Reconstructing silicon pixel hits using neural networks

S. Sekhar, O. Amram, P. Maksimovic, M. Swartz, T. Vami

Johns Hopkins University

APS DPF 2021, 14 July 2021

The CMS Pixel Detector

- Records the 3D trajectory of the charged particles through millions of silicon pixels
  - Robust seeding for track-pattern recognition
  - Reconstructing primary, secondary vertices
  - Fast tracking at trigger level etc.
- Closest to the interaction point, extremely radiation-hard
- Layout covers $-3 < \eta < 3$
  - 4 barrel layers arranged cylindrically
  - 3 layers at endcaps in turbine geometry
- Phase-1 pixel size : $100\mu m$ X $150\mu m$ X $285\mu m$
- Coordinate system: Local $x - y - z$

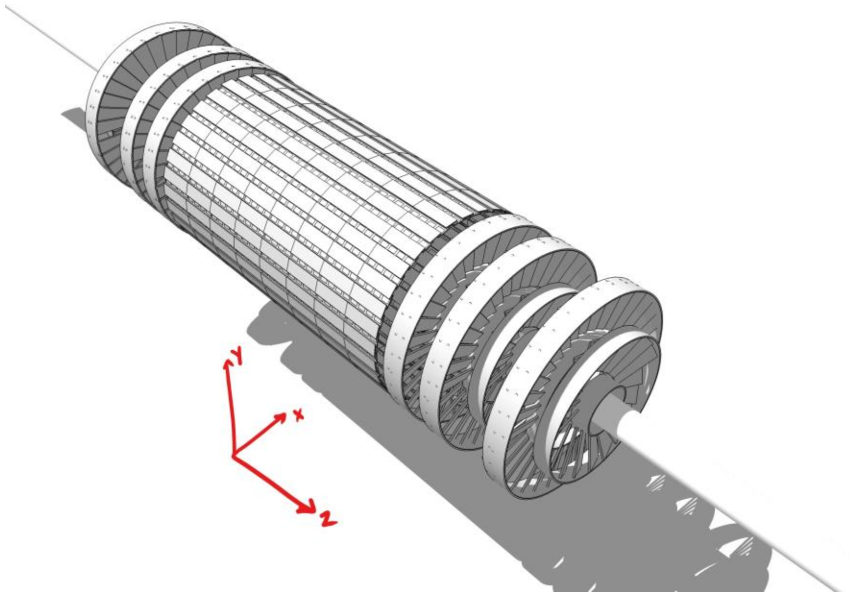


Fig 1. Phase-1 Pixel Detector Layout

Current Local Reconstruction

- Goal: to obtain local x and y position of track hits, and their errors
- Resolution is improved by charge sharing
  - Lorentz drift in local-x (due to magnetic field)
  - Geometry in local-y
- Important parameters: track angles α and β

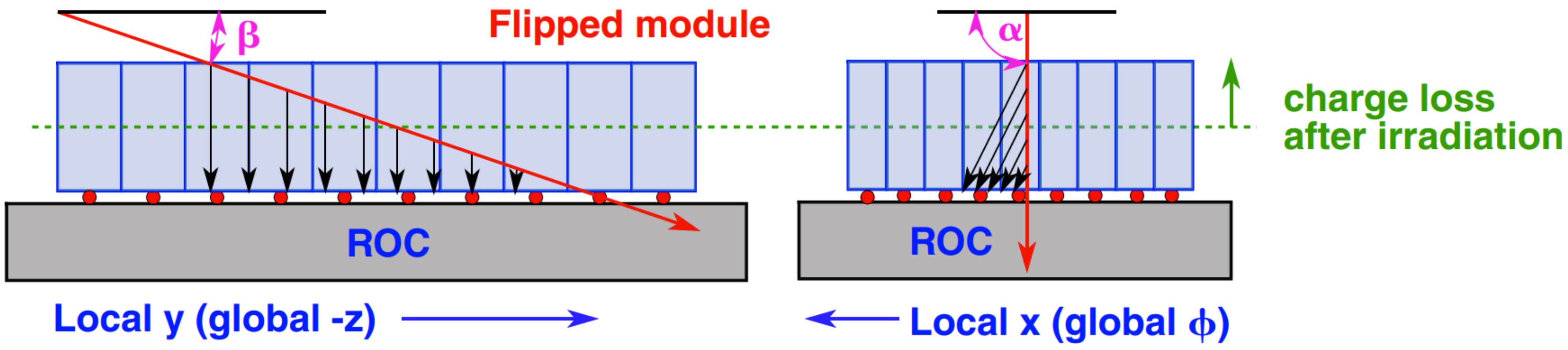


Fig 2. Illustration of track angles in the barrel

Algorithms currently used

- Generic Algorithm (used at High Level Trigger + Offline)
  - Quick, formula-based, does not necessarily require track angle information
  - Has a good performance for unirradiated sensors, does not model radiation damage very well
- Template Algorithm (used Offline)
  - Slow, uses track angle information
  - Fits clusters to projected cluster shapes generated from PIXELAV - a detailed sensor simulation
  - Has a superior performance for all sensors – models radiation damage very well

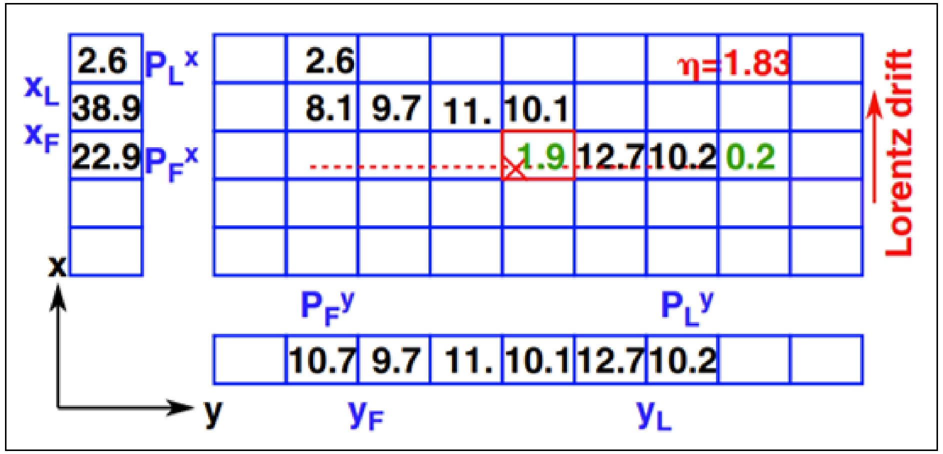


Fig 3. Illustration of a template

Motivation for this study

- CMS is moving to heterogeneous computing at the High Level Trigger from 2022
- Template and generic algos don't work well with GPU integration -> changes to the CPE (Cluster Position Estimate) algos desirable
  - Template object files are too large, cannot be read efficiently into GPUs
  - Performance of the generic algo decreases with heavy radiation damage
- Goal: to devise an algorithm that has the performance of the template algorithm that can perform inference with the speed of the generic algorithm
- Attempt: Neural networks
  - Computations involve millions of vector multiplication operations -> very compatible with GPUs!

