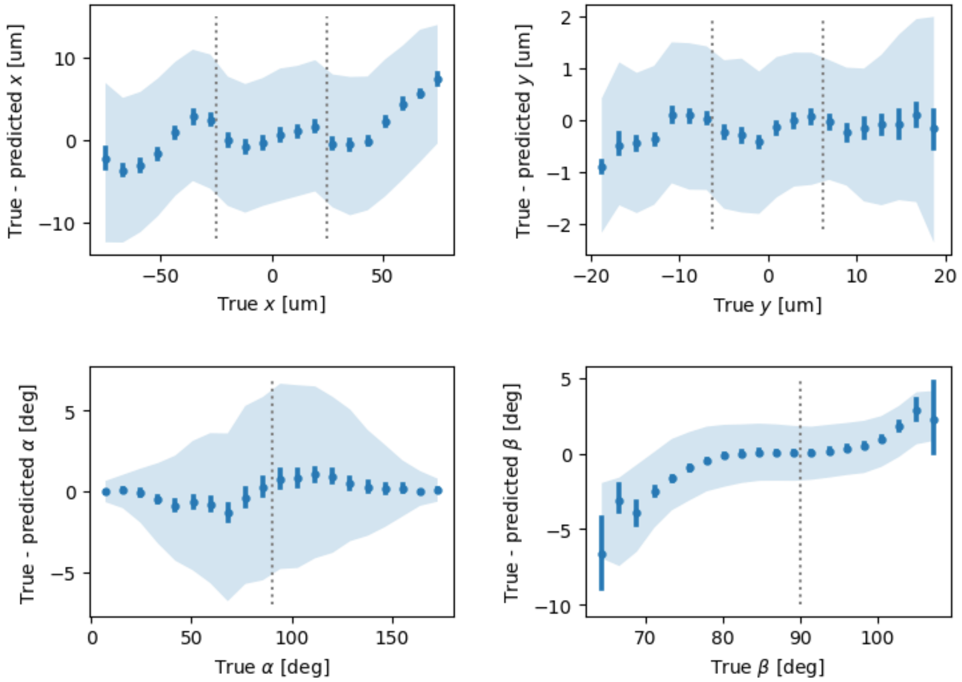
Hit position x, y

Pattern of bias repeats across each pixel

Mean resolution of 10 μm and 1μm in x, y

Performance of the featurization network

- Residuals vs. truth, with band showing mean predicted uncertainty



Hit position x, y

Pattern of bias repeats across each pixel

Mean resolution of 10 μm and 1μm in x, y

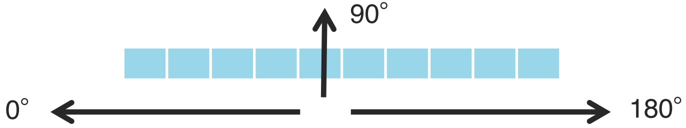
Angles α, β

Largest uncertainty near α=90° due to single pixel hits

Dataset covering limited range in β

Mean resolution corresponding to cone of $\sigma < 5^{\circ}$

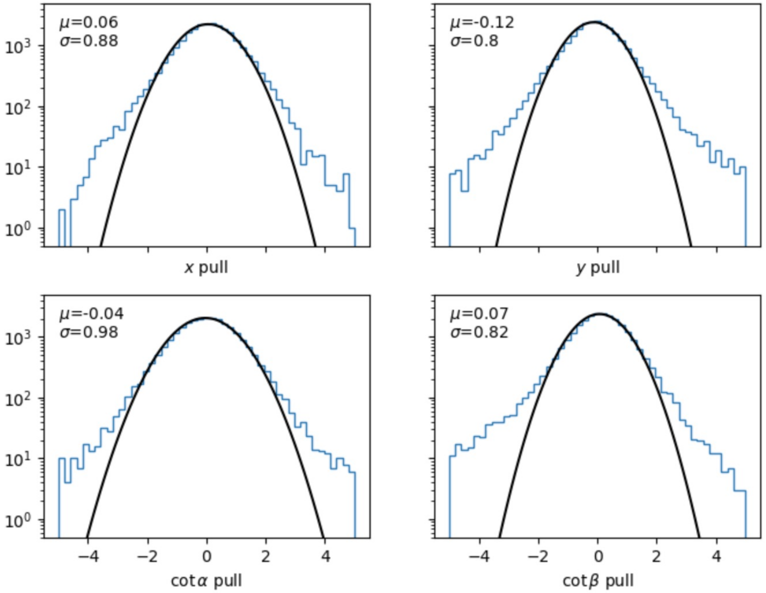
(~0.2% of the full solid angle)

