Particle-Flow End-to-end Reconstruction for Highly Granular Calorimeters

Jan Kieseler, Shah Rukh Qasim, Philipp Zehetner

Work in context of CMS ML4Reco

Motivation - Highly Granular Calorimeters

Particle flow:
Add reconstructed tracks to inputs

End-to-end:
Single step from hits (tracks) to showers

Output:
- Clustered detector hits and tracks
- Perform particle identification
- Energy regression
- Energy uncertainty

Literature:

[1] arXiv:1902.07987 GravNet

[2] arXiv:2002.03605 Object Condensation

[3] arXiv:2204.01681 Full Reconstruction

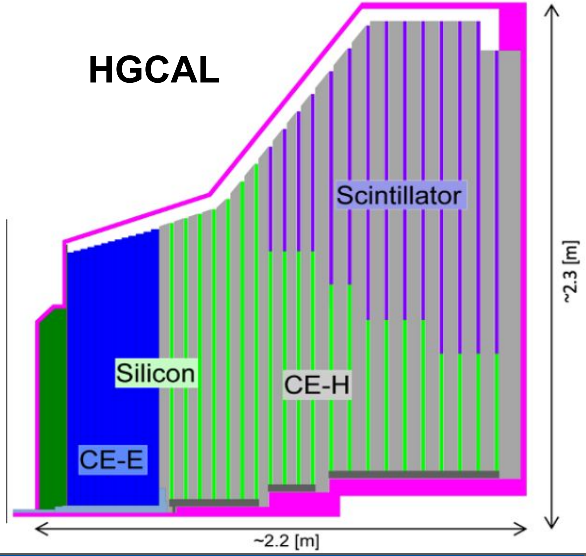
Published work (toy detector)
- Clustering in 200 PU
- with energy prediction
- in toy detector
- See publication [3]
- or ACAT 2022

Ongoing work (toy detector)
- inclusion of tracks
- energy uncertainty
- particle identification
- updated toy detector

CMS - High-Granularity Calorimeter (HGCAL)

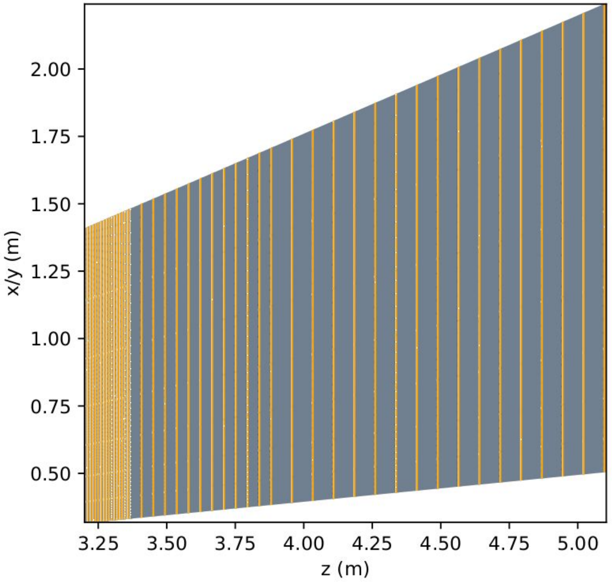
- 6 million readout channels (silicon + scintillators)
- Expected 200 pile-up (PU)
- Around 200k active channels per event

HGCAL



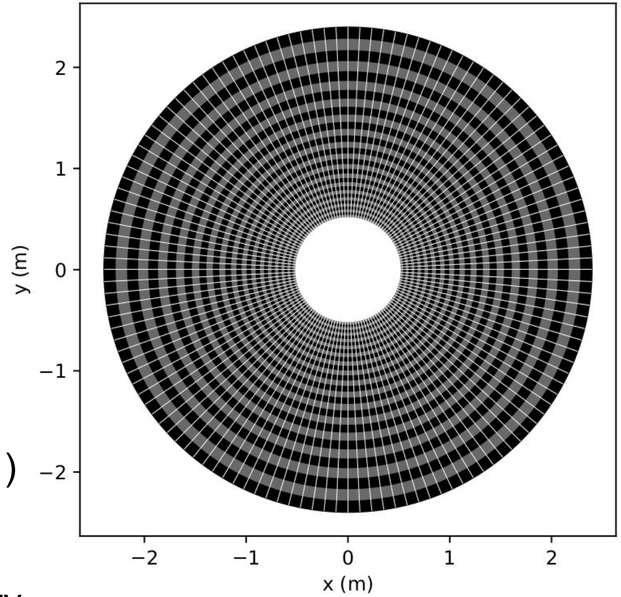
Detector

Toy Detector used for these studies



Longitudinal view [3]