Neural Networks

- Artificial neural networks (NNs) are computing systems that are very good at extracting underlying patterns in data without being explicitly programmed to do so
- Gained a lot of interest since ImageNet challenge in 2012
  - Visual recognition of objects in ~14 million images using neural networks
- CMS and ATLAS are also using deep learning in various tasks now
  - Jet reconstruction/classification
  - Jet substructure
  - b-tagging
  - Many more
- Supervised NNs are collections of nodes/neurons that together approximate a complex non-linear function
  - We can aggregate layers of neurons to extract more features: deep neural network

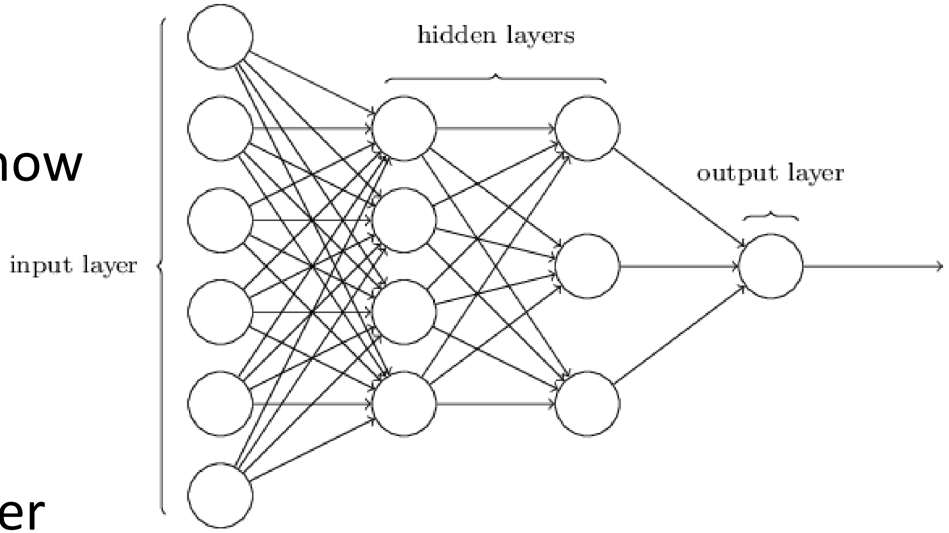


Fig 4. A NN with hidden layers

Convolutional Neural Networks

- Convolutional NNs are special types of NNs:
  - Extremely good at seeking hierarchical patterns in images
  - Require less preprocessing and reduced computational power
- Complex components:
  - Convolutional kernels/filters, Pooling, Batch Normalization, Dropout etc.
  - Fully connected layers just like in DNNs to produce final outputs

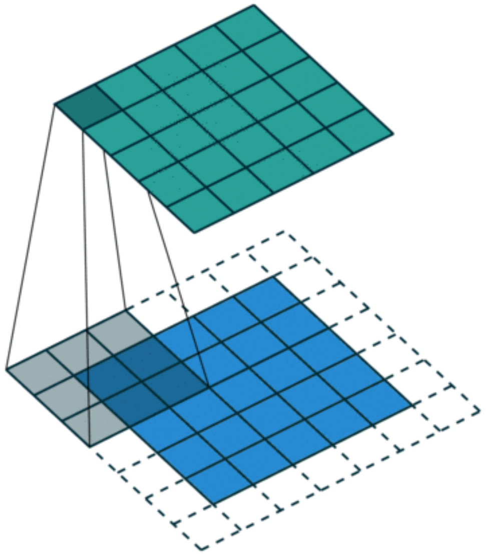


Fig 5. A filter in action

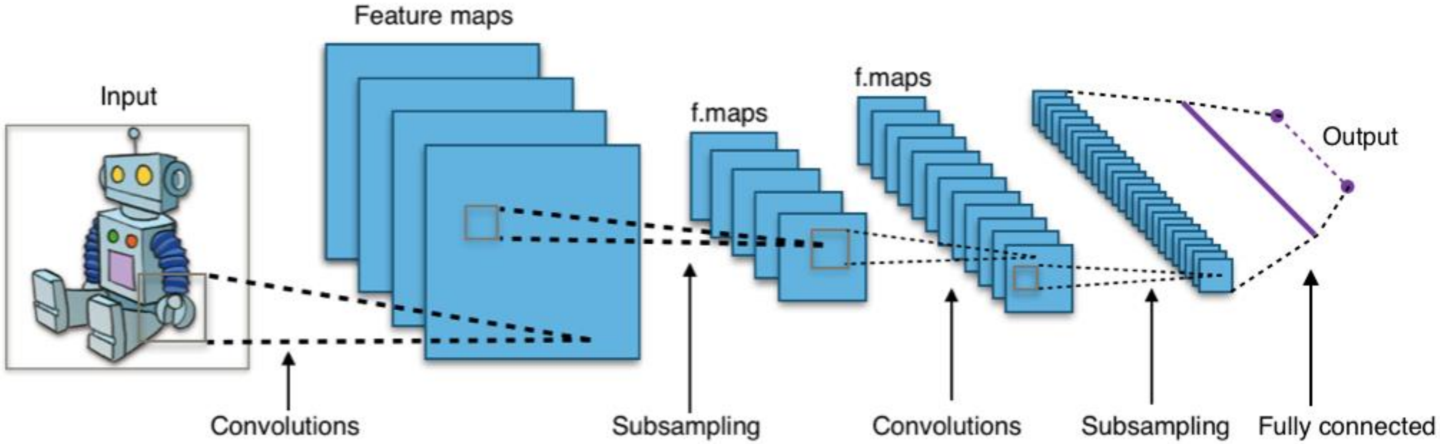


Fig 6. A toy CNN architecture

Neural Networks

- We train three models of NNs to predict the hit position (x,y) of a cluster of single particle hits
  - Hybrid 2D CNN (CNN+deep NN)
  - Hybrid 1D CNN – independent for x and y
  - Deep fully connected NN (DNN) – independent for x and y
- Model details:
  - Optimizer: Adam
  - Activation: ReLU
  - Loss: mean squared error
- Inputs: 13x21 matrix of pixels(or 1D projections) from PIXELAV simulations + Track angles $\cot\alpha, \cot\beta$
  - 10 million clusters for training, ~300k clusters for testing
  - Unirradiated and irradiated clusters $(2.3 \text{ X } 10^{15}\ n_{eq}/cm^2)$
- All clusters have been preprocessed to simulate real clusters in the detector
  - Applied $\tanh$ gain with appropriate noise parameters to describe the true pre-amplifier response, and implemented readout chip threshold
  - Implemented realistic cluster centering: shifted the geometric center of cluster to center of the matrix

