https://github.com/hep-lbdl (code + data)

Accelerating HEP Inference with Deep Neural Networks

BERKELEY LAB

Luke de Oliveira, Benjamin Nachman, Michela Paganini

Joint WLCG and HSF Workshop, Napoli, March 27, 2018

For more details:

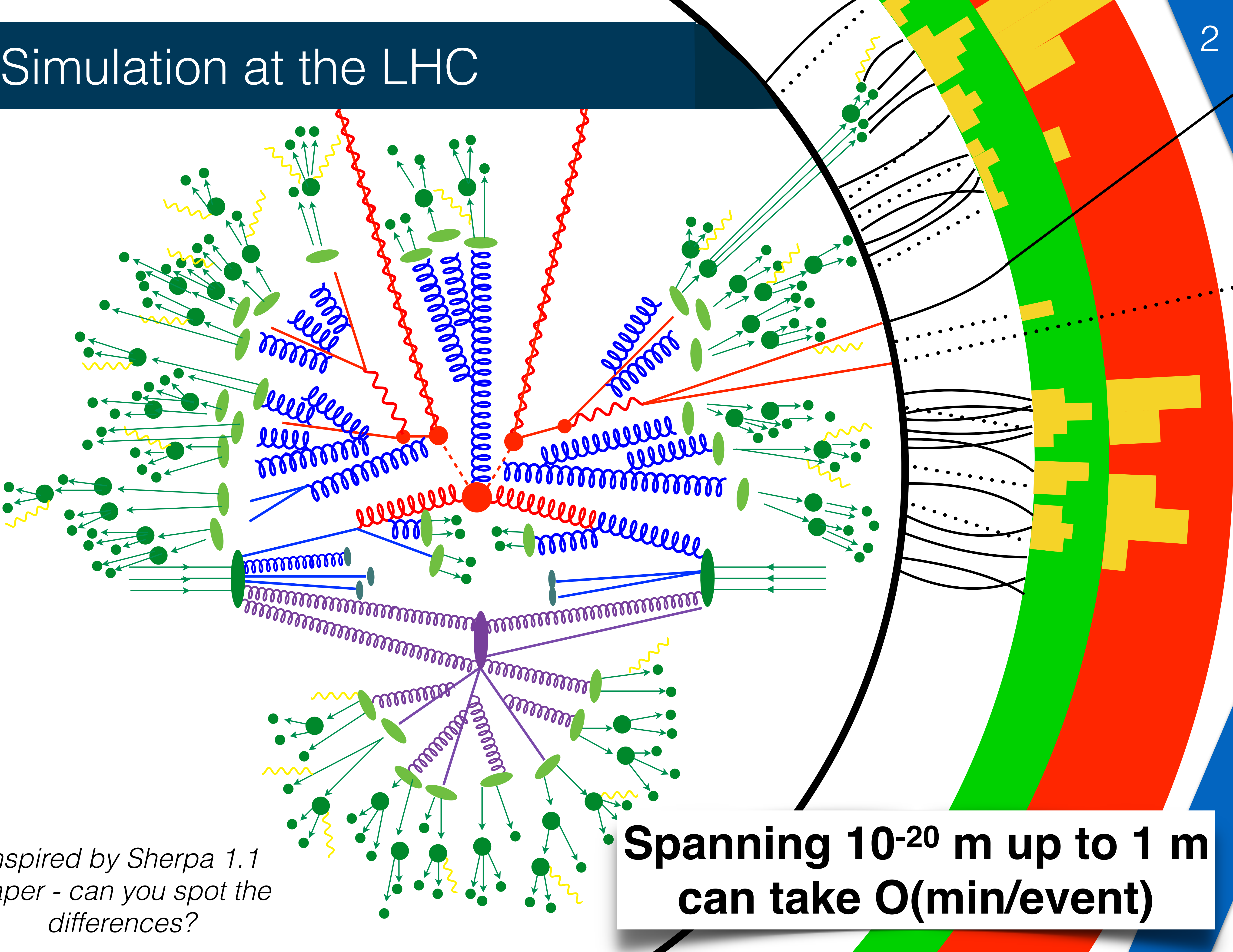
Phys. Rev. Lett. 120, 042003 (2018), 1705.02355

Phys. Rev. D 97, 014021 (2018), 1712.10321

Comput Softw. Big Sci. (2017) 1: 4, 1701.05927

See also related work by S. Vallecorsa et al. (GeantV), C. Guthrie et al. (NYU), W. Wei et al. (LCD dataset group), D. Salamani et al. (Geneva), D. Rousseau et al. (Orsay)

Simulation at the LHC



Inspired by Sherpa 1.1 paper - can you spot the differences?

Spanning 10^{-20} m up to 1 m can take O(min/event)

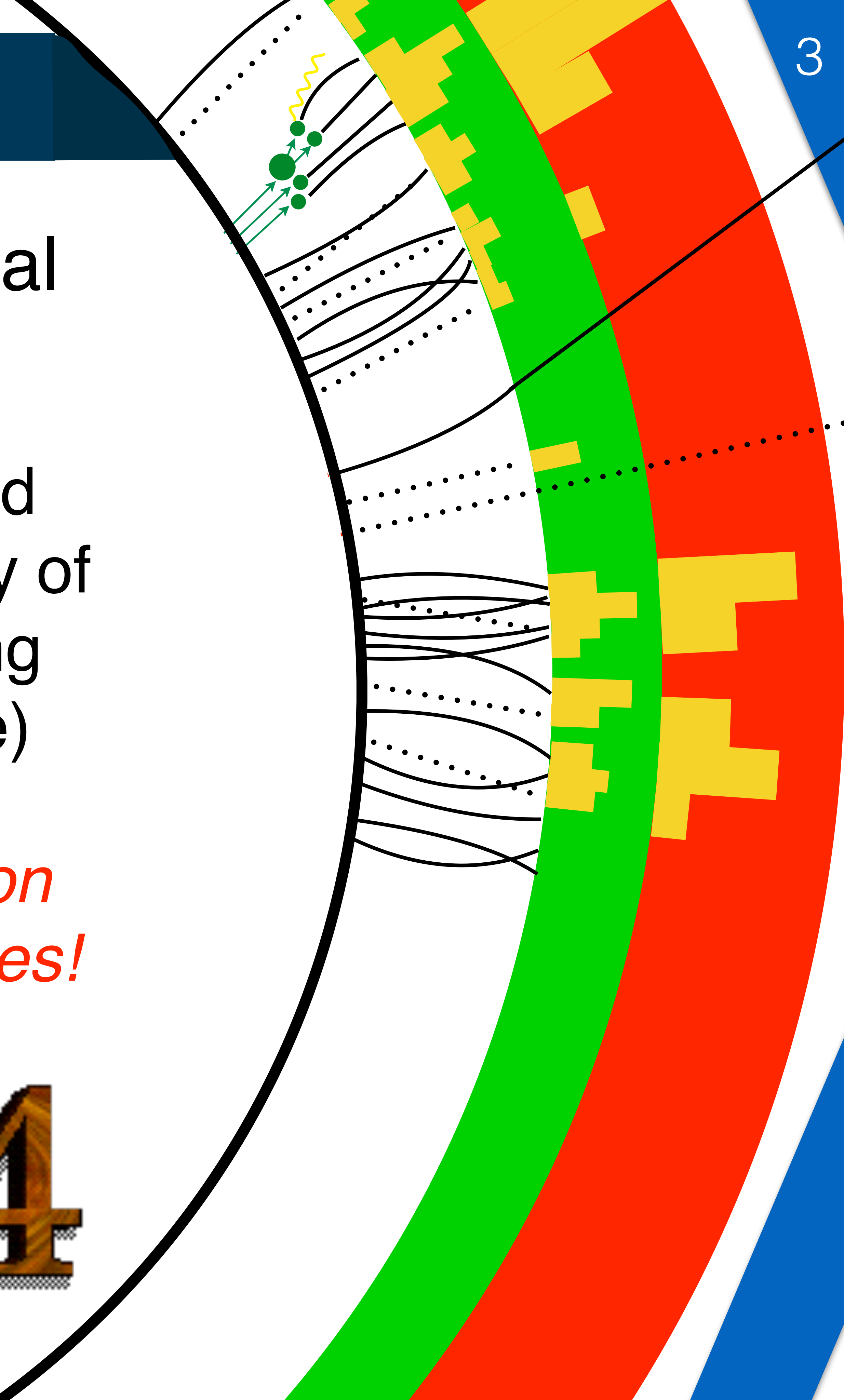
Material Interactions

State-of-the-art for material interactions is Geant 4.

Includes electromagnetic and hadronic physics with a variety of lists for increasing/decreasing accuracy (at the cost of time)

This accounts for O(1) fraction of all HEP computing resources!

