CERN openlab

High Granularity Calorimeter Simulation using Generative Adversarial Networks

3D Generative Adversarial Networks for High Energy Physics Calorimeter Simulation

Gulrukh Khattak Federico Carminati Sofia Vallecorsa

17/4/2019

Overview

- 3DGAN
  - Dataset
  - Architecture
  - Detailed analysis
- Results
  - GAN generated events
  - Physics related quantities
  - Cell energies
  - Sparsity
  - Correlations
  - Optimization function
  - Image Quality tests
- Future Plans
  - Evolutionary Approach
  - Challanges
  - Implementation

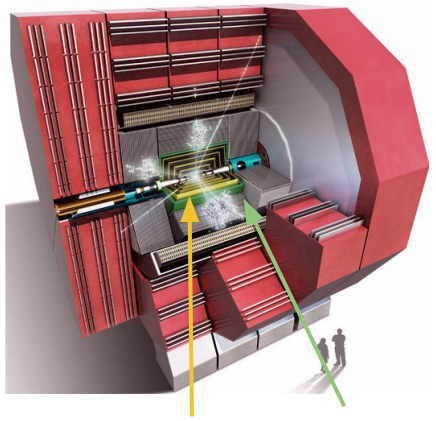
3DGAN

*Three dimensional **G**enerative **A**dversarial **N**etworks[1]*

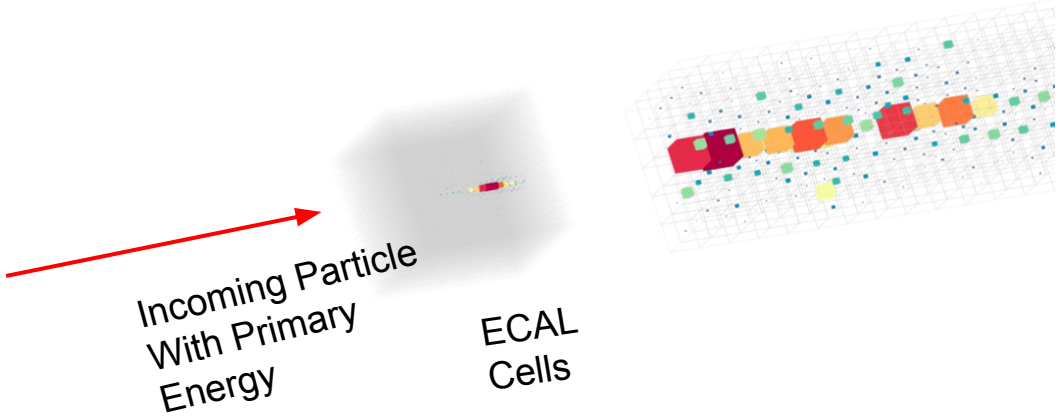
Data set

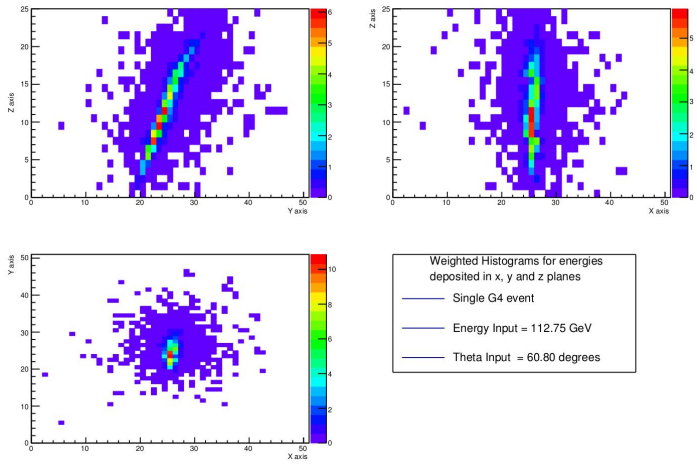
CLIC Calorimeter

- Compact Linear Collider CLIC: **Proposed** linear particle accelerator [2]
- **Open data set** developed for ML applications [3]: Events as selected cells around the barycenter of particle showers simulated using Geant4 [4]
- Fixed angle, Primary particle energy 10-500 GeV (electrons)
  - Event → 25 x 25 x 25 image → 15, 625 cells
    - 200,000 events
- Variable angle (electrons)
  - Event → 51 x 51 x 25 image → 65, 025 cells
    - 120,000 events from 100 to 200 GeV primary energy
    - 400,000 events from 2 to 500 GeV
- Detector response as 3D images
  - Images are sparse
  - Intensities cover a large spectrum over seven orders of magnitude