Residuals

- We discuss our results by studying residuals: distributions of $(x_{pred} - x_{true})$ and $(y_{pred} - y_{true})$
  - x_{pred}, y_{pred}: Hit positions predicted by the neural network algos
  - x_{true}, y_{true}: True hit positions
- The residual distributions are fitted by the Gaussian function
  - Standard deviation (σ) of gaussian fit characterised well the peak of the residual distribution
- However, to characterize the overall behaviour of the algorithm the standard deviation of the residual distribution (denoted by RMS) is also used as it better captures the tails of the distribution

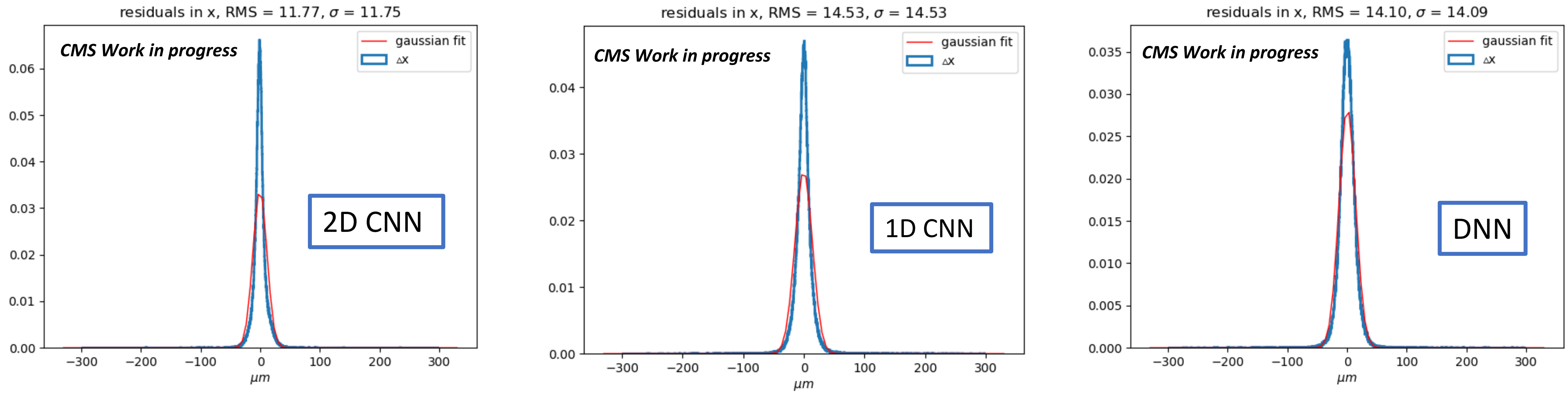


Fig 6. Residuals in x for Phase-1 irradiated clusters. More residual plots can be found in [4]

Results

- We can see that the while the template algorithm does provide superior resolution, the CNNs' predictions have the smallest RMS .

Unirradiated				
	x(μm)		y(μm)	
Algorithm	σ	RMS	σ	RMS
Generic	5.915	17.89	17.68	21.83
Template	6.607	11.8	16.11	20.52
DNN	11.98	11.99	19.99	20.26
1D CNN	11.34	11.37	18.39	18.39
2D CNN	10.56	10.59	20.0	20.0

Irradiated				
	x(μm)		y(μm)	
Algorithm	σ	RMS	σ	RMS
Generic	17.12	22.52	32.1	37.2
Template	14.34	17.2	27.92	29.96
DNN	14.09	14.10	38.97	38.97
1D CNN	15.49	15.50	35.99	36.86
2D CNN	11.75	11.77	26.9	26.95

Table 1: Comparison of results from all the algorithms.

Conclusion

- Neural Networks are doing well as Cluster Parameter Estimators and show promise!
- We have currently ironing out kinks in the implementation of inference within the CMS software framework as well as standalone (python and C++)
- In the long run, we expect that the improved local reconstruction will provide possible improvements to b-tagging and physics with data scouting
  - Precise reconstruction at the pixel detector will improve vertexing and b-tagging, lending usefulness to several physics analyses
  - In Phase-2 runs we will use data scouting by taking advantage of fast and accurate reconstruction at the HLT. Improvements to pixel local reconstruction at the HLT are thus desirable.
- Next steps: Error estimation and training the networks on double width pixels
  - In each ROC, the pixels on the edges are double the width of a regular pixel. Doubling of size -> more charge deposition.

