Towards a differentiable sampling ECAL model with optimal absorber material distribution

Shivay Vadhera (LLR – Summer Student 2022; UM-DAE Centre for Excellence in Basic Sciences).
Dr. Fabricio Jiménez Morales (LLR – Institut Polytechnique de Paris / CNRS)
Dr. Vincent Boudry (LLR – Institut Polytechnique de Paris / CNRS)

Goals

- Overarching Goal: Optimization of SiW-ECAL absorber material distribution.
- Sub-Goal: Find a way to generate events in an “optimizable” manner.
  - Speed would be a nice byproduct.

Goals (Cont’d.)

- Several advantages to using a differentiable model!
- One choice is the Generative Adverserial Network (GAN), to generate hits.
- Tried to keep the design simple – “What is the least complicated thing that works?”

The Prototype

- 15 layers, 32 x 32 channels.
- Tungsten absorber; Si cells (5mm x 5mm)

Dataset Used

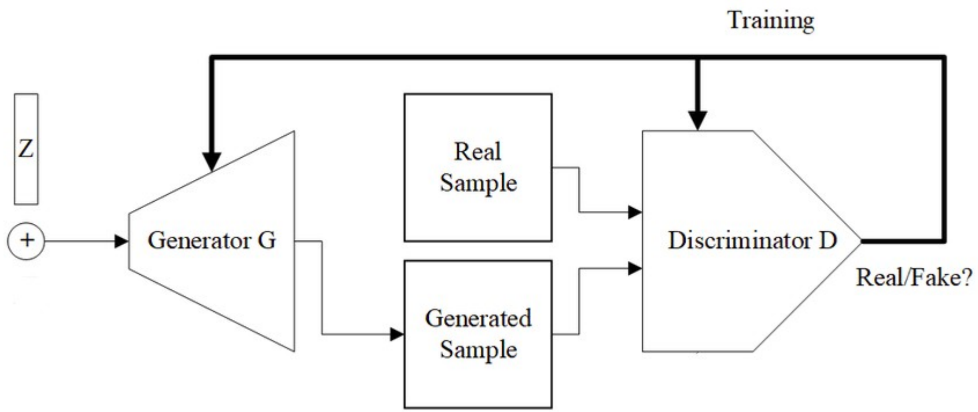
- We produced a simulated dataset for the SiW-ECAL prototype:
- 10 GeV electrons at normal incidence (5k events).
- Arrangement of 15 layers as in last prototype beam test:
  - Si of three different thicknesses (320, 500 and 650 μm)
  - 8 x 4.2mm + 7 x 5.6mm W layers (20.8 X0).
- Not Included:
  - Readout effects.
  - Dead cells.

Up Next

- A 2D WGAN
- A 2D WGAN + Conservation
- A 3D GAN
- DNNs, PCA+WGANs
- Future Work

IDEA #1: A 2D WGAN

A WGAN (Wasserstein GAN, with GP) is used because it is generally more “stable” than a simple GAN.



- Image source: Wikipedia (https://upload.wikimedia.org/wikipedia/commons/6/64/A-Standard-GAN-and-b-conditional-GAN-architecturpn.png)
- Attribution: האדם-החושב, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

IDEA #1: A 2D WGAN

- Dataset: N 15 x 32 x 32 images -> N 15 x 1024 images.
- Generator: 64 x 1 x 1 -> 1 x 15 x 1024
- 3 batches of ConvTranspose2d+BatchNorm+ReLU
- Some other iterations were also tried out; the latest one being ending with a Conv2d+ReLU combination on the output.