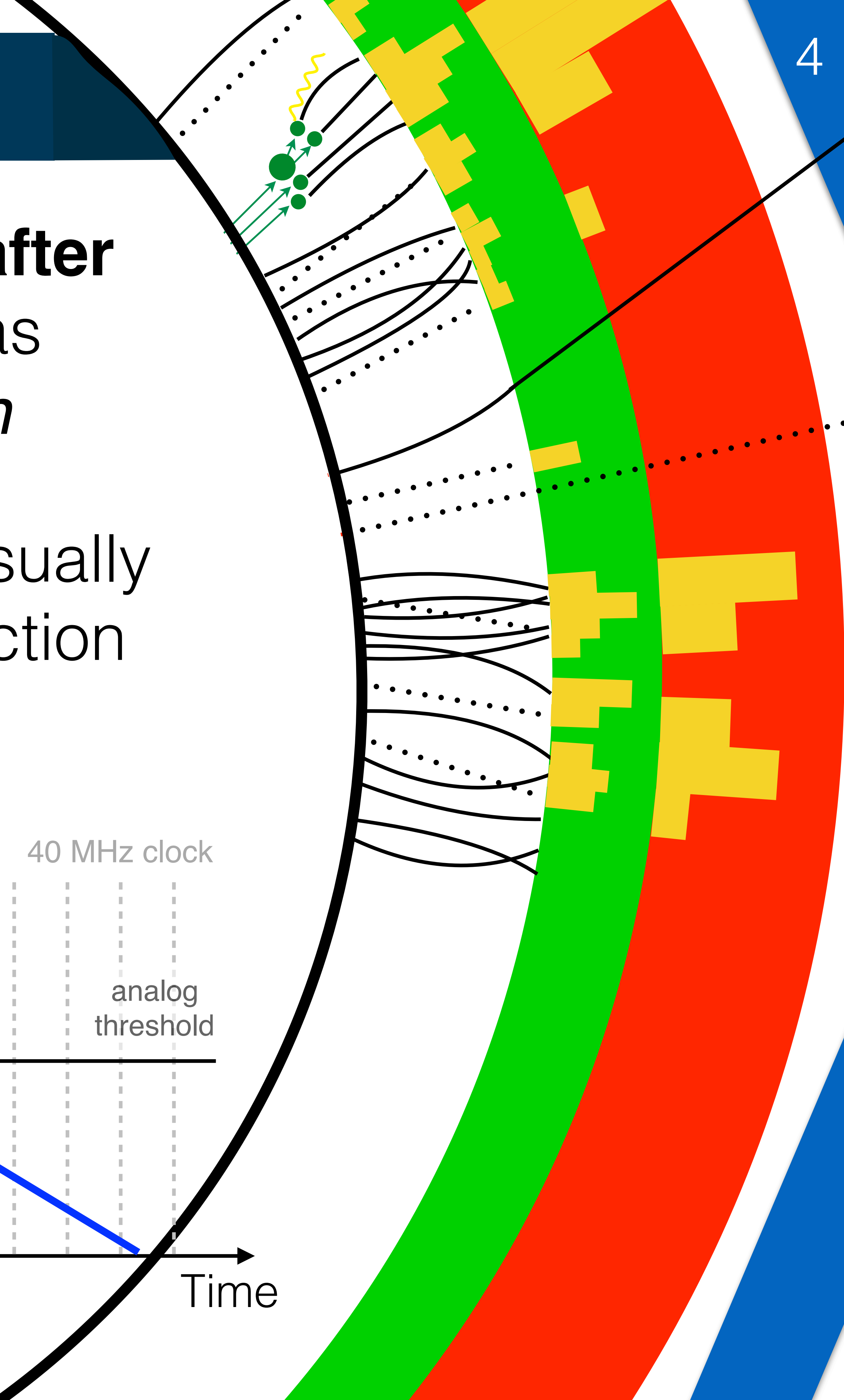
Geant 4

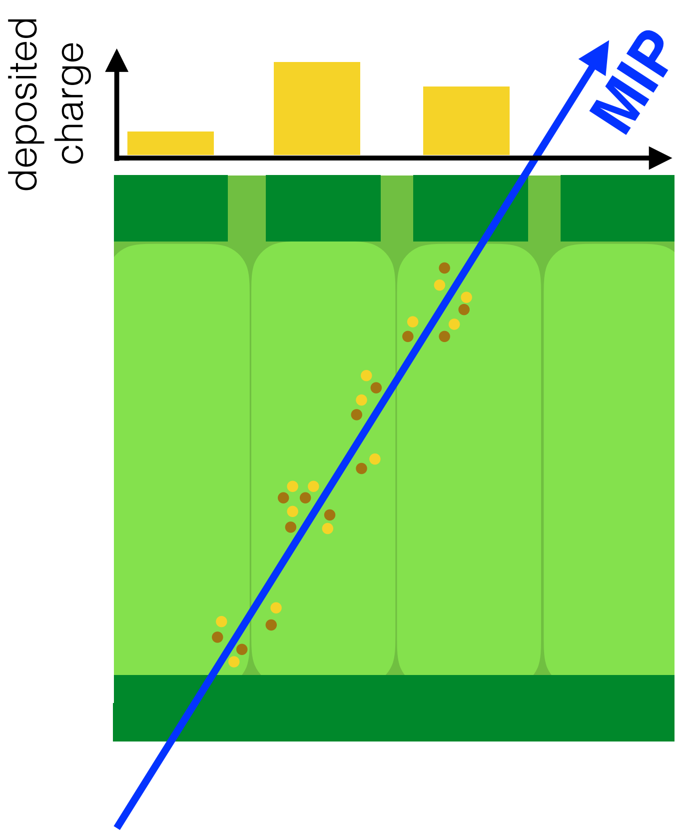


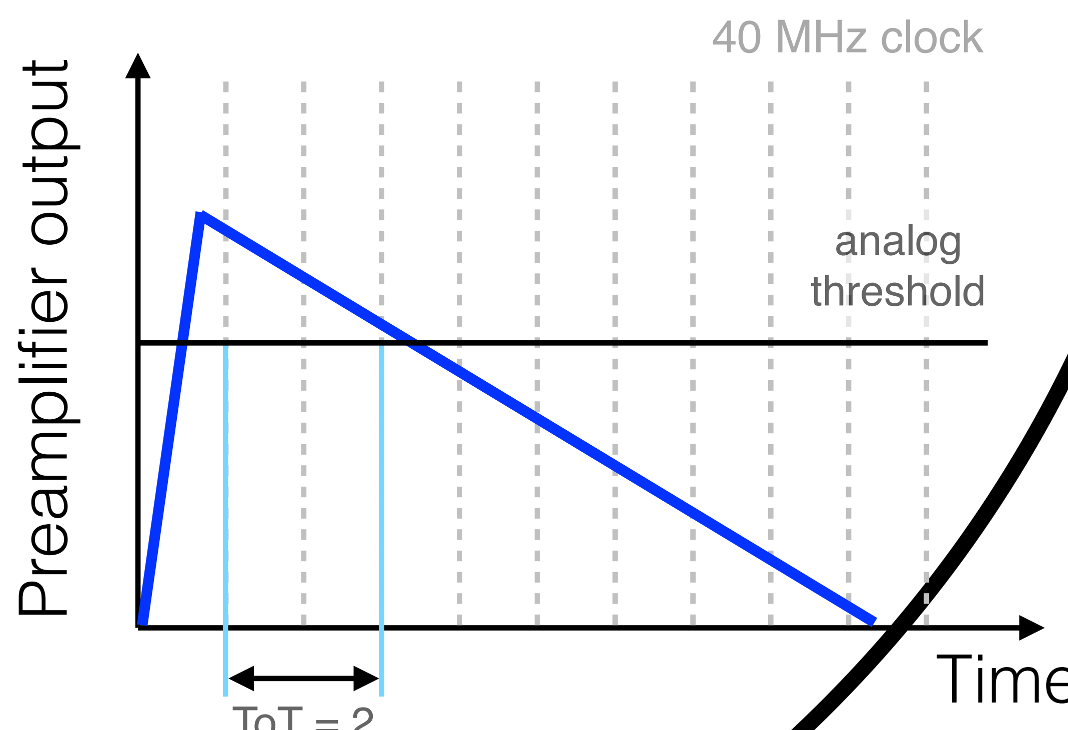
Digitization

It is important to mention that **after** Geant4, each experiment has custom code for ***digitization***

this can also be slow; but is usually faster than G4 and reconstruction



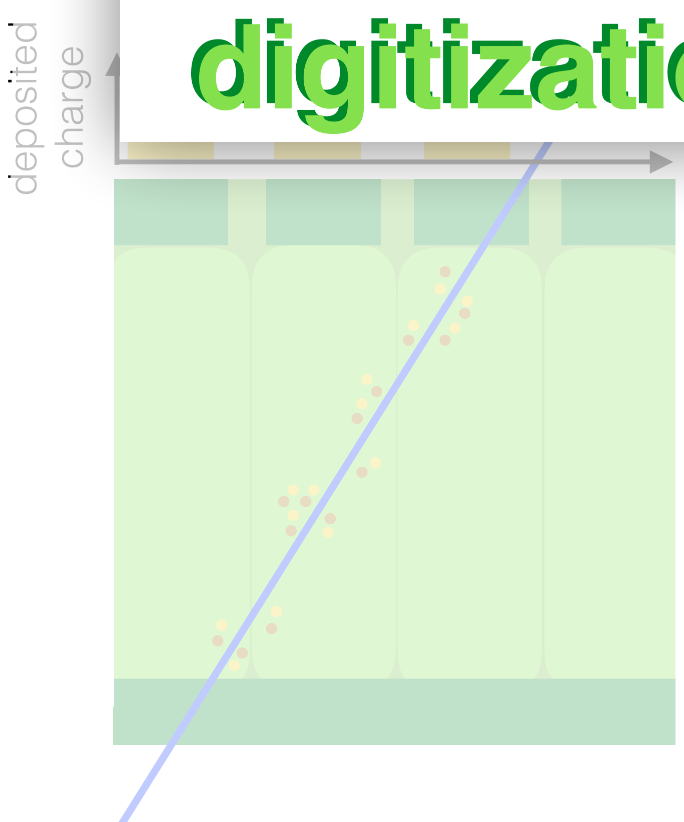


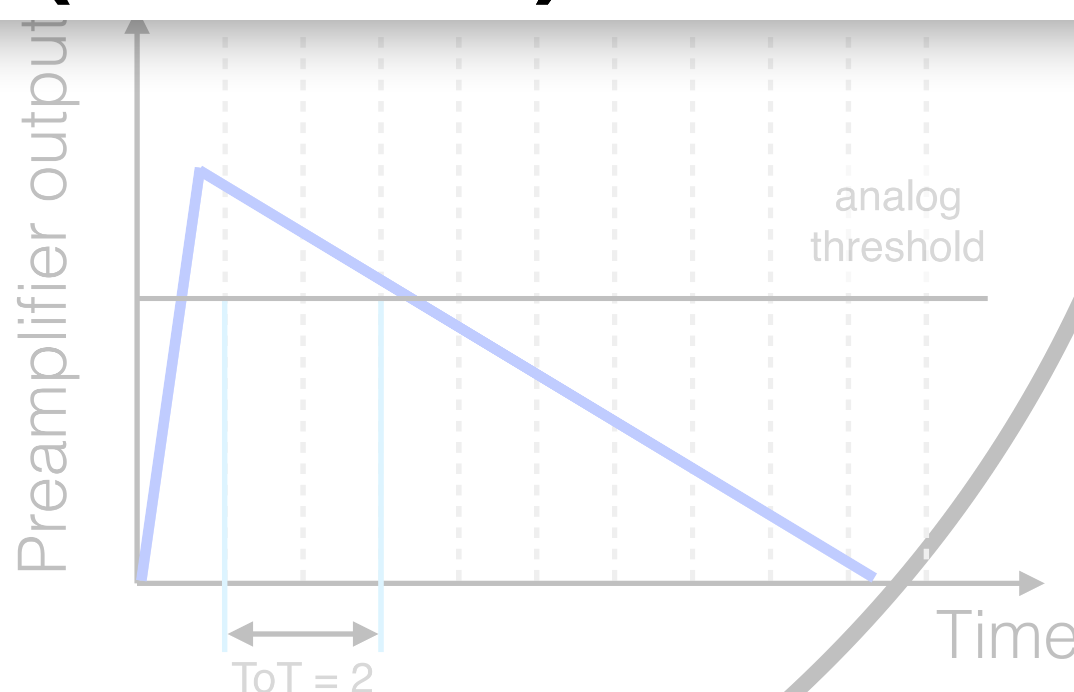


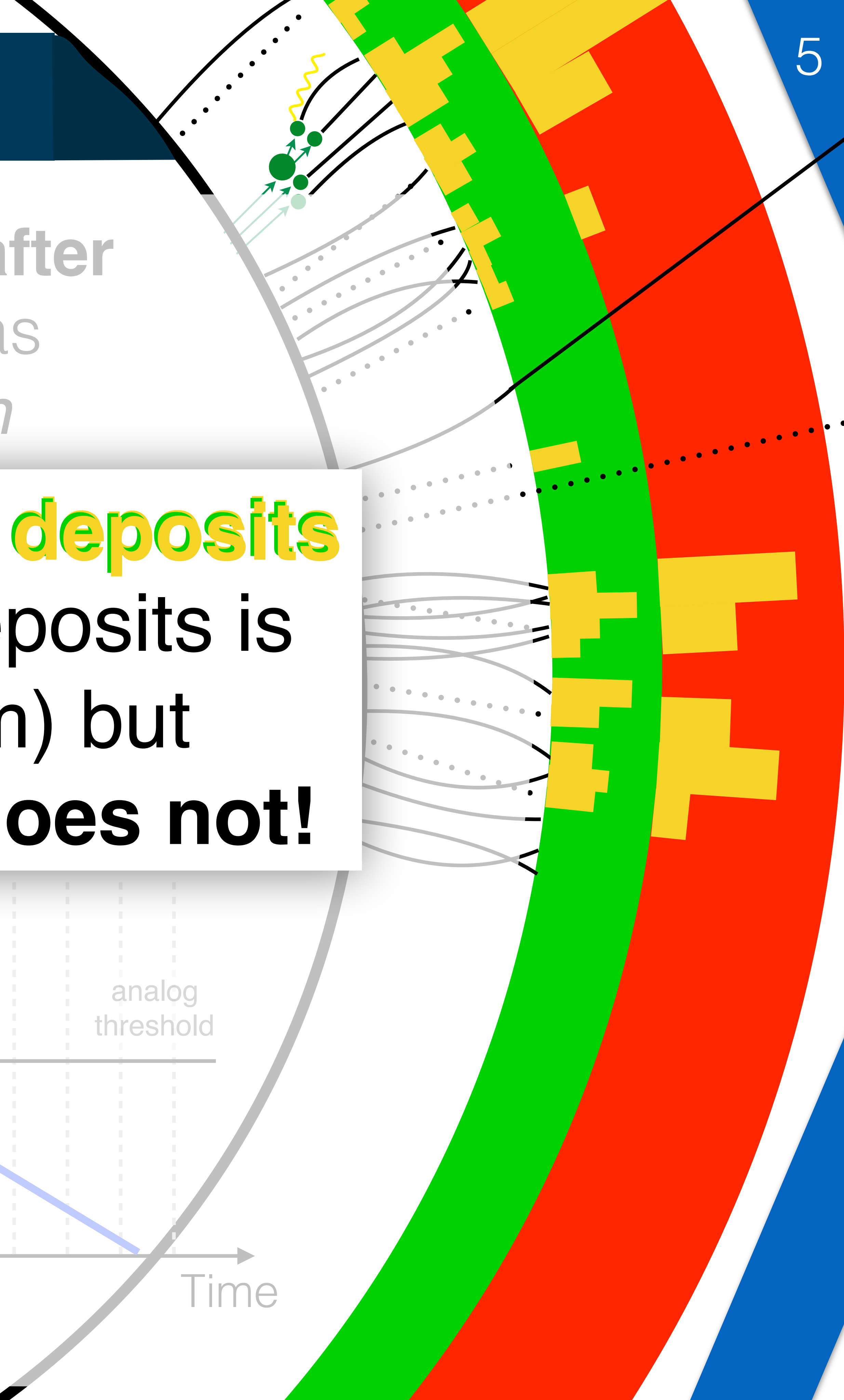
Part IV: Digitization

It is important to mention that **after** Geant4, each experiment has custom code for *digitization*

N.B. **calorimeter energy deposits factorize** (sum of the deposits is the deposit of the sum) but **digitization (w/ noise) does not!**

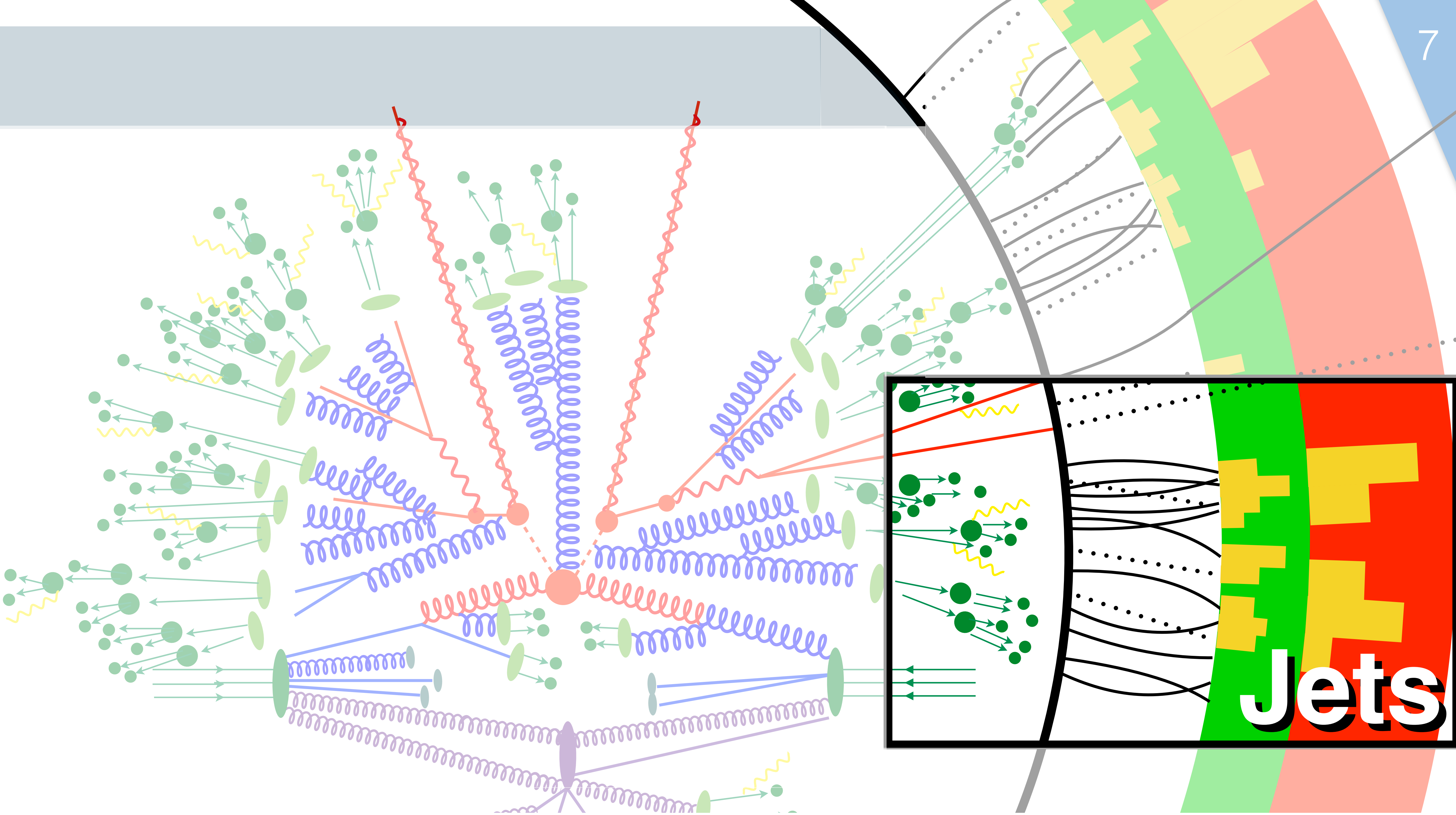






Goal: replace (or augment) simulation steps with a faster, powerful generator based on state-of-the-art machine learning techniques