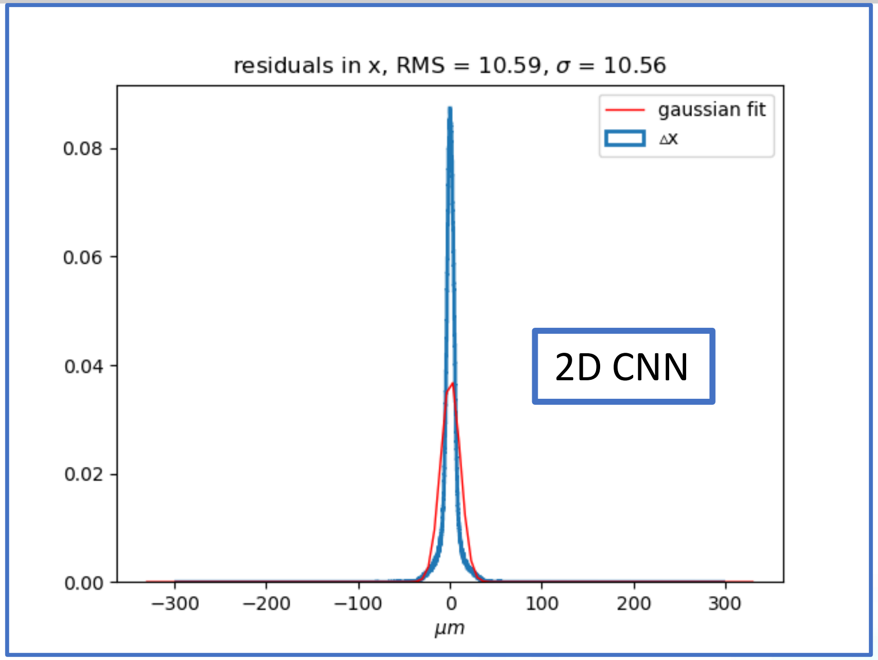
References

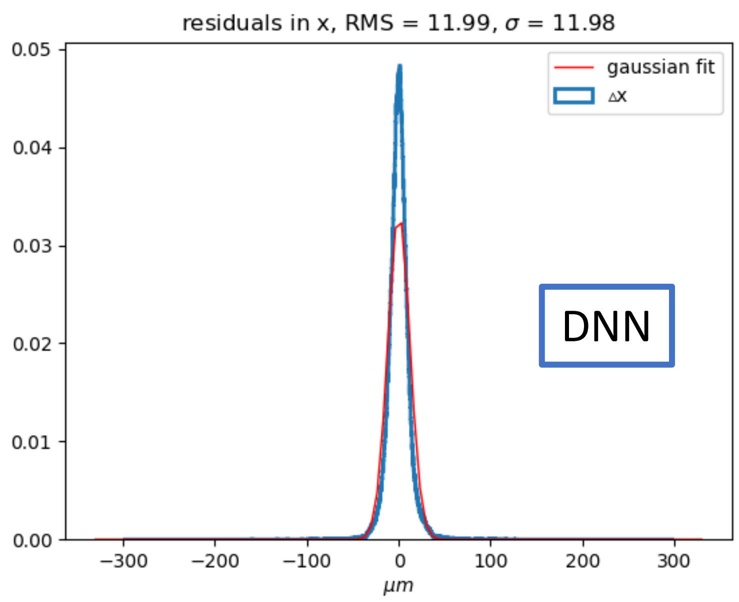
- *A new technique for the reconstruction, validation, and simulation of hits in the CMS pixel detector.* M. Swartz (CERN), D. Fehling (CERN), G. Giurgiu (CERN), P. Maksimovic (CERN), V. Chiochia (CERN). DOI: 10.22323/1.057.0035. Published in: PoS VERTEX2007 (2007), 035S.
- *Position Determination of Pixel Hits*. Susanna Cucciarelli (Basel U.), Danek Kotlinski (PSI, Villigen), Teddy Todorov (CERN)
- *A Detailed Simulation of the CMS Pixel Sensor.* M. Swartz (2002).
- Our presentation to the CMS ML Forum, Feb 24, 2021: https://bit.ly/3fDUeCu

