Prepared for submission to JHEP

CaloGAN: Simulating 3D High Energy Particle Showers in Multi-Layer Electromagnetic Calorimeters with Generative Adversarial Networks

Michela Paganini[a,b], **Luke de Oliveira**[a], **and Benjamin Nachman**[a]

[a] *Lawrence Berkeley National Laboratory, 1 Cyclotron Rd, Berkeley, CA, 94720, USA*

[b] *Department of Physics, Yale University, New Haven, CT 06520, USA*

E-mail: michela.paganini@yale.edu, lukedeoliveira@lbl.gov, bnachman@cern.ch

Abstract: Simulation is a key component of physics analysis in particle physics and nuclear physics. The most computationally expensive simulation step is the detailed modeling of particle showers inside calorimeters. Full detector simulations are too slow to meet the growing demands resulting from large quantities of data; current fast simulations are not precise enough to serve the entire physics program. Therefore, we introduce CaloGAN, a new fast simulation based on generative adversarial neural networks (GANs). We apply the CaloGAN to model electromagnetic showers in a longitudinally segmented calorimeter. This represents a significant stepping stone toward a full neural network-based detector simulation that could save significant computing time and enable many analyses now and in the future. In particular, the CaloGAN achieves speedup factors comparable to or better than existing fast simulation techniques on CPU (100×-1000×) and even faster on GPU (up to $\sim 10^5\times$)) and has the capability of faithfully reproducing many aspects of key shower shape variables for a variety of particle types.

Contents

1 Introduction

The physics programs of all of the experiments based at the Large Hadron Collider (LHC) rely heavily on detailed simulation for all aspects of event reconstruction and data analysis. Simulated events are used to interpret the results of ongoing experiments and estimate the performance of new ones, including upgrades. For example, analysis-specific simulation samples, selected through criteria that maximize a specific signal yield, must be produced to correctly estimate the expected limits on the cross-section of a given physics process under investigation. The production of physics results is often limited by the absence of adequate Monte Carlo (MC) simulation, and the increase in luminosity at the LHC will only exacerbate the problem. For example, the ATLAS and CMS experiments at the high-luminosity LHC (HL-LHC) will each see about 3 billion top quark pair events [1–7]; for a MC statistical uncertainty that is significantly below the data uncertainty, tens or hundreds of billion simulated events would be required. This is not possible using full detector simulation techniques with existing computing resources.

State-of-the-art simulations are able to precisely model detector geometries and physics processes spanning distance scales as small as 10^{-20} m for the initial parton-parton scattering, all the way to the material interactions at meter length scales. The description of particle showers in calorimeters, with Geant4 [8] as the state-of-the-art, is the most computationally demanding part of the whole simulation process. This part of the simulation can take minutes per event on modern, distributed high performance platforms [9, 10]. Currently, full MC simulation occupies 50-70% of the experiments' worldwide computing resources, equivalent to billions of CPU hours per year [11–13]. The relevance of this step has sparked the development of approximate, fast simulation solutions to mitigate its computational complexity. Fast simulation techniques rely on parametrized showers [14–16] for fluctuations and possibly look-up tables for low energy interactions [17]. For many applications, these techniques are sufficient. However, analyses that utilize the detailed structure of showers for particle identification as well as energy and direction calibration may not be able to rely on these simplified approaches [18].

We introduce a Deep Learning-based framework to enable high-fidelity fast simulation of particle showers in electromagnetic calorimeters. Previous work [19] assessed the viability of GAN-based simulation of *jet-images* [20] – sparse, structured, 2D representations of jet fragmentation analogous to a single-layer, idealized calorimeter – and focused on providing architectural guidelines for this regime. Neural network-based generation, including GANs, Variational Auto-Encoders [21], and Adversarial Auto-Encoders [22], have also been tested in other areas of science, such as Cosmology [23, 24], Condensed Matter Physics [25], and Oncology [26]. Longitudinally segmented calorimeter simulation offers unique challenges due to the sparsity of hit cells, the non-uniform granularity, and the sequential structure between layers. In addition to enabling physics analyses at the LHC, the CaloGAN may form a base for solving problems in other domains of science, medicine, and technology.

The paper is organized as follows. Section 2 introduces the dataset of calorimeter showers and Sec. 3 briefly reviews the generic GAN setup. The CaloGAN is described in Sec. 4 and first results of its performance are documented in Sec. 5. The paper ends with conclusions and future outlook in Sec. 6.

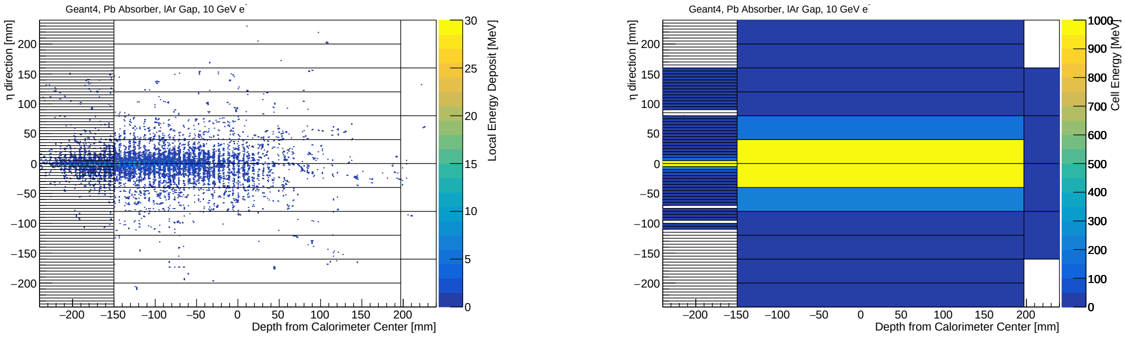
2 Dataset

A detector simulation begins with a list of particles with lifetimes greater than $\mathcal{O}(\mathrm{mm}/c)$. For each particle, we are given its type (e.g. electron, pion, etc.), its energy, and its direction. The particle type will determine when and how the particle will interact with the material along its trajectory. Material interactions with the detector factorize[1]: the energy deposited in a calorimeter by various particles is the sum of the energy from each shower treated independently.

There are two flavors of calorimeters: electromagnetic and hadronic. Electromagnetic calorimeters are designed to stop electrons and photons, which have shallower and narrower showers compared with protons, neutrons, and charged pions. Hadronic calorimeters are thicker and deeper in order to capture penetrating radiation that forms irregular showers from nuclear interactions. As this is the first application of GANs to a longitudinally segmented calorimeter, we will focus only on electromagnetic showers. In addition to already providing the capability to simulate electrons and photons, the electromagnetic shower contains all of the new challenges described in Sec. 1.

Transverse segmentation is critical for particle identification and energy calibration in an electromagnetic calorimeter. For example, the radiation pattern can be used to distinguish prompt photons from $\pi^0 \to \gamma\gamma$, where the distance between the two photons is $\mathcal{O}(\mathrm{cm})$ for a 10 GeV π^0 at one meter

[1] Energy losses factorize, but detector readout does not. Due to threshold and digitization effects, the energy readout from two energy deposits in different detector elements need not be the same as the recorded energy from the two deposits in the same element. In detector simulations, these non-linear effects are treated after accounting for the material interactions and are therefore beyond the scope of the CaloGAN. It may be interesting in future work to consider an end-to-end generator that includes these effects, but it may not save a lot of time since simulation is much more costly than reconstruction.



(a) Each dot represents one energy deposit from Geant4 and the color of the dot encodes the energy. The absorber-gap structure is clearly visible, where most of the energy is lost in the absorber.

(b) Discretized version of (a), in which energy depositions are assigned to individual, discrete detector cells.