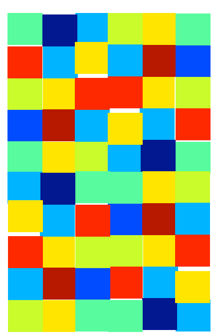
This work: attack the most important part:
Calorimeter Simulation



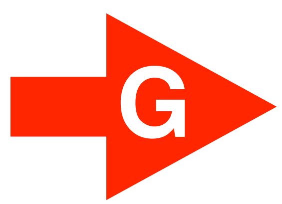
First step: instead of studying the detailed structure of calorimeter showers, we consider **Jet images**

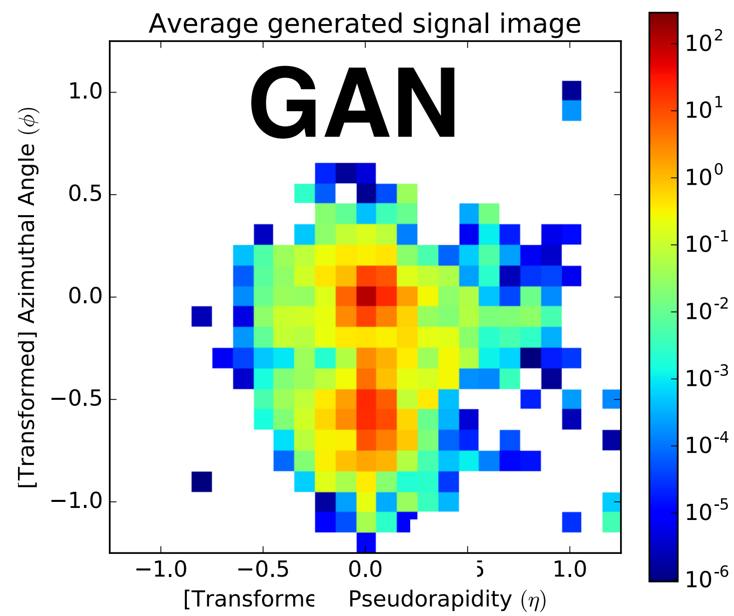
And now: Modern Deep NN's for Generation

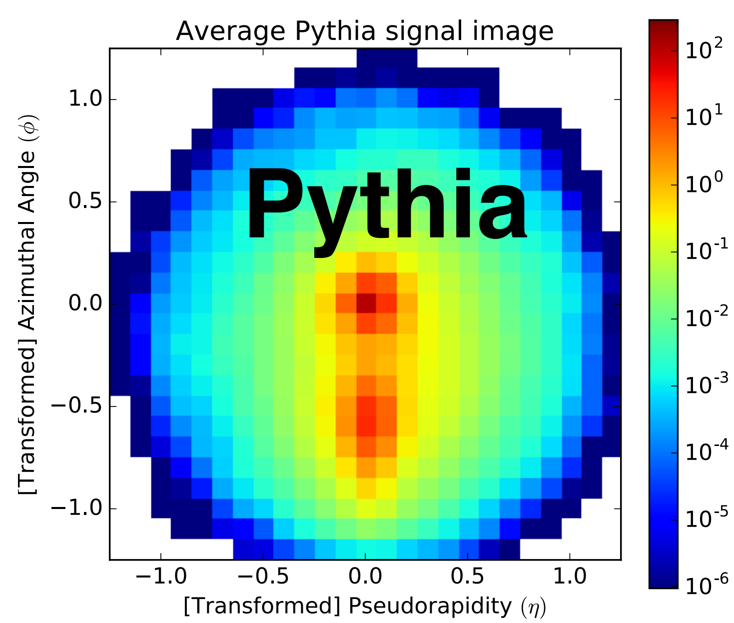
Generative Adversarial Networks (GAN):
A two-network game where one maps noise to images and one classifies images as fake or real.



noise







{real,fake}

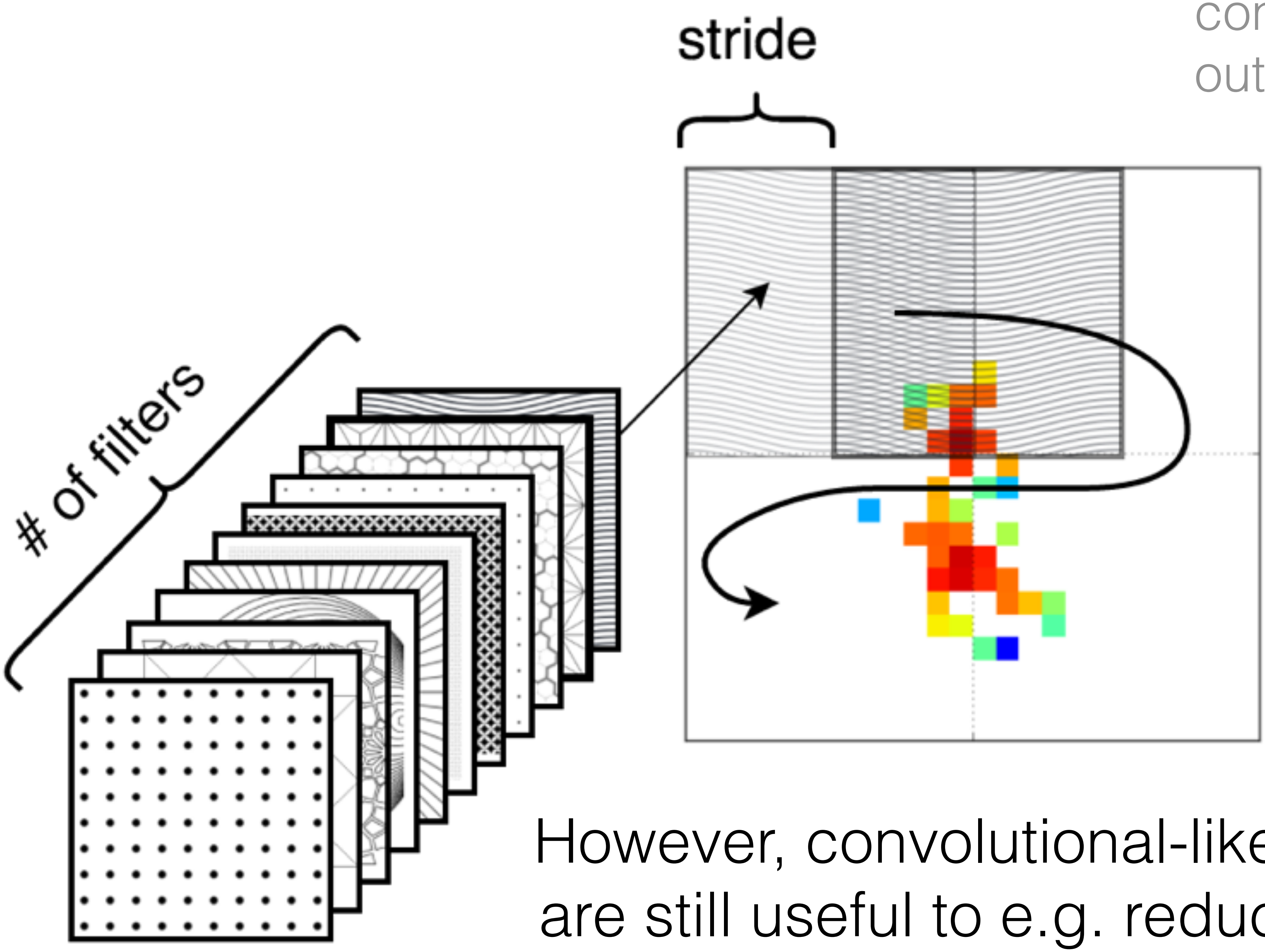
When **D** is maximally confused, **G** will be a good generator

Physics-based simulator

Locally Connected Layers

Due to the structure of the problem, we do not have translation invariance.

Classification studies found fully connected networks outperformed CNNs

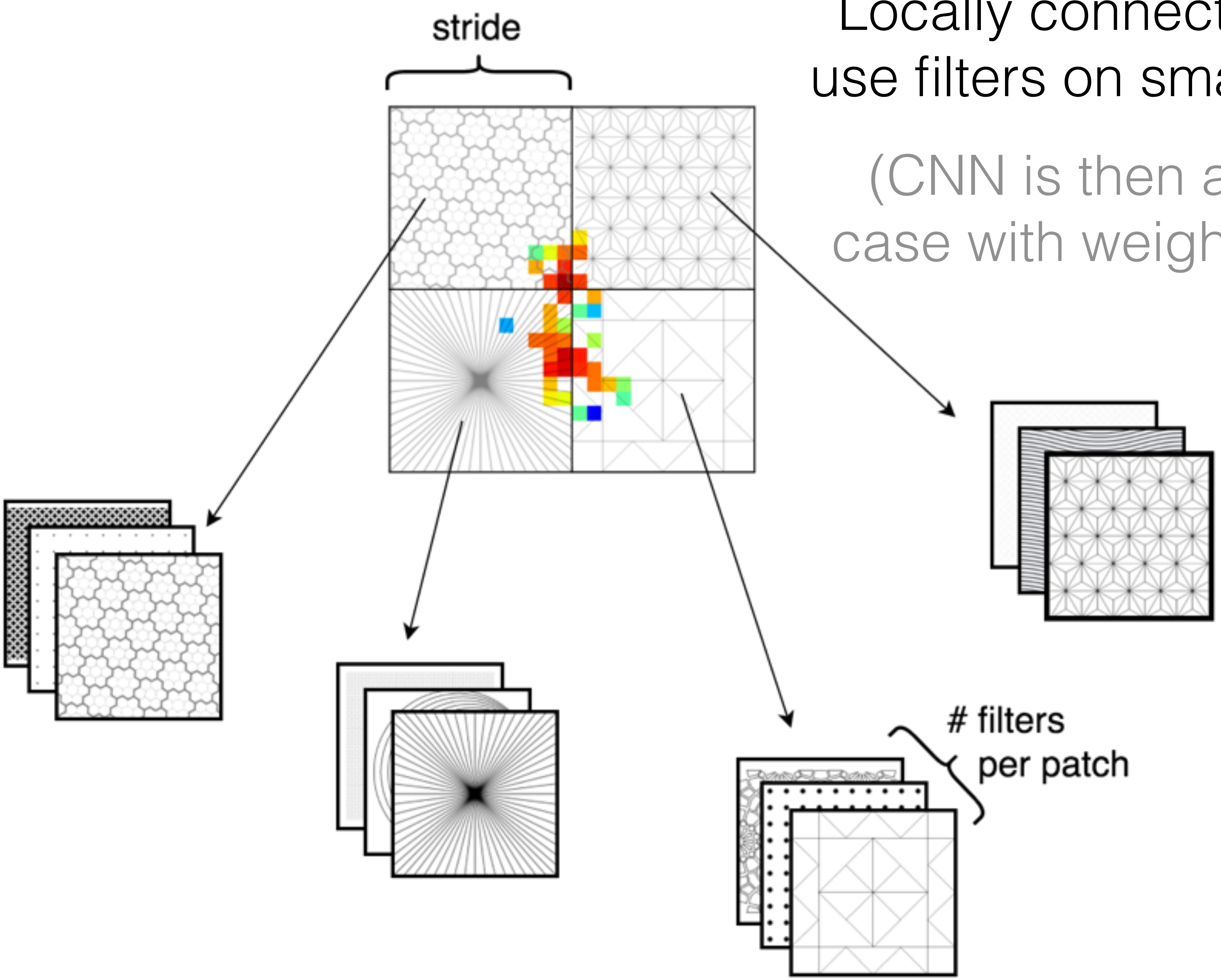


However, convolutional-like architectures are still useful to e.g. reduce parameters