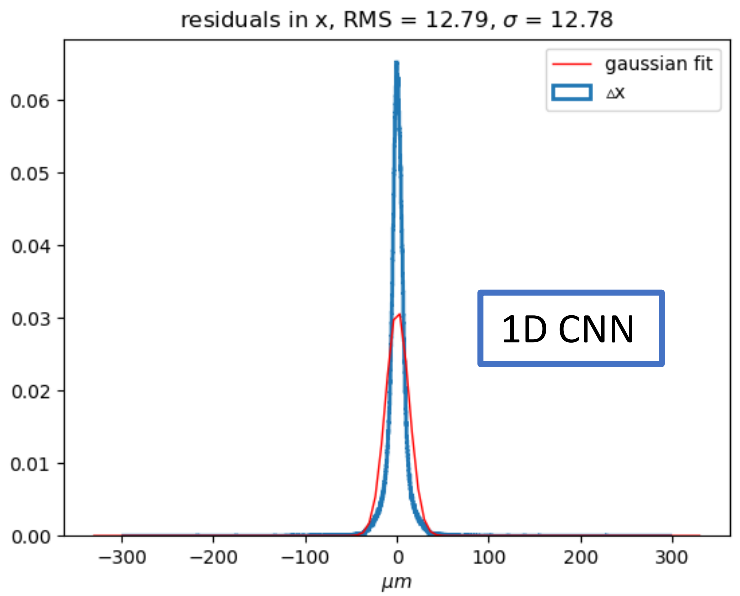
BACKUP

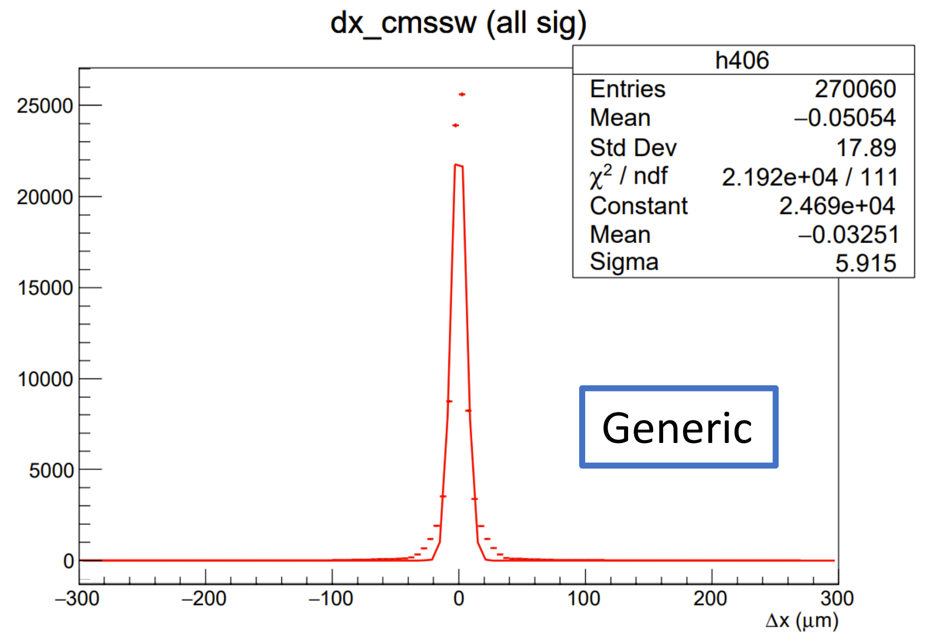
Results: Unirradiated clusters

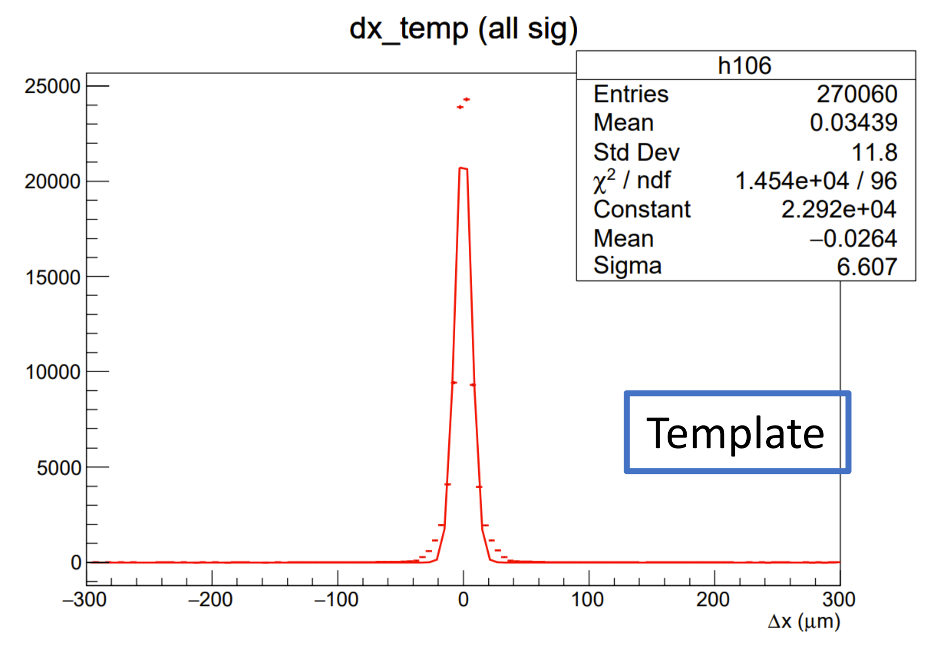
Comparison of NNs to other CPE algos : x





