Signed Distance Function Based Generative Adversarial Network for 3D Shape Generation

Chiyu Jiang*
Department of Mechanical Engineering
University of California
Berkeley, CA 94720
chiyu.jiang@berkeley.edu

Madeleine Traverse †
Department of Computer Science
Cornell University
Ithaca, NY 14850
mjt249@cornell.edu

Quanlai Li‡
Department of Computer Science
University of California
Berkeley, CA 94720
quanlai_li@berkeley.edu

Abstract

Shape representation is a crucial aspect in 3D shape learning. Recent advances in 3D shape learning rely on voxelized volumetric shape representation or rendered 2D views. While these methods have achieved some success in tasks such as classification and shape retrieval, there is much room for improvement in terms of generating visually pleasing results. In this paper, we make the case that signed distance function (SDF) is a more suitable representation for many aspects of 3D shape learning, and that deep 3D convolutional generative adversarial networks (GAN) can be utilized for learning in 3D SDF domain. More interestingly, we show that our neural network is capable of learning physical field properties in an unsupervised manner.

1 Introduction

This study [1] seeks to explore and promote the use of the 3D Signed Distance Function (SDF) for 3D shape based learning tasks. We further apply it to one of the more challenging tasks, shape generation, and show that SDF based algorithms can achieve better results in both the qualitative and quantitative spectrum. We name our neural network SDF-GAN for its use of SDF for data representation. Data representation is a crucial aspect in machine learning. While the form of data representation is generally agreed upon in more conventional fields such as image based and text based learning tasks, it is yet a settled issue in 3D shape learning. Prompted by recent advances in learning algorithms such as Generative Adversarial Network (GAN) and an increasing demand for robust 3D learning algorithms for scene understanding in robotics and self-driving applications, there have been numerous recent studies that have proposed different representations of 3D shapes for learning applications, giving better visual appearance, and measurably better shape features for classification tasks. In computational geometry community, there are several commonly used forms of 3D shape representation, among which there are graph based data structures such as polygonal mesh, grid based data structures such as voxel representation and multi-view 2D images, and structureless

*Author webpage:http://cfd.me.berkeley.edu/people/chiyu-max-jiang/
†Author webpage:https://github.com/mjt249
‡Author webpage:http://www.quanlai.li

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

representation such as point cloud. Graph based and structureless representations are difficult to apply to modern advanced algorithms such as Convolutional Neural Networks (CNNs) which require a grid based data structure whose invariant spatial structures can be aggregated by parameter sharing convolutional kernels. Hence grid based data representations such as the voxel and multi-view rendered images are more commonly adopted in learning tasks. However, voxel and multi-view representations each have their inherent deficiencies. Voxel representation is a binary volumetric representation of shapes which records the binary occupancy of space in a 3D gridded domain. However binarization results in a sparse and discontinuous encoding of shape information. Moreover, in terms of shape generation, voxelized shapes present sharp edges that are not visually pleasing. Multi-view rendered 2D images are limited by the view position and fail to capture information now available to the given views (concavities etc.). Signed Distance Function, on the other hand, possess desirable properties. It can be represented as a continuous scalar function in a 3D gridded domain, performs dense encoding of shape information, and is capable of recovering smooth and fine-resolution details. Hence, it can be directly applied to 3D convolution based learning algorithms, and provide a more rich and dense information encoding than current methods. In the coming sections, section 2 will discuss some related work in the field of 3D shape representation and 3D deep learning. Section 3 will elaborate on the details of geometry processing for converting mesh-based shapes into 3D signed distance function scalar field, section 4 will go into details of the neural network model, section 5 will present the qualitative and quantitative results, and section 6 will conclude the paper and open up further discussions.

2 Related Work

In this section we present a brief overview of related work in the field of 3D shape representation, generative adversarial networks, and 3d deep learning.

2.1 Analyzing and representing 3D shapes

In the vision and graphics community, understanding and generating 3D objects is an important topic. Many researchers have contributed to this research field[1][2]. In the past decade, researchers in machine learning and computer vision have tried new ways to represent 3D objects. Most of the representations are based on meshes and skeletons. A recent approach uses pre-trained models to generate both object structure and surface[3]. Most such algorithms represent the 3D shapes with voxels. Our algorithm uses signed distance function to represent the shapes.

2.2 Generative adversarial networks

Generative adversarial networks have a generative neural network and a discriminative neural network [4]. The two neural networks co-evolve during training and the generative model can produce a better latent probabilistic model. GANs can be used in a wild range of applications. These applications are growing rapidly, especially in the vision community. In recent years, researchers adopted GANs with convolutional neural networks for image process. LAPGAN used Laplacian pyramids of adversarial networks[5]. DC-GAN did unsupervised representation learning with deep learning. Both achieved impressive results[6].