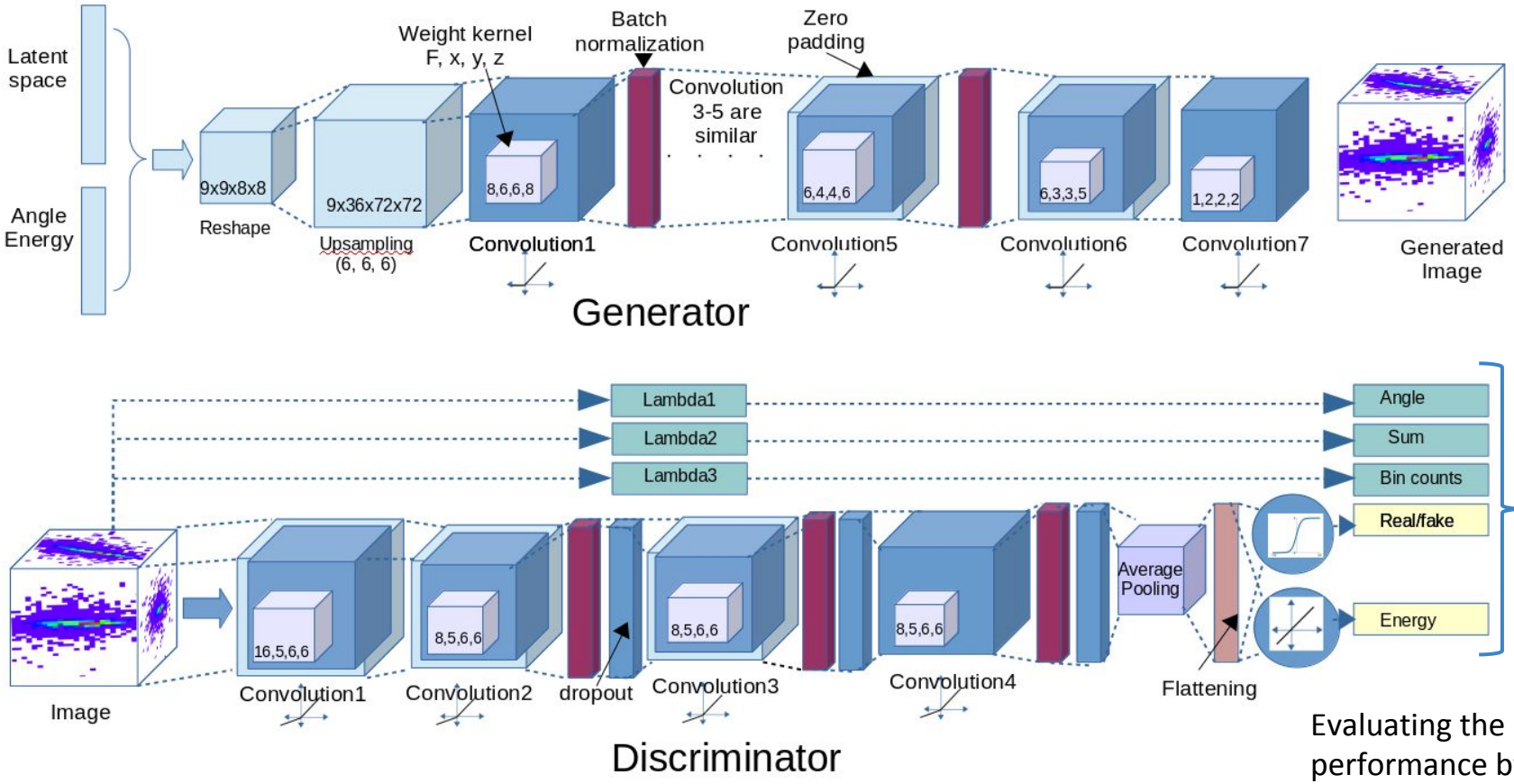


http://clicdp.web.cern.ch/





Architecture



Generator Params 1066 k
Discriminator Params = 87 k

Evaluating the performance by agreement to labels and Physics related constraints

"Caltech ibanks GPU cluster thanks to Prof M. Spiropulu"

Detailed Analysis

Evaluating and tuning performance

- Multiple criteria
- Detailed GAN vs GEANT4 comparison (More than 200 Plots!) for multiple features:
  - Shower Shapes
  - Sampling Fraction
  - Discriminator Primary Energy regression (dense layer)
  - Position of max energy deposition
  - Hits above a threshold (0.0003 GeV)
  - Fraction of energy deposited in different parts of the shower
  - Angle measured from shower
  - Discriminator real/fake probabilities (dense layer)
  - Shower moments
  - Sparsity
  - Cell energy histograms
  - Correlation between different quantities
  - Projections in x, y, z planes to access visually
- Image Quality Analysis

Results

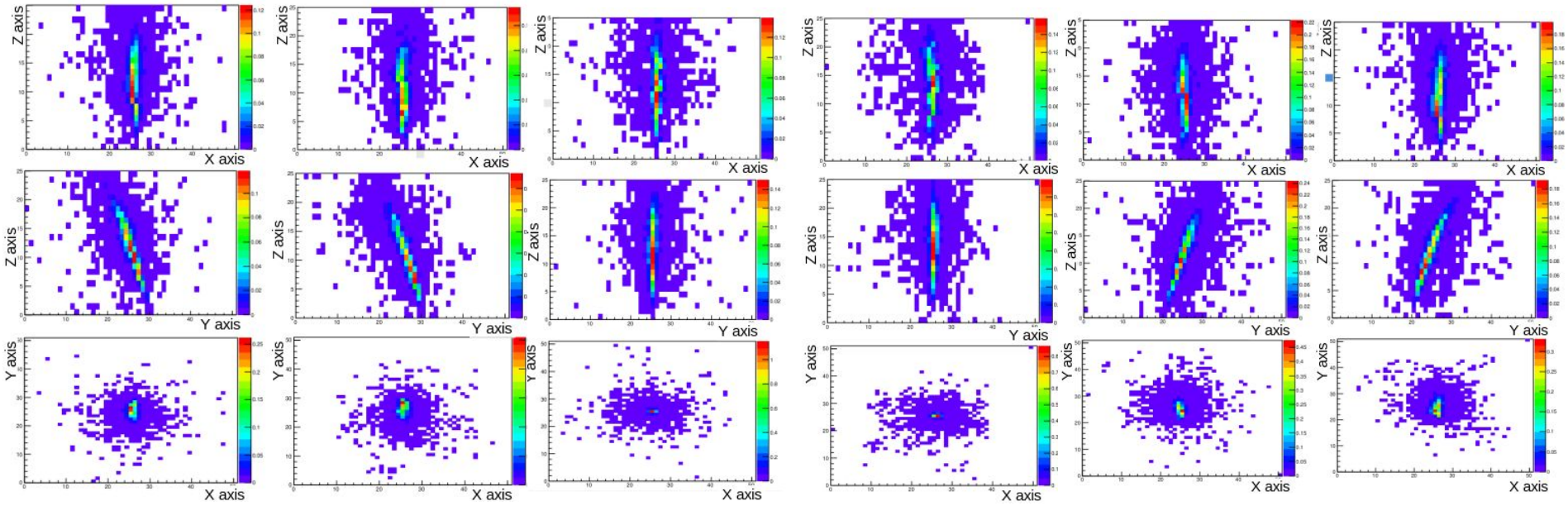
100-200 GeV Primary Energy and 60-120 Degrees angle

GAN generated events

100-200 GeV Primary Energy and 60-120 Degrees angle

- *G4 vs. GAN events 2D histograms for energy deposited in x, y, z planes* with same Primary Particle energy and angle