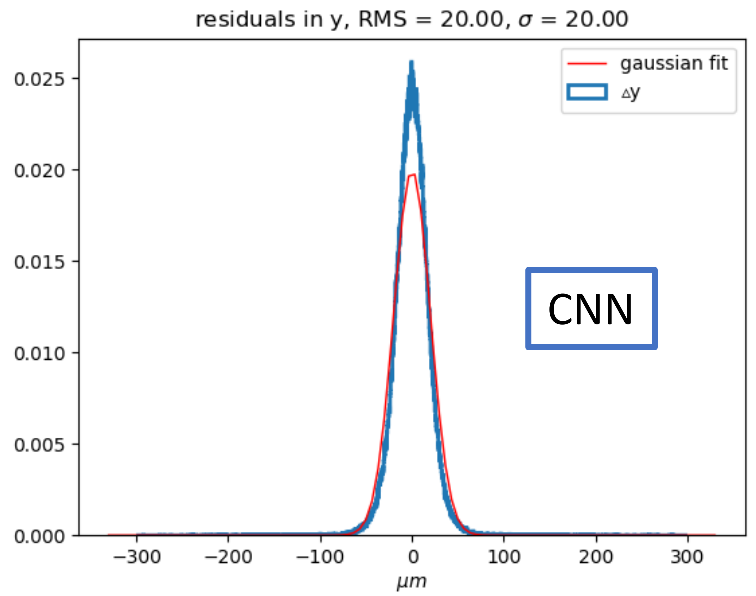


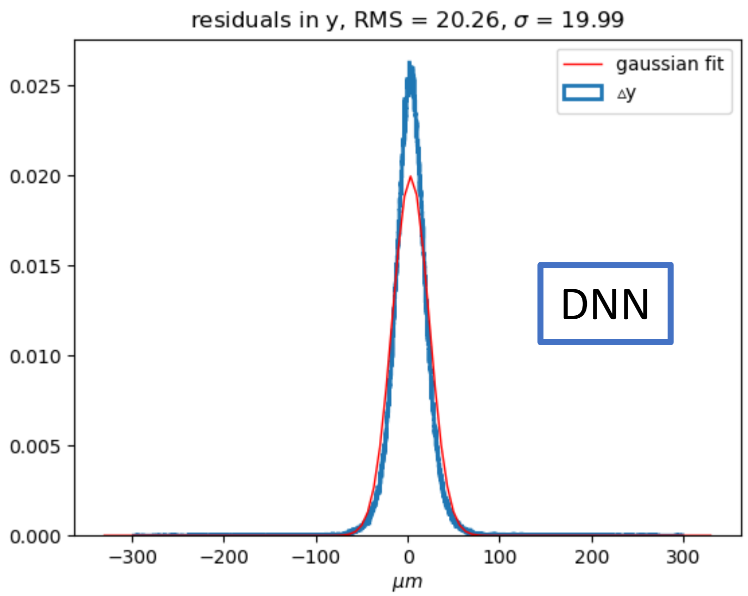


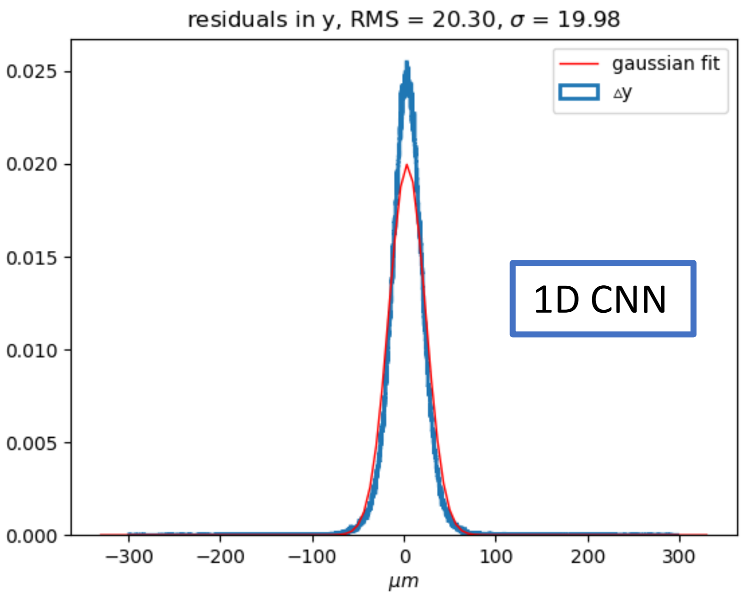


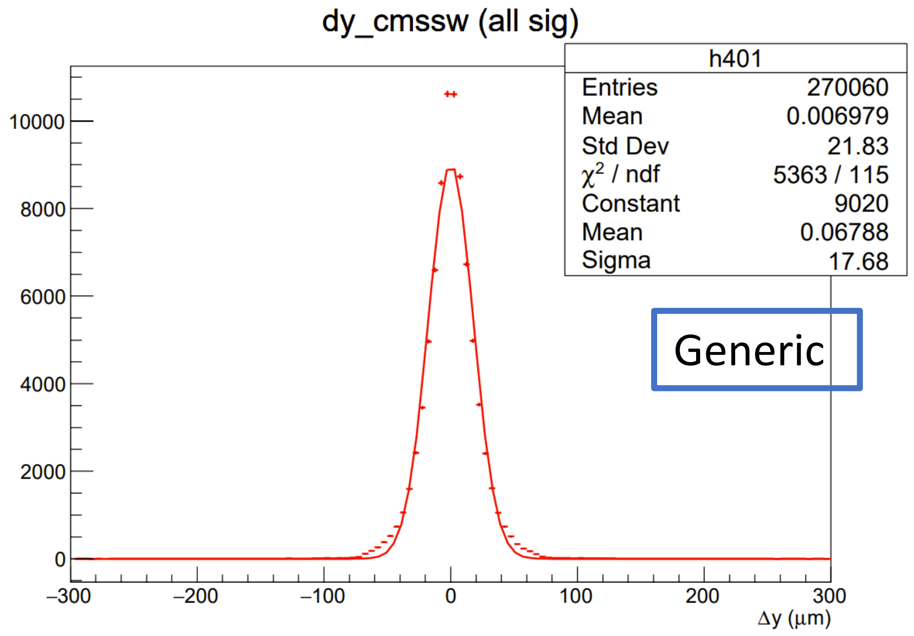
Algo	RMS(μm)	σ(μm)
2D CNN	10.59	10.56
1D CNN	12.79	12.78
DNN	11.99	11.98
Template	11.8	6.607
Generic	17.89	5.915

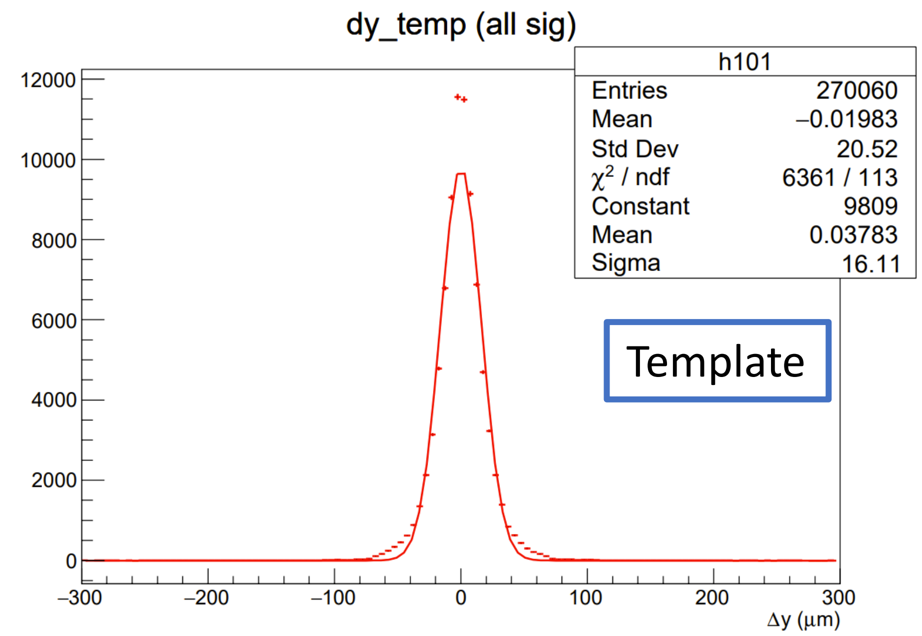
Comparison of NNs to other CPE algos : y