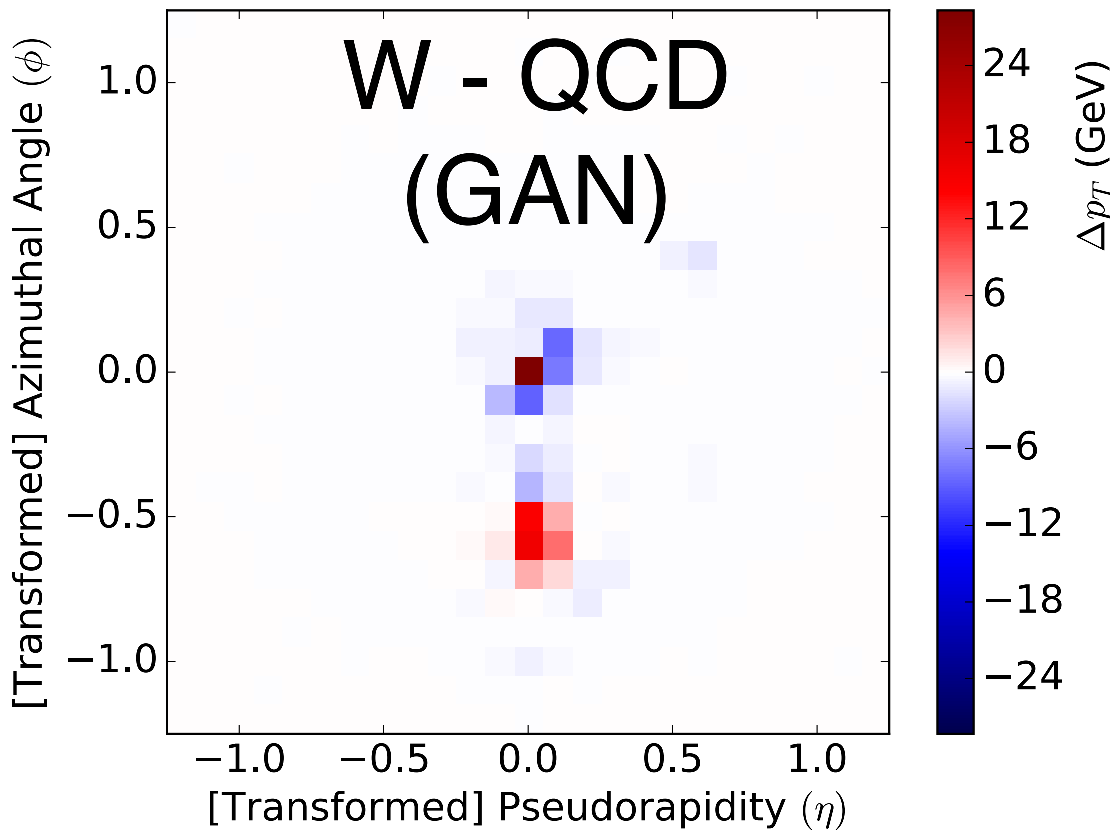
Locally Connected Layers

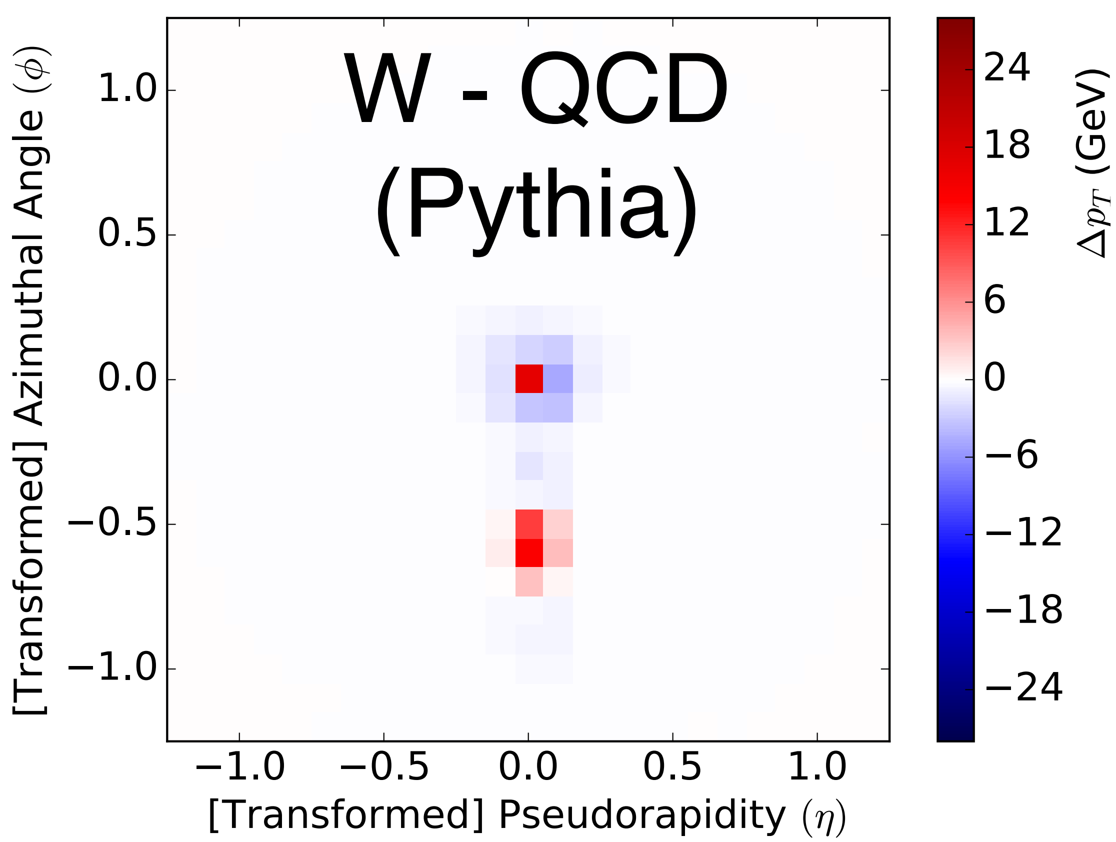
Locally connected layers use filters on small patches

(CNN is then a special case with weight sharing)

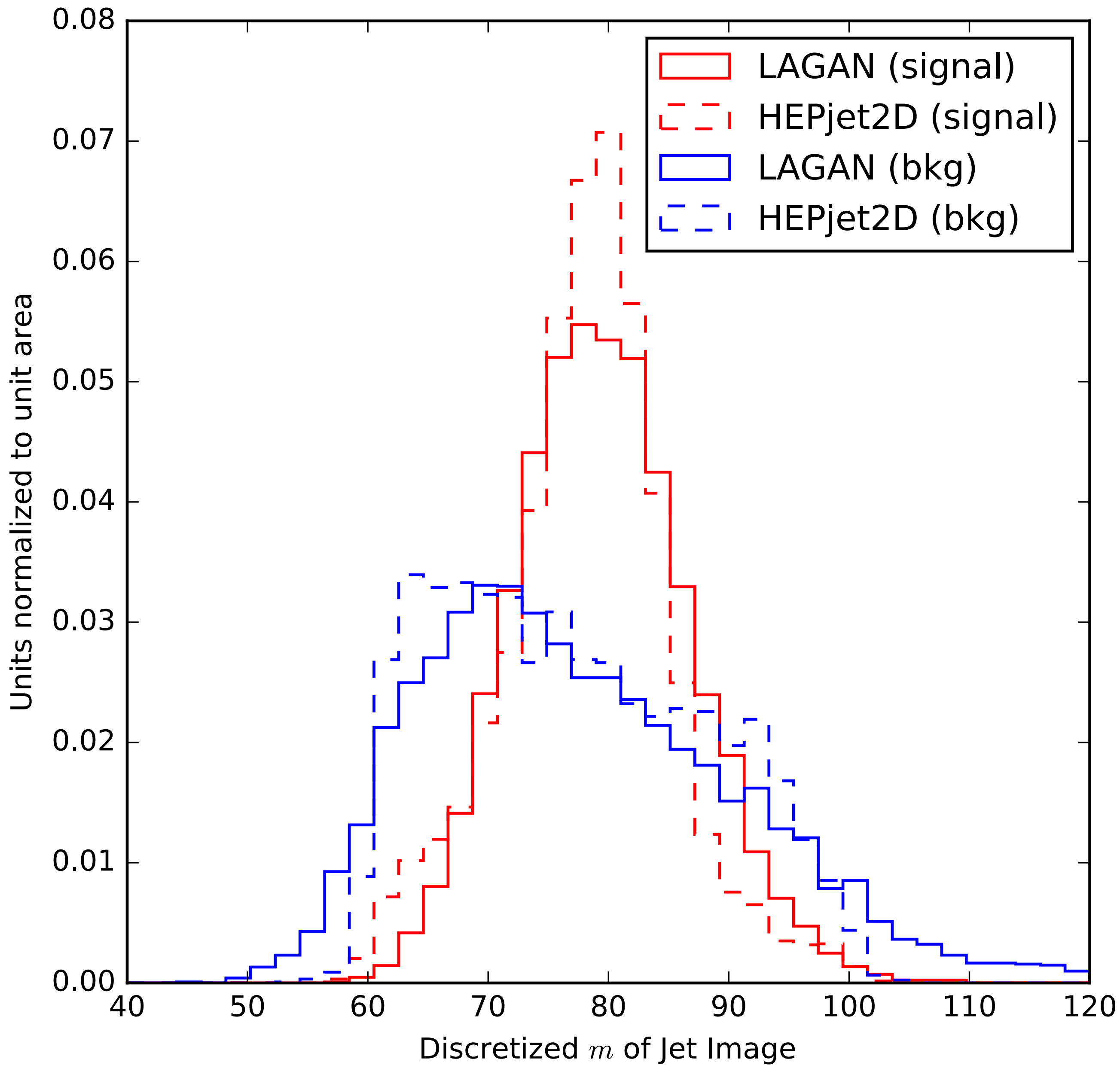


Locally Aware GAN (LAGAN)





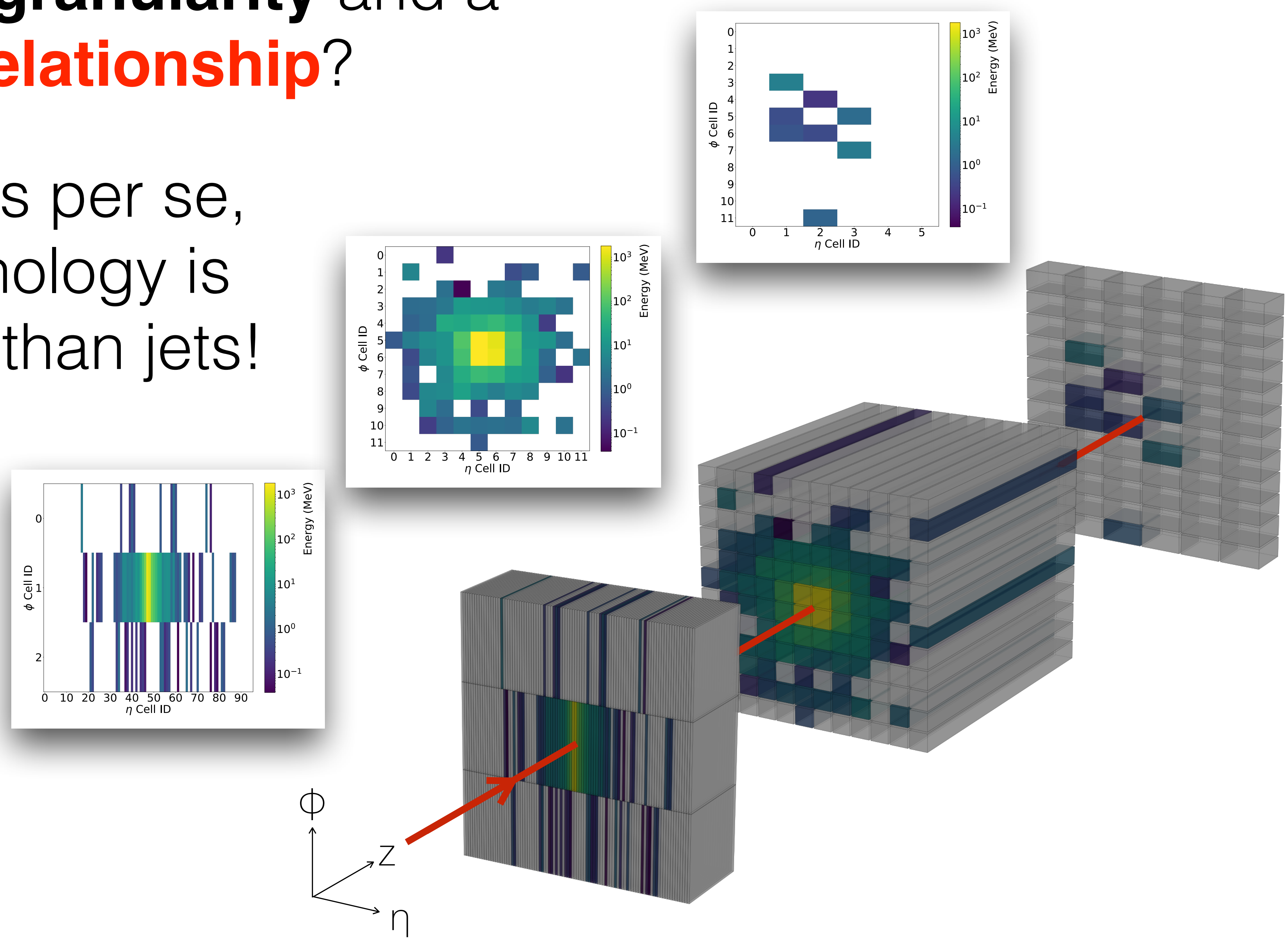
Unlike `natural images', we have physically meaningful 1D manifolds (here, jet mass)



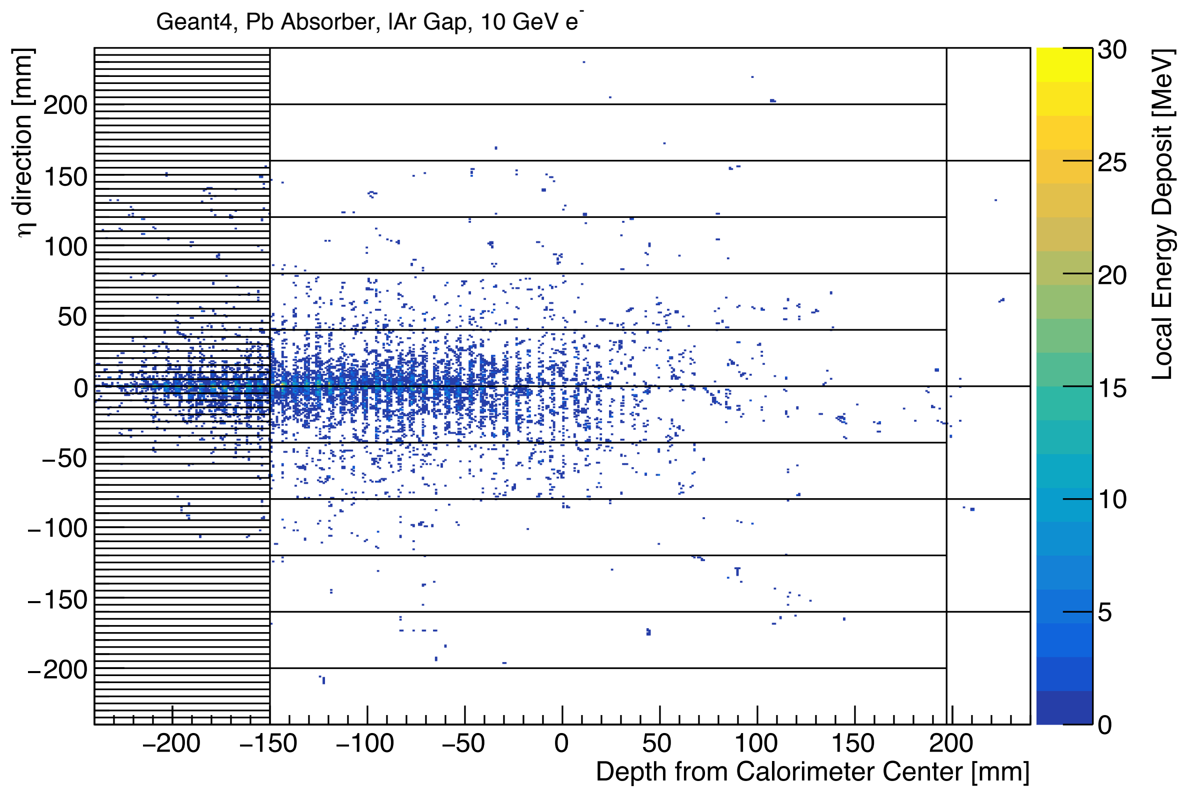
+ More Layers for Generation

What about **multiple layers** with **non-uniform granularity** and a **causal relationship**?