Figure 1: The electromagnetic shower from one 10 GeV electron event. The boundaries of the cells are shown, projecting out the ϕ segmentation.

from the interaction point. Pion rejection and an excellent resolution for photons in the Higgs boson $H \rightarrow \gamma\gamma$ discovery channel were driving factors for the design of the ATLAS Liquid Argon (LAr) electromagnetic calorimeter [27], which will serve as an inspiration for the calorimeter used in this study. In particular, our calorimeter is a cube with size 480 mm^3 with no material in front of it. There are three instrumented layers in the radial (z) direction[2] with thicknesses 90 mm, 347 mm, and 43 mm. The active material is LAr and the absorber material is lead. In contrast to the complex accordion geometry in the actual ATLAS calorimeter, our simplified setup (built on the Geant4 B4 example) uses flat alternating layers of lead and LAr that are 2 mm and 4 mm thick, respectively. Each of the three layers have a different segmentation, which is also not square in the first and third layers. In particular, the cells in the first layer are 160 mm × 5 mm, the cells in the second layer are 40 mm^2, and the cells in the third layer are 40×80 mm^2. The short direction in the first layer (η) corresponds to what would be the pp beam direction in a full experiment. In contrast, the short direction in the third layer (ϕ) is perpendicular to η. Table 1 summarizes the calorimeter geometry.

Layer	z segmentation [mm]	η segmentation [mm]	ϕ segmentation [mm]
0	90	5	160
1	347	40	40
2	43	80	40

Table 1: The dimensions of the calorimeter. The z direction is the direction of particle propagation (radial direction in a full experiment), the η direction would be along the pp beam axis in a full experiment, and ϕ is perpendicular to z and η.

The training dataset is prepared as follows. Geant4 10.2.0 [8] is used to generate particles and simulate their interaction with our simplified calorimeter using the Ftfp_Bert physics list

[2] This is the direction that prompt neutral particles at $\eta = 0$ would enter the calorimeter without any prior material interactions. See Fig. 2.

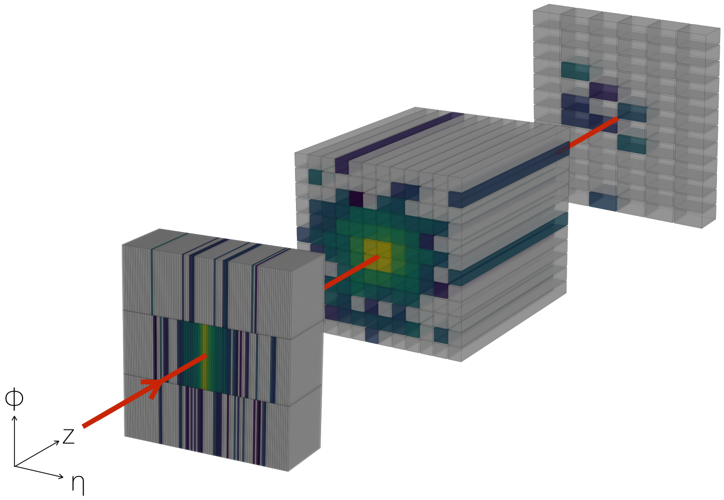


Figure 2: Three-dimensional representation of a 10 GeV e^+ incident perpendicular to the center of the detector. Not-to-scale separation among the longitudinal layers is added for visualization purposes.

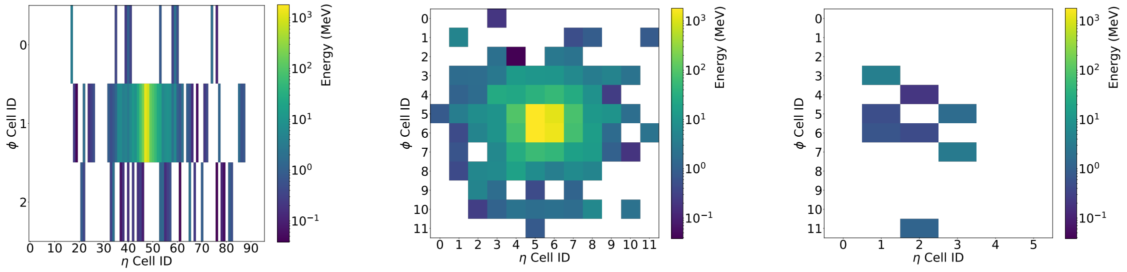


Figure 3: Two-dimensional, per-layer representation of the same shower as in Fig. 2.

based on the Fritiof [28–31] and Bertini intra-nuclear cascade [32–34] models with the standard electromagnetic physics package [35]. Positrons, photons, and charged pions with various energies are incident perpendicular on the center of the calorimeter front. Energies in the training are uniform in the range between 1 GeV and 100 GeV. Fig. 1 shows an example 10 GeV electron event with the exact energy deposits from Geant4 (Fig. 1(a)) and after descretizing them according to our calorimeter geometry (Fig. 1(b)). For visualization purposes, a 3-dimensional particle energy signature (Fig. 2) will be displayed in the rest of this paper as a series of three 2D images in η - ϕ space (Fig. 3), where the pixel intensity represents the sum of the energies of all particles incident to that cell[3]. The first

[3] For the purposes of this study, the cell energy is the sum of the energy deposited in the lead and the argon; in practice, only the LAr energy deposits are measured. Dividing out these two components is left for future work (see Sec. 6).

layer can be represented as a 3 × 96 image, the middle layer as a 12 × 12 image, and the last layer as a 12 × 6 image.

3 Generative Adversarial Networks

Since their first formulation [36], Generative Adversarial Networks have become a rapidly increasing area of attention in the Machine Learning literature, and have recently been found useful in several scientific application domains. Generative Adversarial Networks (GANs) cast the task of training a deep generative model as a two-player non-cooperative minimax game, in which a generator network G is trained concomitantly with an adversary, the discriminator network D, in order to learn a data distribution $\mathcal{N} \sim f$. The generator G learns a map from a latent space z to the space of generated samples $\mathcal{S} \sim \hat{f}$, while D learns a map from the sample space to $[0, 1]$, the probability that a shown sample is real. The game-theoretical basis for this framework [36, 37] ensures that if we extend the space of allowed functions that G and D can draw from to be the space of all continuous functions, then there exists some G that exactly recovers f, i.e., $\hat{f} \to f$, while for every sample produced by the generator, the discriminator is maximally confused and admits a posterior of being real of $\frac{1}{2}$. This formulation of GANs admits the loss function $\mathcal{L}_{\mathsf{adv}}$ shown in Eqn. 3.1.

$$\mathcal{L}_{\mathsf{adv}} = \underbrace{\mathbb{E}[\log(\mathbb{P}(D(I) = 0 \mid I \in \mathcal{S}))]}_{\substack{\text{term associated with the discriminator}\\ \text{perceiving a generated sample as fake}}} + \underbrace{\mathbb{E}[\log(\mathbb{P}(D(I) = 1 \mid I \in \mathcal{N}))]}_{\substack{\text{term associated with the discriminator}\\ \text{perceiving a real sample as real}}} \tag{3.1}$$

Though the GAN framework has shown promise, stability is still a major roadblock, and various ad-hoc and theoretical improvements have been suggested, from architectural guidelines [38–42] to reformulations of the loss specified in Eqn. 3.1 to move away from the Jensen-Shannon divergence [43–48]. As suggested in [49], we are able to impose task-specific metrics which allow us to move away from loss level notions of quality and focus on task-level fidelity measures. We make the conscious decision to utilize the vanilla loss formulation as we find adequate performance with this version.

4 The CaloGAN

Generative Adversarial Networks are explored as a tool to speed up and increase the precision of fast simulation of particle showers in an EM calorimeter. We identify this solution with the name CaloGAN.