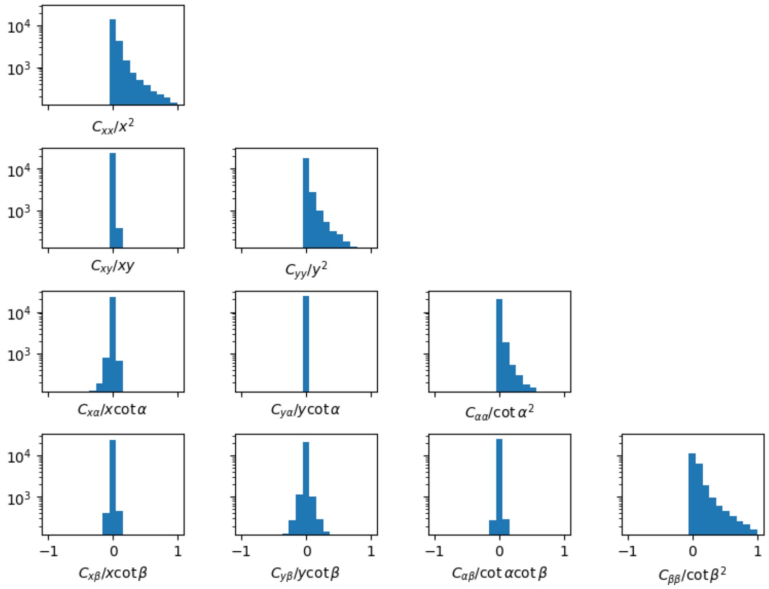


Fermilab

Performance of the featurization network

- Pulls = residual / predicted uncertainty good out to ~ 3σ
- Small correlations between features





Featurization: future plans

- How much can we compress the network?

  Reduce ops, quantization aware training, hls4ml

  Try training on shape of deposited charge sampled at a lower frequency

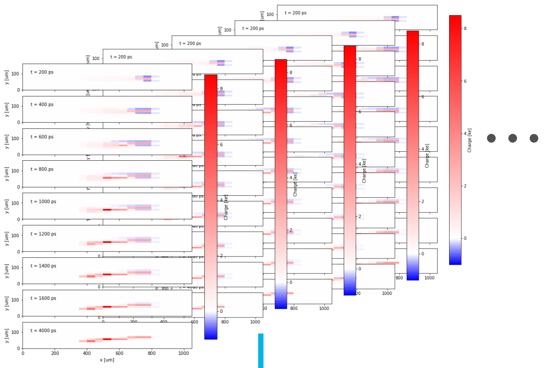
- What should we output?