IDEA #1: A 2D WGAN

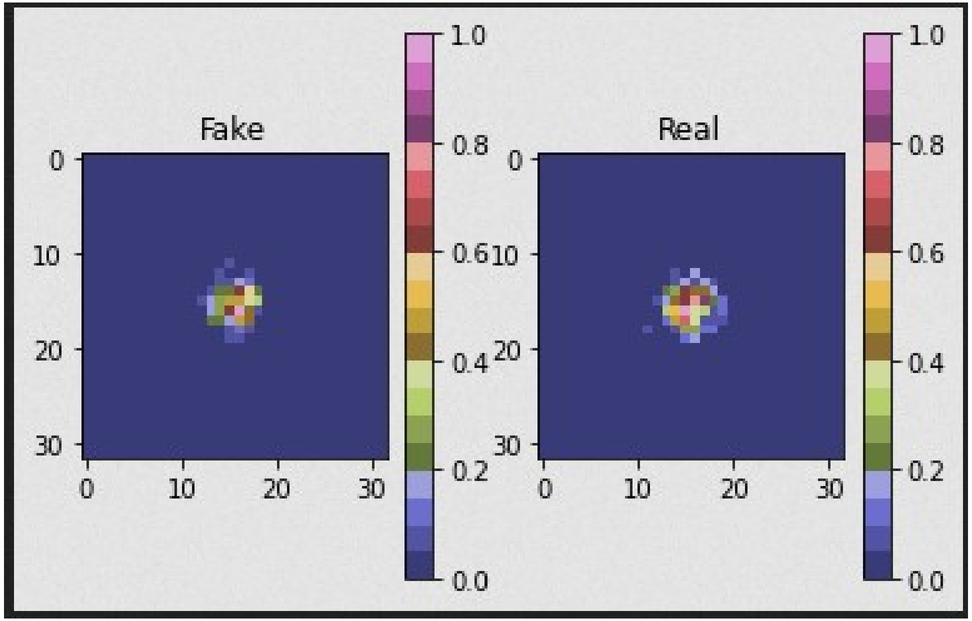
- Critic: 1 x 15 x 1024 -> 1 x 1 x 1
- 3 batches of Conv2d+InstanceNorm+LeakyReLU.
- Only Conv2d on the last step.

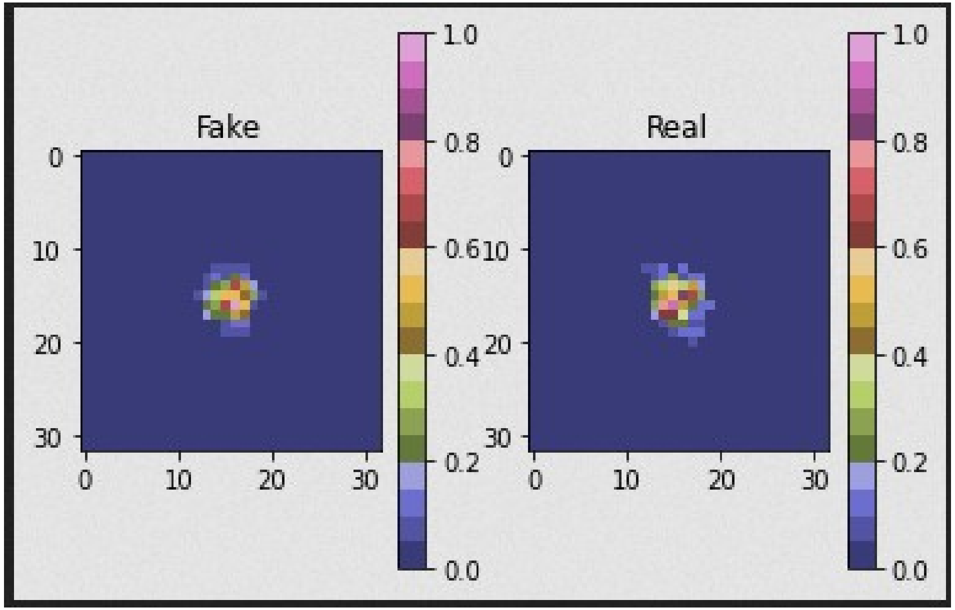
IDEA #1: A 2D WGAN: Results

- Qualitative agreement quite good when compared with the results of all the events together.
- Systematic bias: The first layer never had any hits. The low number of hits in the first layer could explain this.