For it to be useful in realistic physics applications, such a system needs to be able to accept requests[4] for the generation of showers originating from an incoming particle of type P at energy E. We introduce an auxiliary task of energy reconstruction to condition on E, a real valued variable. The Auxiliary Classifier GAN [39] formalism is tested to also condition on class P, but ultimately abandoned in favor of training a specific generative model for each particle type, as the authors expect that versioning and particle-specific improvements will be prioritized in any practical implementation.

In practice, energy is scaled by a factor of 10^2 and multiplied to the 1024-dimensional latent space vector $z \in \mathbb{R}^{1024}$. The generator G then maps this input to three gray-scale image outputs with different numbers of pixels, which represent the energy patterns collected by the three calorimeter layers as the

[4] This covers a significant portion of the challenge; in practice, the the fast simulation must also take the η-ϕ position of incidence and the incidence angles. We leave this to future work (see Sec. 6).

requested particle propagates through them. The discriminator D accepts the three images as inputs, along with E, the chosen value for the particle energy. The inputs are mapped to a binary output that classifies showers into real and fake, and a continuous output which calculates the total energy deposited in the three layers, then compares it with the requested energy E.

4.1 Model Architecture

Given the sparsity levels and high dynamic range in the data described in Section 2, we follow the LAGAN guidelines [19] to modify the DCGAN [38] architecture for this specific regime.

In the generator (shown in Fig. 4), our design combines parallel LAGAN-like processing streams with a trainable attention mechanism that encodes the sequential connection among calorimeter layers. The LAGAN submodules are composed of a 2D convolutional unit followed by two locally-connected units with batch-normalization [50] layers sandwiched in between. The dimensionality and granularity mismatch among the three longitudinal segmentations of the detector demand separate streams of operations with suitably sized kernels. Towards providing a readily adaptable tool, we provide an architecture construction that is simply a function of the desired output image size, as we seek a common denominator that can be readily applied to a variety of particles in order to obtain reasonable baselines in a quick R&D cycle.

Modelling the sequential nature of the relationship among the energy patterns collected by the three layer requires extra care. Drawing inspiration from [42], we choose an attention mechanism to allow dependence among layers, in which we define trainable transfer functions to optimally resize and apply knowledge of the energy pattern in previous layers to the generation of the subsequent layer readout. More specifically, in-painting takes as input a resized image from a previous layer, $\mathcal{I}^{'}$, and the hypothesized image from the current layer, $\mathcal{I}$, and learns a per-pixel attention weight W via a weighting function $\omega(\mathcal{I}, \mathcal{I}^{'})$ such that the pre-ReLU version of the current layer is $W \odot \mathcal{I} + (1-W) \odot \mathcal{I}^{'}$, where $\odot$ is the Hadamard product. This end-to-end trainable unit can utilize information about the two layers to decide what information to propagate through from the previous particle deposition. An alternative architectural choice that includes a recurrent connection will be subject of future studies.

Leaky Rectified Linear Units [51] are chosen as activation functions throughout the system, with the exception of the output layers of G, in which we prefer Rectified Linear Units [52] for the creation of sparse samples [19].

In the discriminator (shown in Fig. 5), the feature space produced by each LAGAN-style output stream is augmented with a sub-differentiable version of sparsity percentage, as well as minibatch discrimination [44] on both the standard locally connected network-produced features and the output sparsity itself, to ensure a well examined space of sparsities. These are represented in Fig. 5 by the purple ‘features’ vector.

The discriminator is further customized with domain-specific features to ensure fidelity of samples. Given the importance of matching the requested energy E, D directly calculates the empirical energy per layer $\hat{E}_i$, $i \in \{0, 1, 2\}$, as well as the total energy $\hat{E}_{\text{tot}}$. Minibatch discrimination is performed on this vector of per-layer energies to ensure a proper distributional understanding. We also add $|E - \hat{E}_{\text{tot}}|$ as a feature, as well as $\mathbb{I}_{\{|E-\hat{E}_{\text{tot}}|>\varepsilon\}}$, a binary, sub-differentiable feature which encodes the tolerance for GAN-produced scatterings to be incorrect in their reconstructed energy.

Further specifications of our exact hyper-parameter and architectural choices as well as software versioning constraints are available in our source code[5].

[5] https://github.com/hep-lbdl/CaloGAN

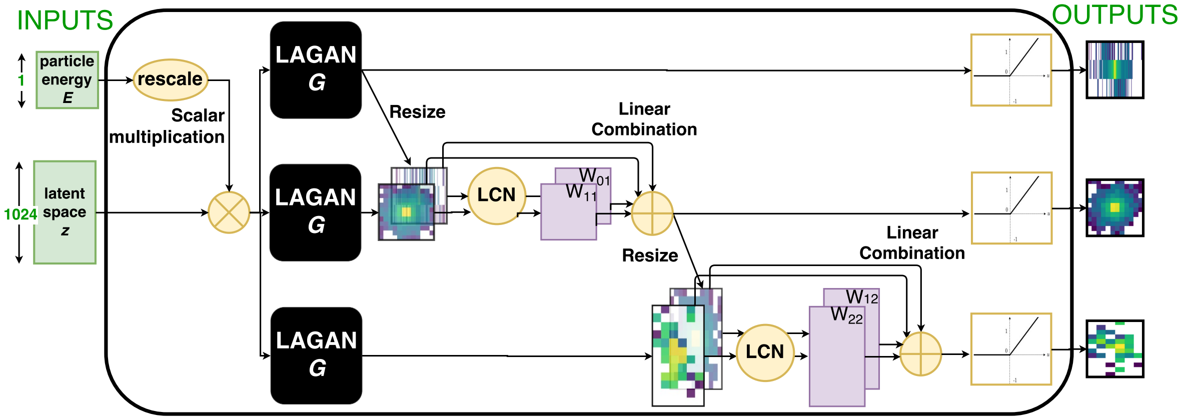


Figure 4: Composite Generator, illustrating three stream with attentional layer-to-layer dependence.

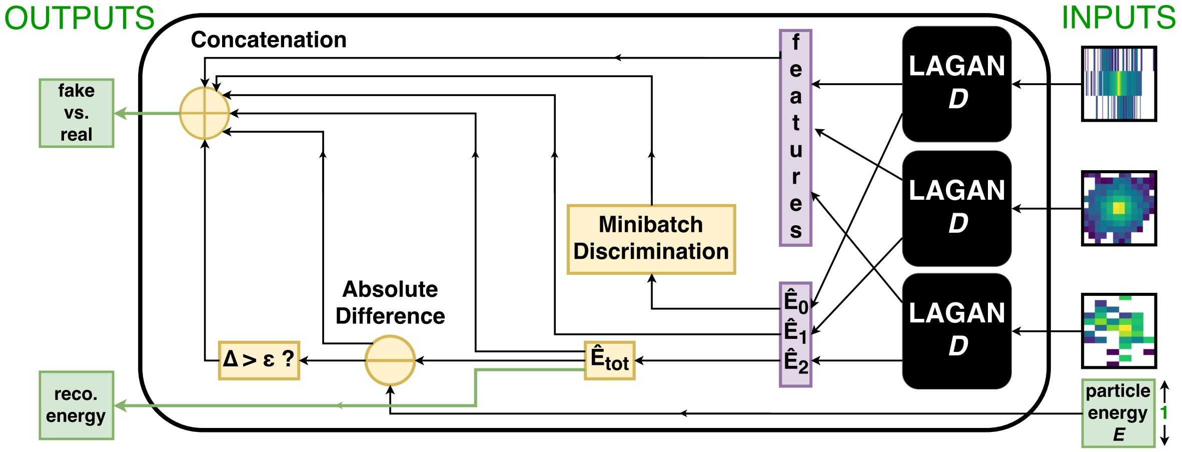


Figure 5: Composite Discriminator, depicting additional domain specific expressions included in the final feature space.

