3D Human Body Shape Generation

Ethan Bonnardeaux[1], Spencer Hill[2], Daniel Stewart[3], Noah Cabral[4]

QMIND – Queen's AI Hub
Queen's University, Kingston, Ontario K7L 3N6, Canada. Queen's

1 e-mail: 18ejb@queensu.ca
2 e-mail: 18sjch@queensu.ca
3 e-mail: daniel.stewart@queensu.ca
4 e-mail: 15nsc4@queensu.ca

Abstract: *Generating point cloud models has become an increasingly popular practice within the machine learning community. Human shape data is the key to producing advancements within medical imaging, virtual reality, gaming, and animation fields. Learning object structure in 3-dimensional space presents many challenges in which deep learning networks have become iteratively capable of resolving. In this paper, we utilize a proven generative modeling technique to learn the approximate representation of human body shapes on point cloud data from the Semantic Body Models Dataset. By leveraging TreeGAN, a tree-based graph convolution generator network, our model is capable of learning the different segments of the human body in an unsupervised fashion. This approach combines the classic Generative Adversarial Framework with a nuanced generator that boosts its feature representation by sequentially accessing historical prediction states. Due to the consistent internal nature of human body shape data, we only sample points from the surface of the body, similarly restricting the model's learned representations*.

---

1. INTRODUCTION

1.1 Motivation

Recently, neural networks involving 3D data have attracted significant research interest. Since the introduction of Point Net in 2016, 3D point clouds have emerged as the most computationally efficient method of interpreting 3D data [1]. While most work has focused on object segmentation, classification, and object detection, in 2019 a novel architecture (TreeGAN) was proposed for 3D point cloud generation [2]. Leveraging the Generative Adversarial Network (GAN) [3] framework and tree-based graph convolution networks (GCNs), TreeGAN achieved seminal results on the ShapeNet40 dataset [2]. However, little to no work has been done to expand this object generation to more impactful datasets.

The immediate application of human body generation is to computer vision and medical imaging fields. The interpretation and generation of the human figure is an essential computer vision task that has received little attention. Furthermore, medical privacy restrictions make novel human body generation beneficial to training medical students and artificial intelligence systems on this data.

1.2 Related Works

1.21 Point Clouds, Neural Networks, and GANs

Most researchers transform point cloud data into 3D voxel grids or collections of images before running the data through deep learning pipelines. Charles et al. [1] proposed PointNet, a novel neural network that directly consumes point cloud data, which well respects the permutation invariance of point clouds. PointNet can be trained to perform 3D shape classification, shape part segmentation and scene semantic parsing tasks. Since the invention of PointNet, point cloud data have been used not just in classification networks but also in generative tasks. For example, Achlioptas et al. [4] proposed a GAN for the generation of 3D points clouds called r-GAN. The generator for r-GAN is based on fully connected layers, leading to r-GAN having difficulty in generating realistic shapes with diversity.

1.22 Improved Training of Wasserstein GANs

A common issue in training GANs is the stability of training. Arjovsky et al. [5] introduced Wasserstein GAN, which uses an efficient approximation of the earth mover's distance function to optimize the discriminator and generator in GAN training. WGAN improved training stability and provided a meaningful loss metric that correlated with the generator's sample quality. However, WGAN still suffered from poor sample generation or a failure to converge, and it has been found that this is due to the weight clipping used to enforce a 1-Lipschitz continuous constraint on the critic. Gulrajani et al. [6] introduced gradient penalty, an alternative to weight clipping. It penalized the norm of the critic gradient with respect to the critics input, improving the sample quality and ability to converge for WGANs.

1.23 Graph Convolutional Networks

Over the past few years, many works have focused on using deep neural networks for graph problems. Defferrard et al. [7] proposed fast-learning convolutional filters for graph-based applications, significantly accelerating one of the main computational bottlenecks in graph convolution problems with large datasets. Kipf and Welling [8] introduced scalable GCNs, where convolution filters use only the information from neighboring vertices instead of from the entire graph. All the GCNs mentioned prior are designed for classification problems, meaning that the connectivity of nodes in the graph were known beforehand. This issue will present challenges for the generation of 3D point clouds, where the connectivity is not known.

1.24 GCNs and GANs for 3D Point Clouds Generation

A number of works have tackled the issue of connectivity. Valsesia et al. [9] dynamically generated adjacency matrices using the feature vectors from each vertex at each layer of graph convolutions during training. Unfortunately, computing this matrix at a single layer incurs a quadratic computational complexity on the number of vertices. This approach is not effective for multi-layer and multi-batch networks. Dong et al. [2] proposed TreeGAN, which, like the other work, requires no prior knowledge regarding connectivity. TreeGAN, however, is much more computationally efficient as it avoids constructing adjacency matrices. It uses a tree-based graph structure, and it exploits this structure by using ancestor information to propagate features over the graph. The tree-based graph structure also has the benefit of allowing the network to generate point clouds for different semantic parts of a model without prior knowledge.