- Sampling calorimeter
- 1.5 ≤ η ≤ 3.0
- 200 µm silicon sensors
- 0.8M sensors (V1)
- 3.1M sensors (V2)
- 180k hits in 200 PU (V1)
- 300k hits in 200 PU (V2)
- square in η and φ
- 28 layers ECAL
  17 radiation lengths (V1)
- 28 layers HCAL
  10 nuclear interaction lengths (V1)
- 50k noise hits (≅ 120 GeV) (V2)

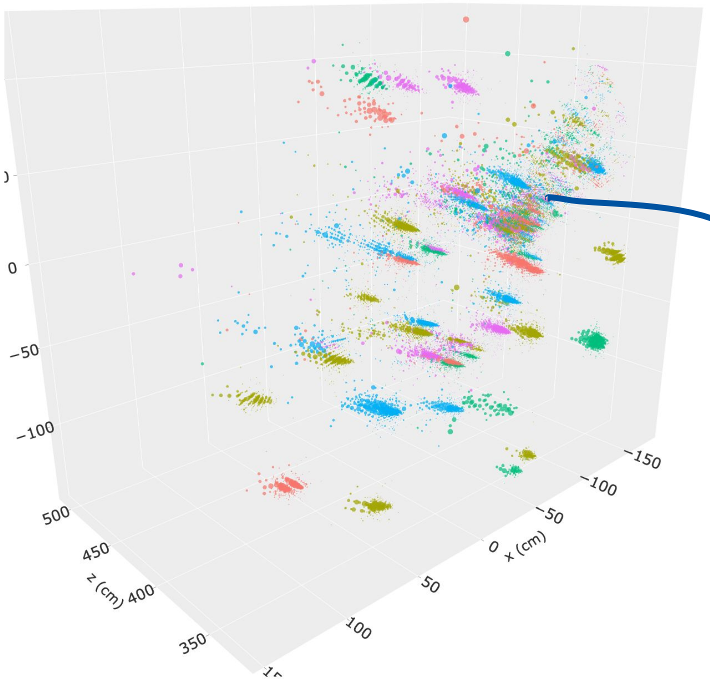


Transverse view [3]

Aiming for events with similar complexity as HGCAL, while simplifying simulation

Training Data (V1)

Example train event - 60 Particles + PU in 30° φ region



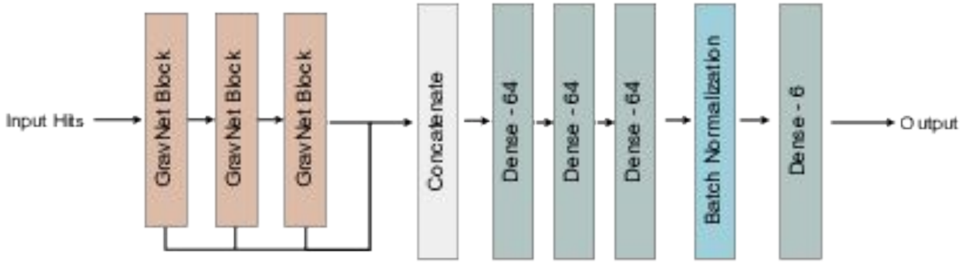
- **Single Particle**
  - Simulated with GEANT4
  - $e^-,\ \gamma,\ \pi^\pm,\ \pi^0,\ \tau^\pm$
  - $E \in [0.1, 200]$ GeV
  - particles generated 1 mm in front of detector (no tracker or magnetic field)
- **Pile Up**
  - Minimum bias proton-proton collisions generated using PYTHIA8
  - √s = 13 TeV
  - only added in random 30° φ region
    - memory constraints while training
    - reduces hits from ~180k to ~34k
    - <u>this will not be applied to test sets</u>

Training events are 60 single particle simulations combined random Gaussian detector noise and 200 PU added in a random 30° φ region

Architecture - GravNet

Architecture is based around GravNet [1] **layers**

- ❖ Graph based approach is natural for sparse data
- ❖ Allows propagation of information through detector
- ❖ Faster than similar approaches e.g. DGCNN [4]

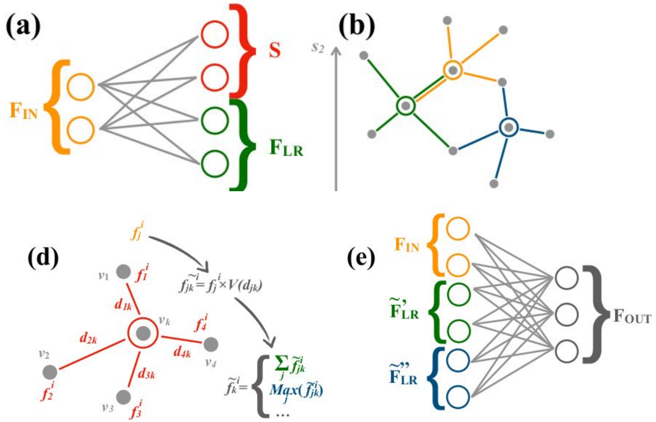


Architecture of our model

GravNet

Speed
- a) Transform input features $\mathbf{F_{IN}}$ into
  - ○ transformed features $\mathbf{F_{LR}}$
  - ○ latent coordinates $\mathbf{S}$
- b) Build graph using coordinates $\mathbf{S}$