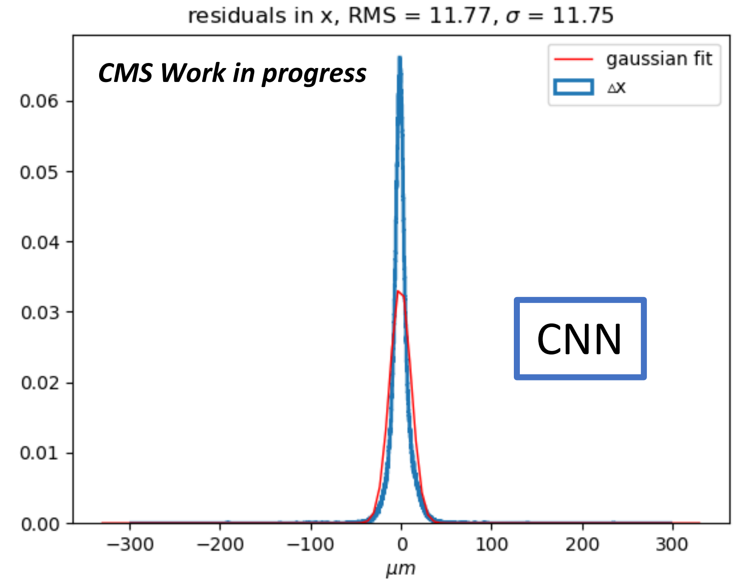


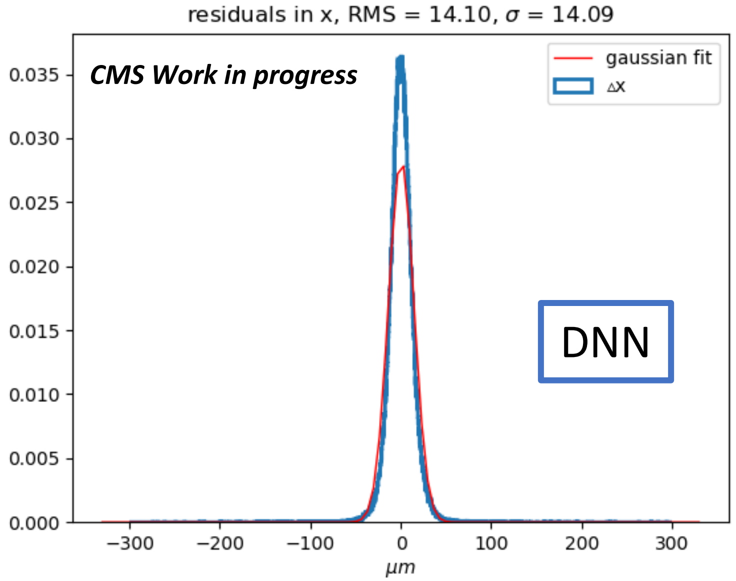


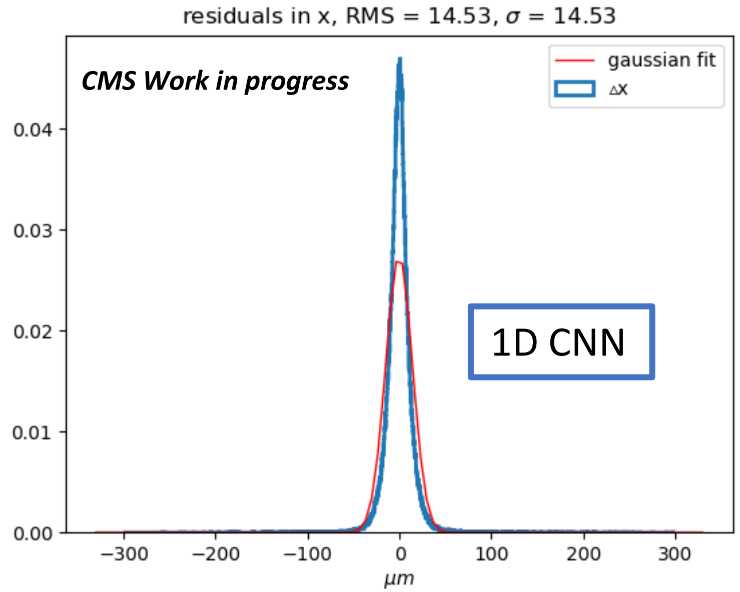
Algo	RMS(µm)	σ(µm)
2D CNN	20.00	10.56
1D CNN	20.30	12.78
DNN	20.26	19.99
Template	20.52	16.11
Generic	21.83	17.68

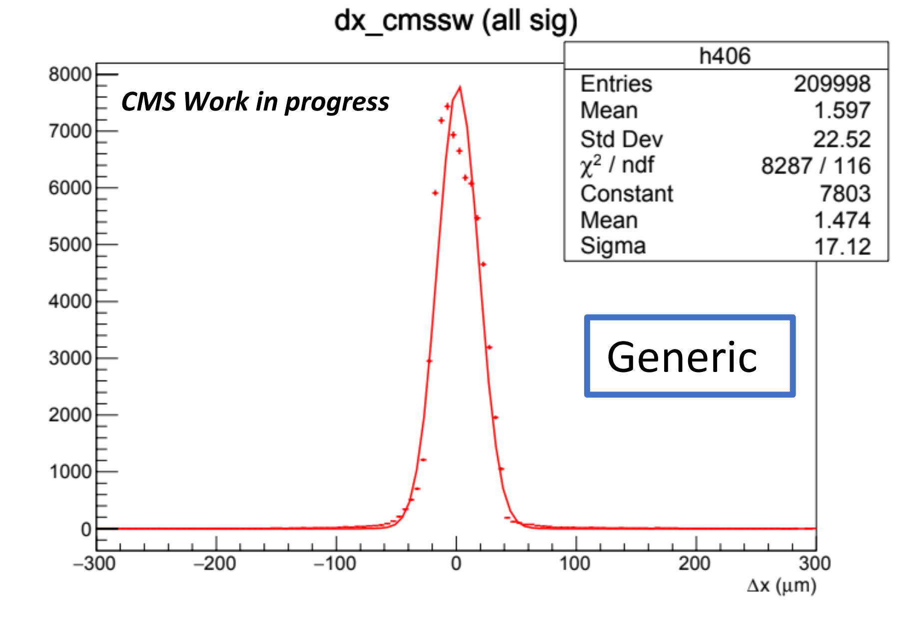
Results: Irradiated clusters

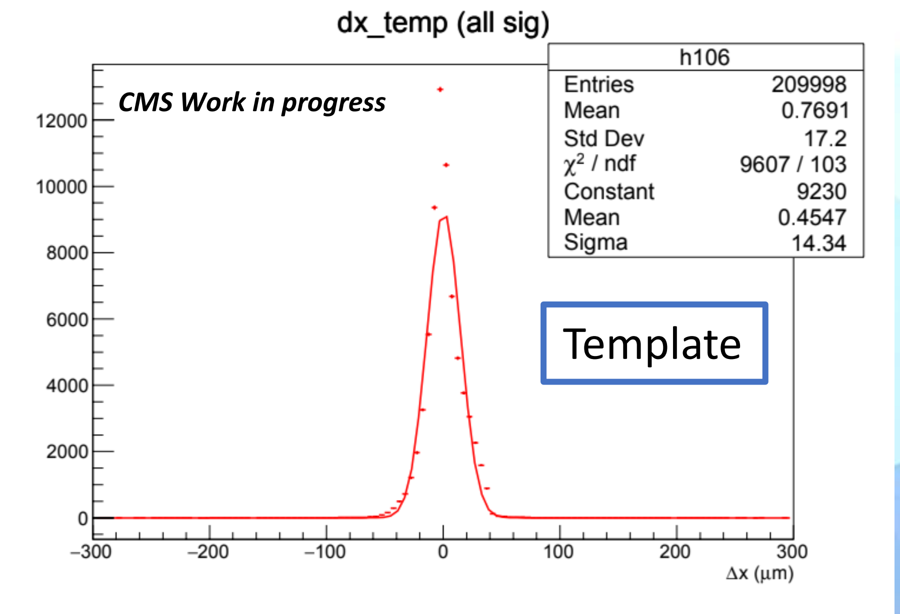
Comparison of NNs to other CPE algos : x