111.07 GeV and 115.54 Degrees	147.49 GeV and 87.83 Degrees	188.95 GeV and 62.96 Degrees

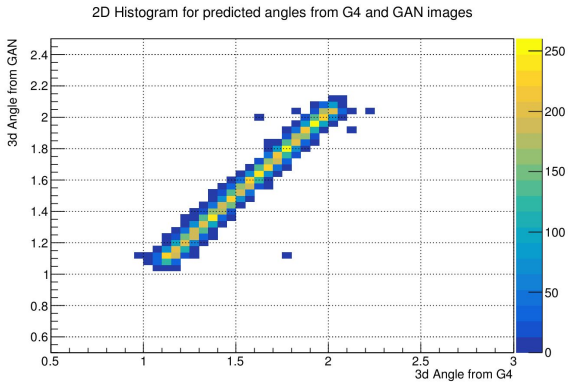


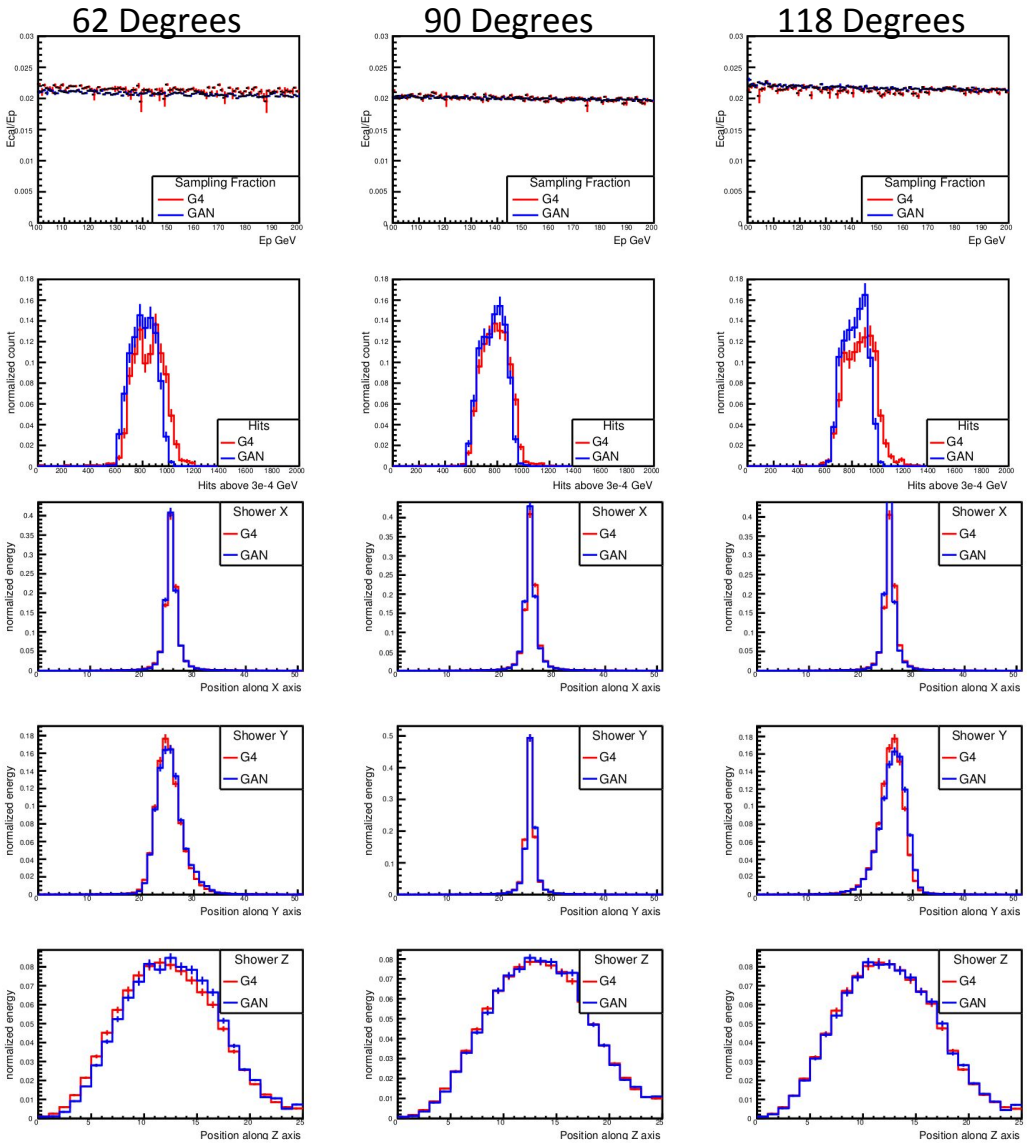
G4 GAN G4 GAN G4 GAN

Physics Quantities

For primary particle energy 100-200 GeV and angle in bins around 62, 90 and 118 Degrees

- Sampling Fraction
- Hits
- Shower Shapes:
  - Energy deposited along x, y and z axis
- Measured Angle

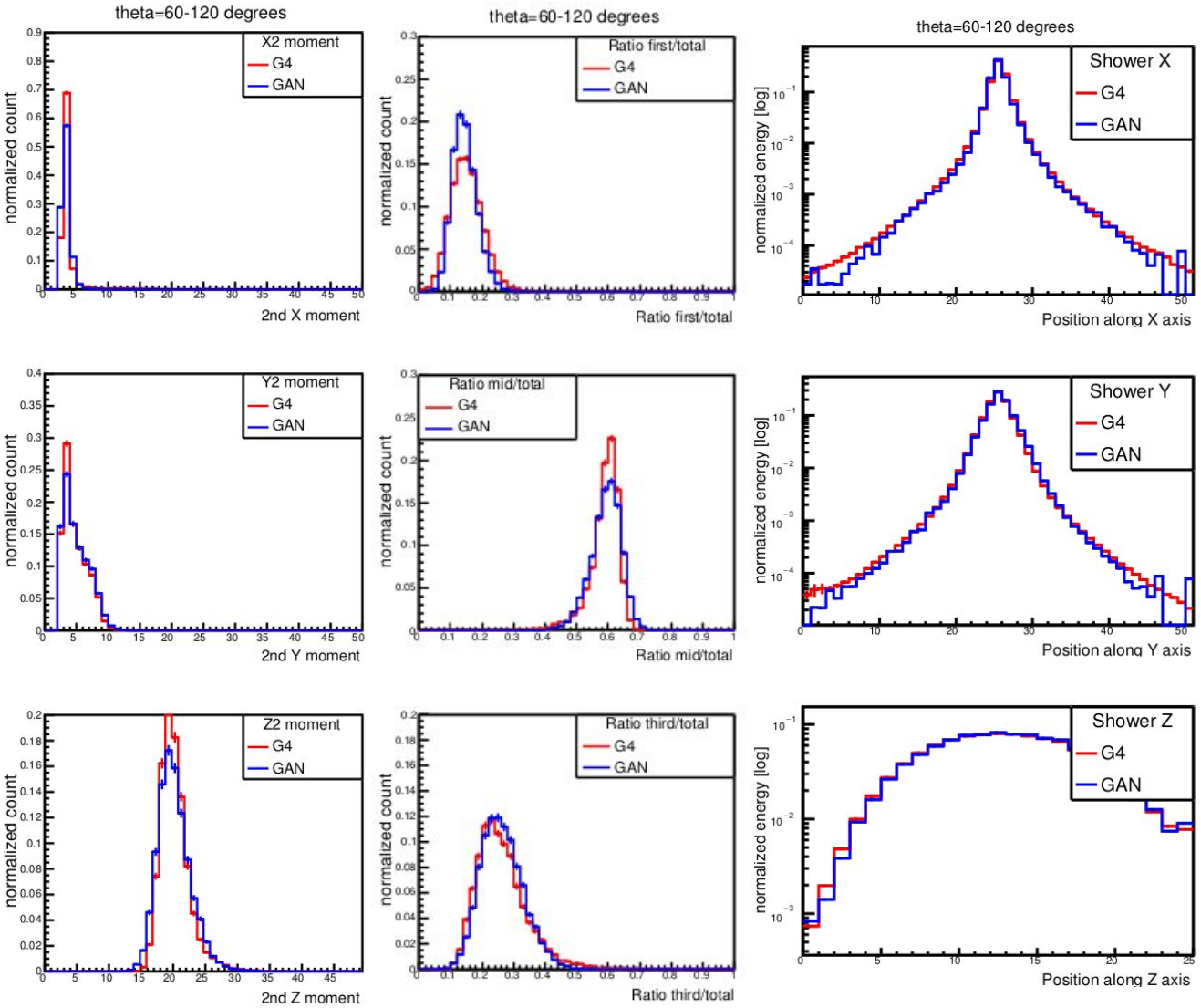




Physics Quantities

Primary particle energy 100-200 GeV and angle 60-120 degrees

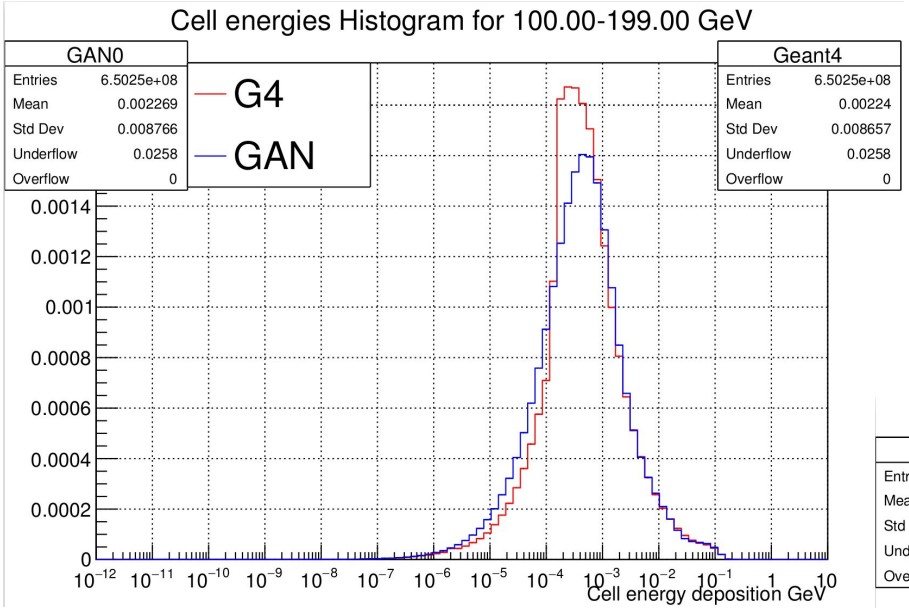
- Shower width
- Fraction of energy deposited in first, second and third part of shower to total
- Shower shapes (log)

