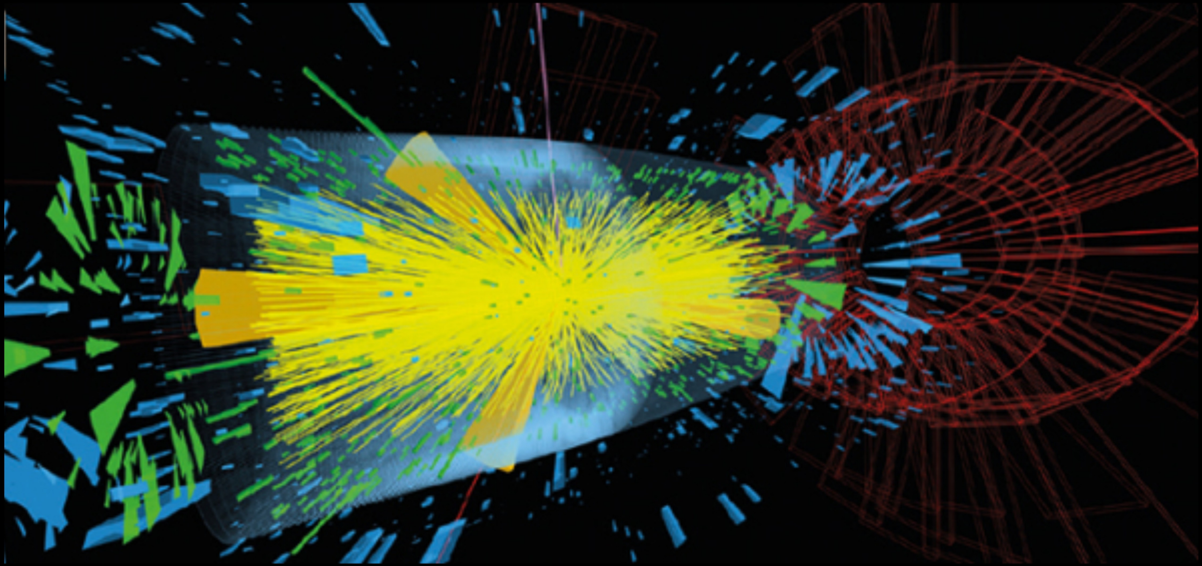
CaloDiffusion



Oz Amram
In collaboration with Kevin Pedro

Fermilab

CaloChallenge Workshop
May 30th, 2023

Diffusion Models

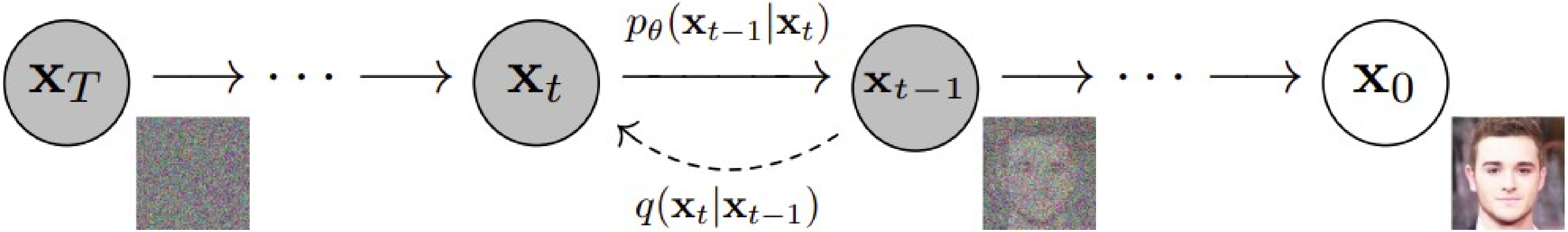
- Diffusion has become the dominant paradigm for ML image generation
  - Dalle-2, Midjourney, Stable Diffusion, etc
- Easy training, high quality results, reasonable computation times

“AI aiding physicists at LHC to analyze data and discover new particles”

Diffusion Models : Technical Details

2006.11239

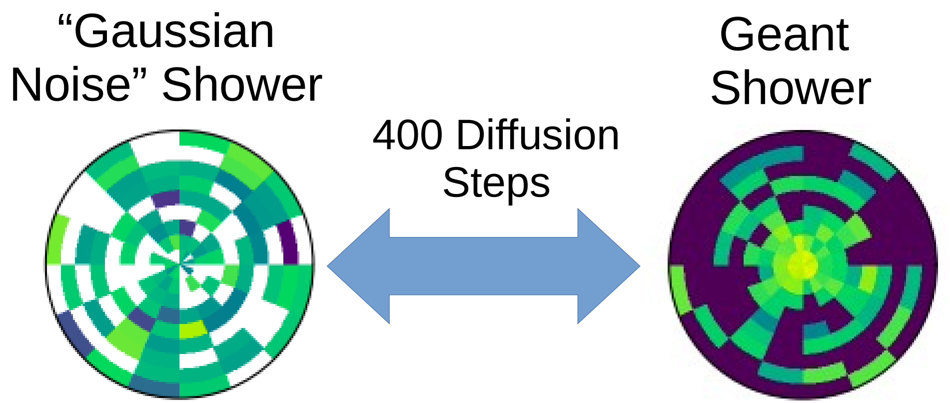
NSteps typically ~few hundred

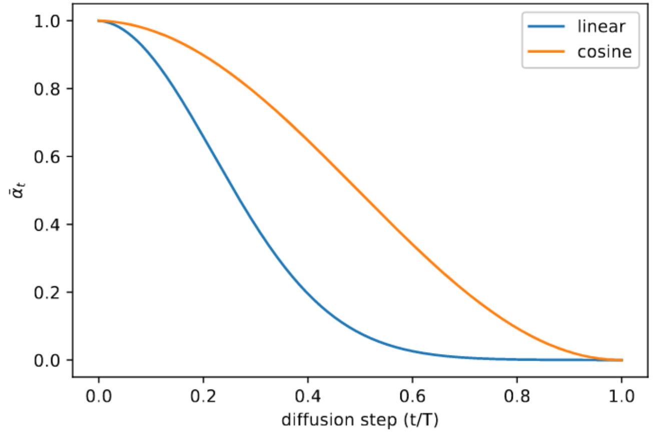


- Diffusion process: Starting with some image, **iteratively add Gaussian noise**, eventually reaching pure noise
- Train a model to **invert the diffusion process**
- Generate by starting from noise image, **iteratively denoise** using trained model
- Can condition on additional input information
  - Eg. text prompt or incident particle energy

‘CaloDiffusion’

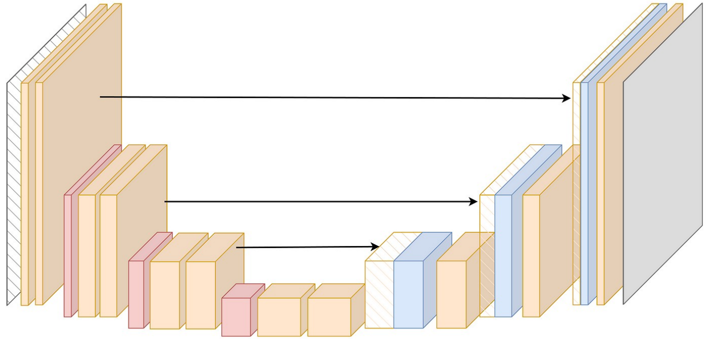
- We train diffusion models to generate synthetic calorimeter showers based on Geant simulations
- We use **400 steps** to interpolate from real shower to Gaussian noise
  - ‘cosine’ noise schedule of 2102.09672
- Preprocessing
  - Voxels divided by incident energy
  - Logit transformation
  - Standard scale so zero mean and unit variance
- Sample with “DDPM algorithm” (2006.11239)





Model Details

- Denoising network is has 'U-net' architecture based on 3D convolutions
  - Primary input: Noisy shower
  - Conditioning inputs: log(incident particle energy) & diffusion noise level
- 6 (8) ResNet blocks,
  - 4x compression in radial / angular dims
- Conditioning inputs embedded into 64 dim vector with 3 layer FCN
- 400k (1.1M) params for datasets 1 and 2 (3)

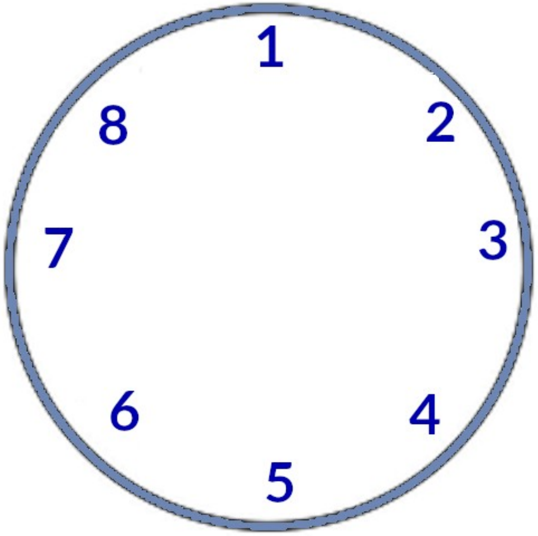


U-nets compress to a smaller dim space but also include skip connections

Optimizing for Cylindrical Data

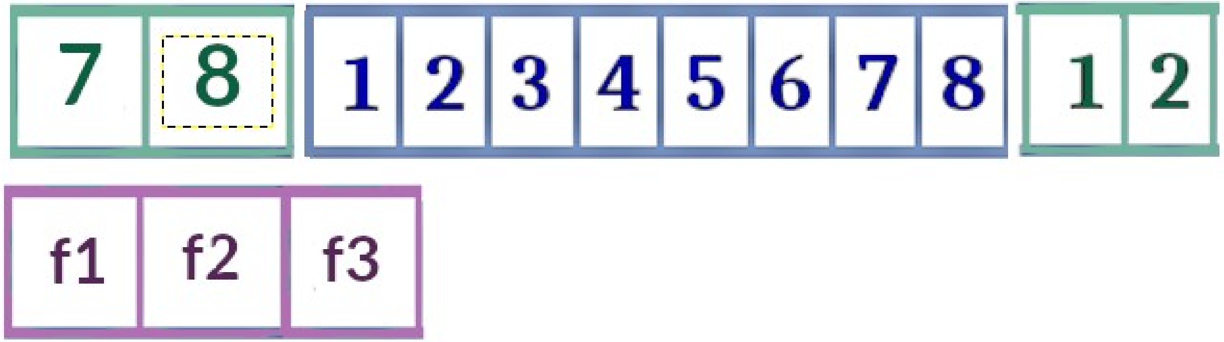
Implement **cylindrical convolutions** to respect periodic boundary of phi