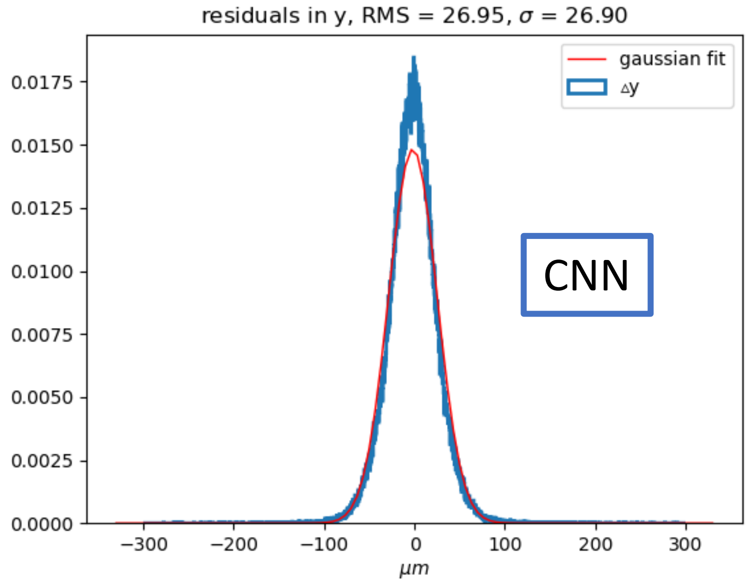


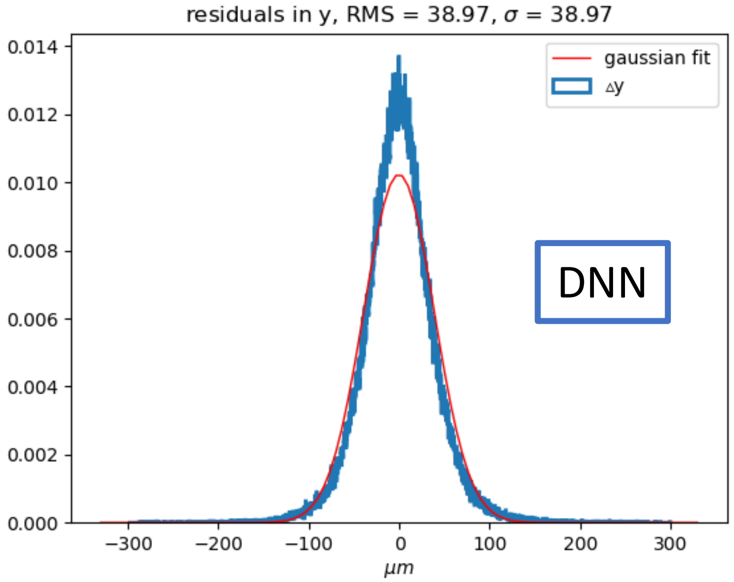


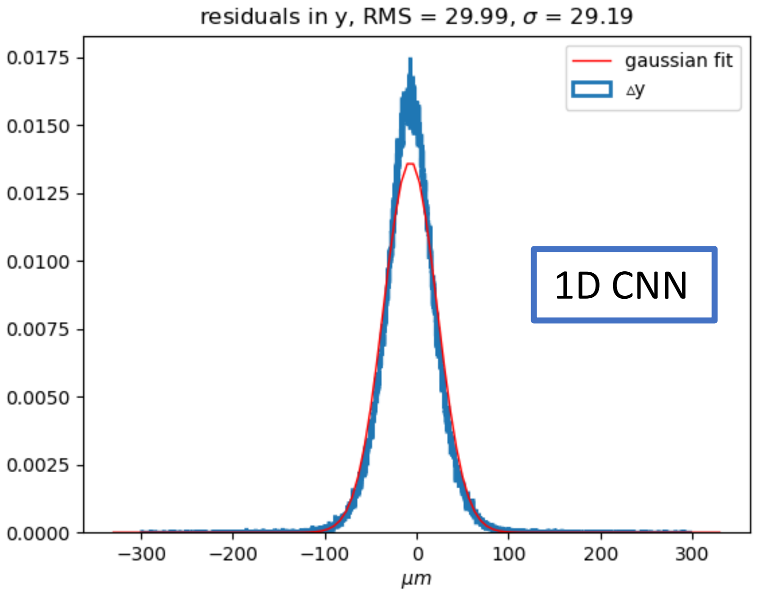


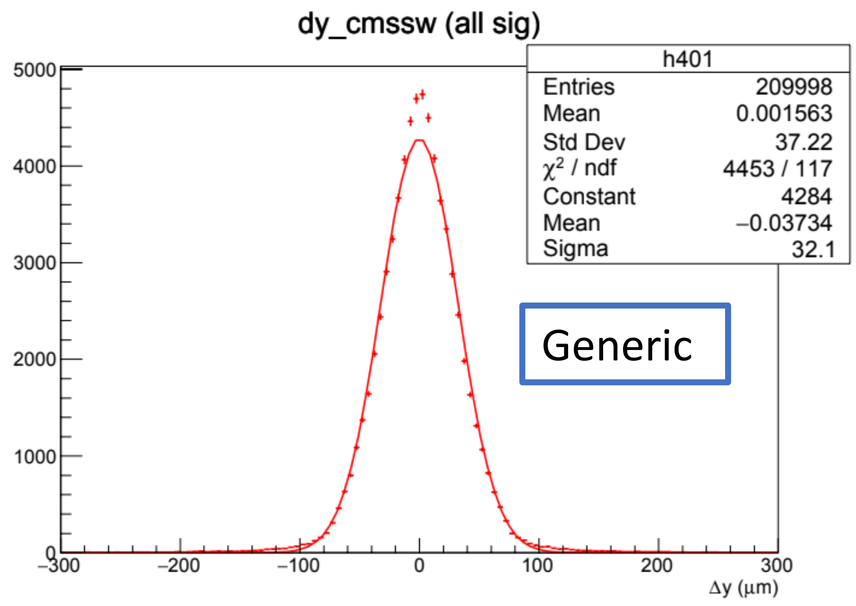
Algo	RMS(μm)	σ(μm)
2D CNN	11.77	11.75
1D CNN	14.1	14.09
DNN	14.53	14.53
Template	17.2	14.34
Generic	22.52	17.12

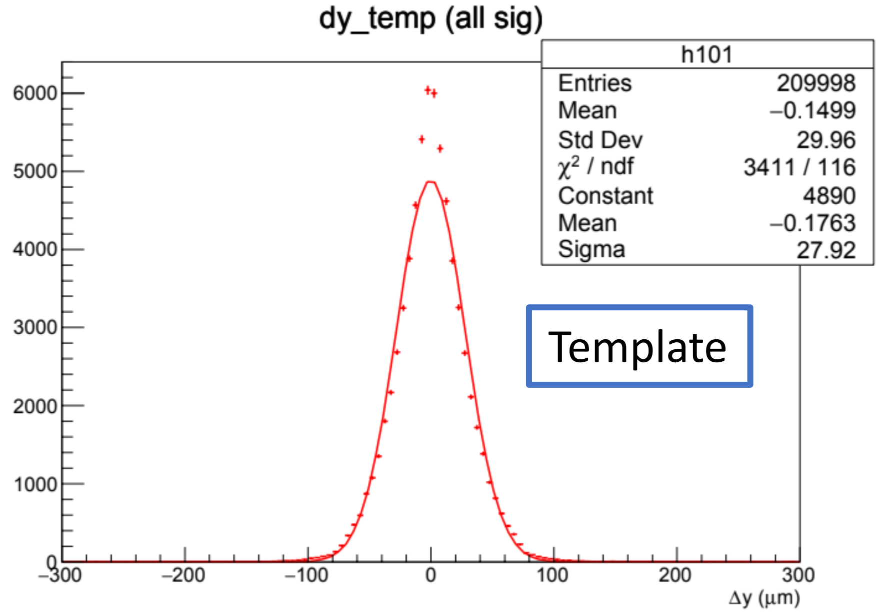
Comparison of NNs to other CPE algos : y











Algo	RMS(μm)	σ(μm)
2D CNN	26.95	26.9
1D CNN	38.97	38.97
DNN	29.99	29.19
Template	29.96	27.92
Generic	37.22	32.1