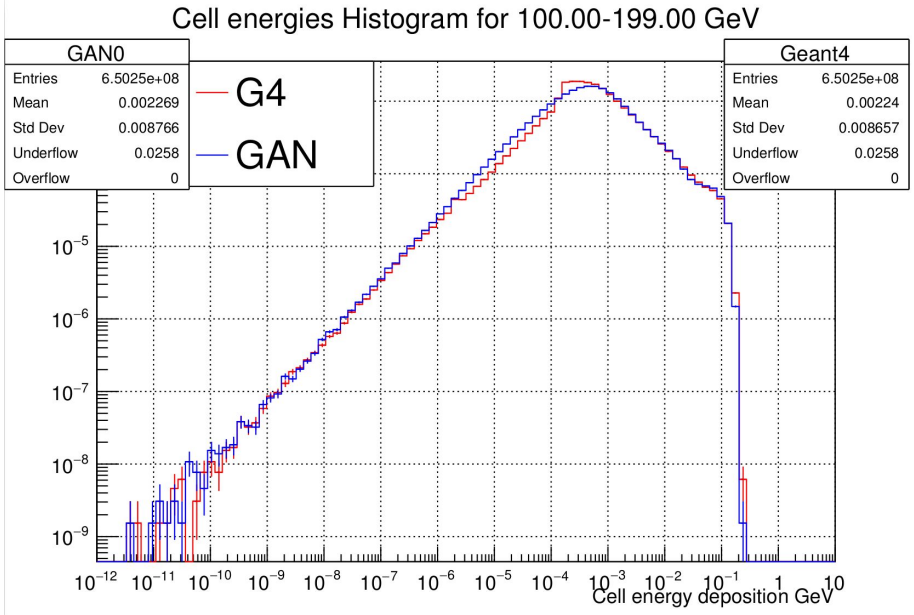


CERN openlab

Cell energies

Energy deposited in individual cells



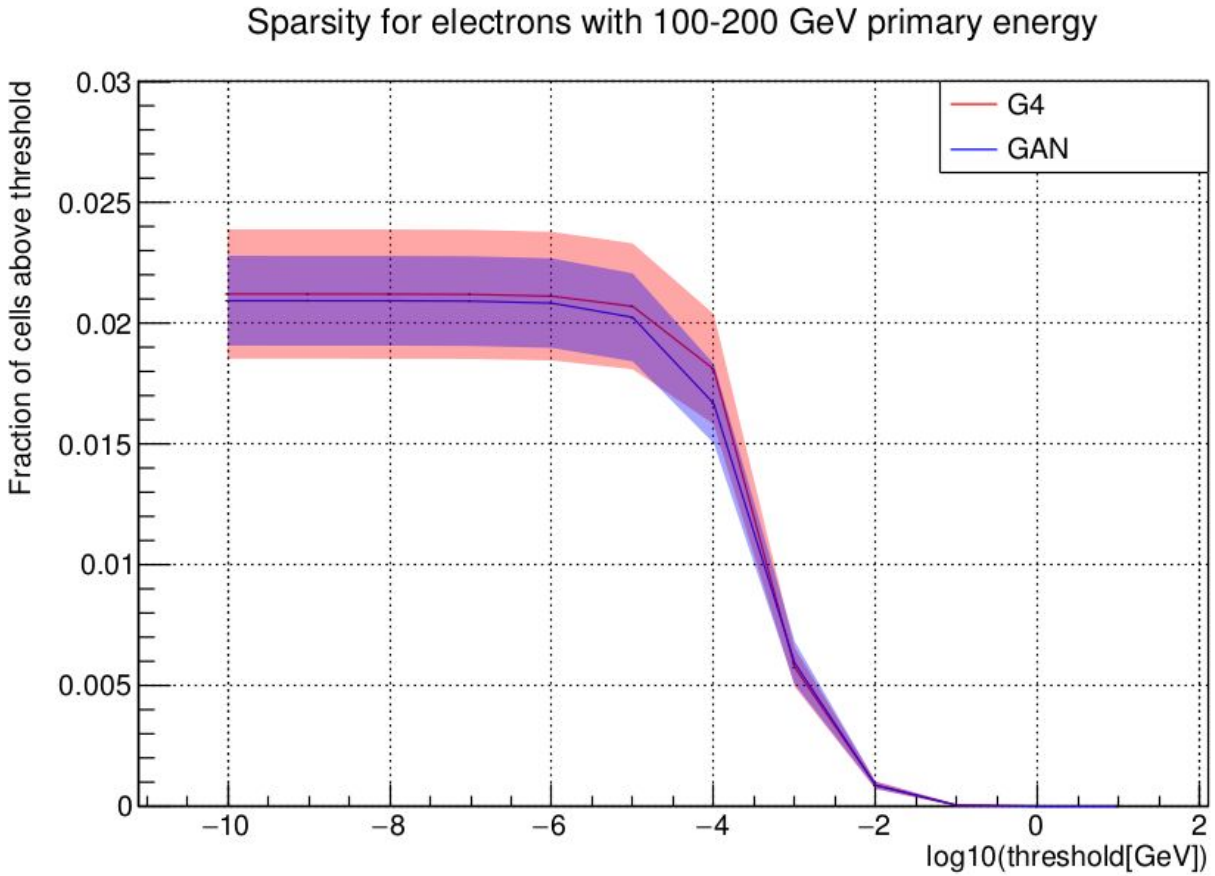


Reasonable agreement to G4 across seven orders of magnitude

Sparsity

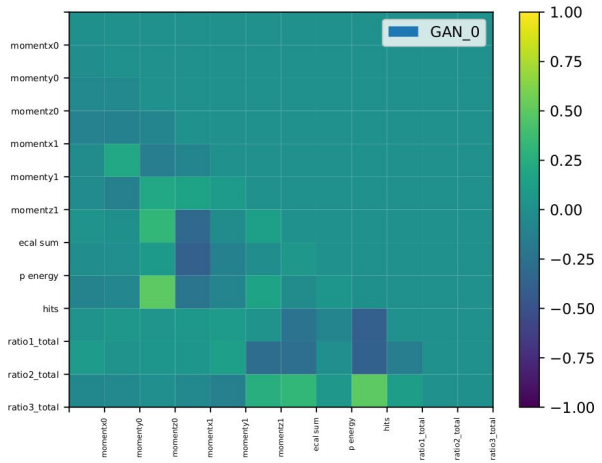
GAN vs. G4

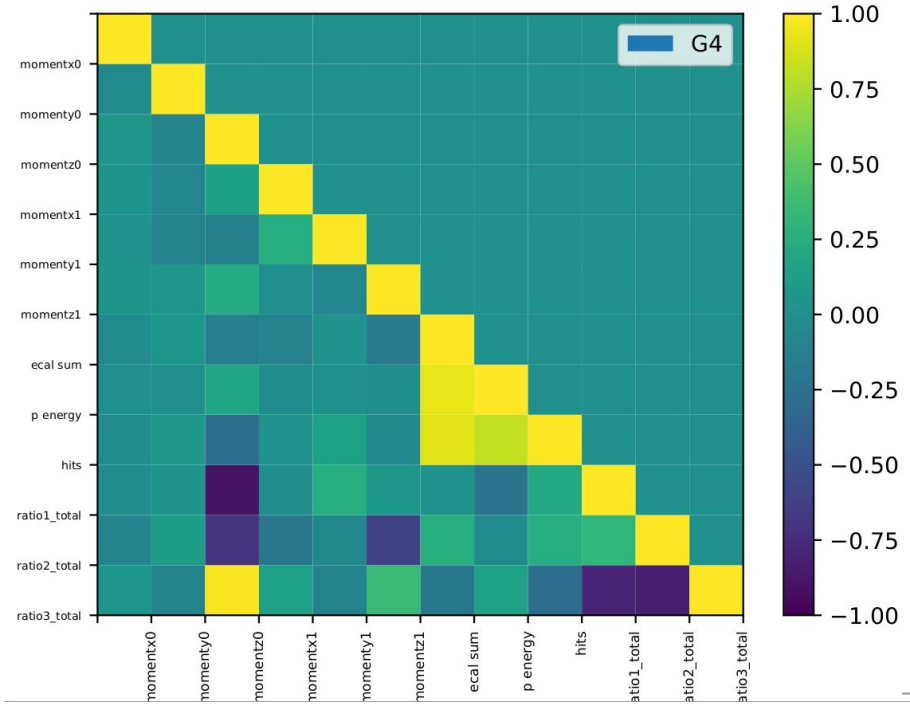
GAN generated images are less sparse but are compatible within error bars

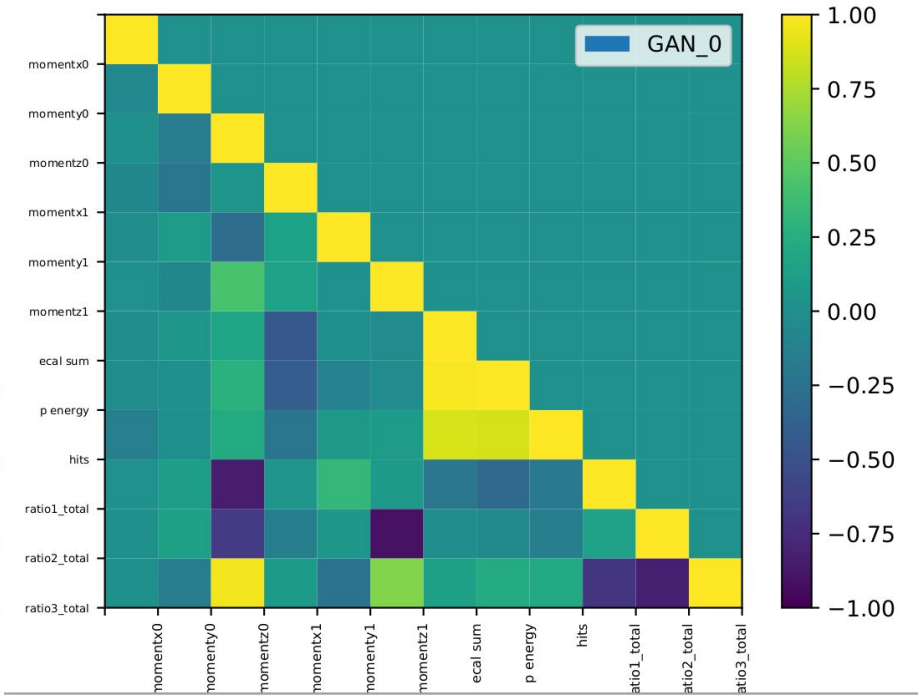


Correlations

G4 vs. GAN







Optimization Function

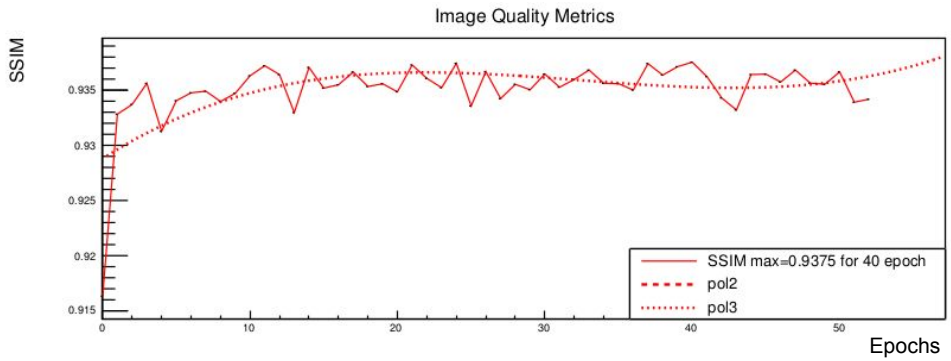
Validation

- GAN is a Minmax game with multiple objectives thus it is very difficult to assess performance on the basis of losses
- A **single figure of merit** to assess performance
  - Compare results of different hyperparameter settings
  - Assess best weights from training results
- Figure of merit to takes into account mean relative errors for:
  - Shower Shapes
  - Moments
  - Measured angles
  - Sampling Fraction
- Wasserstein Distance between these quantities (work in progress)
  - Wasserstein distance for more than one dimension is NP hard
  - Gromove Wasserstein is being investigated

Image Quality Analysis

Structural Similarity Index

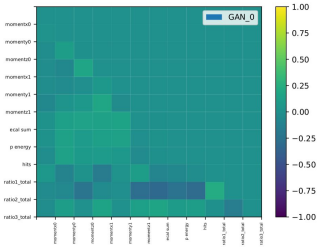
- Structural Similarity Index or **SSIM [5]** is used to assess **similarity** between images commonly used in denoising applications
- For **GAN** SSIM has also been used to measure **diversity** between generated images
- SSIM was computed for images from same energy and angle bin:
  - SSIM **G4** to **G4** = ~0.9445
  - SSIM **GAN** to **GAN** = ~0.9507
  - SSIM **G4** to **GAN** = ~0.9495