Performance
- d) Aggregate weighted features
  - ○ Weights depending on distance
  - ○ Aggregation typically is *mean* or *max*
- e) Concatenate the new features

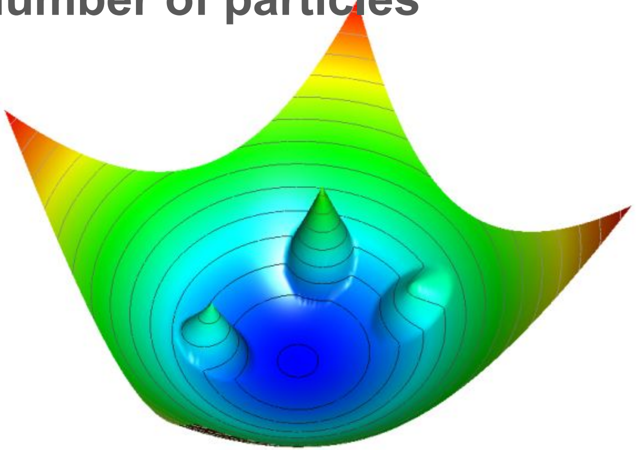


[4] Yue Wang et al. Dynamic Graph CNN for Learning on Point Clouds

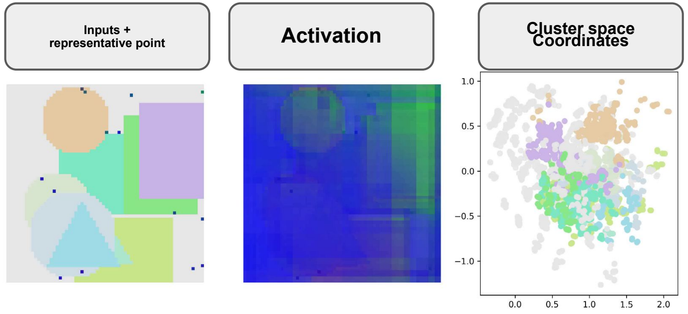
Object Condensation Loss

Enables us to reconstruct an a priori unknown number of particles

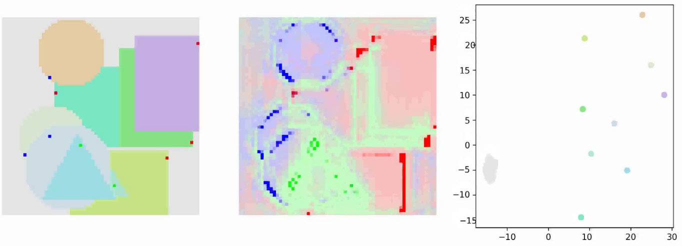
- ❖ Every vertex can represent a shower
- ❖ Encourage model to have one representative vertex (RV) per object
- ❖ In the latent space:
  - ➢ Vertices are pulled towards their RV
  - ➢ Vertices are pushed away from other RVs
  - ➢ Hits of the same shower are clustered together
- ❖ Points around RVs in the latent space are collected as shower
- ❖ RV is then used to predict shower's properties



J. Kieseler, arXiv:2002.03605, Eur. Phys. J. C 80, 886 (2020)



←before Training
after Training →



Energy & Momentum

Energy Regression

- Unfavourable to directly predict showers' energies as
  - energies can differ by orders of magnitude
  - sensitive to splitting or merging showers
- Instead learn a correction factor ψ multiplied to shower's energy