Original Data



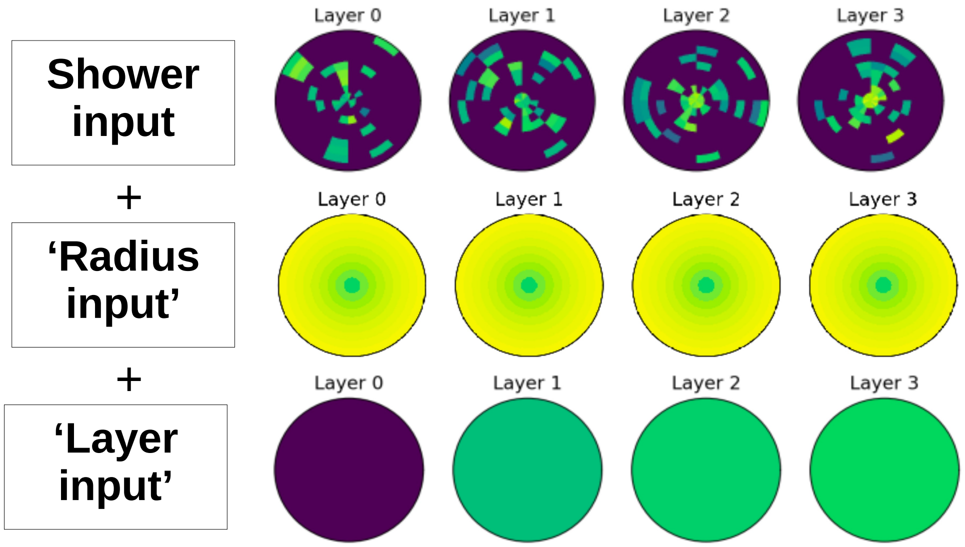
Linear Rep.

Circular Padding



Geometric Conditioning

- Showers are **not** translation invariant along R & Z
- Convolutions are inherently local → will do the same thing across whole geometry
- Instead allow convs. to be conditional on 3D location by adding additional input channels to shower 'image'
  - More efficient for learning



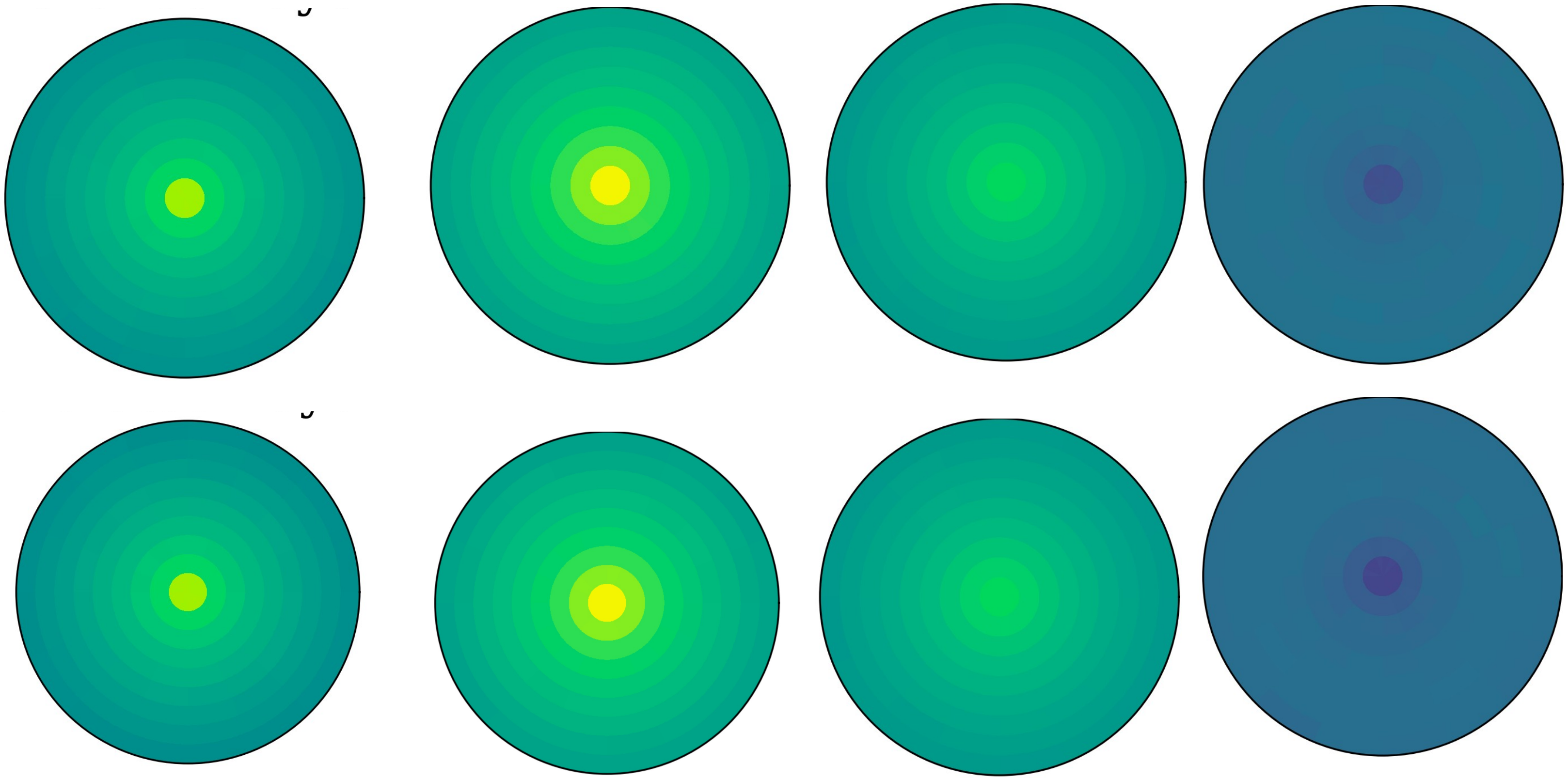
Additional input channels

Dataset 2 Average Showers

Layer 4 | Layer 12 | Layer 25 | Layer 42

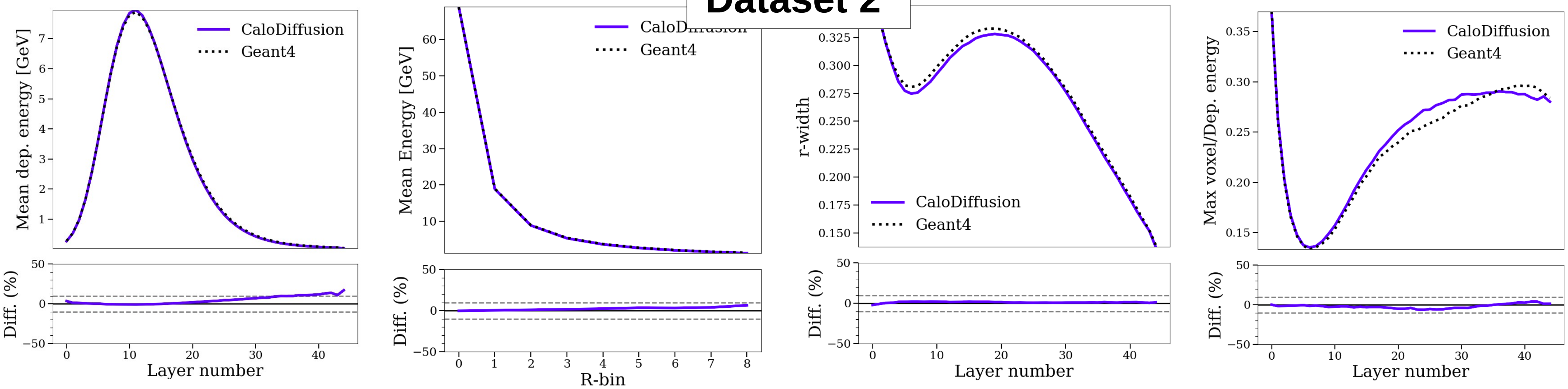
Geant

CaloDiffusion

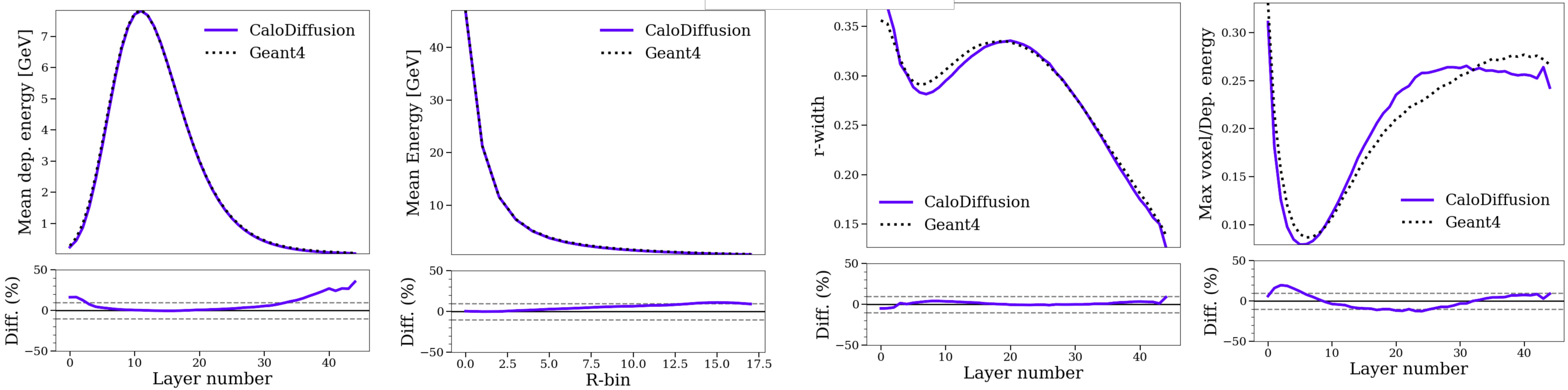


Dataset 2 & 3 Results (1/2)