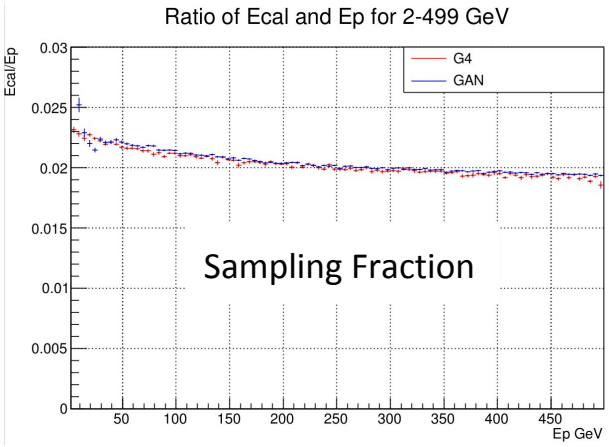


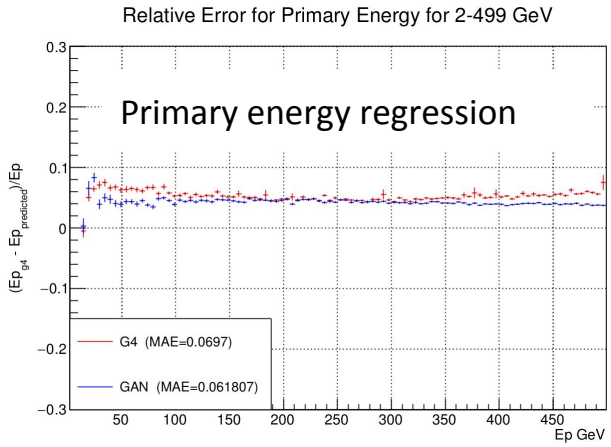
SSIM as training progresses

Transfer Learning

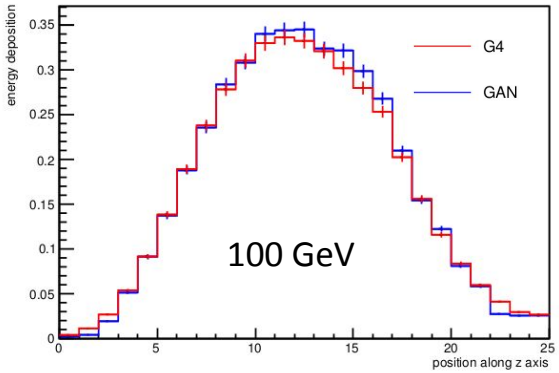
- Training for 2-500 GeV spectrum
  - Starting from pretrained weights (trained for 100-200 GeV)

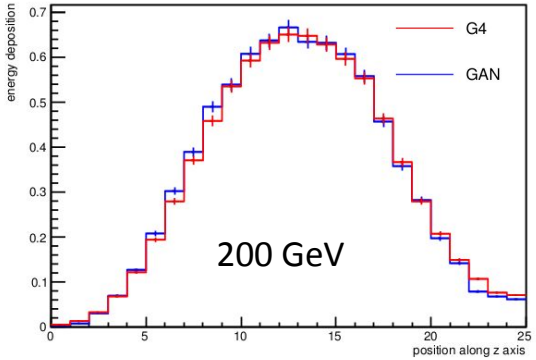


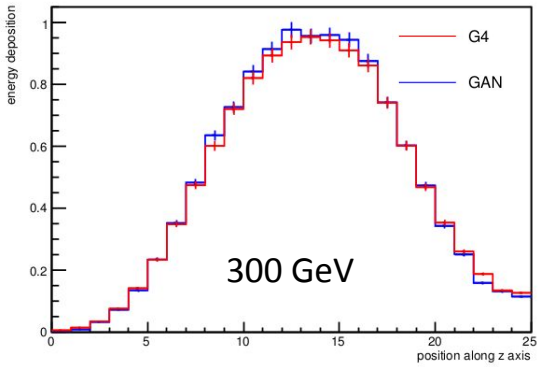


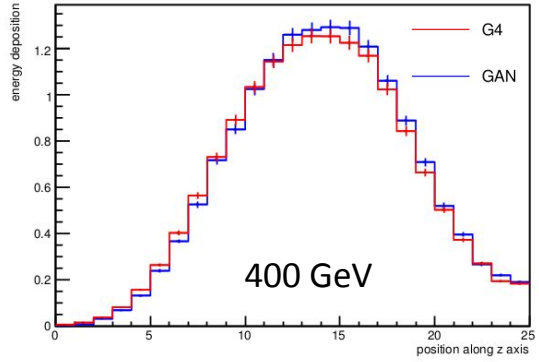


Shower shapes in the longitudinal direction for Different Primary energies









Future Plans.....

Generalization

Generalize 3DGAN

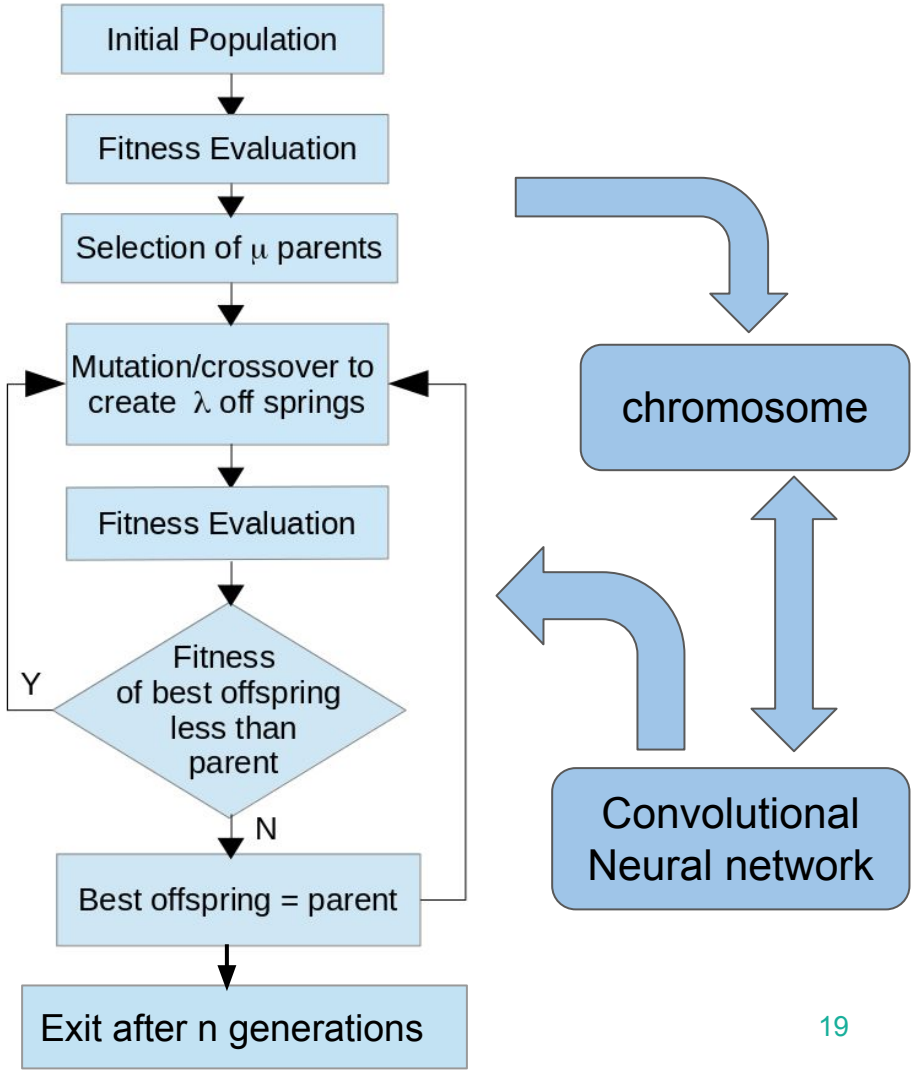
Generalization Challenges

- Generalize the approach so that 3DGAN can be trained and tuned automatically to data from different types of calorimeter
- Challanges
  - Potentially large number of possible configuration
    - Training related Hyper parameters:
      - e.g learning rates, loss weights, optimizer, batch size, latent size etc.
    - Architecture related parameters:
      - Number of layers, number of filters, filter sizes, use of dropout, batch normalization, pooling and upsampling
  - **Long training times**
- Proposed approaches:
  - **Hyper-parameter** scan using distributed training [5]
  - Training and Optimization at the same time using **evolution**