IDEA #1: A 2D WGAN: Results

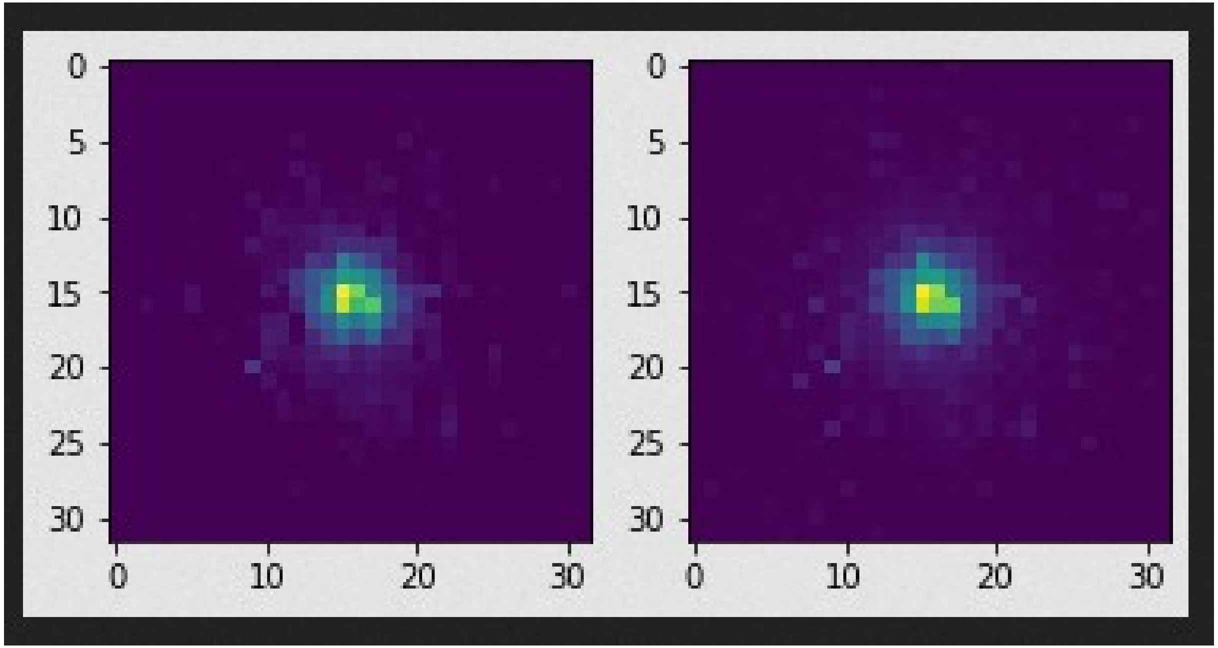
- The result for layers 3 and 4 with 64 events is shown below (the maximum hit_E for both is normalized to 1):





IDEA #1: A 2D WGAN: Results

- Side Note: Trying with 5k pion (10GeV) events together, gives a much better result. (left-fake, right-real).



IDEA #2: Including Conservation

- Penalize diff in sum_E and nhits. Start with sum_E.
- Tried penalizing the generator with the difference in absolute sum_E – the (normalized to 1) energies in both cases have at least an order of magnitude agreement (same for penalizing the critic) .

```
Means: fake: 0.0006203119410201907, real: 0.0005407848511822522. Difference(%): 14.705864941312269
Std: fake: 0.01062756683677435, real: 0.009805705398321152. Difference(%): 8.38146166000367
Sum of all values: fake: 609.7914428710938, real: 531.6131591796875, Difference (%): 14.705859390696848
```

IDEA #2: Including Conservation: Results

- Next, included difference between total number of hits as well, and penalized the critic.
- Again, an order-of-magnitude similarity is evident.
- Cutoff was 5% of maximum hit energy.

```
Loaded Checkpoint
Total Number of real hits: 5460, with total energy: 66044.8828125
Total Number of fake hits: 4321, with total energy: 51328.16796875
Mean of fake hit_E:11.87877082824707, Mean of real hit_E: 12.09613227844238З
Energy/event(fake):401.0013122558594, Energy/event(real): 515.9756469726562
Std. of fake hit_E:9.416844367980957, Std. of real hit_E:10.957155227661133
```

IDEA #3: Going 3D

- To get a better idea of the spatial correlations in the cell, we tried to work with ConvTranspose3D and Conv3D layers in a simple GAN.
- Penalized *relative* differences in sum_E and nhits.
- Did not work as well as the previous ones (not even close!), even qualitatively, besides being slower.

IDEAS #4 & #5: Some not-so-clever ideas

- Using a DNN to map (layer, x, y) -> hit_E.
  - Too smooth an interpolation.
- Using PCA for dimensionality reduction (dealing with sparsity)
  - Matrices obtained are too random and have no discernable structure.

Lessons Learnt

- Sparsity is a big issue which is not handled well by conventional models, which require and work with “smoothness” in some sense. (Side Note: This *may* work better for high-energy particles, since the issue of sparsity may be partly mitigated by their energies).
- The curse of dimensionality is real; but naive dimensionality reduction is not the solution.
- Simple networks are too smooth; we *need* sophistication for the problem.

Related Work

- Sebastian Bieringer, Erik Buhmann, Sascha Diefenbacher et. al. - working on fast simulation of showers in highly-granular calorimeters, see talk by Peter McKeon:

  https://agenda.linearcollider.org/event/9800/contributions/51163/attachments/38337/60227/ILD_SFT_31_08_22.pdf

Future Work

- More sophisticated models (such as graph networks) are needed.
- Attention Mechanisms may be useful in learning inter-layer correlations better. Both this, and graph networks have been tried in past work by some people, to reasonable success.
- Other possible methods include state-of-the-art techniques such as diffusion models and Normalizing Flows.