Two additional architectural modifications were tested in order to build a particle-type conditioning system directly into the learning process. Neither the AC-GAN [39] nor the conditional GAN [40] frameworks were able to handle the substantial differences among the three particle types. We suspect that both a significantly richer model and a larger latent space could alleviate some problems associated with conditioning using the investigated approaches. Although building a fully joint model is an interesting Machine Learning challenge, the practicality and flexibility of this application may suffer from having one single model for all particle showers.

4.2 Loss Formulation

In this work, we augment the classical adversarial loss term $\mathcal{L}_{\text{adv}}$ (Eqn. 3.1) – which penalizes the system whenever D fails to classify samples originating from the true and generated data distributions – by conditioning on a continuous variable.

We define a simple mean-absolute-deviance in energy $\delta_{\mathcal{E}}(e, e') = |e - e'|$ which allows us to penalize instances of too little or too much deposited energy. This solution not only helps ensuring the confinement of the generated energy to a desirable range, but it also allows to encode a 'soft' physical notion of conservation of energy, according to which no more energy than the initial E of the incoming particle can be physically collected by the detector. This is used to define a new loss in with respect to the requested energy E and a reconstructed energy $\hat{E}(I)$:

$$\mathcal{L}_E = \mathbb{E}[\delta_{\mathcal{E}}(E, \hat{E}(I))|I \in \mathcal{S}))] \tag{4.1}$$

Note, however, that this formulation discourages, but does not forbid, a deposition of more energy than was requested. We can remedy this unphysical result by sampling from a conditional distribution until energy preservation is met. This issue is further addressed in Sec. 5.2.

During training, the generator will maximize Eqn. 4.2, and the discriminator will maximize Eqn. 4.3.

$$\mathcal{L}_{\mathsf{generator}} = \lambda_E \mathcal{L}_E - \mathcal{L}_{\mathsf{adv}} \tag{4.2}$$

$$\mathcal{L}_{\mathsf{discriminator}} = \lambda_E \mathcal{L}_E + \mathcal{L}_{\mathsf{adv}} \tag{4.3}$$

4.3 Training Strategy

To down-weight the importance of $\mathcal{L}_{\mathsf{adv}}$ compared to $\mathcal{L}_E$, and to rescale with energy range, λ_E is set to 0.05. This hyper-parameter can be tuned in a systematic way, but with minimal tuning, we were able to find a reasonable value.

The weights in the generator and discriminator are optimized in an alternating fashion over a set of 100,000 Geant-simulated events for each particle type in batches of 256, using the Adam optimizer. The discriminator has a learning rate of 2×10^{-5}, and the generator has a learning rate of 2×10^{-4}. We note that outside of initial rough hyperparameter tuning, we perform no dedicated optimization per particle type, and simply apply the same training parameters to all three networks.

Each system is trained for 50 epochs. Sixteen NVIDIA K80 graphics cards are used for initial hyperparameter sweeps, with two Titan X Pascal Architecture cards used for final training. Keras `v2.0.3` is used to construct all models, with the TensorFlow [53] backend (`v1.1.0`).

5 Performance

As discussed in [49], there exist several methods to qualitatively and quantitatively assess the performance of generative networks, but not all evaluation criteria are equally suitable and reliable for all applications. In this paper, we choose application-driven methods focused on sample quality, as opposed to a likelihood based one. A first qualitative assessment will be accompanied by a quantitative evaluation based on physics-driven similarity metrics. The choice reflects the domain specific procedure for data-simulation comparison.

5.1 Qualitative Assessment

As a first level of sanity checks, we examine the average calorimeter deposition per pixel per layer. In particular, we look for irregular patterns and inconsistency as cross checks against physical reasoning and intuition. Figures 6, 7, and 8 show, at a very high level, that on average the systems learn a fairly complete picture of the underlying physical processes governing the cascades of e^+, γ, and π^+ respectively. In these figures, the particle energy is uniform between 1 GeV and 100 GeV.

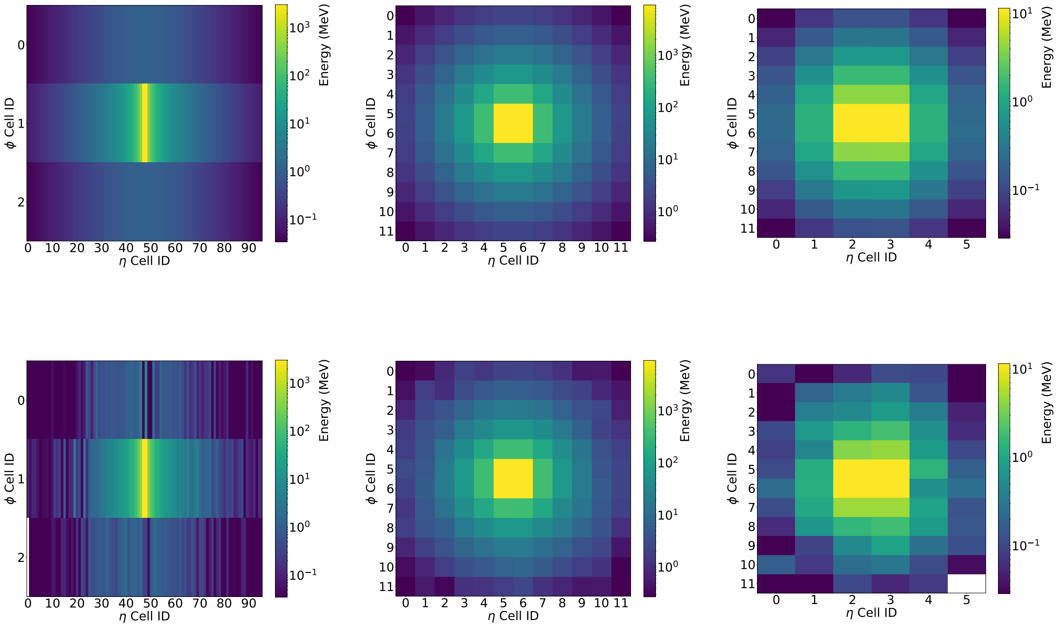


Figure 6: Average e^+ Geant shower (top), and average e^+ CaloGAN shower (bottom), with progressive calorimeter depth (left to right).

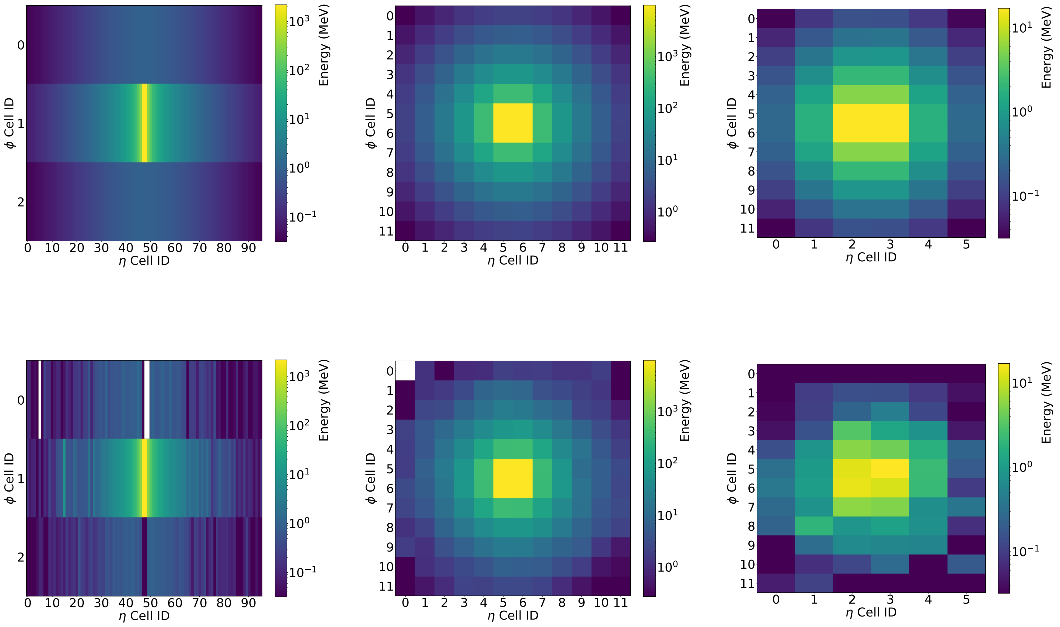


Figure 7: Average γ Geant shower (top), and average γ CaloGAN shower (bottom), with progressive calorimeter depth (left to right).