Evolutionary Approach

Genetic Algorithm to train and optimize neural networks at the same time

- **Genetic Algorithm** will be used to train a Neural Network
- **Generic Tool** for different detectors
- **Global** instead of local minima
- **Complex and indirect cost** functions are possible
- **Highly Scalable**



Challenges

GA for GAN

- Network Size:
  - Trainable parameters in **millions** for 3DGAN model
- Big Data
- Required Resources:
  - **HPC resources** will be essential.
- Architecture Optimization:
  - Flexible and stable
- Adversarial Training:
  - Simultaneous training of **two networks**
- Inexact solution:
  - A **hybrid approach** can incorporate SGD as a callback

Implementation

Step by step implementation

- This approach will be implemented in **three phases**:
- Initially we will reduce the problem complexity and focus on genetic training by:
  - Converting **3D data to 2D** images, slicing the detector volume along the direction of particle propagation.
  - focusing on the **discriminator** network only
  - **Fixed** architecture
- During a second phase architecture hyper-parameters will also be encoded in chromosomes and optimised
  - Test **indirect encoding** for weight encoding to reduce the number of parameters and stabilize training.
- Finally the complete GAN scenario will be implemented.

Conclusions

3DGAN

- Showers were generated conditioned on primary particle **energy** and **angle**.
- Physics performance within few percent of Geant4 simulation
- **Transfer learning** successfully demonstrated for entire energy spectrum
- **Investigation of different approaches for generalization……**

Thank you !!!

References

[1] I. J. Goodfellow, et al. Generative Adversarial Networks. ArXiv e-prints , June 2014.

[2] CERN Accelerating Science, https://home.cern/ , accessed 27.10.18.

[3] M. Pierini, “The linear collider detector dataset,” https://indico.hep.caltech.edu/indico/getFile.py/access?contribId=16&sessionId=9&resId=0&materialId=slides&confId=102, accessed 27.10.18

[4] S. Agostinelli et al. GEANT4: A Simulation toolkit. Nucl. Instrum. Meth. , A506:250–303, 2003.

[5] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. “Image quality assessment: from error visibility to structural similarity “. *Trans. Img. Proc.* 13, 4 (April 2004), 600-612. DOI=http://dx.doi.org/10.1109/TIP.2003.819861

[6] S. Vallecorsa, D. Moise, F. Carminati. IEEE HiPC 2018.

Bonus Slides

SSIM for 3DGAN

Structural Similarity Index

$$SSIM = (2 \mu_x \mu_y + c_1)(2 \sigma_{xy} + c_2) / (\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)$$

for two windows from image x and y with $\mu_x \mu_y$ being means and $\sigma_x \sigma_y$ being variances from x and y

$c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$

k1 = 0.01, k2 = 0.03

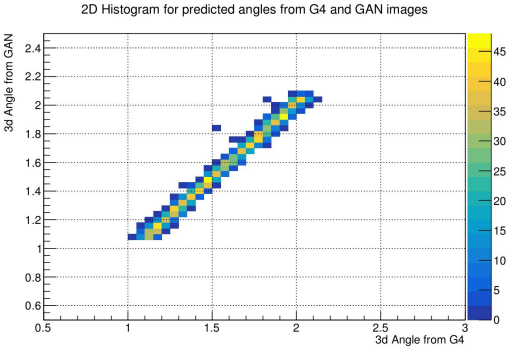
L= dynamic data range for integer values and set as 1 for float and double intensities [7]

SSIM for Different values of L

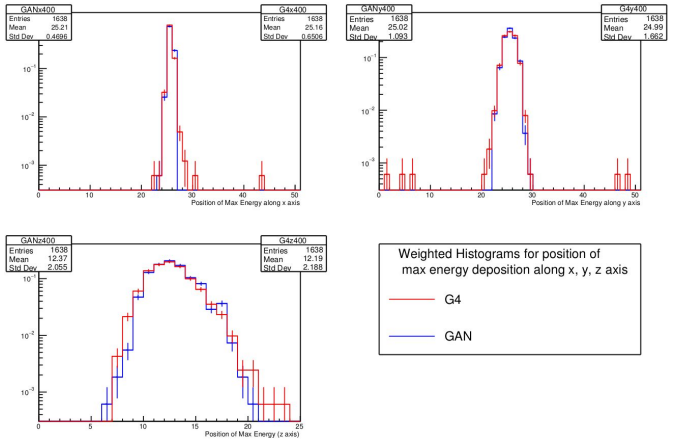
No.	L	G4 vs. G4	GAN vs. GAN	G4 vs. GAN
1	1	0.9445	0.9507	0.9495
2	1e-2	0.2133	0.2542	0.2285
3	1e-4	0.0454	0.0611	0.0465
4	1e-6	0.0453	0.0517	0.0457