Dataset 2

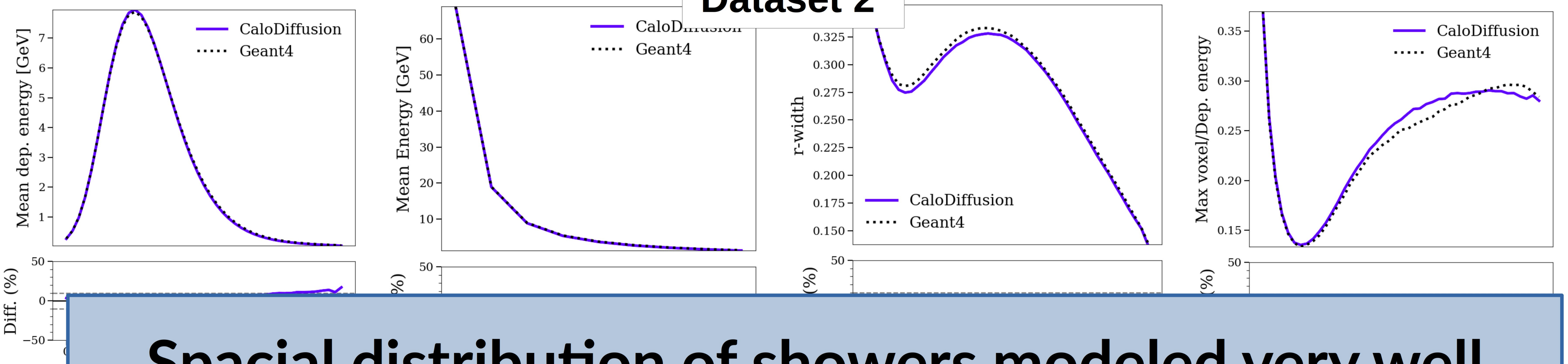


Dataset 3



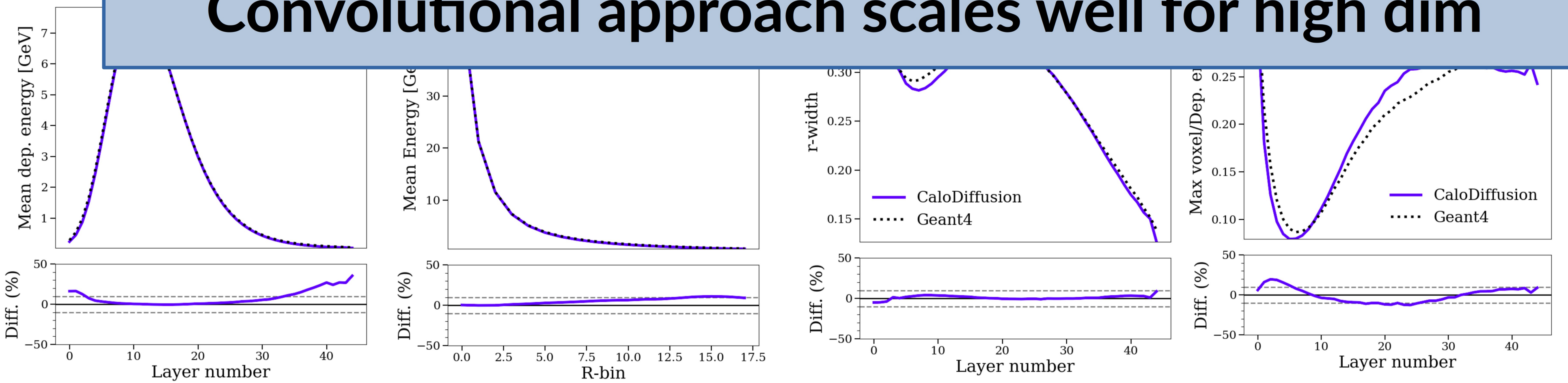
Dataset 2 & 3 Results (1/2)

Dataset 2



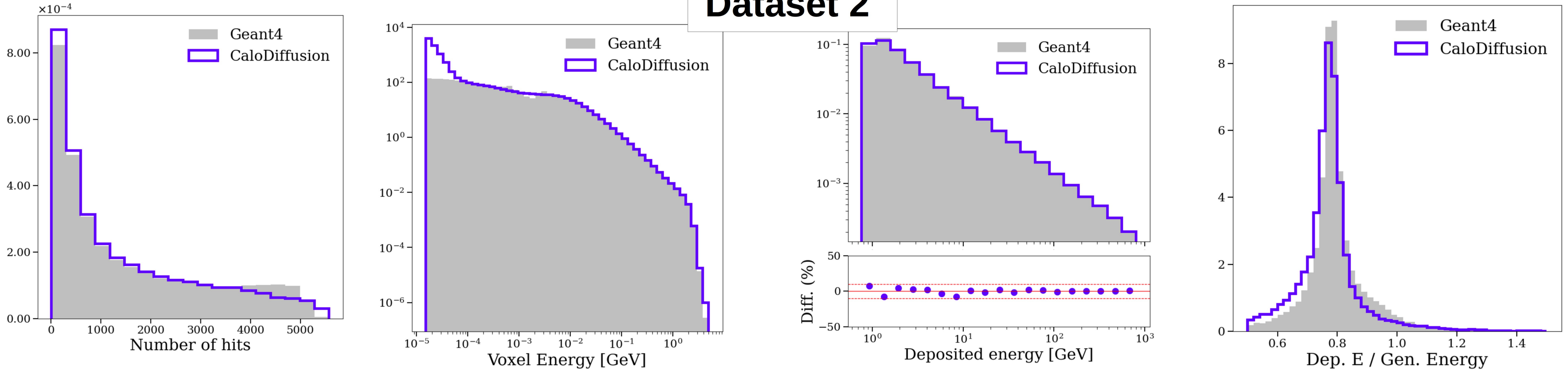
Spacial distribution of showers modeled very well

Convolutional approach scales well for high dim

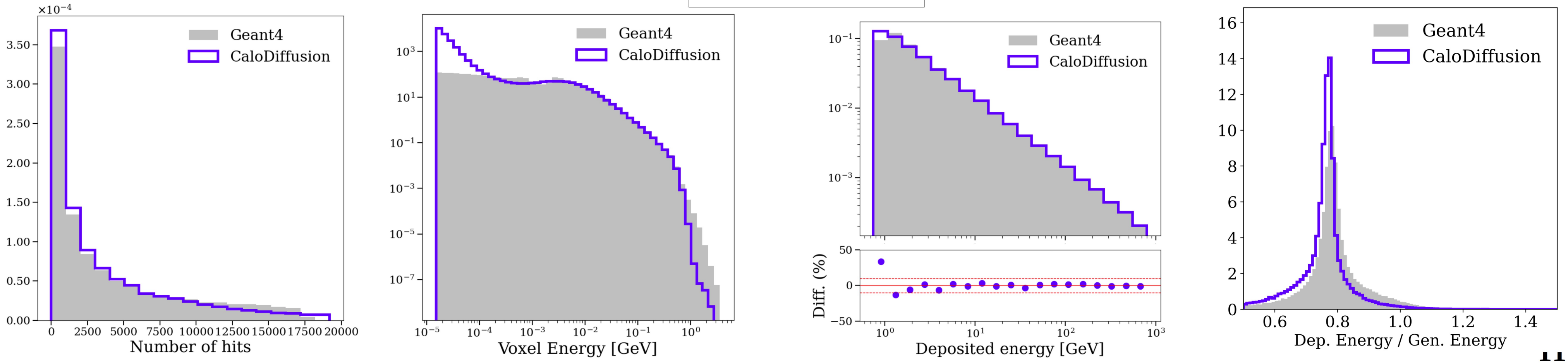


Dataset 2 & 3 Results (2/2)