Not jet images per se, but the technology is more general than jets!



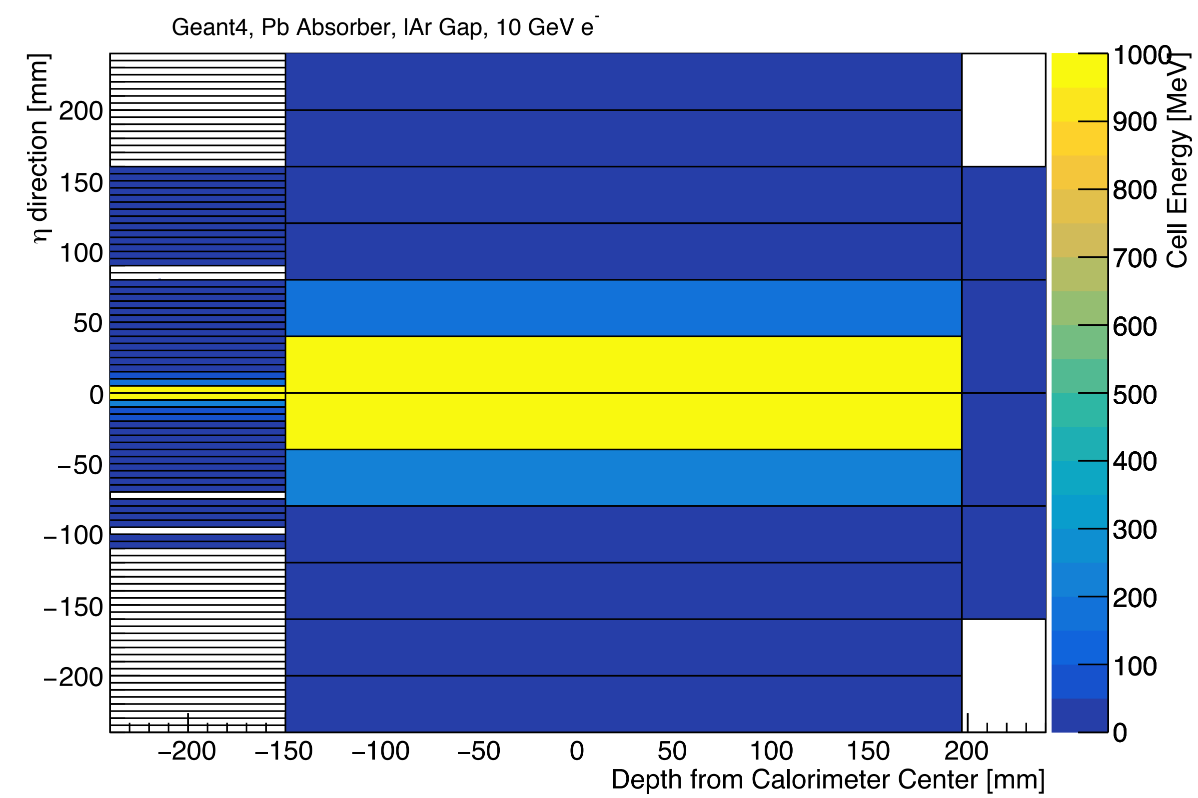
Calorimeter Simulation



We take as our model a 3-layer LAr calorimeter, inspired by the ATLAS barrel EM calorimeter

A single event may have $O(10^3)$ of particles showering in the calorimeter - too cumbersome to do all at once (now)

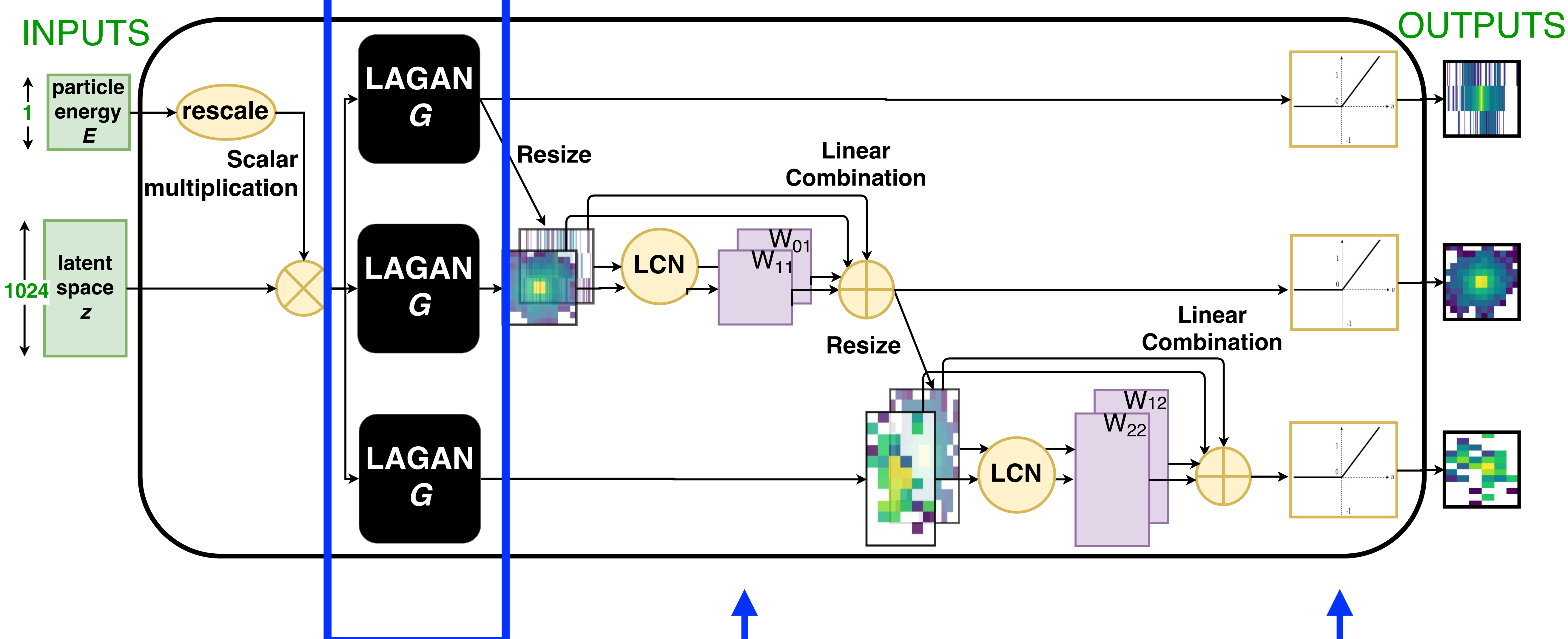
We exploit factorization of energy depositions



Generator Network for CaloGAN

One 'jet image' per calo layer

One network per particle type; input particle energy

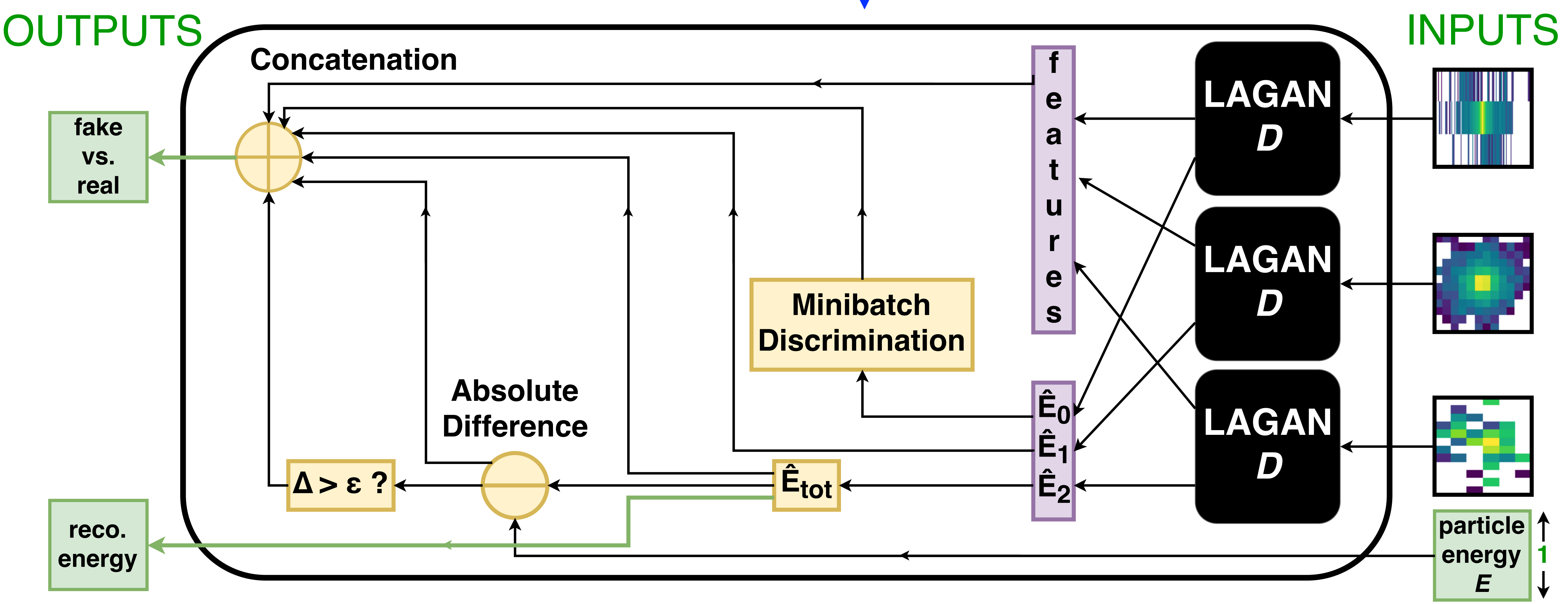


use layer i as input to layer i+1

ReLU to encourage sparsity

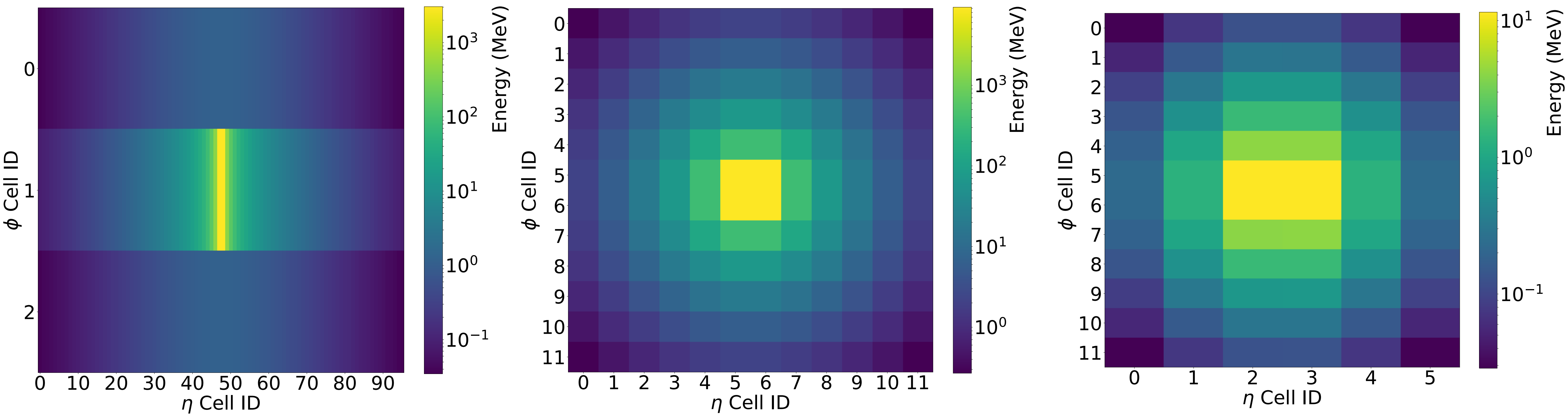
Discriminator Network for CaloGAN

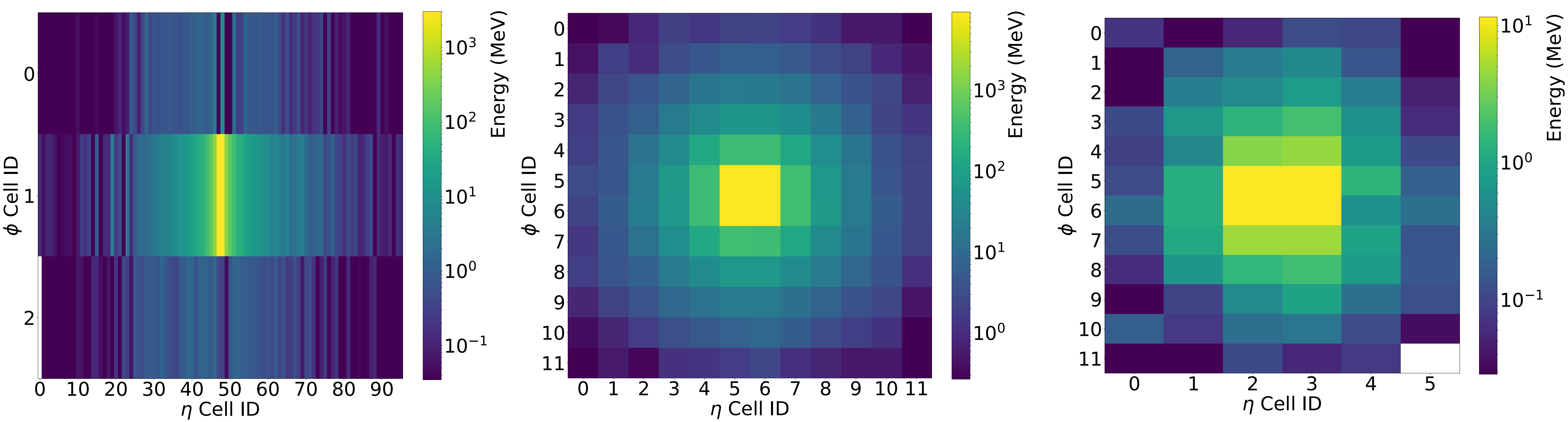
help avoid
'mode collapse'