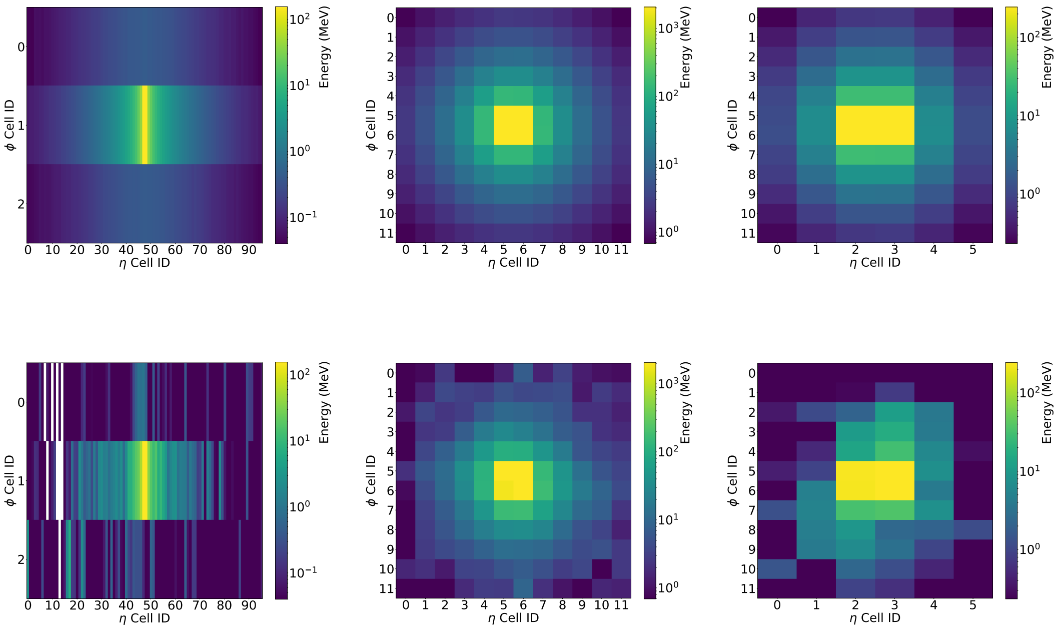


Figure 8: Average π^+ GEANT shower (top), and average π^+ CALOGAN shower (bottom), with progressive calorimeter depth (left to right).

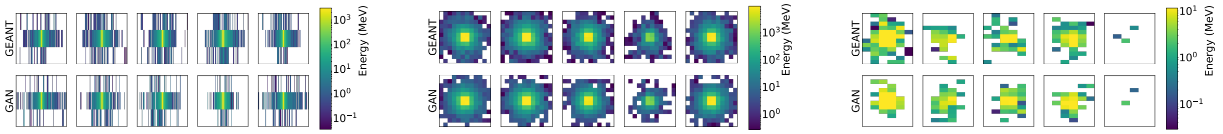


Figure 9: Five randomly selected e^+ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CALOGAN candidates.

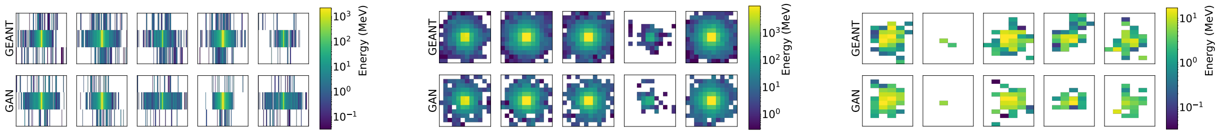


Figure 10: Five randomly selected γ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CALOGAN candidates.

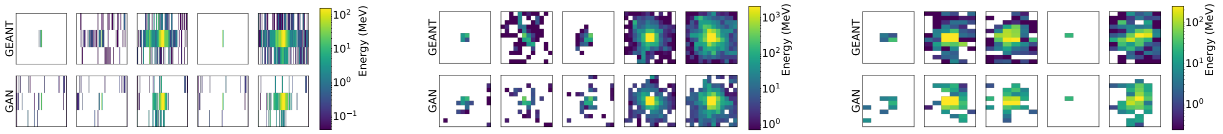


Figure 11: Five randomly selected π^+ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CALOGAN candidates.

Diversity and overtraining concerns can be investigating by considering the nearest neighbors among the training and generated datasets. Figures 9, 10, and 11 shows five randomly selected events and their GAN-generated nearest neighbors for all three calorimeter layers for e^+, γ and π^+ showers respectively. Good qualitative agreement can be found between the two distributions across all layers, without obvious signs of mode collapse. Compared to the other two particle types explored in this application, at the individual image level, charged pions clearly display a higher degree of complexity and diversity in their showers. Some π^+ deposit energy in all cells of a given layer, some only hitting a handful of them. This is because charged pions undergo nuclear interactions in addition to electromagnetic interactions and partially justifies the desire to have separate models per particle type.

5.2 Shower Shapes

Electron and photon classification and energy calibration use properties of the calorimeter shower [54–57]. These same features can be used to quantitatively assess the quality of the GAN samples. The list of features used for evaluation is provided in Table 3 in Appendix A. The key physical quantity that governs the shapes of these distributions is the number of radiation lengths X_0 that are traversed by the particle. By definition, X_0 is the distance an electron will travel before its energy is reduced to $1/e$ on average. The equivalent distance for photons is slightly further (by $9/7$ [58]) and is set by the mean free path for pair production. The transverse shower size is also proportional to X_0. For a brief review, see e.g. [58].

The 1-dimensional distributions for Geant- and GAN-generated samples are available in Fig. 12. Although the sparsity levels per layer are only roughly matched, notice how, for the majority of the remaining variables, the GAN picks up on complex features in the distributions across several orders of magnitude and all particles types. The unique features that pions exhibit, compared to the other particles, may be part of what makes it unfavorable to train a single model for multiple particle types.

Note that shower shape variables were not explicitly part of the training, which is based only on the distribution of pixel intensities and energy. In the future, one can integrate the shower shape distributions into the loss-function itself. For now, we have left them out for a comprehensive validation assessment.

In addition to comparing shower shapes to reference distribution under a flat prior, we want to measure the quality of conditioning on energy. As outlined in Sec. 4.2, we cannot explicitly impose conservation of energy, but one can devise a simple sampling system to only keep simulated showers that obey this constraint. As can be noted in the e^+ example in Fig. 13, our loss formulation coupled with the uniform training distribution admits an approximately symmetric conditional output energy distribution. In Fig. 13, note that the vertical lines that approximately coincide with the mode of each distribution represent the requested energy, and could easily be used as a threshold on selecting physical events. A noteworthy feature of this system is that one can request energies that lie outside the trained region (capped at 100 GeV in this application), to which a trained CaloGAN will return samples around the requested energy level – though with broader width, and mode shifted towards the training domain. Whether or not these extrapolated samples obey shower shape distributions and other metrics is left as future work.