$$E_{pred} = \psi \sum_{h \in \text{shower}} e_h$$

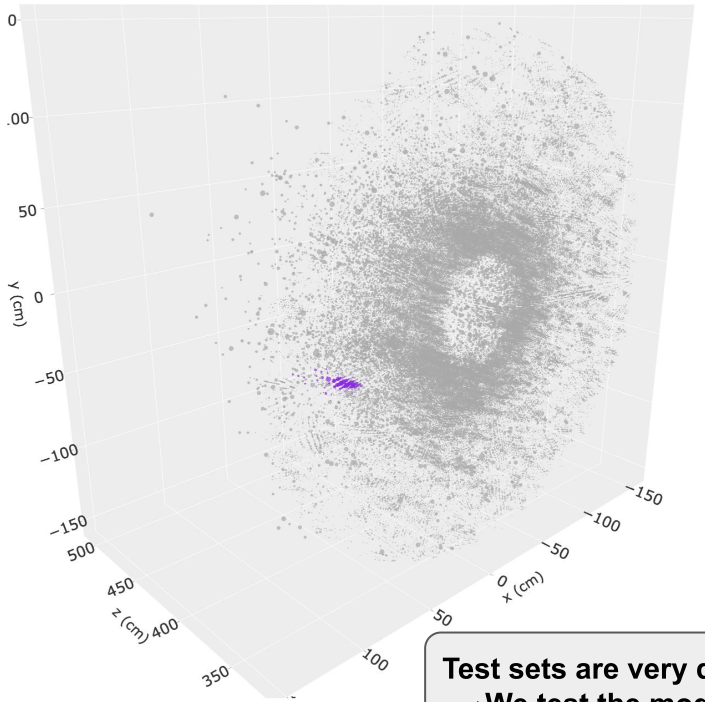
Transverse Momentum

- Calculated from energy using the showers energy weighted mean position
- For consistency this is used for
  - $p_{T\text{pred}}$ (using E_{pred})
  - $p_{T\text{truth}}$ (using E_{truth})

$$p_T = E / \cosh \eta$$

Test Data

Example test event - Single Particle + 200 PU

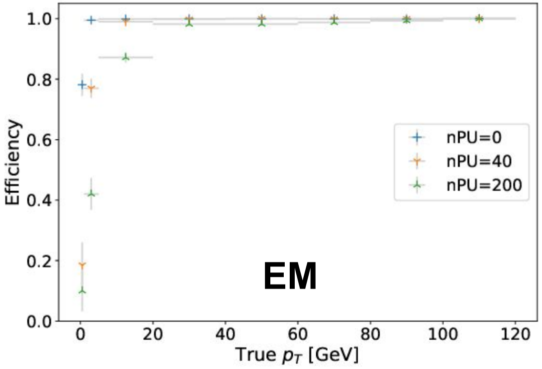


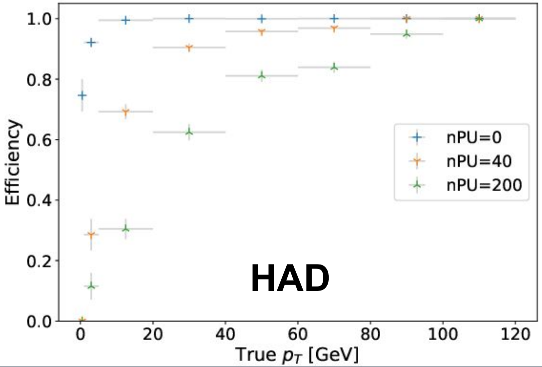
- **Single Particle**
  - Simulated with GEANT4
  - e^-, γ, π^+
  - $E \in [0.1, 200]$ GeV
  - $\eta \in [1.6, 2.9]$
- **Jets**
  - $q\bar{q} \rightarrow t\bar{t}$
  - generated at √s = 13 TeV using PYTHIA8
- **Pile Up**
  - Minimum bias proton-proton collisions generated using PYTHIA8
  - √s = 13 TeV

Test sets are a single particle or jet events combined with random Gaussian detector noise and up to 200 PU

Test sets are very different from training set
→We test the model's ability to generalize

Efficiency & Unmatched Rate





Efficiency EM

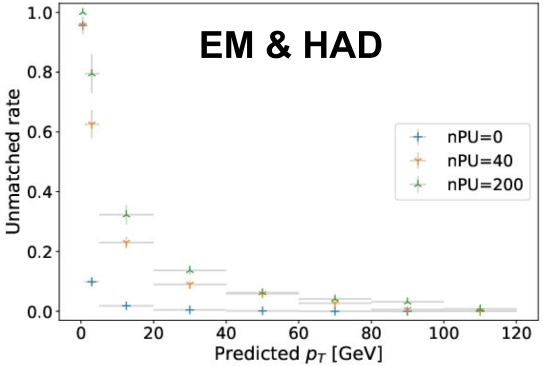
- ❖ Efficiency quickly approached 100% with high p_T
- ❖ PU reduces efficiency for showers with small p_T (< 20 GeV)

Efficiency HAD

- ❖ PU has larger impact on reconstruction efficiency
- ❖ 200 PU hadronic showers are the most challenging case