Average Images

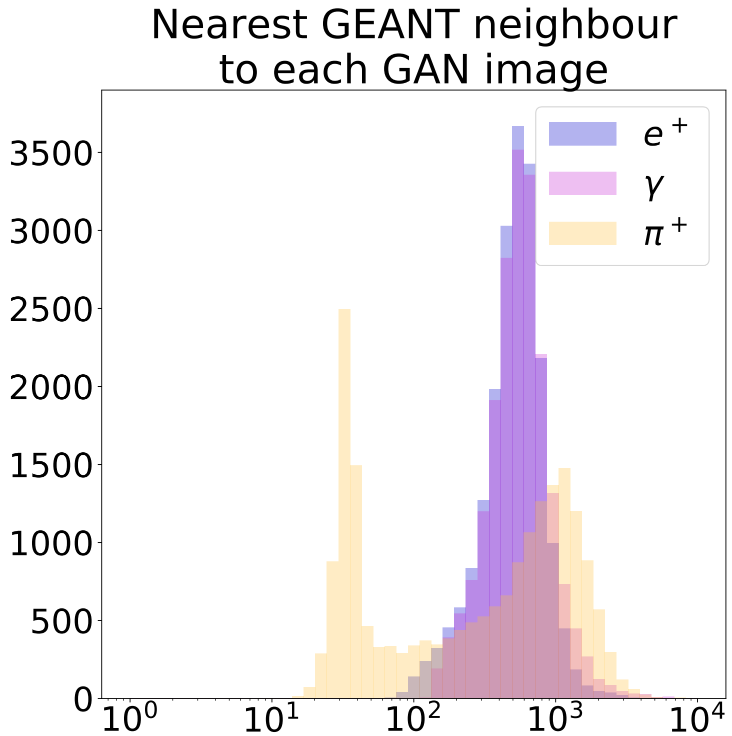
Geant4



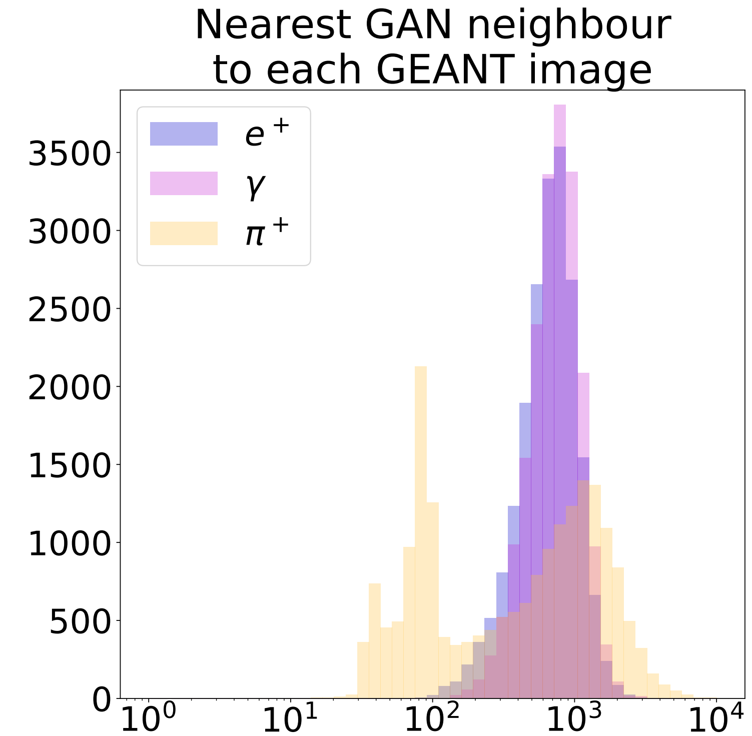


CaloGAN

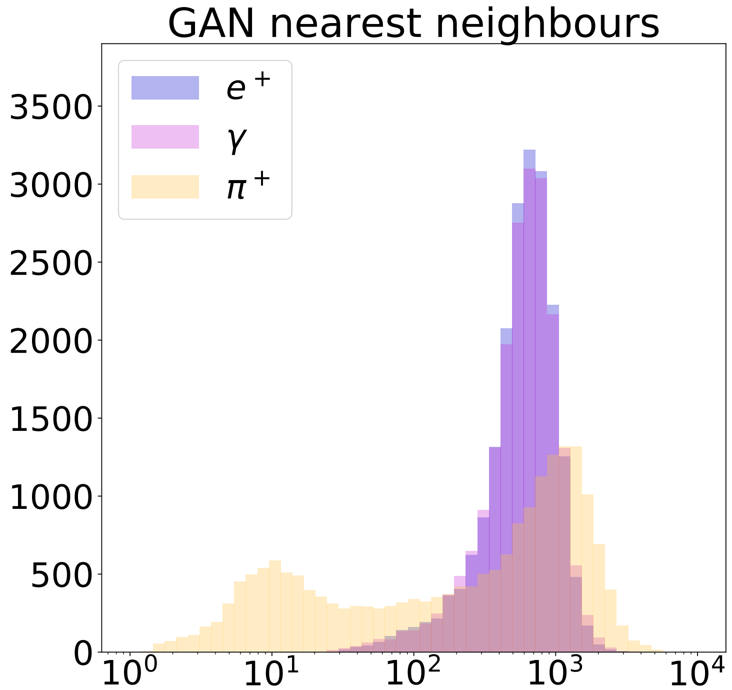
“Overtraining”



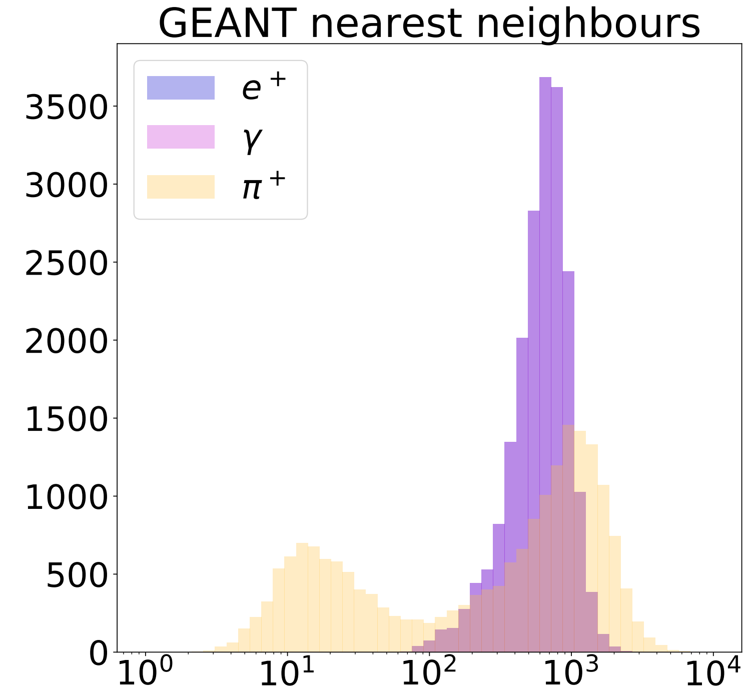
not
memorizing



A key challenge in training GANs is the diversity of generated images. This does not seem to be a problem for CaloGAN.

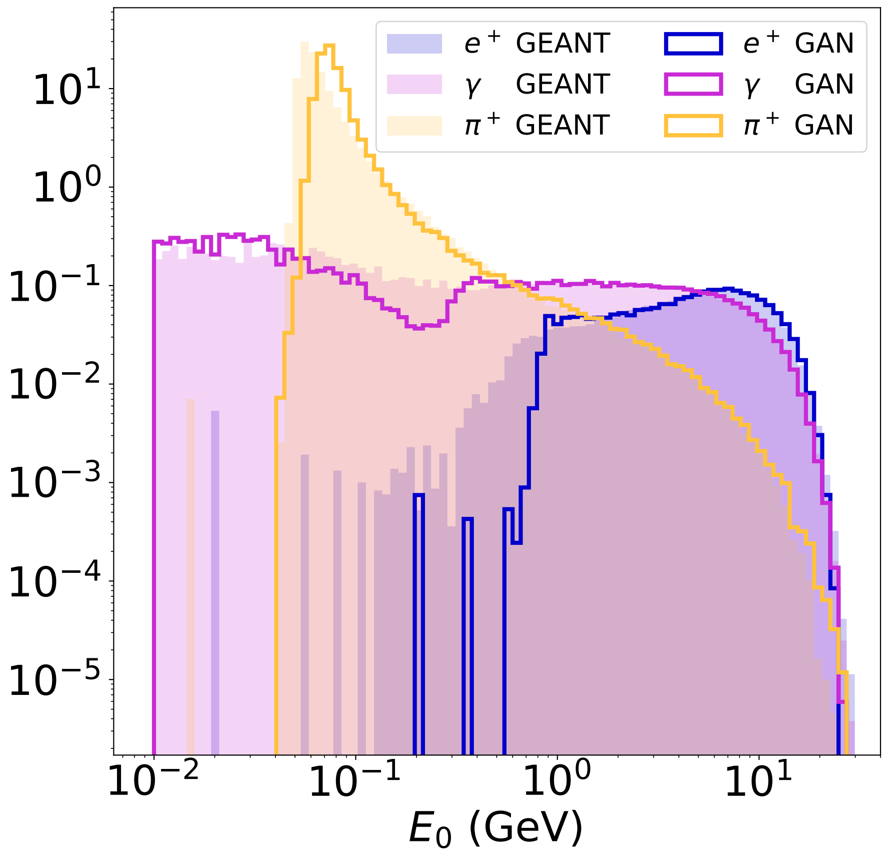


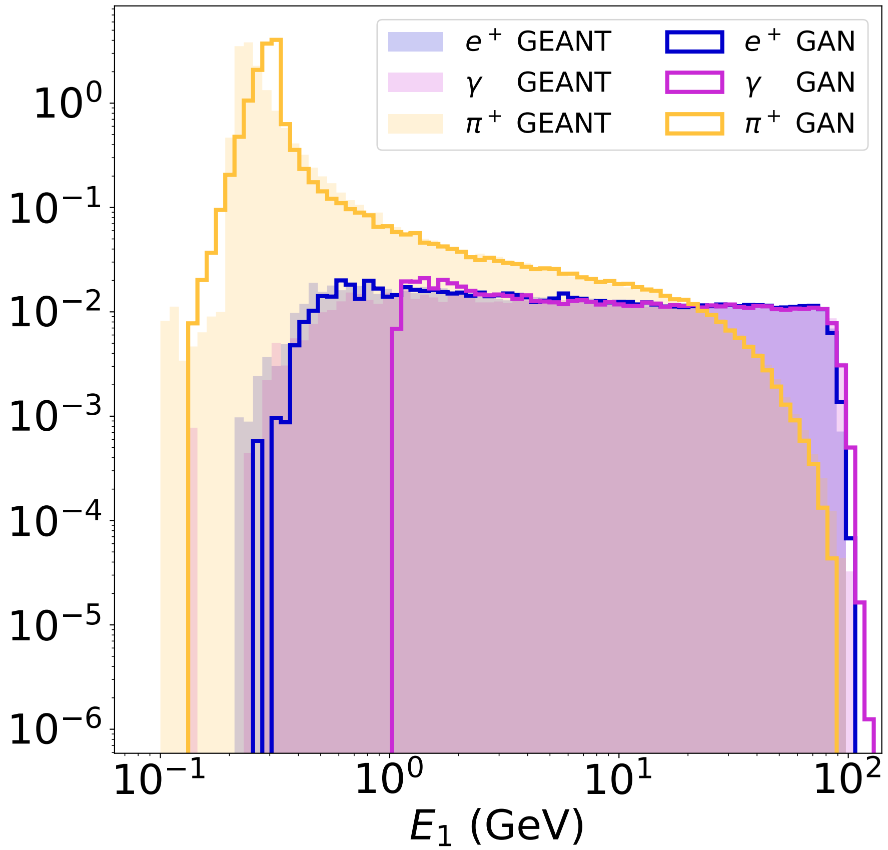
no mode
collapse

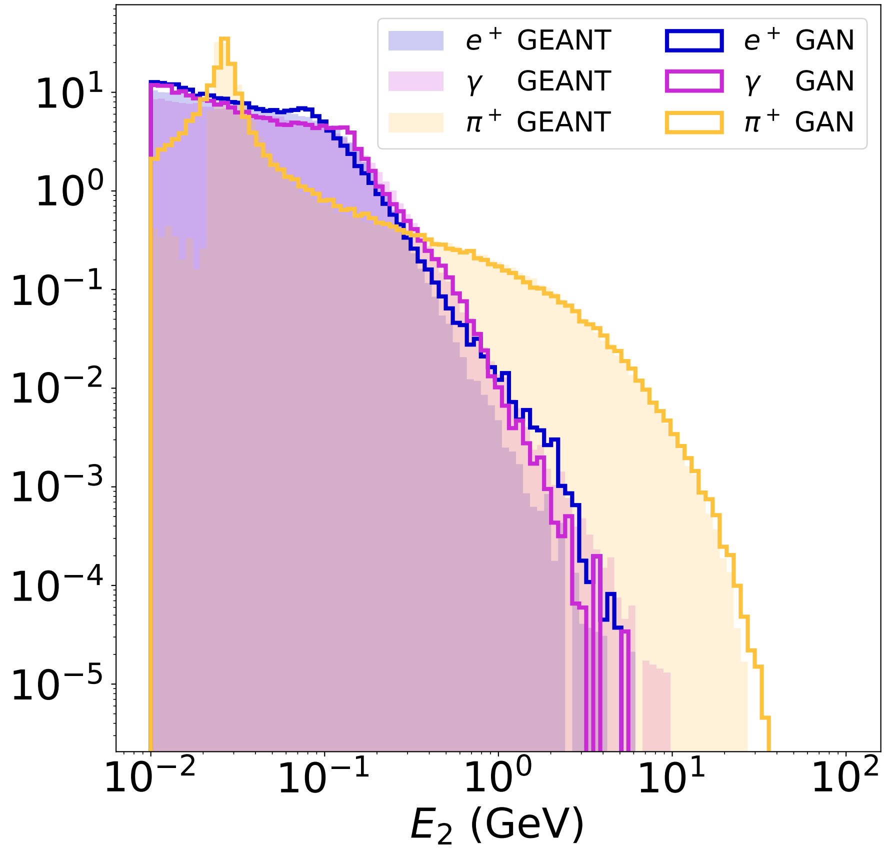


Energy per layer

Pions deposit much less energy in the first layers; leave the calorimeter with significant energy