5.3 Classification as a Performance Proxy

Transferability of classification performance from GAN-generated samples to Geant-generated samples can be used as a proxy both for CaloGAN image quality and potential utility in a practical fast simulation setting. Preliminary studies show encouraging results in this direction.

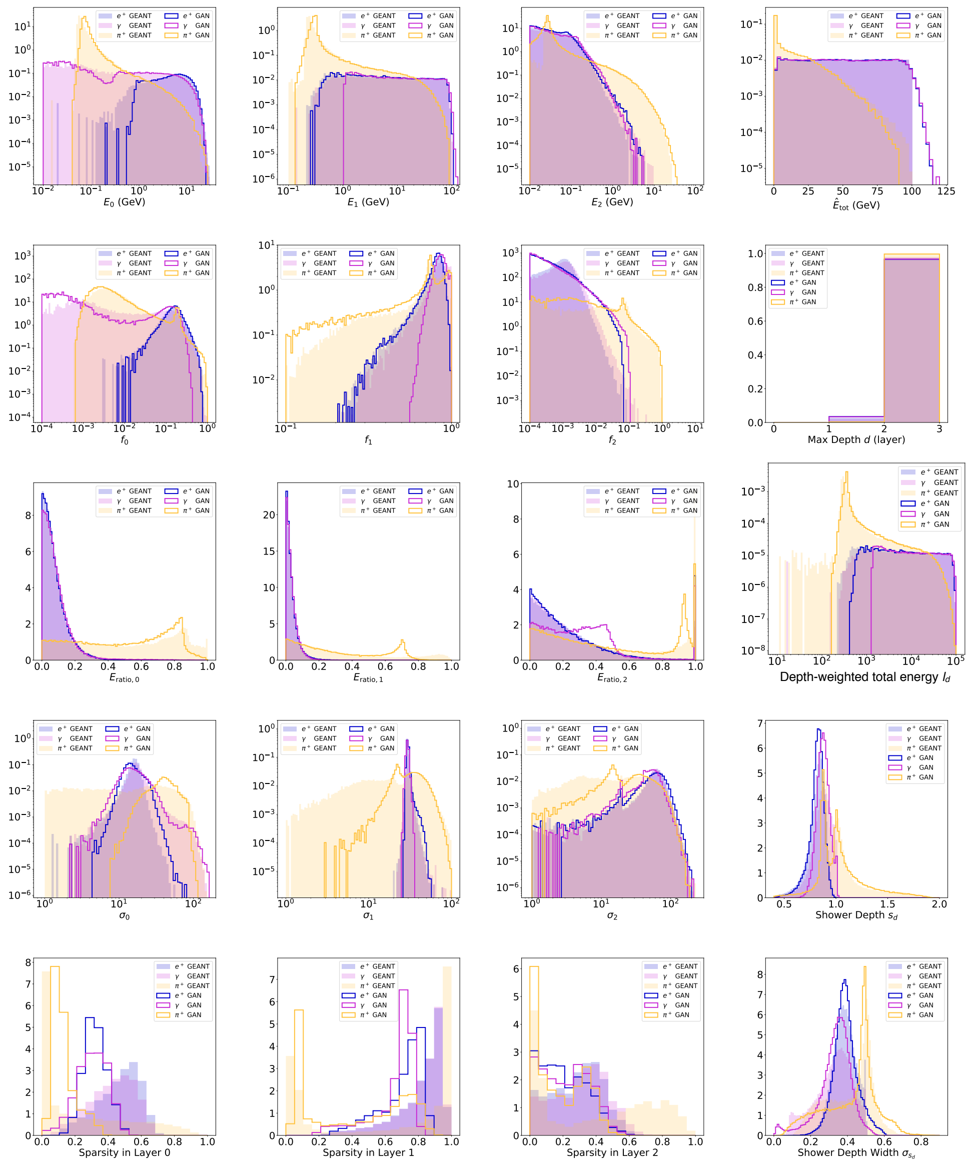


Figure 12: Comparison of shower shape variables, introduced in Table 3, and other variables of interest, such as the sparsity level per layer, for the Geant and CaloGAN datasets for e^+, γ and π^+.

5.4 Computational Performance

In addition to the promise of being a high-fidelity fast simulation paradigm and respecting shower shape variables, the CaloGAN affords many orders of magnitude in computational speedups. We benchmark generation time on e^+ with incident energy drawn uniformly between 1 GeV and 100 GeV. Geant4 and CaloGAN on CPU are benchmarked on nearly identical compute-nodes on the PDSF distributed cluster at the National Energy Research Scientific Computing Cluster (NERSC) [59], and numerical results are obtained over an average of 100 runs. CaloGAN on GPU hardware is benchmarked on

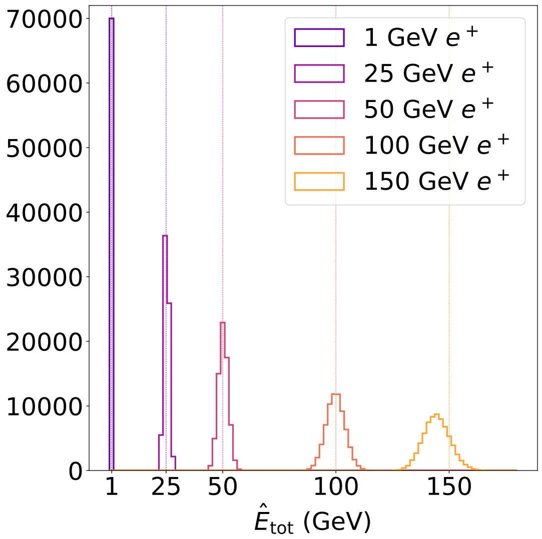


Figure 13: Post energy-conditioned empirical energy response for e^+ incident at 1, 25, 50, 100, and 150 GeV. Though our model is only trained on the unform range between 1 and 100 GeV, it still admits a compelling peak at 150 GeV.

an Amazon Web Service (AWS) `p2.8xlarge` instance, where a single NVIDIA K80 is used for the purposes of benchmarking.

In Table 2, we show the time-to-generate a single particle shower in milliseconds. We provide different batch sizes for CaloGAN, as we expect different use-cases will have different demands around batching computation. We note that a batch can accept any number of different requested energies. With the largest batch sizes on GPU, our method admits a speedup of 5 orders of magnitude compared to the Geant4 benchmark. In addition, generation time with Geant scales with incident energy, whereas computational time is flat as a function of incident energy for the CaloGAN.

Generation Method	**Hardware**	**Batch Size**	**milliseconds/shower**
Geant4	CPU	N/A	1772
CaloGAN	CPU	1	13.1
		10	5.11
		128	2.19
		1024	2.03
	GPU	1	14.5
		4	3.68
		128	0.021
		512	0.014
		1024	0.012

Table 2: Total expected time (in milliseconds) required to generate a single shower under various algorithm-hardware combinations.

5.4.1 Implementation Notes

As noted previously in Sec. 4.1, separating per-particle-type CaloGAN architectures and implementations affords many benefits. It is easy to imagine a situation where the life cycles surrounding models for different particle types are very different. In addition, this allows for total independence of versioning, framework, or language – i.e., an ecosystem of RPC clients around various generators for specific tasks.

When possible, any GAN should maximally employ batching – we imagine most applications can request all showers from one event simultaneously, maximally taking advantage of CPU/GPU while minimizing data transfer overhead.

6 Conclusions and Future Outlook

We generated three-dimensional electromagnetic showers in a multi-layer sampling LAr calorimeter with uneven spatial segmentation, while attempting to preserve spatio-temporal relation among layers. Our approach has infused Physics domain knowledge and is able to reproduce many aspects of key shower shape properties comparable to the ones in the Geant full simulation. We show the possibility of up to 5 orders of magnitude increase in computational efficiency.