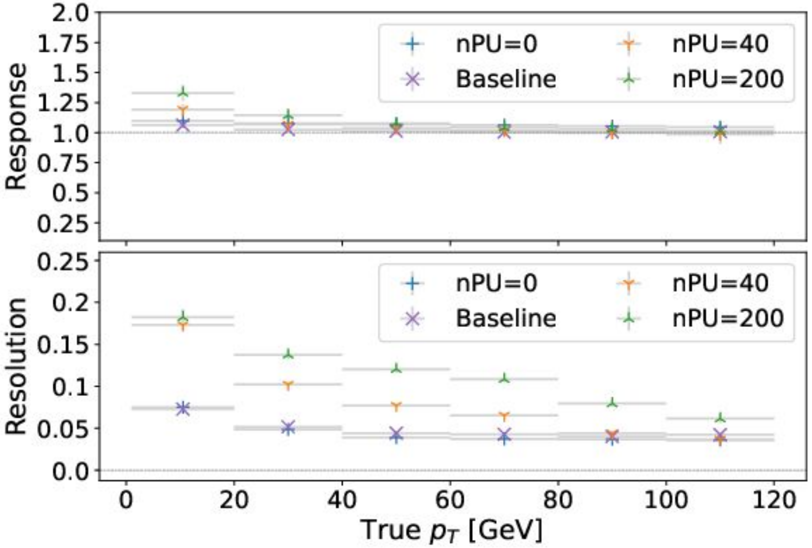
Unmatched Rate

- ❖ High PU causes low p_T showers to be unmatched

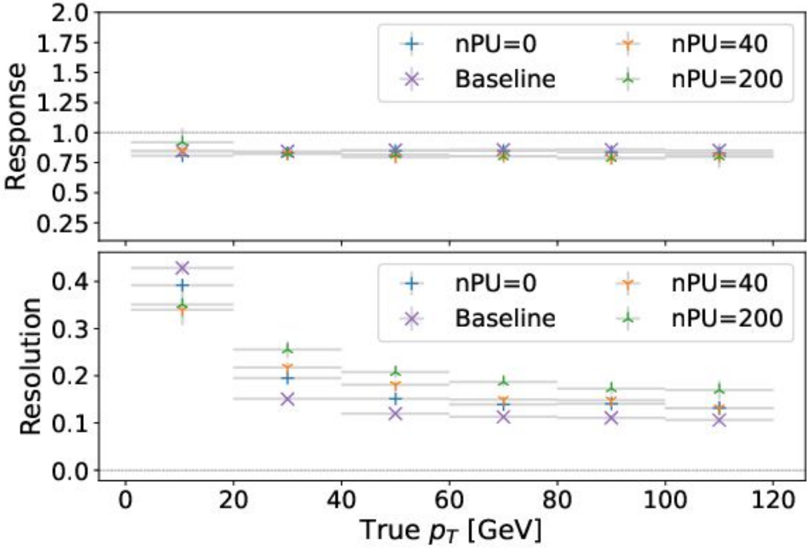


Response & Resolution

EM



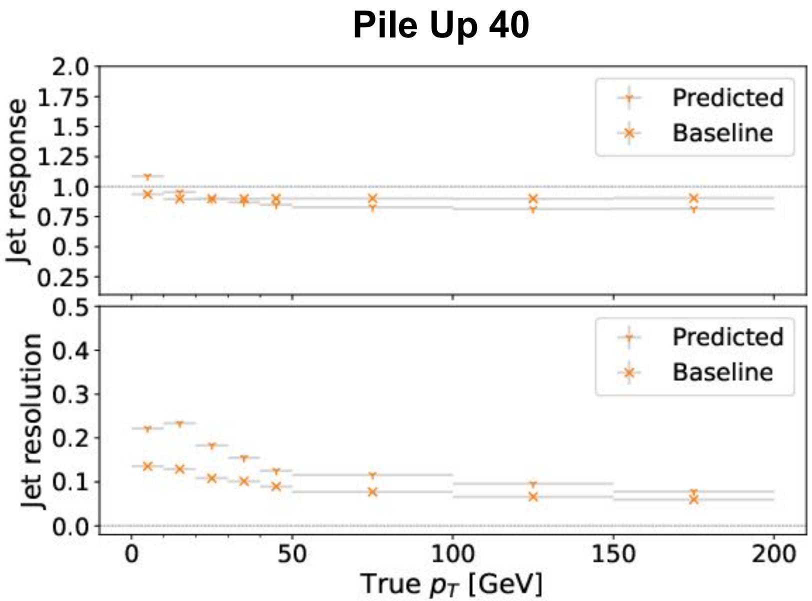
HAD

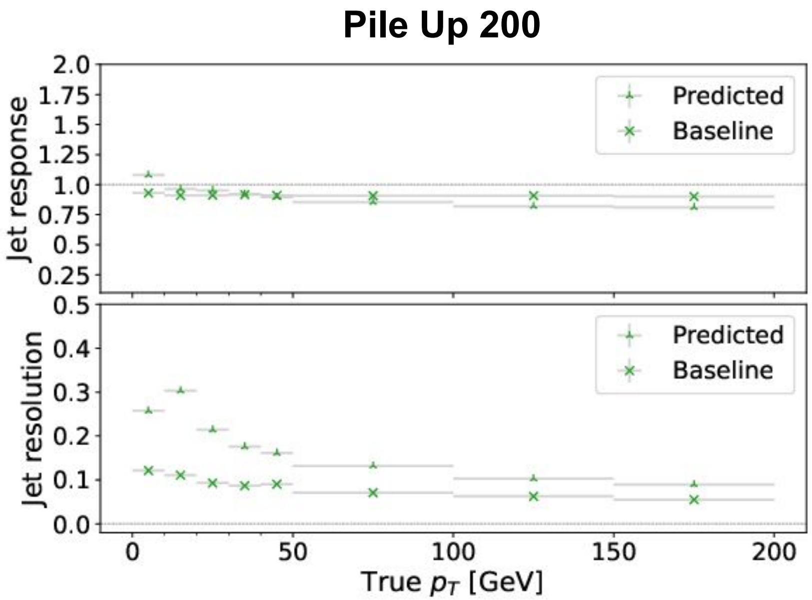


- ❖ Without PU resolution and response match **truth assisted** base line
- ❖ PU influences resolution, but only affects response for low energies