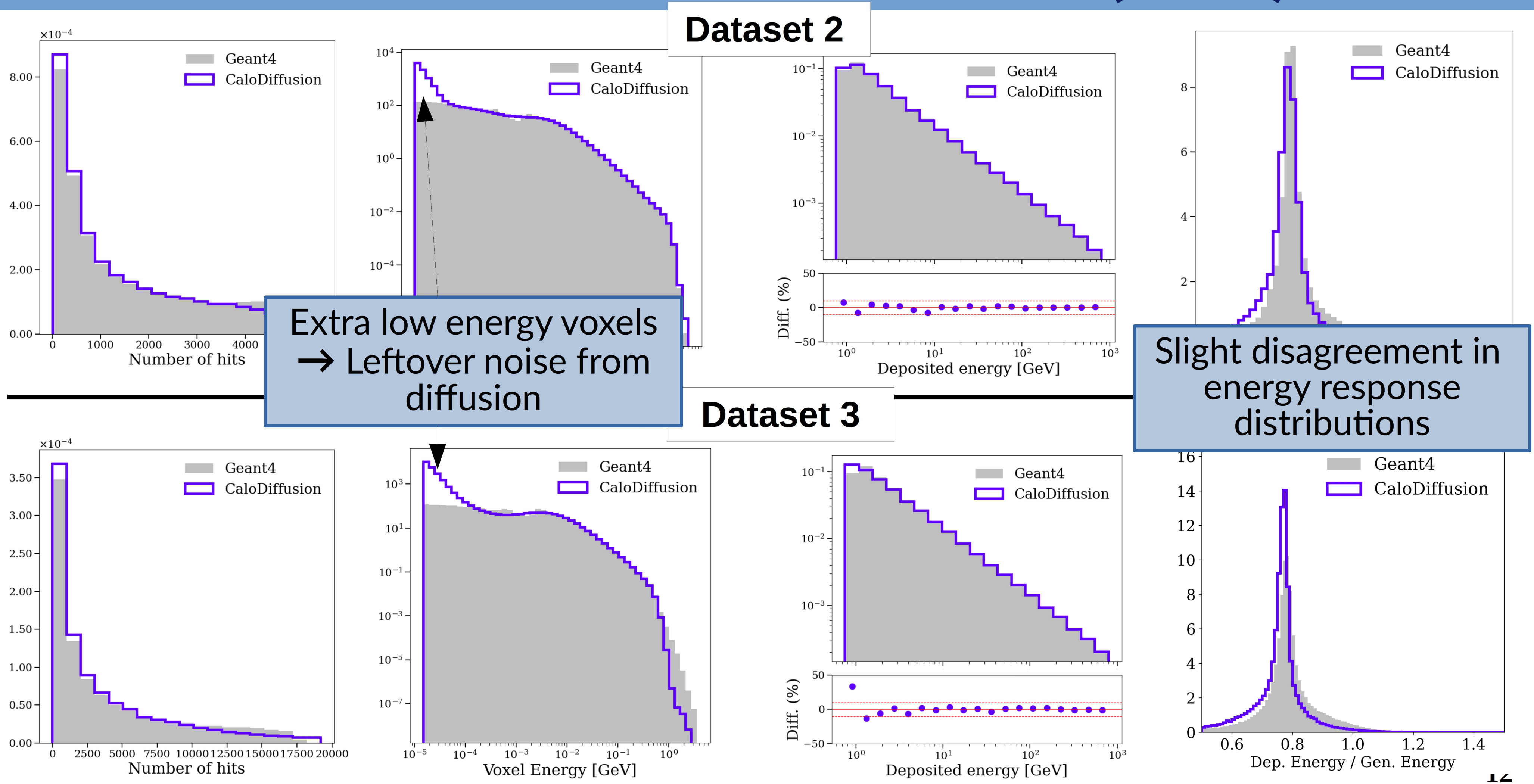
Dataset 2



Dataset 3



Dataset 2 & 3 Results (2/2)

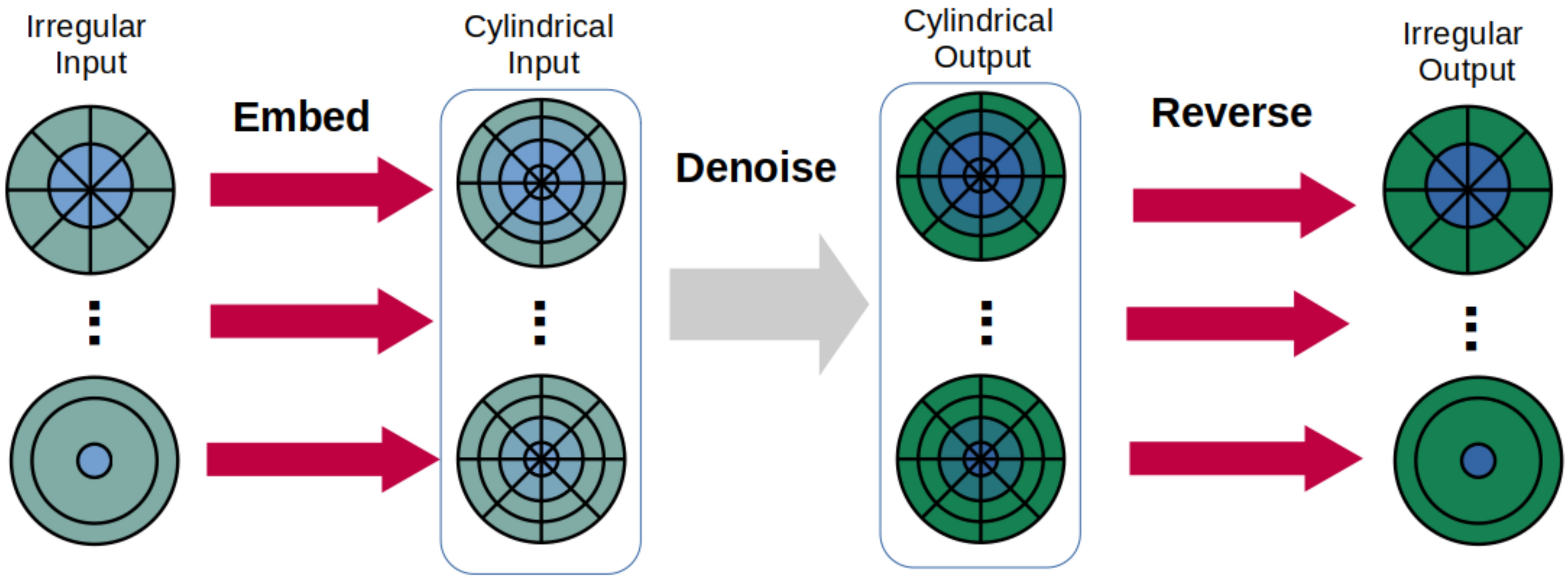


Extra low energy voxels → Leftover noise from diffusion

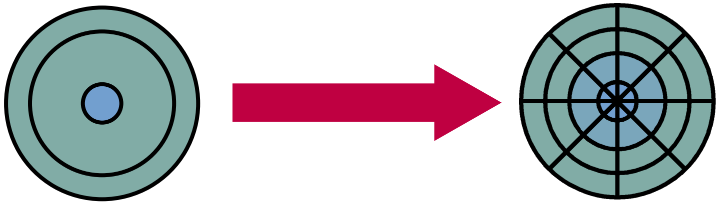
Slight disagreement in energy response distributions

Embedding Irregular Geometries

- Dataset 1 (ATLAS detector) is cylindrical but has **irregular structure** in layers
  - Different radial / angular bins in each layer → can't apply cylindrical convolutions
- GLaM : Learn an **embedding** that maps input into **regular cylindrical structure**



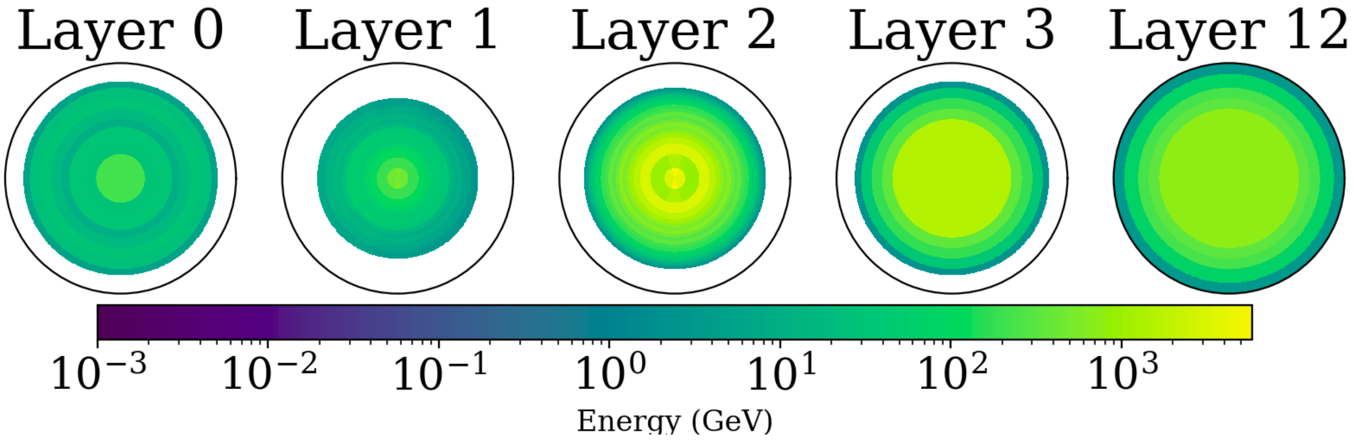
GLaM : Geometry Latent Mapping



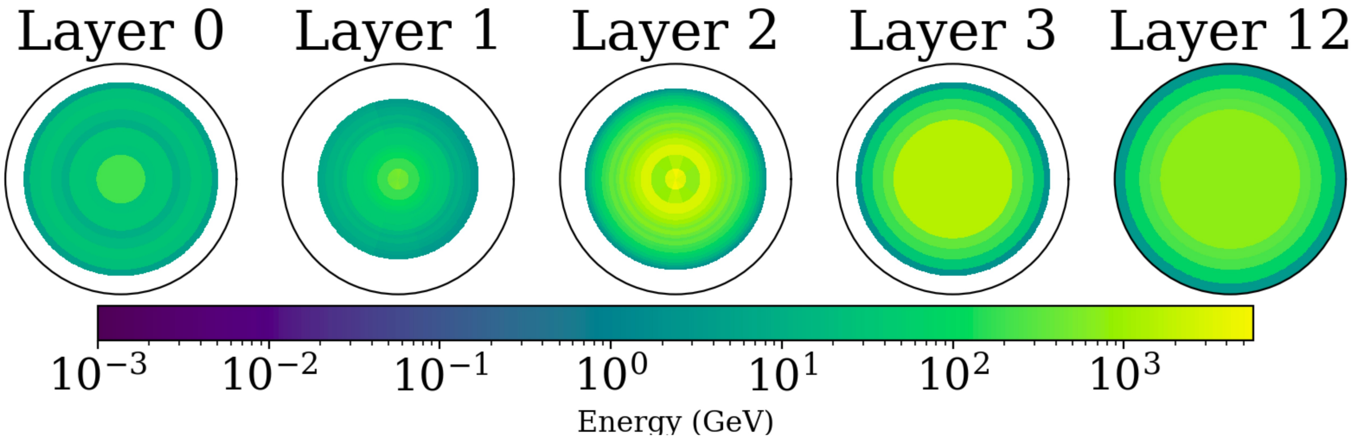
- Decide on regular cylindrical shape → maximal set of all radial + angular bins
  - Significant increase in dimensionality (368→ 5x10x30)
- For each layer, map from irregular binning to regular structure
  - Enforce angular symmetry → split evenly among angular bins
  - Parameterize radial mapping with a **single learnable matrix** per layer, optimized during diffusion training
  - Initialized to **geometric overlap** between bins + O(10^-5) noise
- Embedding is only ~3k params for dataset 1!