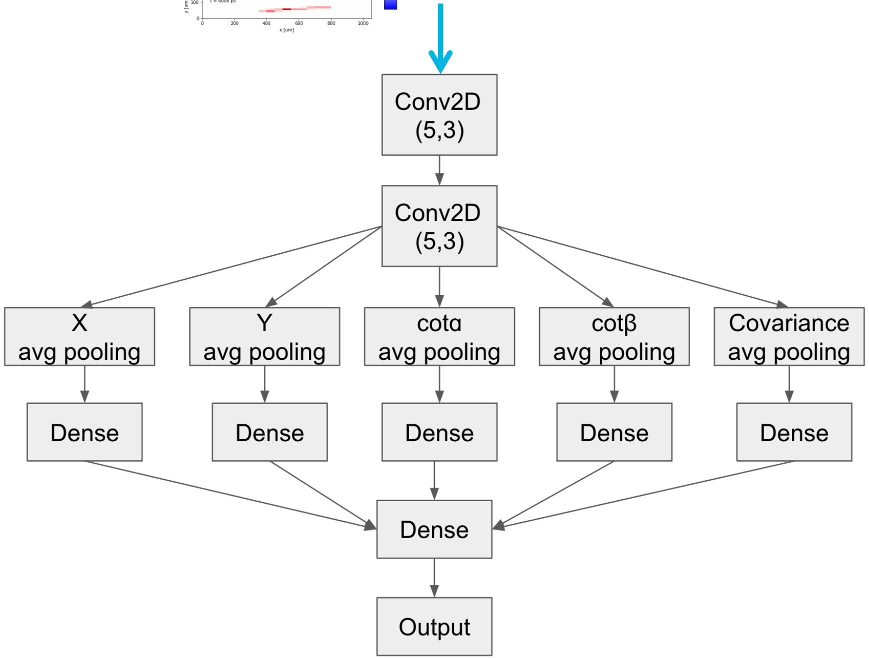
  So far: 14 physics parameters

  Outputting latent space would give lots of flexibility

- Applications for future colliders

  What might we do differently at an e^+e^- or muon collider?





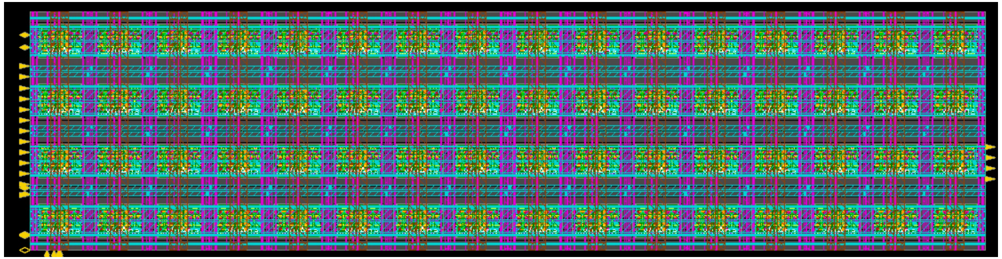
Summary & next steps

- AI on-chip has great potential to **reduce data rates to manageable levels** at the HL-LHC and beyond
  - Co-design with focus on preserving information that is useful for physics
- First implementation of the **p_T filtering** looks very promising!
- **Feature extraction** for x, y, α, β and full covariance is possible!
- Leverage **emerging technologies** to improve energy efficiency and accuracy:
  - Analog multiplication
  - Neuromorphic / spiking networks
  - 3D stacking