- ❖ Lower resolution for hadronic particles also for **truth assisted** base line
- ❖ PU again mostly affects resolution

Jet Reconstruction



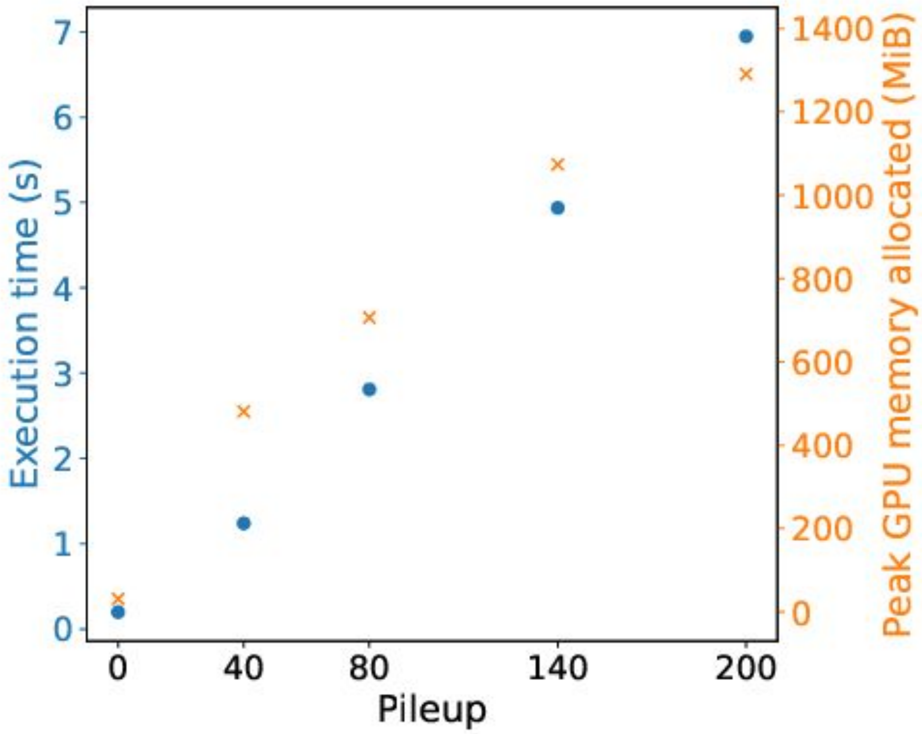


Baseline:
based on true deposited energy of incident non-pileup particles

- Response < 1 due to large hadronic contributions
- Comparable response to baseline
- Resolution approaching 10% in both PU scenarios

Computational Requirements

- Inference time and memory both scale linear with number of hits in detector
- Less than 10 seconds inference time for 200 PU (NVIDIA V100 GPU)
- Less than 1.5 GB peak memory usage for 200 PU
  →Can be deployed on low-end GPUs
- Ongoing work on inclusion of small clustering models to compress input indicate potential for significant speed ups



Ongoing Work - Overview

Addition of Tracks

- Tracks assigned to charged particles
- Energy corresponds to particles' true energy with Gaussian noise added
- Boolean flag to separate tracks from hits
- Network embeds hits and tracks in first layers

Updated Detector

- Finer granularity, number of channels as well as number of active hits now larger than expected for HGCAL
- More realistic noise level, both in terms of noise hits and noise-related energy deposits
- More absorber material for ECAL