Dataset 1-Photons Average Showers

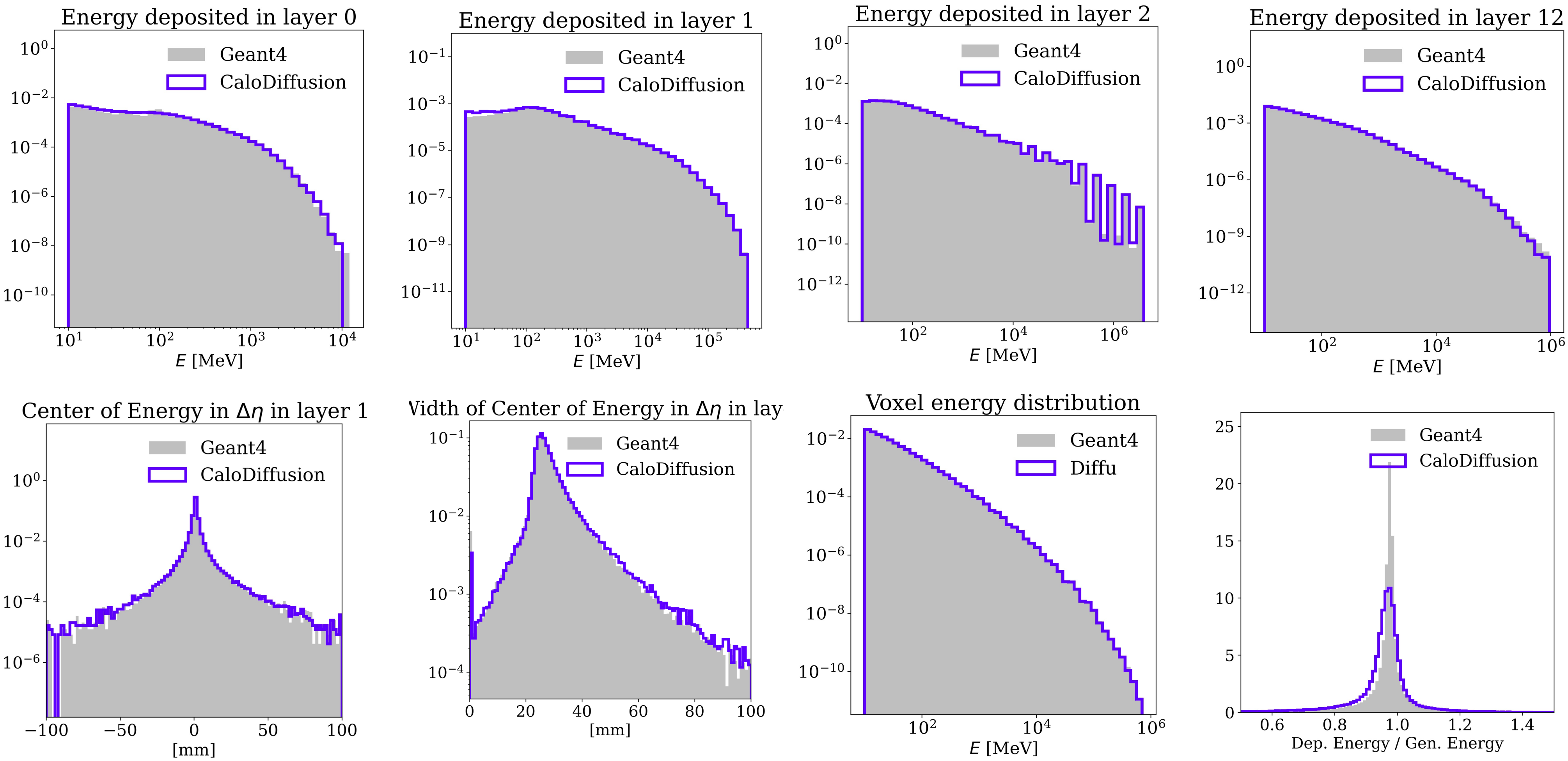
Geant



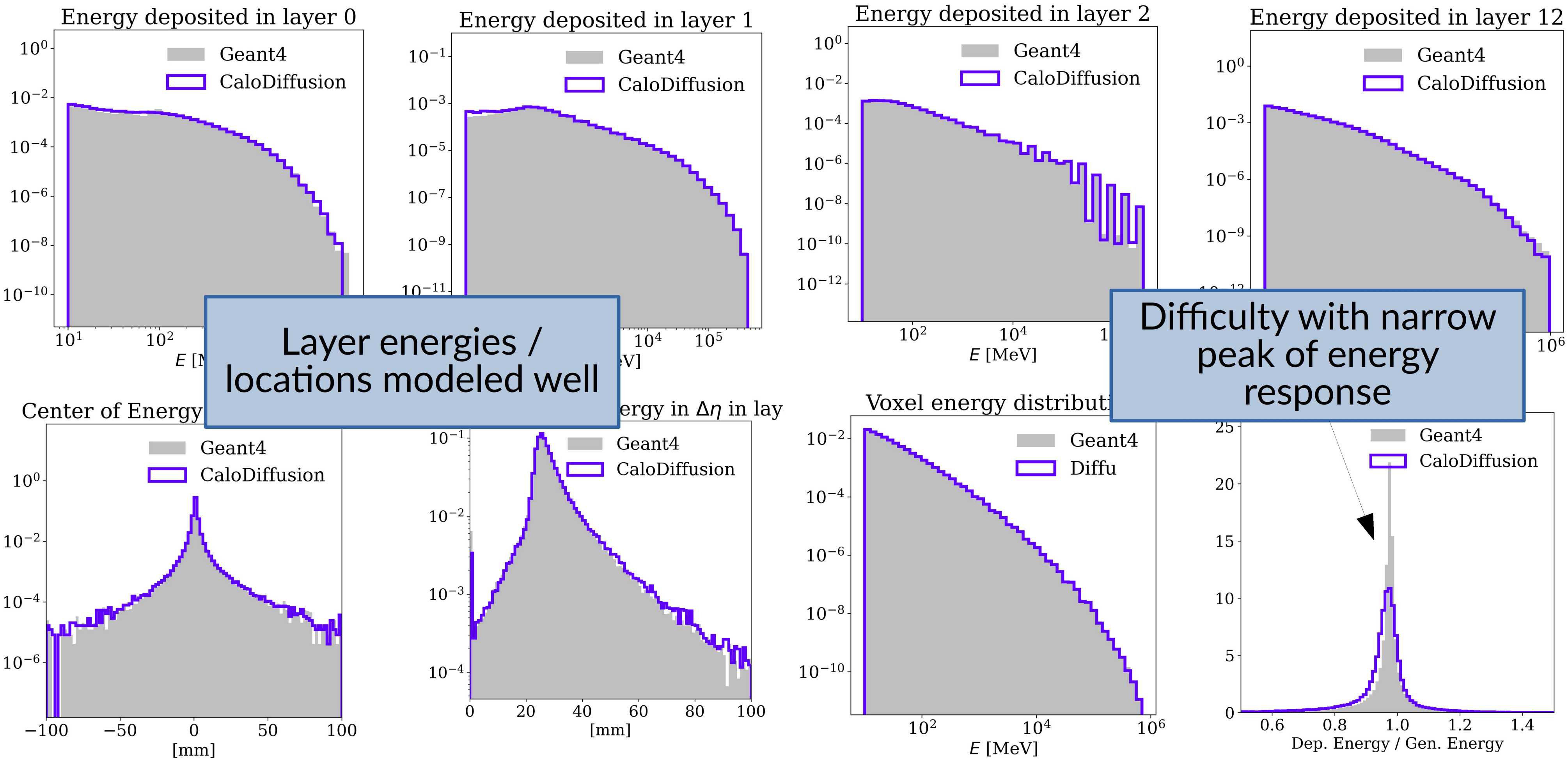
Diffusion



Dataset 1-Photons Results



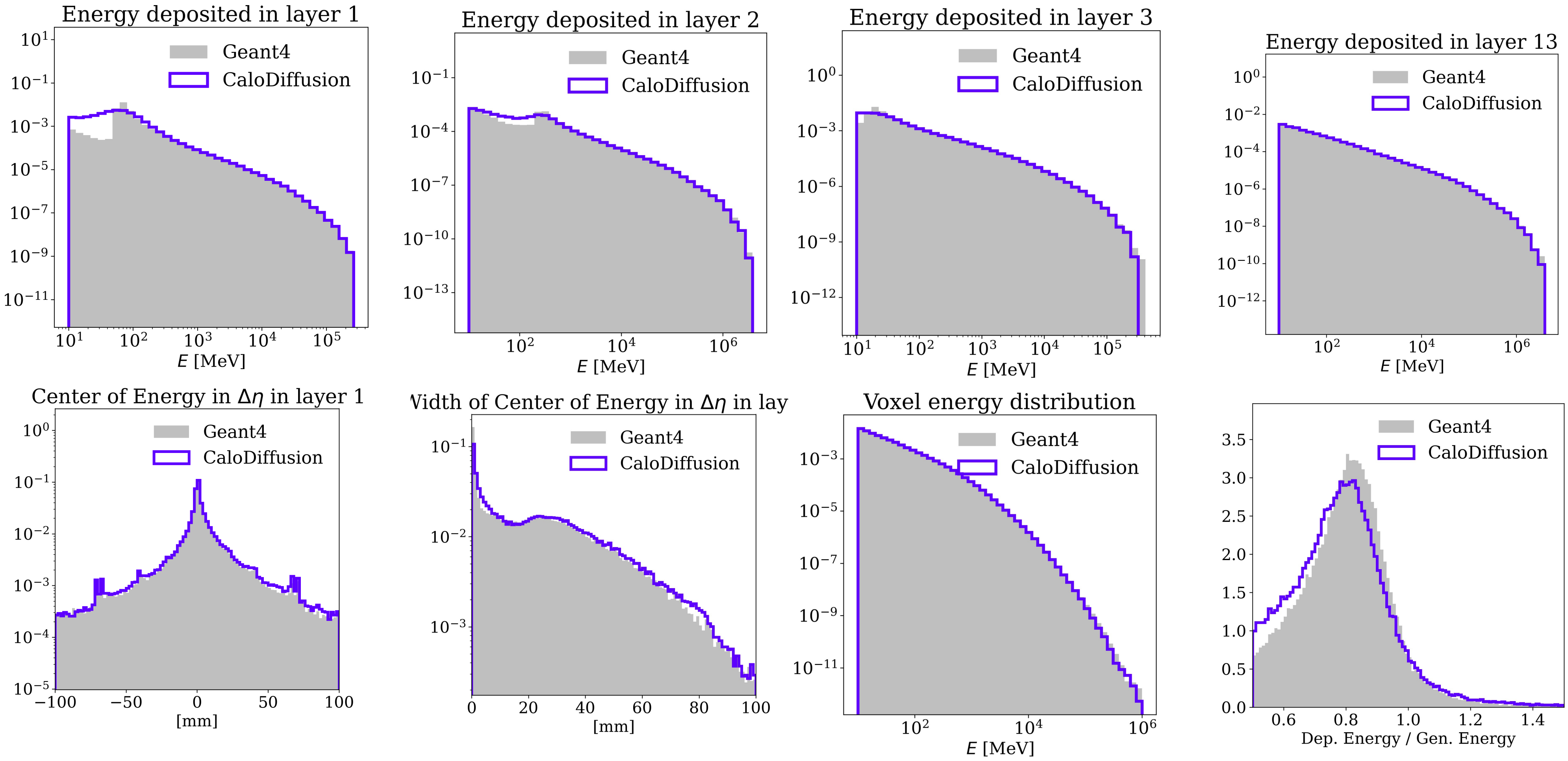
Dataset 1-Photons Results



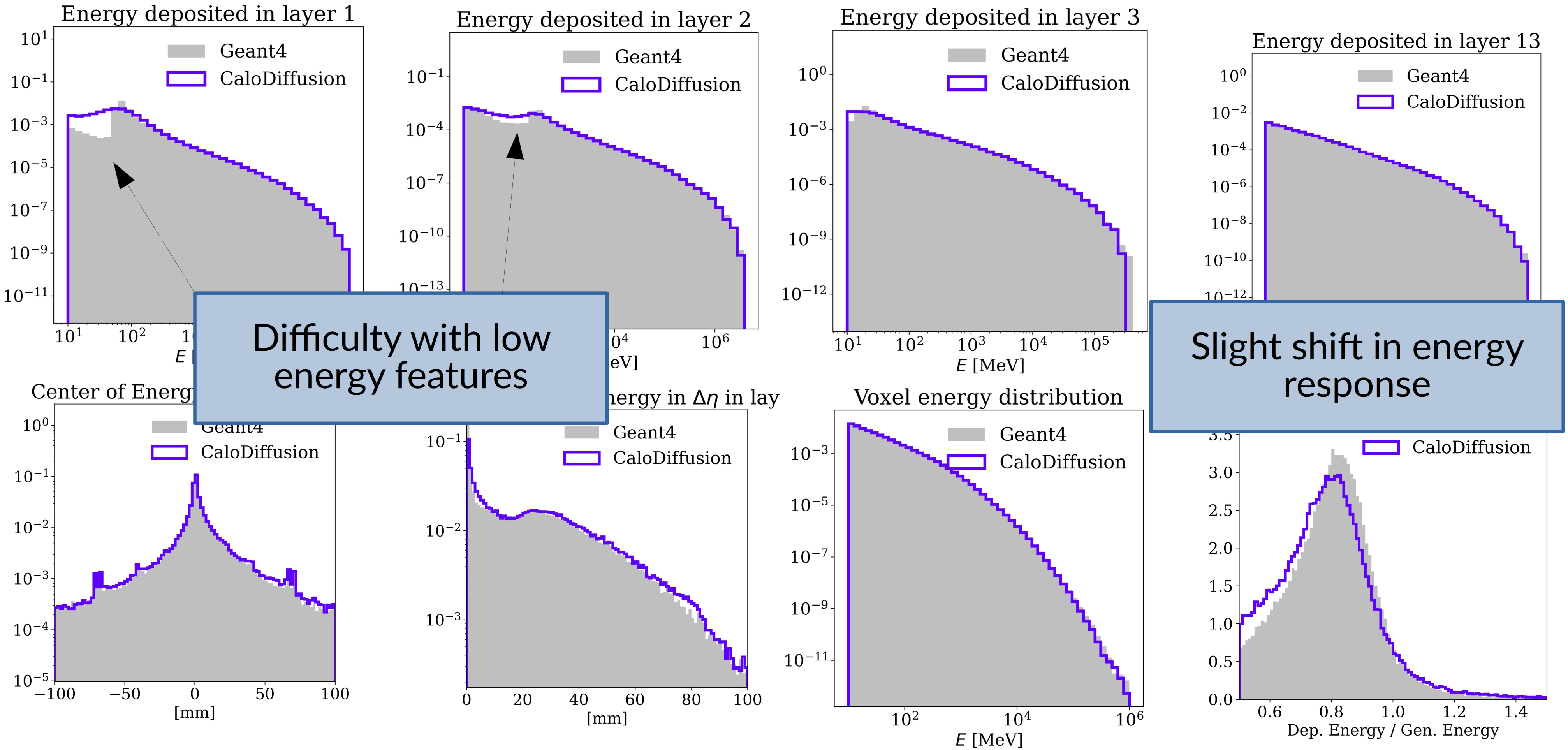
Layer energies / locations modeled well

Difficulty with narrow peak of energy response

Dataset 1-Pions Results



Dataset 1-Pions Results



Difficulty with low energy features

Slight shift in energy response

Classifier Metric

- Train a NN classifier to distinguish between Geant showers and CaloDiffusion showers
  - 2 hidden layers of 512 nodes, dropout = 20%
- Similar results for high-level and low-level input features

	Dataset 1-pions	Dataset 1-photons	Dataset 2	Dataset 3
AUC (low-level / high-level)	**0.64 / 0.74**	**0.64 / 0.67**	**0.61 / 0.61**	**0.73 / 0.77**

AUC's much less than 1 for all datasets!

Timing

- Evaluated generation time of our model using on CPU (Intel E5-2650v2) & GPU (NVIDIA V100)

Dataset	Batch Size	Time / Shower, CPU [s]	Time / Shower, GPU [s]
1-photons (368 voxels)	1	5.3	3.0
	10	1.3	0.3
	100	0.7	0.08
1-pions (533 voxels)	1	5.7	3.0
	10	1.3	0.4
	100	0.7	0.07
2 (6.5k voxels)	1	9.6	2.6
	10	3.4	0.3
	100	3.2	0.2
3 (40.5k voxels)	1	52.7	4.1
	10	44.1	1.4
	100	-	1.3

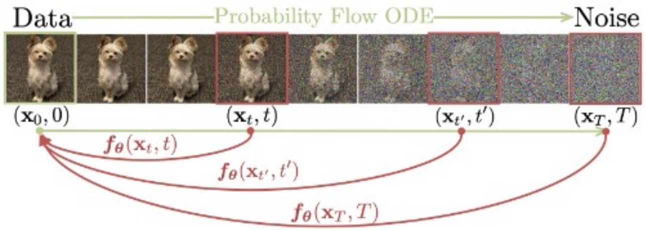
Future Work I

- Very minimal **hyperparam optimization** done so far, likely significant room for improvement
  - Pre-processing closer to std normal would likely help for diffusion
- Lots of room to explore in GlaM approach,
  - Picked simplest setup that worked