1.3 Problem Definition

In this paper, we introduce point cloud generation of human body shape representations from randomized latent vectors. We explore the semantic parametric reshaping of human body models dataset [10] (a derivative of the Caesar dataset) to train our model. Historically, point cloud generation has been explored solely on the ShapeNet40 dataset. This dataset contains 40 different object classes and enables the generator models to produce a wide range of outputs. Currently, the TreeGAN paper has achieved state-of-the-art results on this dataset. We aim to train a generator on a single object class with a higher point-cloud resolution (3072 points) to produce increasingly granular results.

2. METHODOLOGY

2.1 Dataset

The dataset used to train and evaluate our model is composed of 3048 scanned body models. More precisely, there are 1531 male models and 1517 female models. To generate this dataset, we collected the mesh models from the publicly accessible dataset: Semantic Body Models Dataset [10]. The decision to develop our dataset from the Semantic Body Models Dataset is driven by the fact that it is open-sourced and completely available to the public and research communities. Further, the meshes in this dataset are pose-invariant, thus, allowing for more efficient learning of the true biological features of the human form rather than differences in pose.

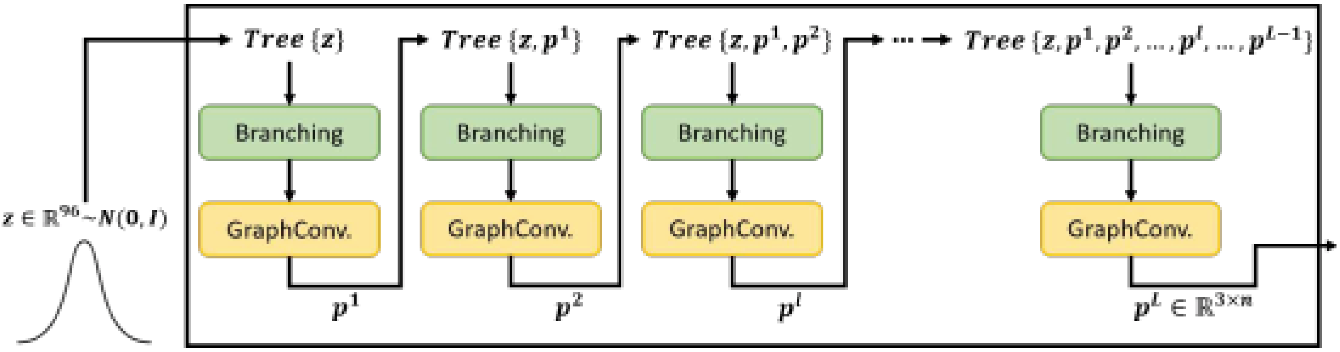

Figure 1 TreeGan Generator Architecture [2]

Mesh objects are highly memory-intensive due to the nature of their vertex-facet construction. Thus, point clouds from scanned meshes were built using an even surface sampling method to construct the point clouds from 3072 evenly spaced points on the surface of the mesh. This method of surface sampling was done for two reasons. The data points within the volume of the mesh do not significantly contribute to the learned features of the human form under the assumption that models are not hollow; Surface sampling maximizes the resolution of features using a point cloud representation while also minimizing memory and compute costs.

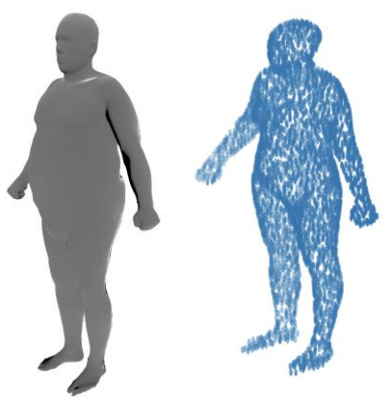

Figure 2 Data example from the Semantic Parametric Reshaping of Human Body Models dataset and a point cloud sample used in model training.

2.2 Model Architecture

Our model is built on the GAN framework, in which a generator and discriminator model train sequentially according to respective loss functions introduced in Wasserstein GAN [5].

$$L_{gen} = -E_{z\sim Z}\left[D\left(G(z)\right)\right], \qquad (1)$$

$$L_{disc} = E_{z\sim Z}[D(G(z))] - E_{z\sim R}[[D(x)] + \lambda_{gp} E_{\hat{x}}\left[\left(\left|\left|\nabla_{\hat{x}} D(\hat{x})\right|\right|_2 - 1\right)^2\right]. \qquad (2)$$

G and D denote the generator and discriminator networks, z is a latent vector created using a normal distribution, $\hat{x}$ are line segments between real and fake point clouds, $x' \sim G(z)$ and x represent real and fake point clouds respectively, and R is the real data distribution. We also apply a gradient penalty, λ_{gp}, to satisfy the 1-Lipshitz condition for GANs.

The generator leverages tree convolutions defined by

$$p^{l+1} = \sigma\left(F_K^l\left(p_i^l\right) + \sum_{q_j \in A\left(p_i^l\right)} U_i^l q_j + b^l\right), (3)$$

which is thoroughly described in (tree-GAN).