Future work will build on this result and continue expanding the complexity of the training dataset to include incoming particles at different locations and angles within the detector, as well as hadronic calorimetery. Concurrent plans include contributing to testing the computational performance on high performance computing (HPC) clusters, and porting these solutions into the simulation packages used at CERN, in order for the various LHC experiments to be able to maximally benefit from this new technology.

Acknowledgments

We would like to thank NERSC for access to computational resources on the PDSF cluster.

References

[1] M. Botje *et. al.*, *The PDF4LHC Working Group Interim Recommendations*, 1101.0538.

[2] M. Czakon and A. Mitov, *Top++: A Program for the Calculation of the Top-Pair Cross-Section at Hadron Colliders*, *Comput. Phys. Commun.* **185** (2014) 2930 [1112.5675].

[3] A. D. Martin, W. J. Stirling, R. S. Thorne and G. Watt, *Uncertainties on alpha(S) in global PDF analyses and implications for predicted hadronic cross sections*, *Eur. Phys. J.* **C64** (2009) 653–680 [0905.3531].

[4] J. Gao, M. Guzzi, J. Huston, H.-L. Lai, Z. Li, P. Nadolsky, J. Pumplin, D. Stump and C. P. Yuan, *CT10 next-to-next-to-leading order global analysis of QCD*, *Phys. Rev.* **D89** (2014), no. 3 033009 [1302.6246].

[5] R. D. Ball *et. al.*, *Parton distributions with LHC data*, *Nucl. Phys.* **B867** (2013) 244–289 [1207.1303].

[6] **CMS** Collaboration, V. Khachatryan *et. al.*, *Measurement of the $t\bar{t}$ production cross section using events in the $e\mu$ final state in pp collisions at $\sqrt{s} =$ 13 TeV*, *Eur. Phys. J.* **C77** (2017) 172 [1611.04040].

[7] **ATLAS** Collaboration, M. Aaboud *et. al.*, *Measurements of top-quark pair differential cross-sections in the $e\mu$ channel in pp collisions at $\sqrt{s} = 13$ TeV using the ATLAS detector*, 1612.05220.

[8] GEANT4 Collaboration, *Geant4 - a simulation toolkit*, *Nuclear Instruments and Methods in Physics Research A* **506** (2003) 250.

[9] **ATLAS** Collaboration, G. Aad *et. al.*, *The ATLAS Simulation Infrastructure*, *Eur. Phys. J.* **C70** (2010) 823–874 [1005.4568].

[10] R. Rahmat, R. Kroeger and A. Giammanco, *The fast simulation of the cms experiment*, *Journal of Physics: Conference Series* **396** (2012), no. 6 062016.

[11] J. Flynn, *Computing Resources Scrutiny Group Report*, Tech. Rep. CERN-RRB-2015-014, CERN, Geneva, Mar, 2015.

[12] E. Karavakis, J. Andreeva, S. Campana, S. Gayazov, S. Jezequel, P. Saiz, L. Sargsyan, J. Schovancova, I. Ueda and the Atlas Collaboration, *Common accounting system for monitoring the atlas distributed computing resources*, *Journal of Physics: Conference Series* **513** (2014), no. 6 062024.

[13] C. Bozzi, *LHCb Computing Resource usage in 2014 (II)*, Tech. Rep. LHCb-PUB-2015-004. CERN-LHCb-PUB-2015-004, CERN, Geneva, Jan, 2015.

[14] G. Grindhammer and S. Peters, *The Parameterized simulation of electromagnetic showers in homogeneous and sampling calorimeters*, in *International Conference on Monte Carlo Simulation in High-Energy and Nuclear Physics - MC 93 Tallahassee, Florida, February 22-26, 1993*, 1993. hep-ex/0001020.

[15] C. ATLAS, M. Beckingham, M. Duehrssen, E. Schmidt, M. Shapiro, M. Venturi, J. Virzi, I. Vivarelli, M. Werner, S. Yamamoto and T. Yamanaka, *The simulation principle and performance of the ATLAS fast calorimeter simulation FastCaloSim*, Tech. Rep. ATL-PHYS-PUB-2010-013, CERN, Geneva, Oct, 2010.

[16] G. Grindhammer, M. Rudowicz and S. Peters, *The Fast Simulation of Electromagnetic and Hadronic Showers*, *Nucl. Instrum. Meth.* **A290** (1990) 469.

[17] E. Barberio, J. Boudreau, B. Butler, S. L. Cheung, A. Dell'Acqua, A. D. Simone, E. Ehrenfeld, M. V. Gallas, A. Glazov, Z. Marshall, J. Mueller, R. Plaakyte, A. Rimoldi, P. Savard, V. Tsulaia, A. Waugh and C. C. Young, *Fast simulation of electromagnetic showers in the atlas calorimeter: Frozen showers*, *Journal of Physics: Conference Series* **160** (2009), no. 1 012082.

[18] *Performance of the Fast ATLAS Tracking Simulation (FATRAS) and the ATLAS Fast Calorimeter Simulation (FastCaloSim) with single particles*, Tech. Rep. ATL-SOFT-PUB-2014-001, CERN, Geneva, Mar, 2014.

[19] L. de Oliveira, M. Paganini and B. Nachman, *Learning Particle Physics by Example: Location-Aware Generative Adversarial Networks for Physics Synthesis*, 1701.05927.

[20] J. Cogan, M. Kagan, E. Strauss and A. Schwartzman, *Jet-Images: Computer Vision Inspired Techniques for Jet Tagging*, *JHEP* **02** (2015) 118 [1407.5675].

[21] D. P. Kingma and M. Welling, *Auto-Encoding Variational Bayes*, *ArXiv e-prints* (Dec., 2013) [1312.6114].

[22] A. Makhzani, J. Shlens, N. Jaitly and I. J. Goodfellow, *Adversarial autoencoders*, *CoRR* **abs/1511.05644** (2015).

[23] S. Ravanbakhsh, F. Lanusse, R. Mandelbaum, J. Schneider and B. Poczos, *Enabling Dark Energy Science with Deep Generative Models of Galaxy Images*, 1609.05796.

[24] K. Schawinski, C. Zhang, H. Zhang, L. Fowler and G. K. Santhanam, *Generative adversarial networks recover features in astrophysical images of galaxies beyond the deconvolution limit*, (May, 2017) L110–L114 [1702.00403].

[25] L. Mosser, O. Dubrule and M. J. Blunt, *Reconstruction of three-dimensional porous media using generative adversarial neural networks*, *ArXiv e-prints* (Apr., 2017) [1704.03225].

[26] A. Kadurin, A. Aliper, A. Kazennov, P. Mamoshina, Q. Vanhaelen, K. Khrabrov and A. Zhavoronkov, *The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology*, *Oncotarget* **8** (2016), no. 7.

[27] **ATLAS Collaboration** Collaboration, *ATLAS liquid-argon calorimeter: Technical Design Report*. Technical Design Report ATLAS. CERN, Geneva, 1996.

[28] B. Andersson, G. Gustafson and B. Nilsson-Almqvist, *A model for low-pt hadronic reactions with generalizations to hadron-nucleus and nucleus-nucleus collisions*, *Nuclear Physics B* **281** (1987), no. 1 289 – 309.

[29] B. Andersson, A. Tai and B.-H. Sa, *Final state interactions in the (nuclear) fritiof string interaction scenario*, *Zeitschrift für Physik C Particles and Fields* **70** (1996), no. 3 499–506.

[30] B. Nilsson-Almqvist and E. Stenlund, *Interactions Between Hadrons and Nuclei: The Lund Monte Carlo, Fritiof Version 1.6*, *Comput. Phys. Commun.* **43** (1987) 387.