- Some **“global” properties** (ie total shower energy), can still be improved
  - Hard to specifically optimize in diffusion training
  - Could try ‘distributional’ MMD loss with large batch size
  - Or separate network to learn total energy distribution, normalize diffusion shower to match

Future Work II

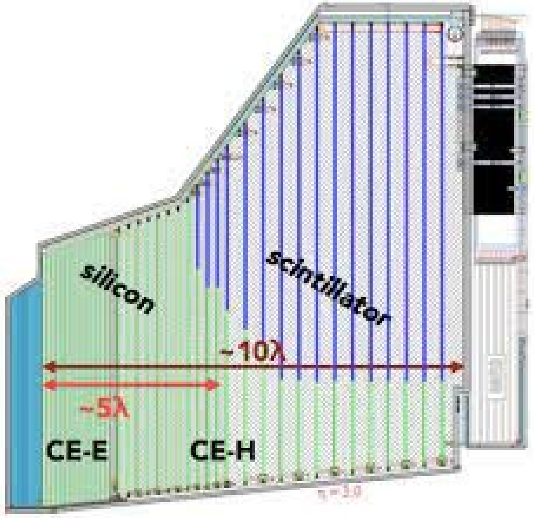
- Generation time could be improved
  - General prob with diffusion models → active area of research
    - Improved sampling algos
    - Compression to a latent space
    - **Distillation methods**
      - Already demonstated in 2304.01266

“Consistency Models” distill diffusion model to allow ~few step generation



- Perhaps starting generation from **approximate shower** instead of pure noise will be faster / easier
  - “Cold Diffusion”, 2208.09392

- Extend to more complicated geometries e.g. **CMS HGCal**



Outlook

- **CaloDiffusion** able to generate high quality showers for **all datasets**
  - Convolutional approach scales well

Lookout for paper on arxiv soon!

- Several novelties
  - Optimizations for cylindrical data
  - GlaM lightweight **embedding** for irregular geometries
- Promising future directions for improvement

Acknowledgements

- Co-author : Kevin Pedro

- This work was performed with support of the U.S. CMS Software and Computing Operations Program under the U.S. CMS HL-LHC R&D Initiative.
- Additional support from the Fermi National Accelerator Laboratory, managed and operated by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy
- Thank you to the CaloChallenge organizers for organizing everything!

Thanks!