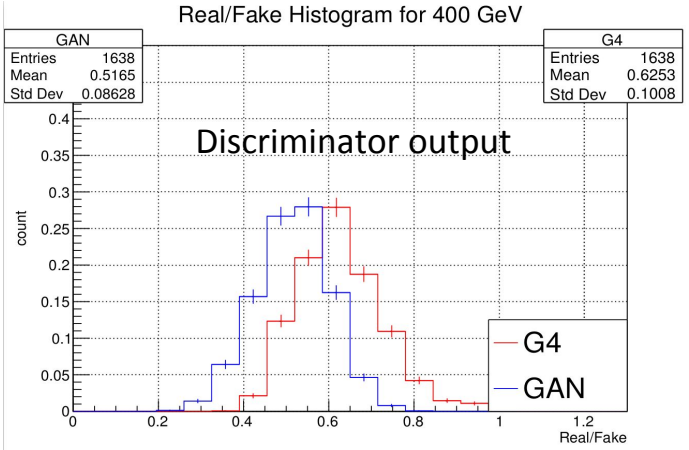
Additional Plots

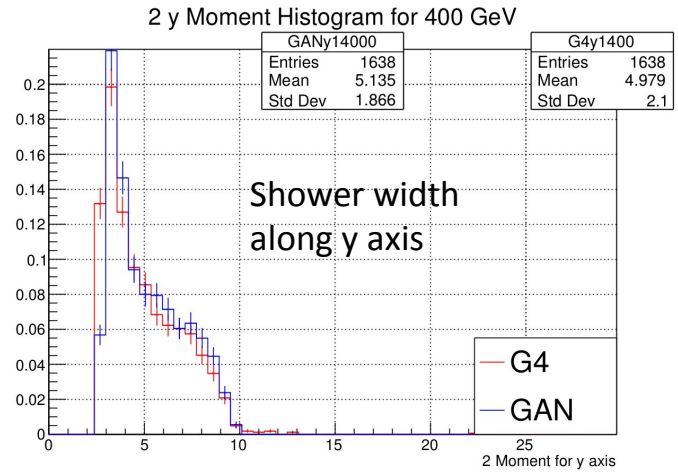
400 GeV

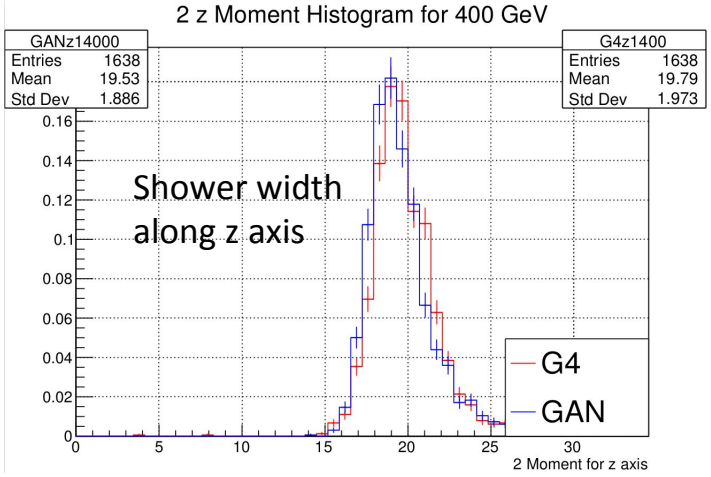


Position of maximum energy deposition









Timing

Fixed Angle

- Fixed Angle:
  - Inference is four order of magnitude faster
  - Training time < 1 hour GTX 1080
- Variable Angle:
  - Training time 2-3 hours GTX 1080

Time to create an electron shower for Fixed angle		
Method	**Machine**	**Time/Shower (msec)**
Full Simulation Fixed Energy (Geant 4)	Intel Xeon Platinum 8180	17000
3DGAN (Fixed Angle) (batch size 128)	Intel Xeon Platinum 8180(TF 1.12)	1
	GTX 1080	0.1
3DGAN (Variable Angle) (batch size 64)	GTX 1080	4.6

Challanges

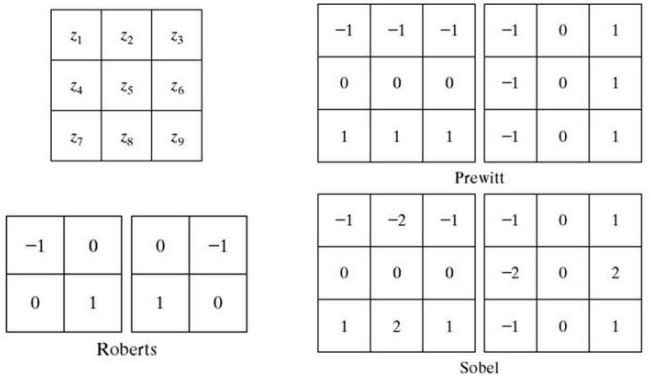
GA for GAN

- Network Size:
  - Trainable parameters in **millions** for 3DGAN model
  - Deep GA [8] has been able to train successfully over four million parameters
- Big Data:
  - Large data for evaluation on entire training data.
  - **Batch training** has been successfully implemented LEEA [9].
- Required Resources:
  - Due to network and data sizes **HPC resources** will be essential.
- Architecture Optimization:
  - Design mechanism to offer greater flexibility
  - **Non conventional connections** between convolutional filter nodes EXACT [10].
  - I**ndirect Weight encoding** similar to HyperNEAT [11]
- Adversarial Training:
  - Simultaneous training of **two networks** competing with each other
  - Evolutionary approach is frequently used in **multiplayer games**
- Inexact solution:
  - Evolutionary approach can discover **neighbourhood of global optima** but cannot arrive at **exact solution** of a local search
  - A **hybrid approach** can incorporate SGD as a callback after no improvement for predefined number of generations to help arrive at exact solution.

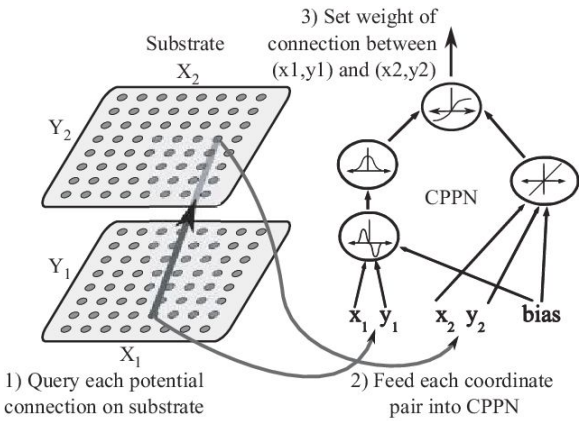
Indirect Weight Encoding

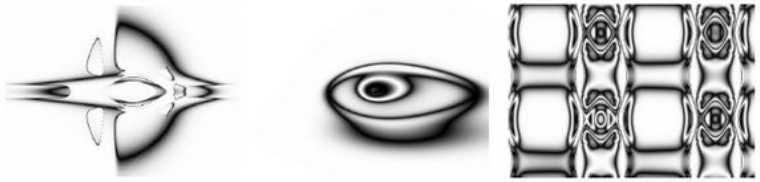
Evolving Filters

- Convolutional Neural networks use the idea that filter kernels can be used to extract interesting features from an image.
- Digital filters can perform operations like smoothing and edge detection etc.
- Image is convolved with a filter kernel
- The weights in a kernel are meaningful in their relation with other weights and its position
- If the filter size is changed the trained weight is not meaningful any more
- Thus instead of direct coding, indirect coding can be used. Thus the underlying relation of weights will be learnt instead of fixed values
- **Hyper Neat** uses indirect coding for neural network evolutionary training as Compositional Pattern Producing Networks **CPPNs**
- Same concept can be extended to CNN



Some commonly used edge detection filters





HyperNEAT CPPN

Asynchronous Update

Genetic Training

- A pool of random parent chromosomes is created
- Fitness for each parent is evaluated
- Offsprings are obtained (mutation and/or cross over)
- Each thread evaluates a single child
- As soon as child is evaluated the fitness is compared with lowest fitness of current pool
  - If child fitness is same or higher the child replaces the parent with lowest fitness
- Different children can take different time for evaluation
- No need for communication between threads
- There will be no loss in performance with number of threads

References

[7] Matlab, SSIM, https://fr.mathworks.com/help/images/ref/ssim.html

[8] F. Petroski Such. et. al. “Deep Neuroevolution: Genetic Algorithms Are a competitive Alternative for Training Deep Neural Networks for Reinforcement Learning”. 2018. arXiv:1712.06567

[9] G. Morse, K. O. Stanley. “Simple Evolutionary Optimization Can Rival Stochastic Gradient Descent in Neural Networks”. GECCO 2016

[10] T. Desell. “Large Scale Evolution of Convolutional Neural Networks Using Volunteer Computing”. IEEE eScience. 2017.

[11] K. O. Stanley, D. D’ Ambrosio, J. Gauci. “A Hypercube-Based Indirect Encoding for Evolving Large-Scale Neural Networks” Artificial Life Journal, MIT Press, 2009.

[12] Odena, et al., "Deconvolution and Checkerboard Artifacts", Distill, 2016. http://doi.org/10.23915/distill.00003