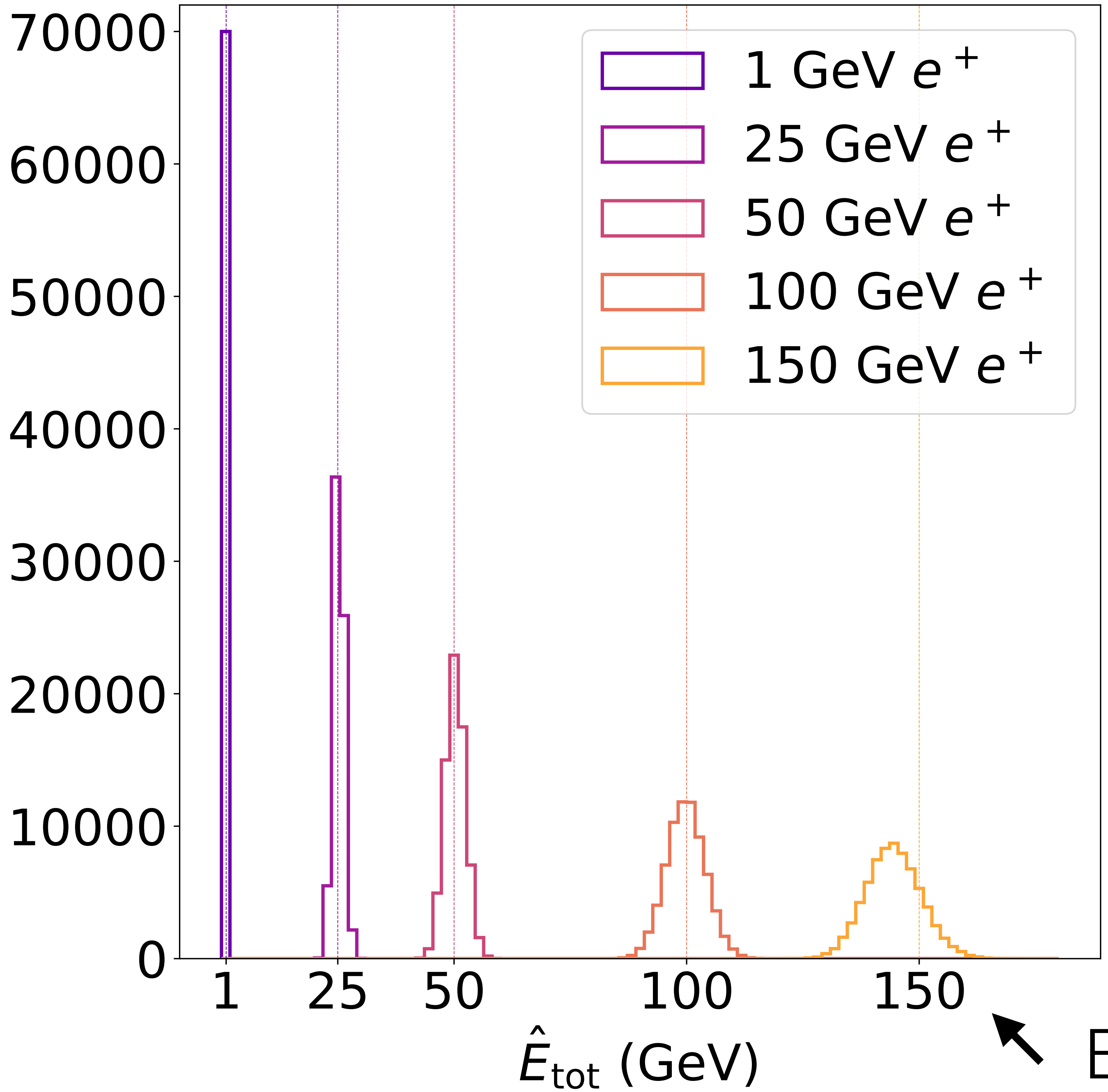






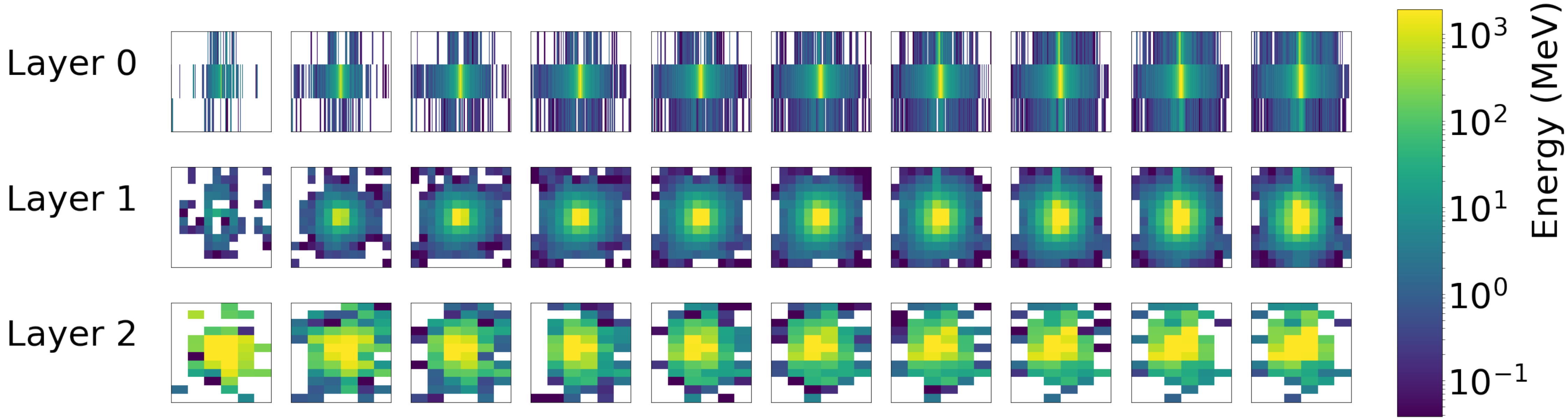
N.B. can always add these (and others) explicitly to the training

Shower Energy

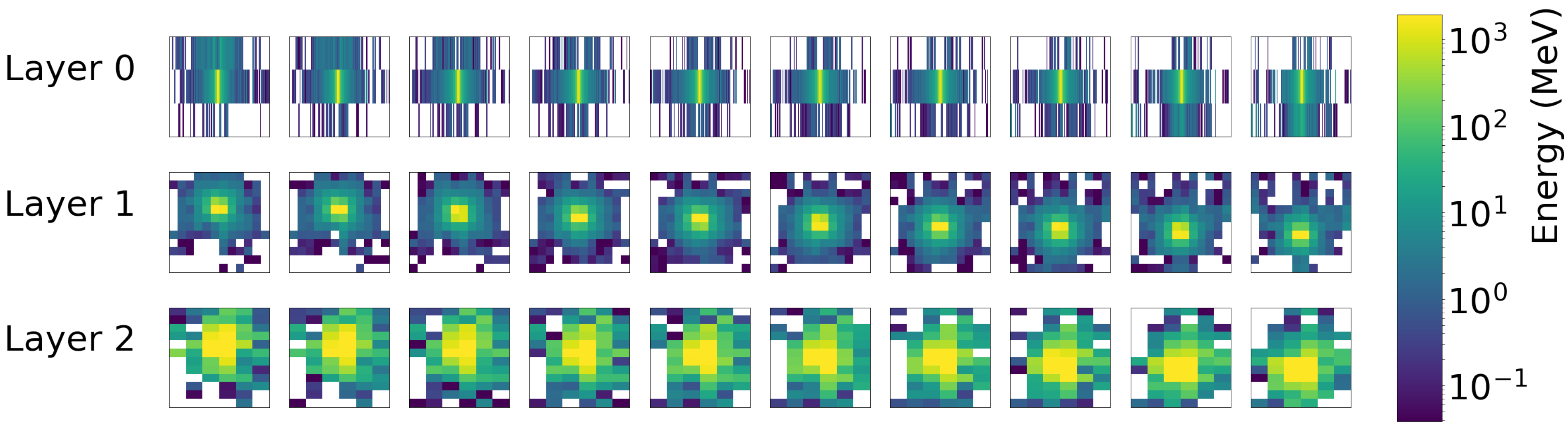


Beyond our training sample!