Backup

Technical Details

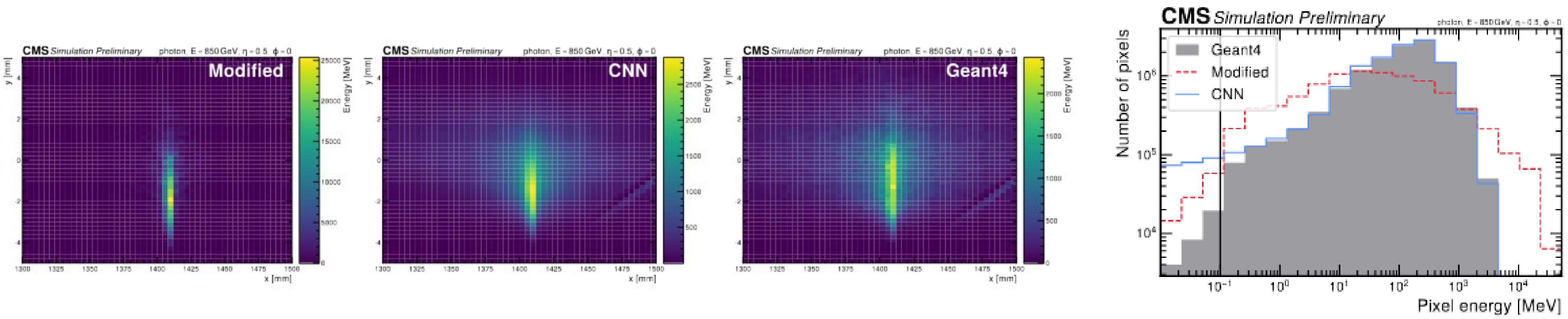
- 'logit' transformation of voxel energies and then standard scale to zero mean and unit variance
  - Correct preprocessing important for diffusion process, related to scale of added noise
- Denoising network uses 'U-net' architecture with cylindrical convolutions
  - Two conditional inputs : shower energy and diffusion step
  - ~400k params for dataset1 and 2, 1.1M for dataset3
- 400 diffusion steps, 'cosine' noise schedule (2102.09672)
- Choices for training objective:
  - Datasets 1 and 2 : Network is trained to predict noise component of image
  - Dataset 3 : Network trained to predict weighted average of noise component and un-noised image,
    - More stable, recommended by 2206.00364
- Sampling uses DDPM algorithm (2006.11239)

Additional Metrics

- Distance metrics:
  - Frechet Particle Distance and Kernel Particle Distance (proposed in 2211.10295)
    - Use implementation proposed for CaloChallenge, based on high level shower features
  - We find that the computation of FPD is slightly biased, ie non-zero values even comparing different random samples of Geant to each other
  - Compare scores for Diffu-Geant (D-G) vs Geant-Geant (G-G)

	Dataset 1 Photons	Dataset 2	Dataset 3
FPD (D-G / G-G)	0.035 / 0.008	0.095 / 0.008	0.275 / 0.011
KPD (D-G / G-G)	0.007 / 0	0.0001 / 0	0.0007 / 0

Previous Work (arXiv:2202.05320)



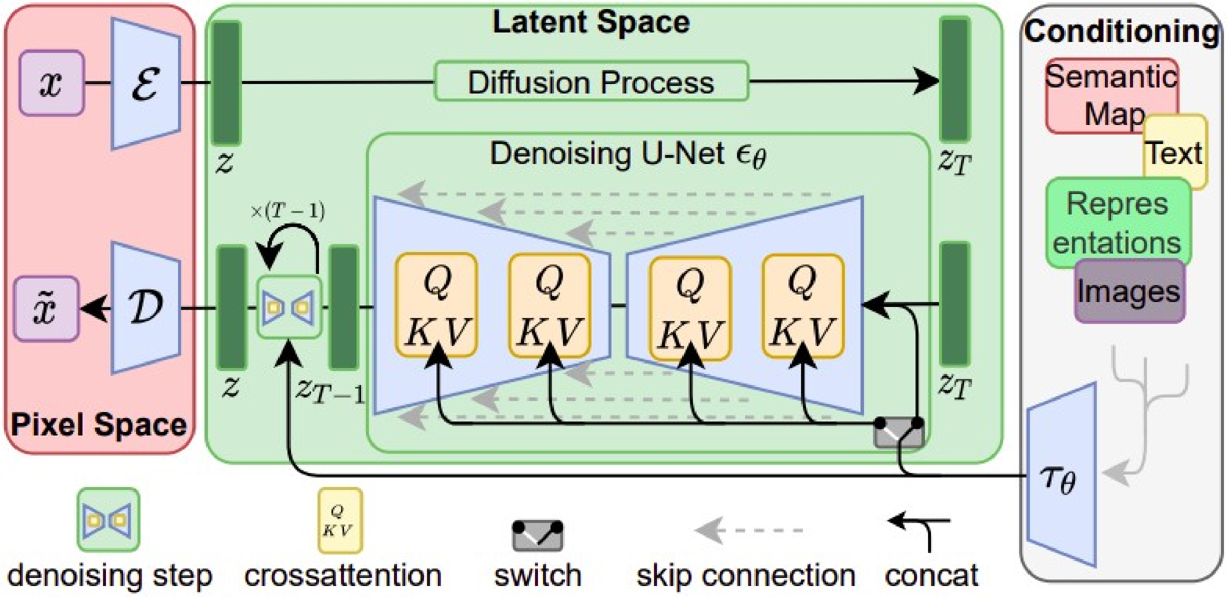
- Generated calorimeter showers with regular & ‘fast’ version of Geant4
- Use a CNN network to ‘denoise’ fast-sim shower image to match high granularity one
- Decent performance in a relatively simple setup
  - Studies showed adding more info to the network beyond ‘energy image’ only moderately improved performance
    - Tried multiplicity, time of energy deposit, other Geant info

Latent Diffusion Models

- Key advantage is that costly diffusion steps done in smaller latent space
- Relies on encoder not losing any important info
  - ‘perceptual loss’ supposed to reduce blurriness
  - Small regularization of latent space (std. normal KL or vector quantization) during AE training
- Conditioning setup very flexible
  - Text prompts using some language model
  - Image conditioning
  - …

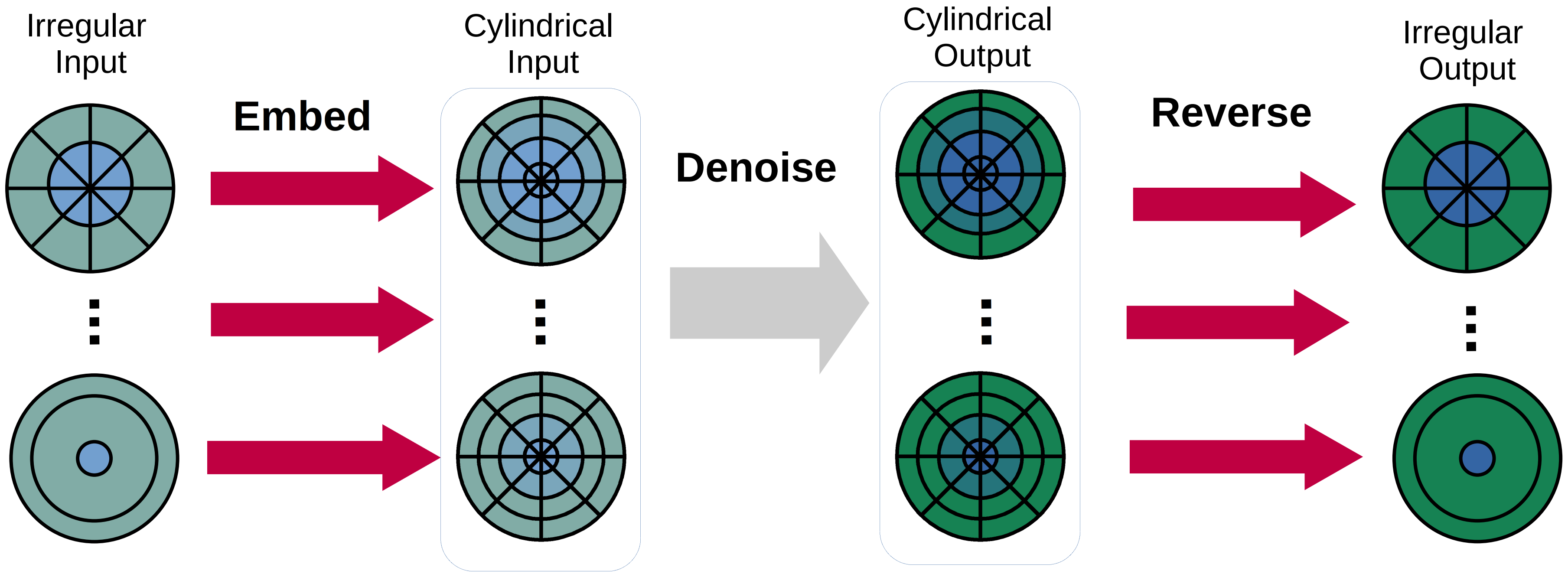
Stable Diffusion (aka Latent Diffusion)

2112.10752

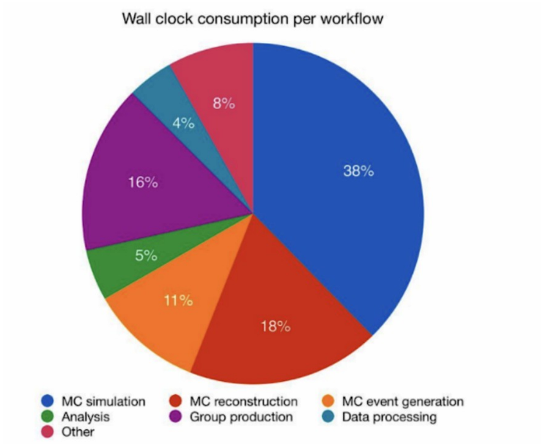


- First encode your image with an autoencoder to a smaller latent space
  - They used a factor of 4 or 8 for each dimm.
- Transform your conditioning data into a latent rep
- Denoising performed on the latent representation of your image, using conditioned data
  - Conditioning done using an attention mechanism
- Decode back into pixel space