The generator takes as input a 96-dimensional latent vector, and through the convolution defined above, conventional convolutional neural network loop terms, and upsampling through defined branching, outputs a set of 3072 3D points. Figure 1 shows the generator built with tree-GCN layers.

As standard in GAN training, the Adam optimizer was used with the custom loss functions shown in (1) and (2).

3. RESULTS AND DISCUSSION

GAN evaluation metrics are an ongoing discussion within the research community as quantitative evaluation methods are continuously being introduced to measure crucial elements of a Generator's performance. Given the nature of this project, the appropriate evaluation metrics are Jensen-Shannon-Divergence (JSD), Coverage (COV-ED, COV-MMD), and Minimum Matching Distance (MMD) [11]. These metrics require a MMD comparison between the reference data and the generator's closest representation per reference example. Our model is currently on ~epoch 500 within the training process and produces outputs as seen in figures 3, 4. Per qualitative examination, the generated results are not yet at comparable to the training data and minimum matching distance would provide inaccurate pairings between the generated and reference examples, resulting in a misleading evaluation of the model's efficacy.

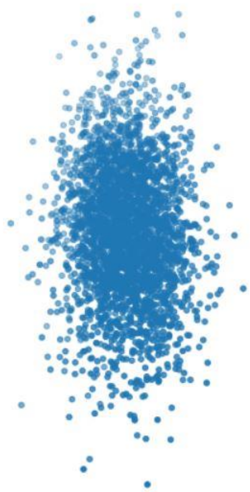

Figure 3 Generated Human Body Shape rotated 45 degrees along Z-axis.

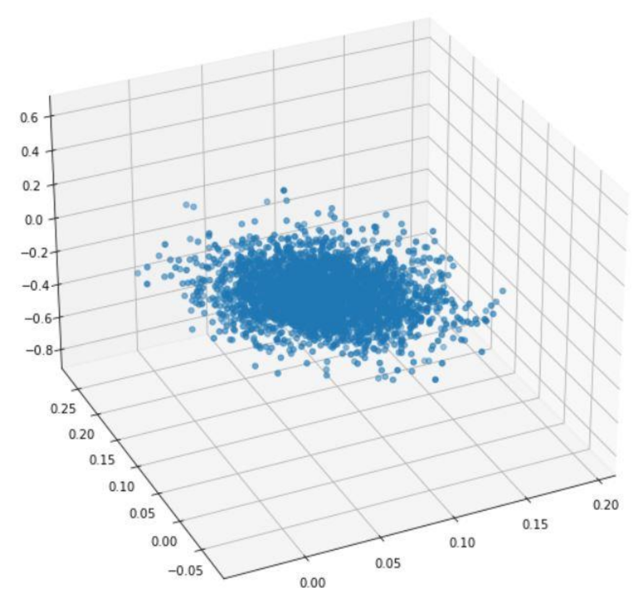

Figure 4 Alternate Generated Human Shape (Side View)

Upon observation, the model has clearly learned the basic features of a human body. Specifically, it has begun representing the chest, arms, legs, and head. As the training progresses, the amount of noise in the generated examples is expected to significantly decrease. The TreeGAN architecture was designed for non-hollow 3D point-cloud data. Due to the hollow composition of human shapes within our dataset, the generator has had difficulty minimizing its loss on the cylindrical-like components of the bodies.

4. CONCLUSIONS AND FUTURE WORK

In this project, we trained a TreeGAN model to introduce the generation of human body shapes to the machine learning community. We discovered a drawback when applying this architecture to hollow shapes. Future work on this problem should involve alterations within the TreeGAN architecture to effectively handle hollow data. Interpolation would also be an interesting area of exploration for a final model to permit controllable generation.

REFERENCES (IEEE format)

[1] Charles R. Qi et al. (2017). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation.

[2] Dong Wook Shu et al. (2019). 3D Point Cloud Generative Adversarial Network Based on Tree Structured Graph Convolutions

[3] Ian J. Goodfellow et al. (2014). Generative Adversarial Networks

[4] Panos Achlioptas et al. (2018). Learning Representations and Generative Models for 3D Point Clouds

[5] Martin Arjovsky et al. (2017). Wasserstein GAN

[6] Ishaan Gulrajani et al. (2017). Improved Training of Wasserstein GAN

[7] Michaël Defferrard et al. (2017). Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering

[8] Thomas N. Kipf et al. (2017). Semi-Supervised Classification with Graph Convolutional Networks

[9] Valsesia, Diego & Fracastoro et al. (2019). Learning Localized Generative Models for 3D Point Clouds via Graph Convolution.

[10] Yipin Yang et al. (2014) . *Semantic Parametric Reshaping of Human Body Models*. In 3DV Workshop on Dynamic Shape Measurement and Analysis.

[11] Panos Achlioptas et al. (2018). Learning Representations and Generative Models for 3D Point Clouds