[31] B. Ganhuyag and V. Uzhinsky, *Modified FRITIOF code: Negative charged particle production in high energy nucleus nucleus interactions*, *Czech. J. Phys.* **47** (1997) 913–918.

[32] M. P. Guthrie, R. G. Alsmiller and H. W. Bertini, *Calculation of the capture of negative pions in light elements and comparison with experiments pertaining to cancer radiotherapy*, *Nucl. Instrum. Meth.* **66** (1968) 29–36.

[33] H. W. Bertini and M. P. Guthrie, *News item results from medium-energy intranuclear-cascade calculation*, *Nucl. Phys.* **A169** (1971) 670–672.

[34] V. A. Karmanov, *LIGHT FRONT WAVE FUNCTION OF RELATIVISTIC COMPOSITE SYSTEM IN EXPLICITLY SOLVABLE MODEL*, *Nucl. Phys.* **B166** (1980) 378–398.

[35] H. Burkhardt, V. M. Grichine, P. Gumplinger, V. N. Ivanchenko, R. P. Kokoulin, M. Maire and L. Urban, *Geant4 standard electromagnetic package for hep applications*, in *IEEE Symposium Conference Record Nuclear Science 2004.*, vol. 3, pp. 1907–1910 Vol. 3, Oct, 2004.

[36] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, *Generative adversarial networks*, *ArXiv e-prints* (2014) [1406.2661].

[37] I. J. Goodfellow, *On distinguishability criteria for estimating generative models*, *ArXiv e-prints* (Dec., 2014) [1412.6515].

[38] A. Radford, L. Metz and S. Chintala, *Unsupervised representation learning with deep convolutional generative adversarial networks*, *ArXiv e-prints* (2015) [1511.06434].

[39] A. Odena, C. Olah and J. Shlens, *Conditional Image Synthesis With Auxiliary Classifier GANs*, *ArXiv e-prints* (Oct., 2016) [1610.09585].

[40] M. Mirza and S. Osindero, *Conditional generative adversarial nets*, *ArXiv e-prints* (2014) [1411.1784].

[41] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever and P. Abbeel, *Infogan: Interpretable representation learning by information maximizing generative adversarial nets*, *ArXiv e-prints* (2016) [1606.03657].

[42] H. Zhang, T. Xu, H. Li, S. Zhang, X. Huang, X. Wang and D. Metaxas, *StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks*, *ArXiv e-prints* (Dec., 2016) [1612.03242].

[43] M. Arjovsky, S. Chintala and L. Bottou, *Wasserstein GAN*, *ArXiv e-prints* (Jan., 2017) [1701.07875].

[44] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford and X. Chen, *Improved techniques for training gans*, *ArXiv e-prints* (2016) [1606.03498].

[45] S. Nowozin, B. Cseke and R. Tomioka, *f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization*, *ArXiv e-prints* (June, 2016) [1606.00709].

[46] G. Qi, *Loss-sensitive generative adversarial networks on lipschitz densities*, *CoRR* **abs/1701.06264** (2017).

[47] X. Mao, Q. Li, H. Xie, R. Y. K. Lau and Z. Wang, *Multi-class generative adversarial networks with the L2 loss function*, *CoRR* **abs/1611.04076** (2016).

[48] M. Lin, *Softmax GAN*, *ArXiv e-prints* (Apr., 2017) [1704.06191].

[49] L. Theis, A. van den Oord and M. Bethge, *A note on the evaluation of generative models*, *ArXiv e-prints* (Nov., 2015) [1511.01844].

[50] S. Ioffe and C. Szegedy, *Batch normalization: Accelerating deep network training by reducing internal covariate shift*, 1502.03167.

[51] A. L. Maas, A. Y. Hannun and A. Y. Ng, *Rectifier Nonlinearities Improve Neural Network Acoustic Models*, .

[52] X. Glorot, A. Bordes and Y. Bengio, *Deep sparse rectifier neural networks*, *Journal of Machine Learning Research* **15** (2011).

[53] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu and X. Zheng, *TensorFlow: Large-scale machine learning on heterogeneous systems*, 2015. Software available from https://www.tensorflow.org/.

[54] **ATLAS** Collaboration, G. Aad *et. al.*, *Electron performance measurements with the ATLAS detector using the 2010 LHC proton-proton collision data*, *Eur. Phys. J.* **C72** (2012) 1909 [1110.3174].

[55] **ATLAS** Collaboration, G. Aad *et. al.*, *Electron reconstruction and identification efficiency measurements with the ATLAS detector using the 2011 LHC proton-proton collision data*, *Eur. Phys. J.* **C74** (2014), no. 7 2941 [1404.2240].

[56] **ATLAS** Collaboration, M. Aaboud *et. al.*, *Measurement of the photon identification efficiencies with the ATLAS detector using LHC Run-1 data*, *Eur. Phys. J.* **C76** (2016), no. 12 666 [1606.01813].

[57] **ATLAS** Collaboration, M. Aaboud *et. al.*, *Electron efficiency measurements with the ATLAS detector using 2012 LHC proton-proton collision data*, *Eur. Phys. J.* **C77** (2017), no. 3 195 [1612.01456].

[58] **Particle Data Group** Collaboration, K. A. Olive *et. al.*, *Passage of Particles Through Matter in Review of Particle Physics*, *Chin. Phys.* **C38** (2014) 090001.

[59] *Nersc compute nodes*, Apr., 2016.

A Shower Shape Variables

Table 3 contains the description and mathematical definition of the shower shape variables used to compare the true and generated data distributions. These are defined as functions of $\mathcal{I}_i$, the vector of pixel intensities for an image in layer i, where $i \in \{0, 1, 2\}$.

Shower Shape Variable	Formula	*Notes*
E_i	$E_i = \sum_{\text{pixels}} \mathcal{I}_i$	Energy deposited in the i^{th} layer of calorimeter
E_{tot}	$E_{\text{tot}} = \sum_{i=0}^{2} E_i$	Total energy deposited in the electromagnetic calorimeter
f_i	$f_i = E_i / E_{\text{tot}}$	Fraction of measured energy deposited in the i^{th} layer of calorimeter
$E_{\text{ratio},i}$	$\dfrac{\mathcal{I}_{i,(1)} - \mathcal{I}_{i,(2)}}{\mathcal{I}_{i,(1)} + \mathcal{I}_{i,(2)}}$	Difference in energy between the highest and second highest energy deposit in the cells of the i^{th} layer, divided by the sum
d	$d = \max\{i : \max(\mathcal{I}_i) > 0\}$	Deepest calorimeter layer that registers non-zero energy
Depth-weighted total energy, l_d	$l_d = \sum_{i=0}^{2} i \cdot \mathcal{I}_i$	The sum of the energy per layer, weighted by layer number.
Shower Depth, s_d	$s_d = l_d / E_{\text{tot}}$	The energy-weighted depth in units of layer number.
Shower Depth Width, σ_{s_d}	$\sigma_{s_d} = \sqrt{\dfrac{\sum_{i=0}^{2} i^2 \cdot \mathcal{I}_i}{E_{\text{tot}}} - \left(\dfrac{\sum_{i=0}^{2} i \cdot \mathcal{I}_i}{E_{\text{tot}}}\right)^2}$	The standard deviation of s_d in units of layer number.
i^{th} Layer Lateral Width, σ_i	$\sigma_i = \sqrt{\frac{\mathcal{I}_i \odot H^2}{E_i} - \left(\frac{\mathcal{I}_i \odot H}{E_i}\right)^2}$	The standard deviation of the transverse energy profile per layer, in units of cell numbers.

Table 3: One-dimensional observables used to assess the quality of the GAN samples.