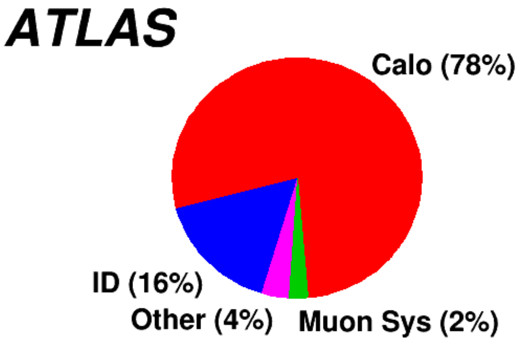
Geometry Diagram



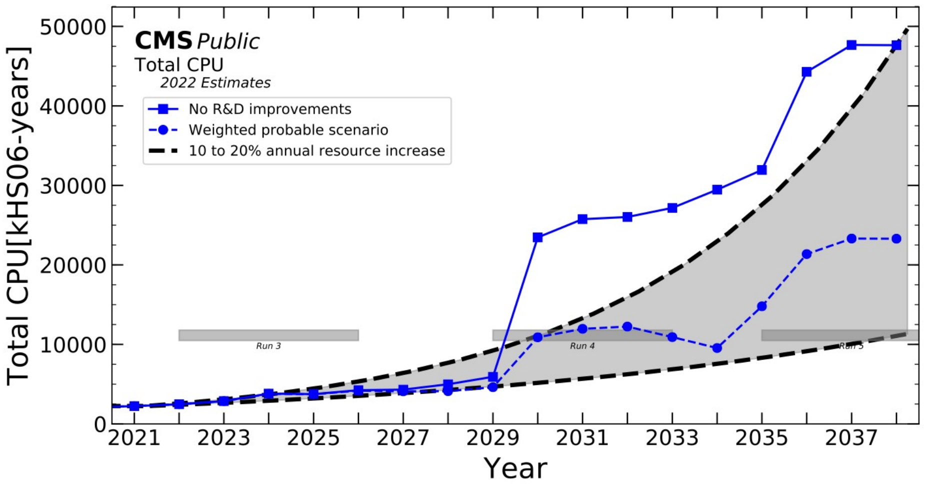
The Need for Fast Simulation



ATLAS CPU hours used by various activities in 2018

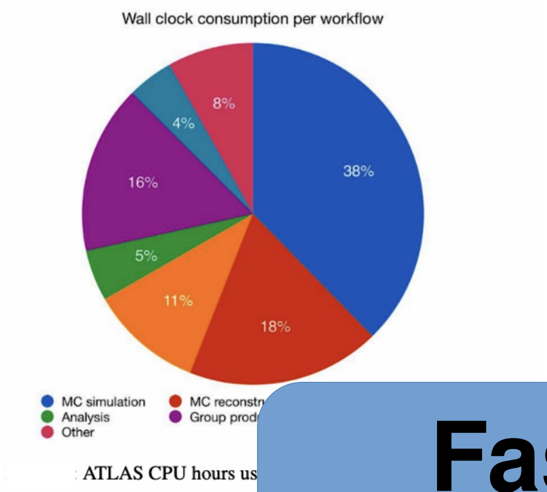


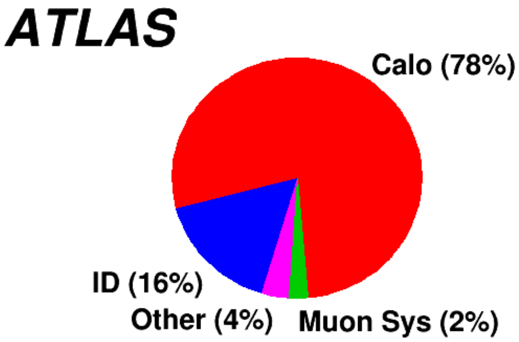
Subdetector CPU fraction for 50 ttbar events
MC16 Candidate Release

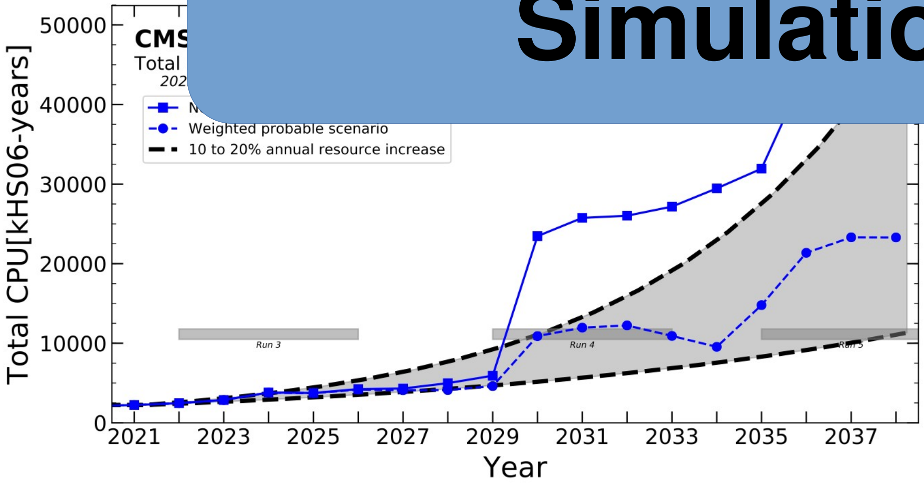


- Geant4 calo simulation is a significant part of ATLAS computing budget
  - CMS will face similar needs with HGCAL in HL-LHC
- For HL-LHC, computing simulation more crunched
  - Reconstruction usage will scale ~linearly with pileup → less resources for sim.

The Need for Fast Simulation





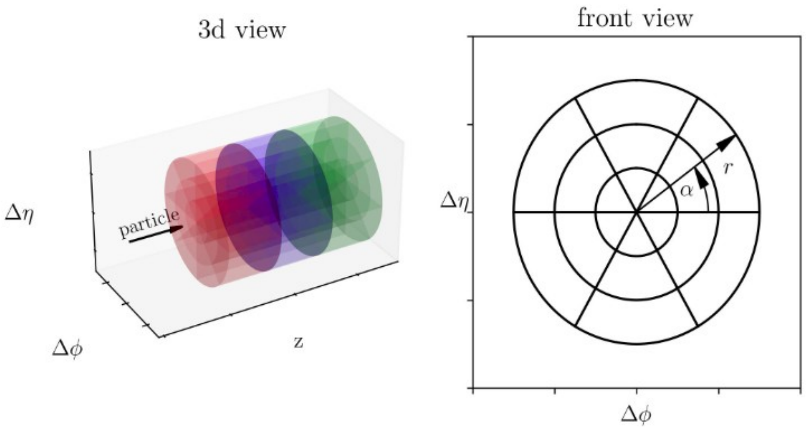


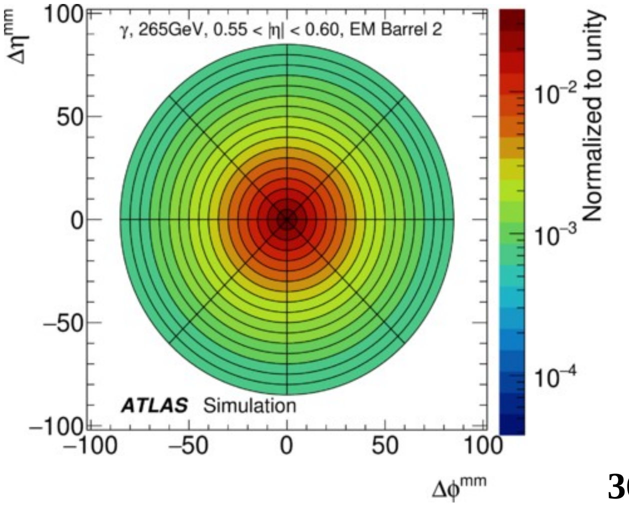
Fast & Accurate Calorimeter Simulation is Needed!

- Geant4 calo simulation is a significant part of ATLAS computing budget
  - CMS will face similar needs
    ...HC
- ...ing simulation more crunched
  - Reconstruction usage will scale ~linearly with pileup → less resources for sim.

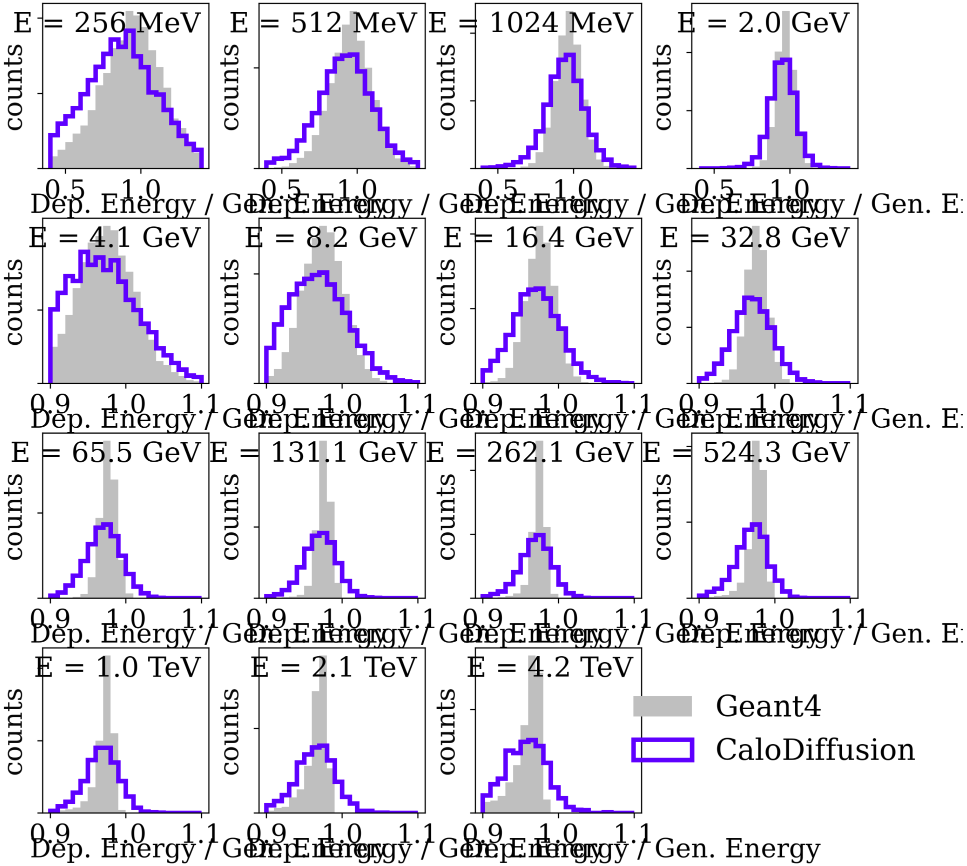
Dataset: Calo Challenge

- Community challenge to compare generative models for Calorimeter simulation
- Standard datasets to allow comparison
  - Dataset1: ATLAS-like geometry, 5 layer cylinder with **irregular binning**, 368 voxels
  - Dataset2: 45 layers, 6480 total voxels
  - Dataset3: 45 layers, 40,500 total voxels





Dataset-1 Photons Energy Response



Dataset-1 Pions Energy Response