Fix noise, scan latent variable corresponding to energy



Fix noise, scan latent variable corresponding to x-position



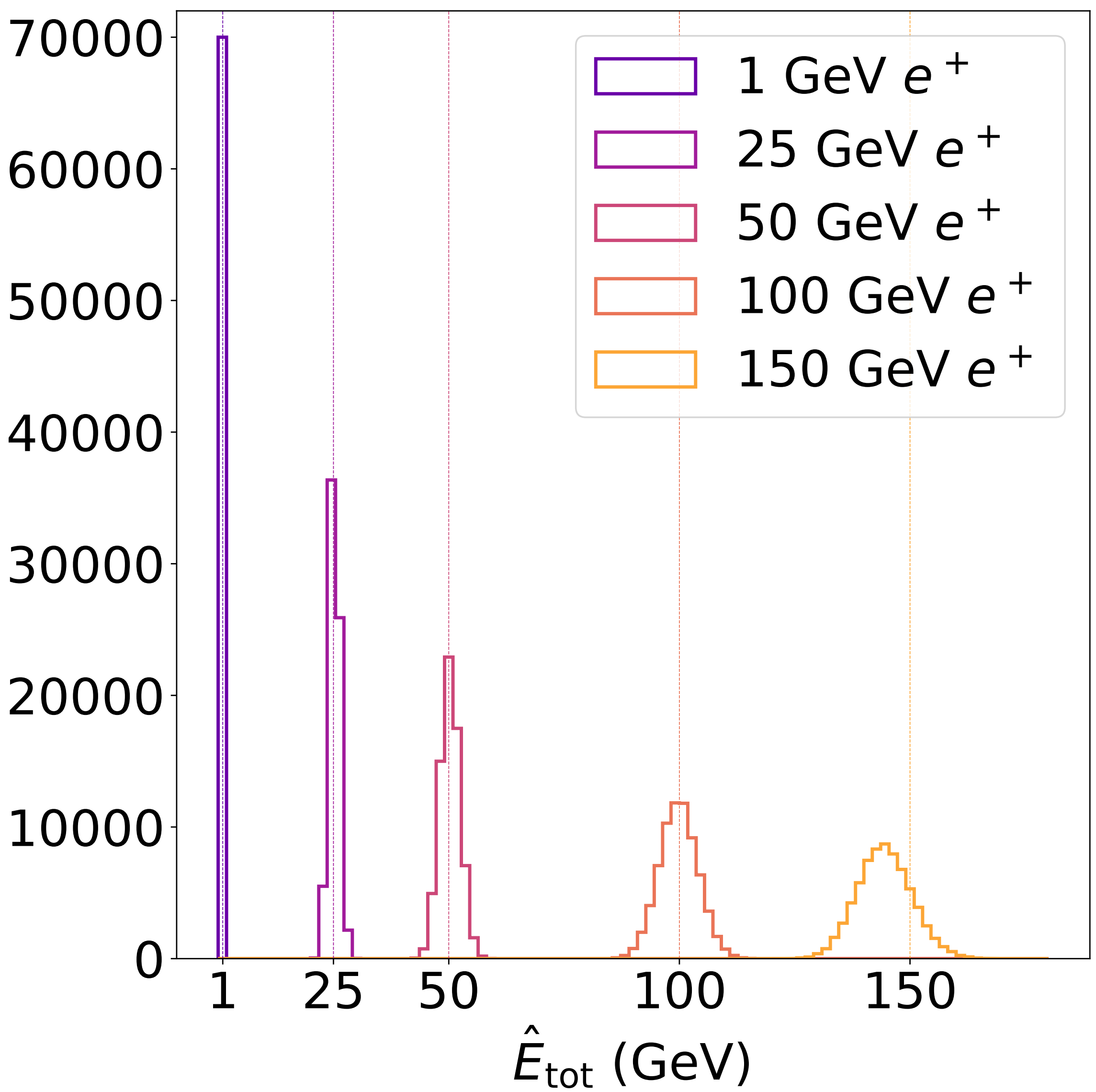
Timing

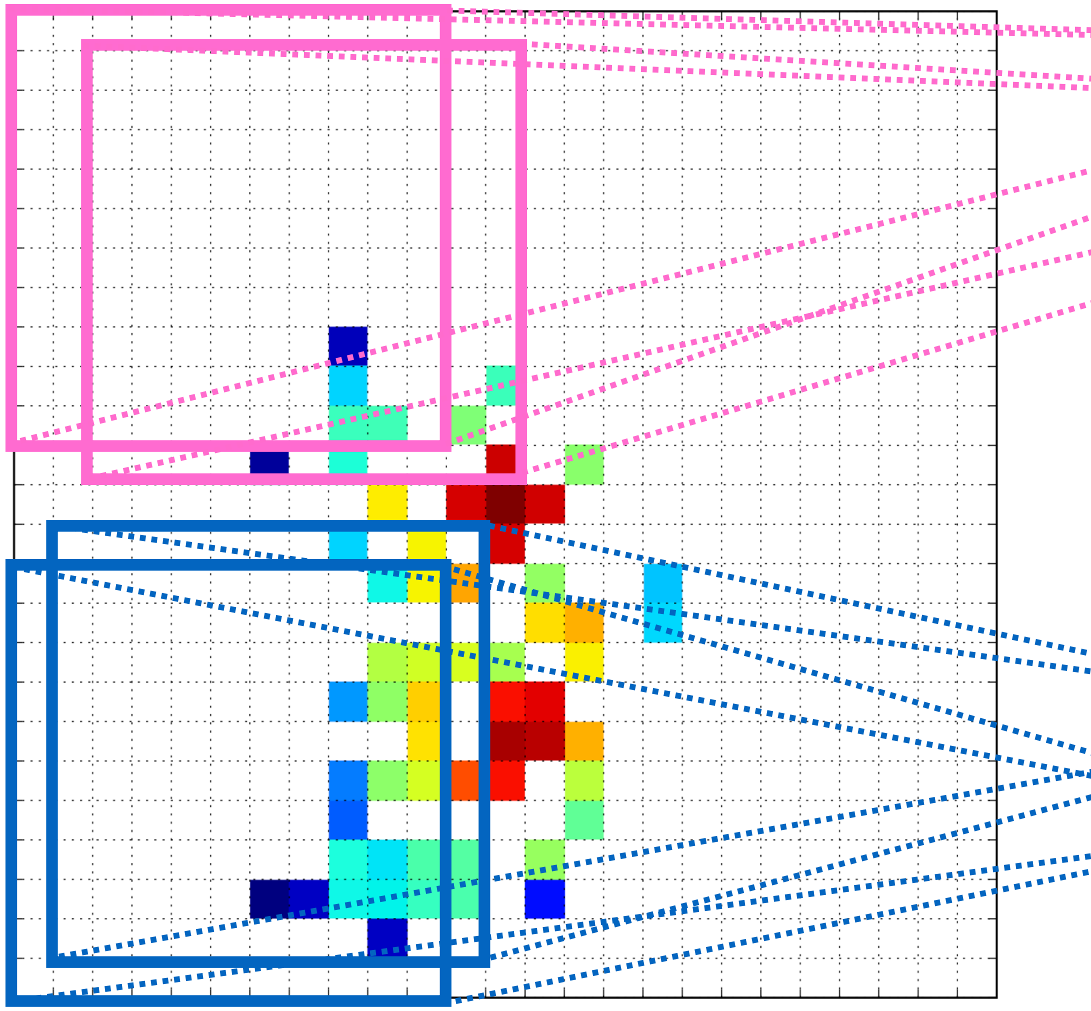
Generation Method	Hardware	Batch Size	milliseconds/shower
GEANT4	CPU	N/A	1772
CALOGAN	CPU *Intel Xeon E5-2670*	1	13.1
		10	5.11
		128	2.19
		1024	2.03
	GPU *NVIDIA K80*	1	14.5
		4	3.68
		128	0.021
		512	0.014
		1024	0.012

(clearly these numbers will change as both technologies improve - this is simply meant to be qualitative and motivating!)

Conclusions and Outlook

Neural-network generation is a systematically improvable path toward a high(er) fidelity simulator.

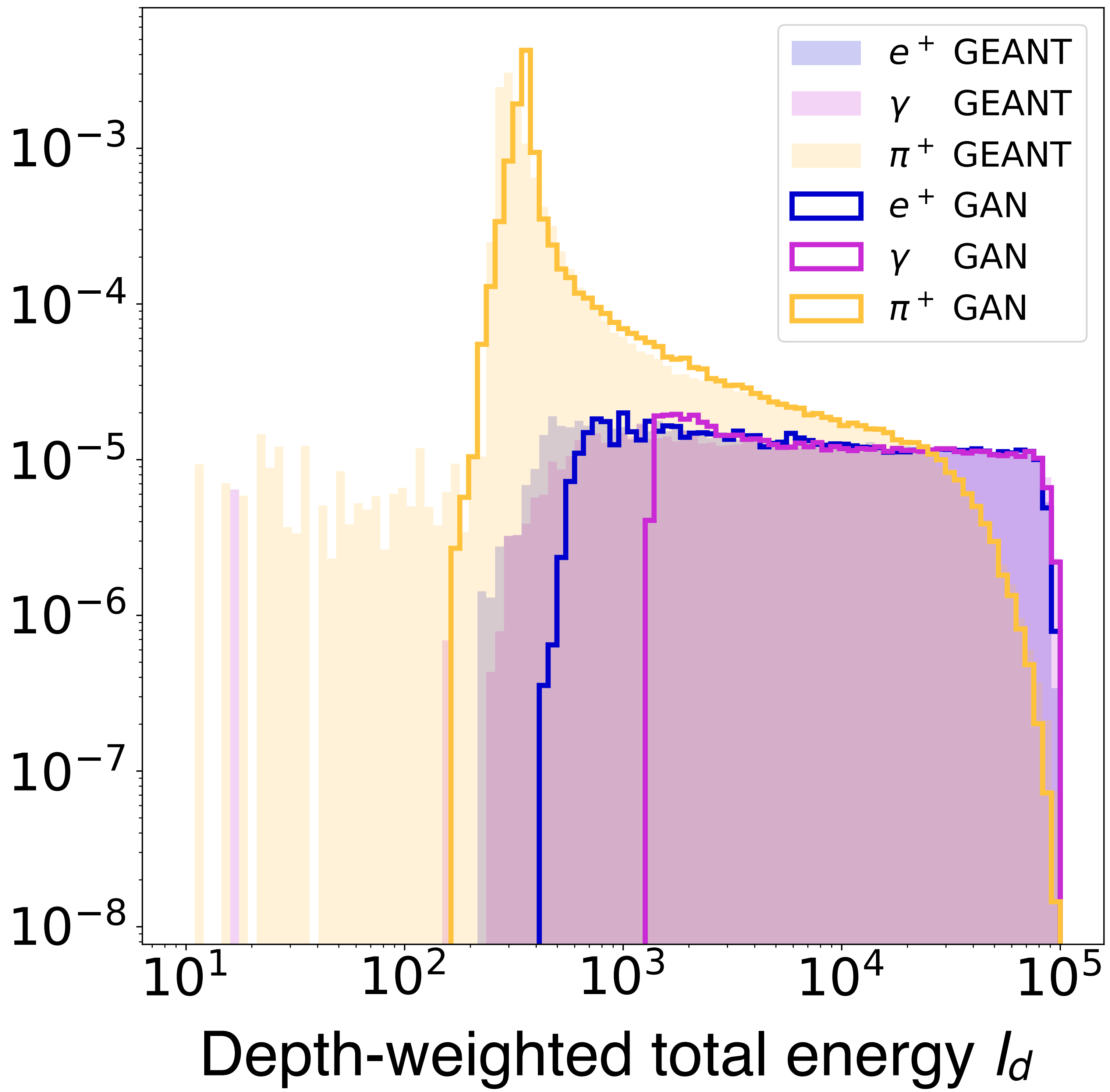




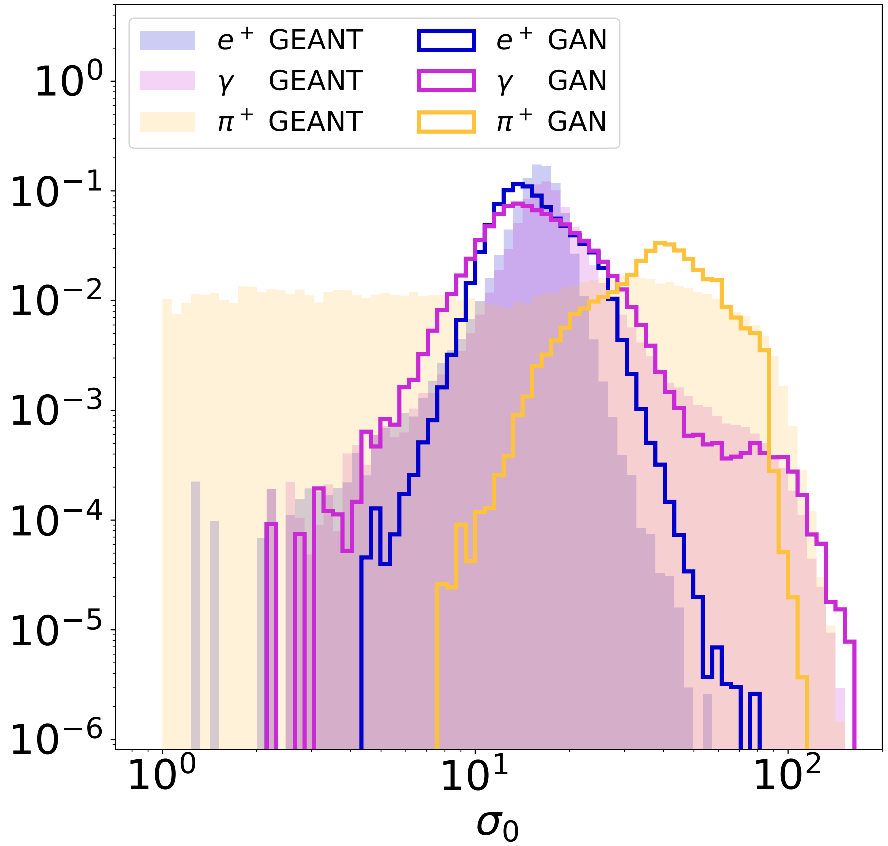
Implementing these tools in an experimental workflow is a key challenge but a lot of active R&D efforts ongoing!

Backup

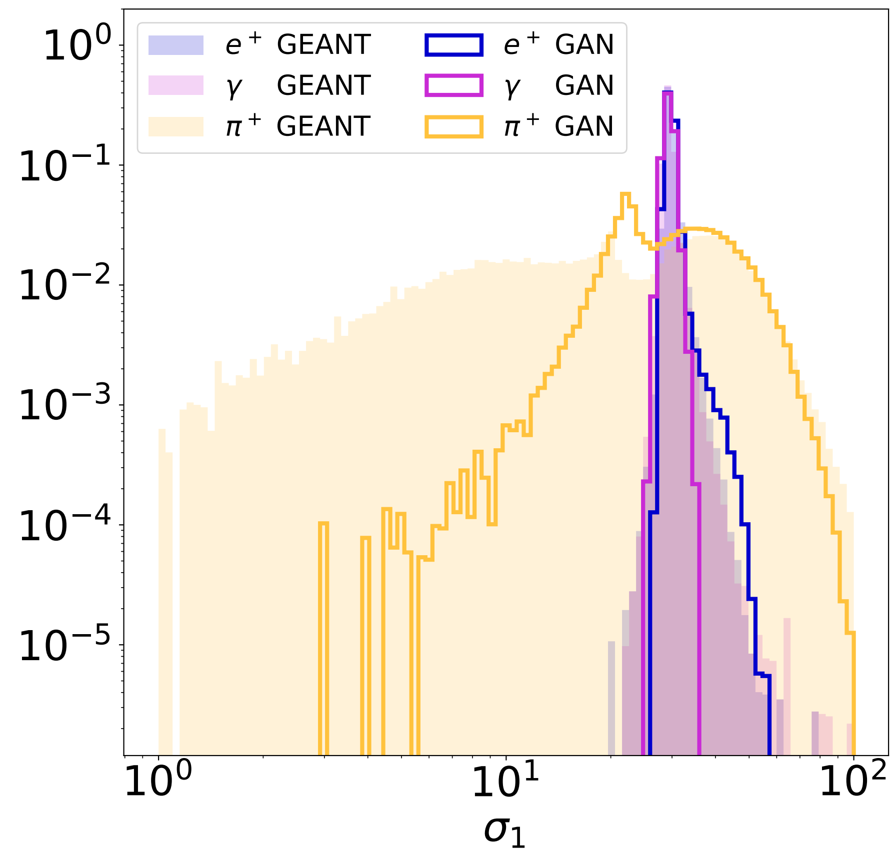
Depth of the shower

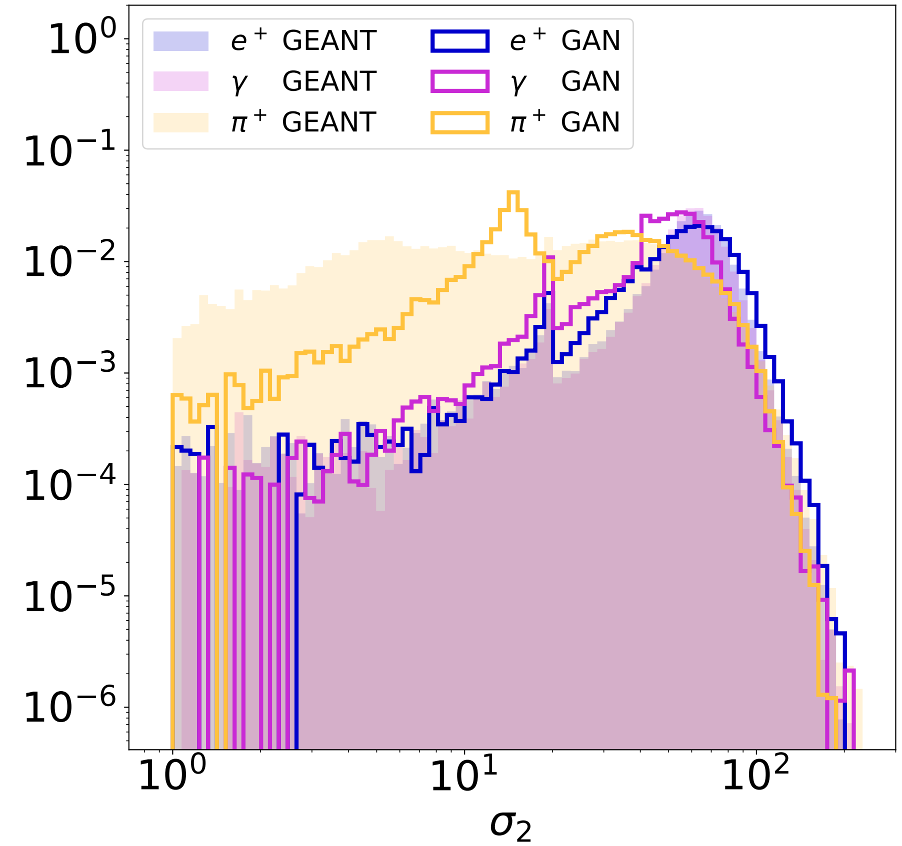


Lateral spread



The much larger variation in the pion showers is a challenge for the network.





These moments and others are useful for classification; we have also tested this as a metric (NN on 3D images)