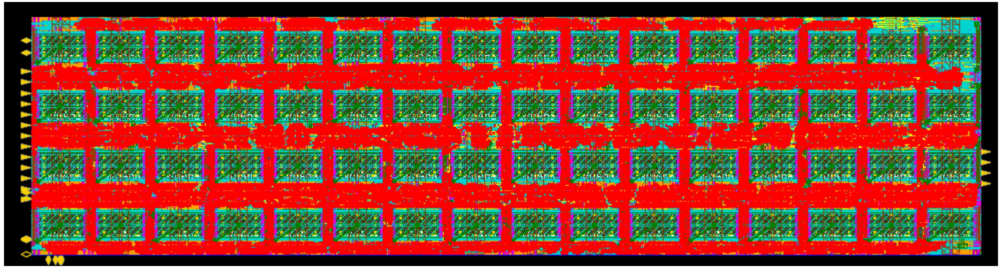
Backup

Implementation in 28nm CMOS

- Floorplan with analog pixels with power and bias grid

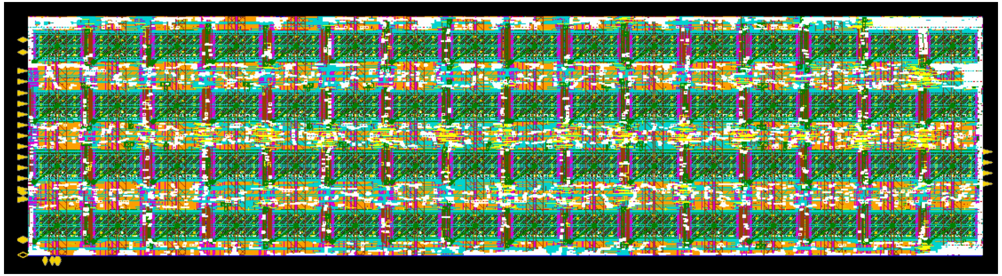


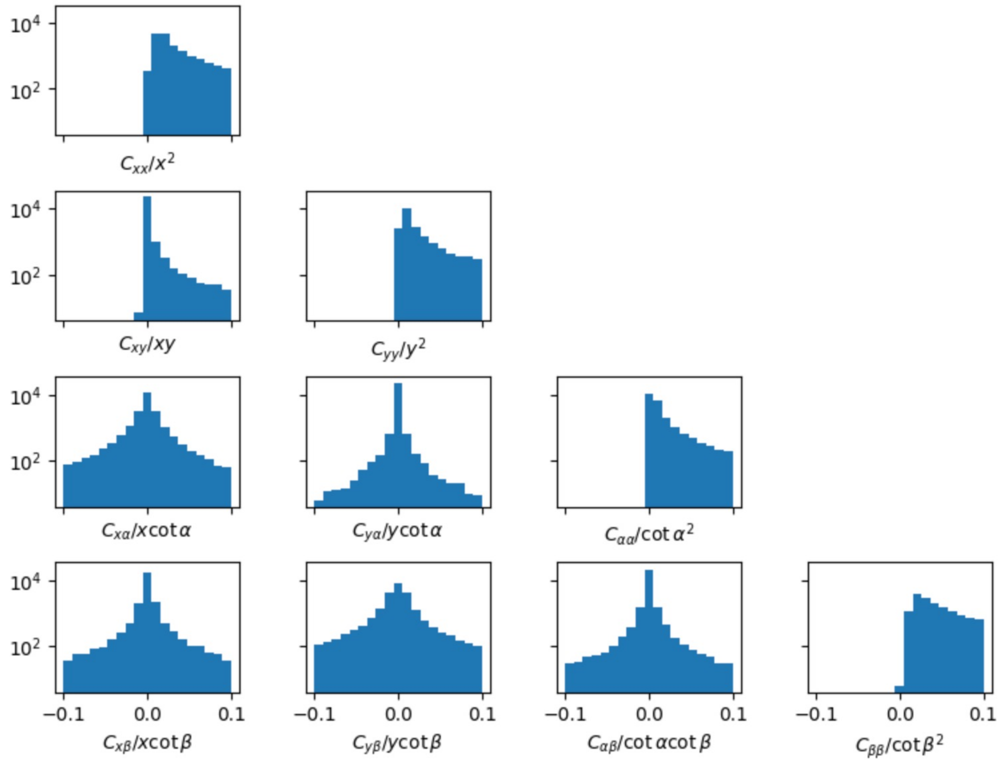
- Red: classifier algorithm



- White: registers for programming the neural network weights

  Triple redundancy to protect against single event upset




10^4
10^2
C_{xx}/x^2
C_{xy}/xy
C_{yy}/y^2
$C_{x\alpha}/x\cot\alpha$
$C_{y\alpha}/y\cot\alpha$
$C_{\alpha\alpha}/\cot\alpha^2$
−0.1
0.0
0.1
$C_{x\beta}/x\cot\beta$
$C_{y\beta}/y\cot\beta$
$C_{\alpha\beta}/\cot\alpha\cot\beta$
$C_{\beta\beta}/\cot\beta^2$