Particle Identification

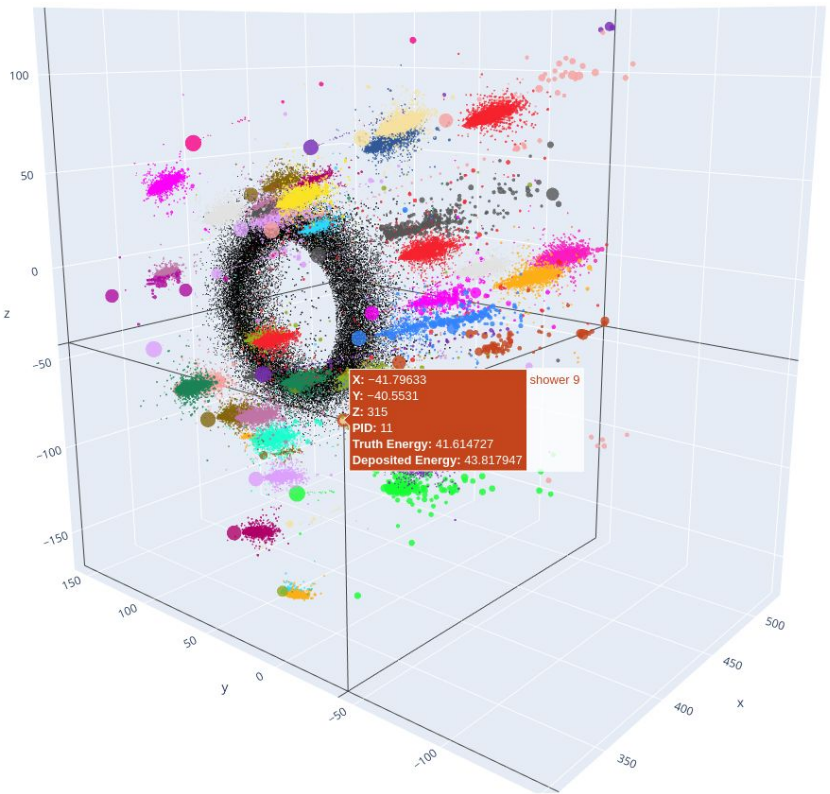
- Classifying between: *photon, electron, muon, charged hadron and neutral hadron*
- Prediction on each individual hit, condensation point decides prediction for full shower
- Categorical cross-entropy loss, non-linear scaled with β

Energy Uncertainty

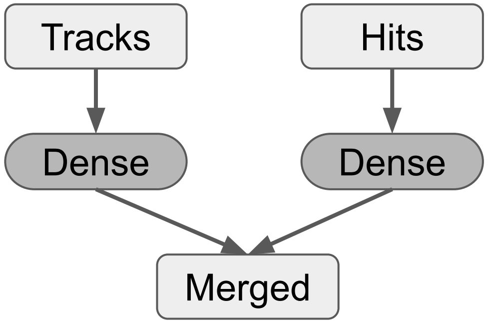
$$L_{\text{Energy}} = \ln \sigma_{\text{E}}^2 + \frac{(E_{\text{pred}} - E_{\text{true}})^2}{\sigma^2}$$

- Predicted energy $E_{\text{pred}} = c \cdot E_{\text{dep}}$ where *c* is learned as correction factor multiplied to the deposited energy E_{dep}
- Predicted uncertainty $\sigma_E = c_\sigma \cdot E_{\text{dep}} + 1\,\text{GeV}$ where c_σ is a learned correction factor
- Minimizing L_{Energy} should give reasonable estimates for the uncertainty

Particle Flow with Tracks



- Tracks are treated like detector hits in front of calorimeter
- Tracks contain the particle's full energy
- Flag is used to embed tracks and hits at beginning of network
- Tracks will be clustered together with hits (ideally never more than one)
- Different strategies on choosing best energy (especially with realistic track resolution)



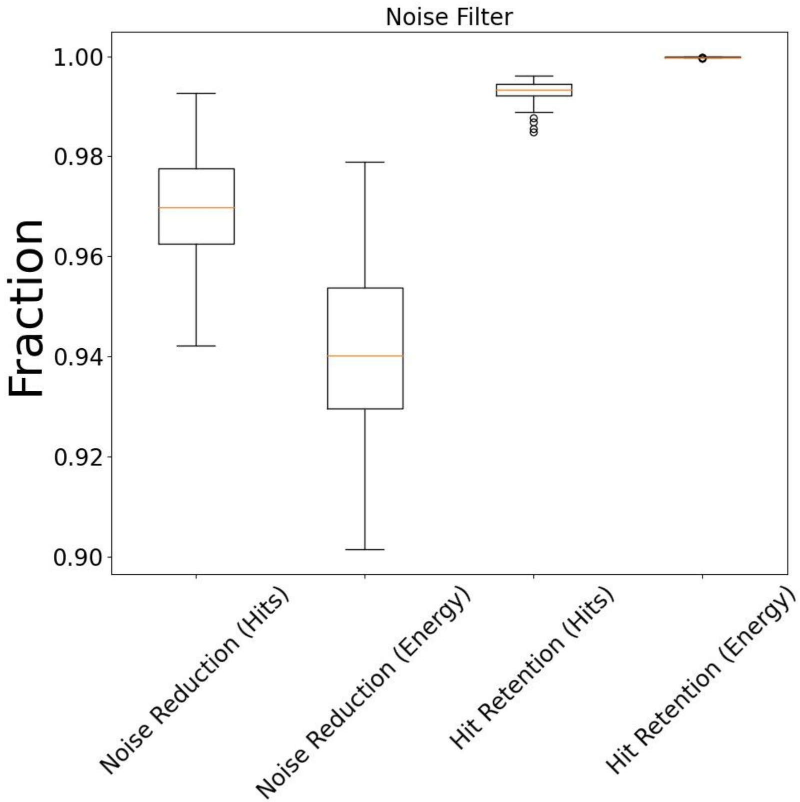
Detector and Noise

New toy detector:

- More channels compared to HGCAL
- More expected hits than in HGCAL
- Significantly more noise than previous version

Noise Filter

- Reduces the number of hits by ~50k
- Simple message passing in low-dimensional cluster space
- Trained separately at beginning, afterwards weights are fixed
- Very efficient and does not significantly reduce real hits



Summary and Outlook

Summary

- Demonstrated end-to-end reconstruction of particles and jets in up to 200 PU
- Promising performance, often close to a truth assisted base line
- Demonstrated generalization over different types of events
- Fast execution time scaling linear with detector hits

Outlook

- Adding track information to use a particle-flow approach
- Upgraded detector geometry towards even higher granularity
- Significantly higher noise levels taken care of by noise filter
- Uncertainty estimate for energy regression

Planned to be published toward the end of the summer

Backup

EIOU: **E**nergy-weighted hit-**i**ntersection **o**ver hit-**u**nion

EIOM: **E**nergy-weighted hit-**i**ntersection **o**ver hit-**m**inimum

$$\mathrm{EIOU}(t,p) = \frac{\sum_{h \in H_t \cap H_p} e_h}{\sum_{h \in H_t \cup H_p} e_h}$$

$$\mathrm{EIOM}(t,p) = \frac{\sum_{h \in H_t \cap H_p} e_h}{\min\left(\sum_{h \in H_t} e_h,\ \sum_{h \in H_p} e_h\right)}$$

$$\hat{p} = \operatorname{argmax}_{p \in P} \left(\mathrm{EIOU}(\hat{t},p)\right)$$

Baseline

$$E_{\text{baseline}} = \sum_{h \in H_t} e_h$$

This baseline will be hard to match as it uses the truth information of the showers

Efficiency:

% of true showers where $\mathrm{EIOU}(\hat{t},\hat{p}) \geq 0.5$

Unmatched Rate:

% of predicted showers where $\mathrm{EIOU}(\hat{t},p) < 0.5$

$\mathrm{EIOM}(\hat{t},p) > 0.9$

Response

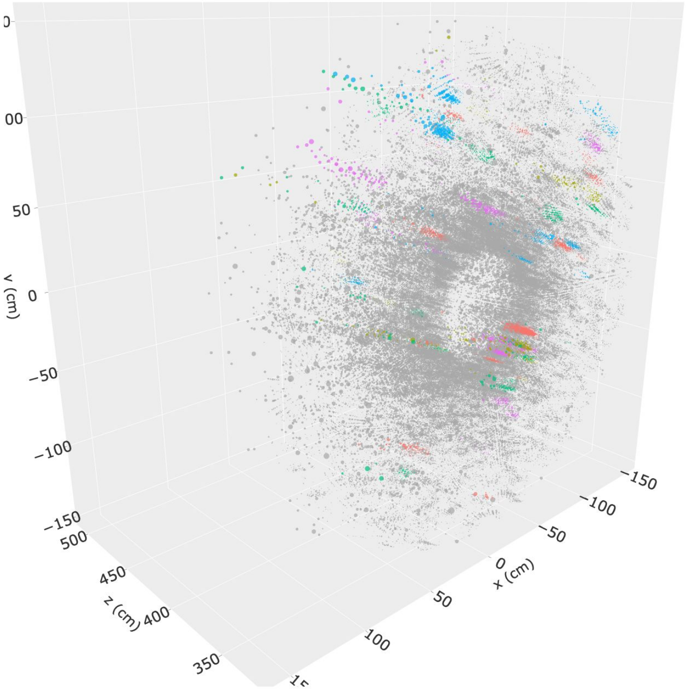
$$< p_{T_{pred}} / p_{T_{truth}} >$$

Mean-corrected resolution

$$\sigma\left(p_{T_{pred}} / p_{T_{truth}}\right) / < p_{T_{pred}} / p_{T_{truth}} >$$

Jet Reconstruction

Example test event: $q\bar{q} \rightarrow t\bar{t}$ + 200 PU



Only consider truth showers from non-pileup particles

Remove predicted showers with > 90% of their energy from pileup interaction (assume existence of pileup identification algorithm)

Distance parameter R=0.4

Based on $\Delta R = \sqrt{\Delta\eta^2 + \Delta\Phi^2}$
Among jets with
$|\Delta p_T|/p_{T_{true}} < 0.5$
$\Delta R < 0.3$
select best match by ΔR

truth showers → **remove pileup**

predicted showers → **filter pileup**

anti-kt jet-clustering[5]

truth jets | **predicted jets**

matching

[5] M. Cacciari, G. P. Salam, and G. Soyez. The anti-kt jet clustering algorithm. Journal of High Energy Physics, 2008(04):063, 2008.