GANs can also deal with a lot of other problems. One group of researchers designed a recurrent adversarial network to generate images[7]. Others developed a Markovian GAN for real-time texture synthesis[8]. GANs are also used for generative visual manipulation[9].

Most of the previous applications of GAN to computer vision focused on 2D images. In this paper we explore the use of GAN in 3D objects, for both shape generation as well as unsupervised learning of latent shape representations.

2.3 Deep learning for 3D objects

Deep learning for 3D tasks has been growing rapidly in recent years. In 3D object recognition field, it is introduced that learning a joint embedding of 3D shapes and synthesized images[10][8]. Xiang et al, Wu et al. and Choy et al. developed a 3D object reconstruction system for in-the-wild images based on recurrent neural network[11][12][13]. Girdhar et al. and Sharma et al. discussed learning voxelized object representations based on auto-encoder networks[10][14]. Yan et al. and Rezende et al. explored deep learning for 3D object generation, some with 2D images for training and a layer for 3D to 2D projection[15][16]. Many of these approaches can also be used for 3D shape classification, retrieval, and reconstruction. Most of these approaches requires supervision. In comparison, our neural network does not rely on a labeled dataset for supervision.

3 Geometry Processing

In this section, we present the robust geometry processing algorithm that is needed to convert large amount of mesh based shape data to a learnable format (3D Signed Distance Function field). This section is dedicated to providing a brief introduction to the properties of the signed distance function and an outline for our algorithm to perform the data conversion.

3.1 Signed Distance Function

Signed distance functions (SDF) have been widely used in the computer vision community for applications such a rendering and segmentation. A mathematical definition of the signed distance function is given as follows:

$$f(x) = \begin{cases} d(x, \partial\Omega) & \text{if} x \in \Omega \\ -d(x, \partial\Omega) & \text{if} x \in \Omega^c \end{cases} \tag{1}$$

where $\partial\Omega$ denotes the boundary of Ω. For any $x \in X$,

$$d(x, \partial\Omega) := \inf_{y \in \partial\Omega} d(x, y) \tag{2}$$

where $\inf$ denotes the infimum. In Euclidean space, the signed distance function has some desirable properties, such as for piecewise smooth boundary, the signed distance function is differentiable almost everywhere, and its gradient satisfies the Eikonal Equation:

$$|\nabla f| = 1 \tag{3}$$

In Section 5.3, we will further show that our GAN model is capable of implicitly learning such field properties through examples in the training data.

3.2 Processing Algorithm

This study involves the use of two 3D geometry datasets, namely the ShapeNet dataset [17] and ModelNet (10/40) [18] dataset. The shape data in the two datasets are both given as triangular mesh format (`.obj, .off`), hence a geometry processing pipeline is needed for the conversion of triangular mesh to gridded signed distance field.

First, we center and normalize the imported triangular mesh and set up a unit size 3D spatial grid around the geometry. Then we use the data structure AABB (Axis-aligned Bounding Box) tree for efficient point-to-mesh distance query. After calculating the point to mesh distance everywhere in the domain, we compute the winding number [19] for each point in the domain to determine the sign. The algorithm is robust in that it is functional even in the case of a non watertight mesh.

In this study, we leveraged the computational geometry code infrastructure provided in the open-source library libigl [20] to facilitate this process.

4 Model

In this section, we present an overview of the generative adversarial network model we adopted for shape generation: SDF-GAN. We also describe the training procedures involved.

4.1 Model Architecture

The Generative Adversarial Network (GAN) is a class of unsupervised learning algorithms that consists of a system of two neural networks: the discriminator and the generator. It was first proposed

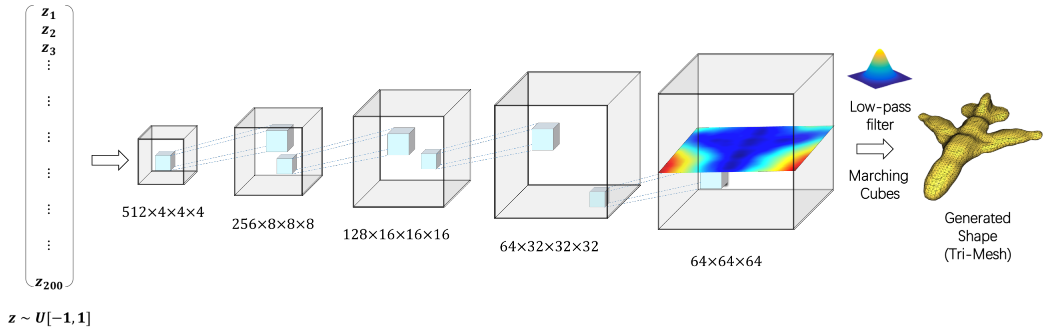

Figure 1: Scaling efficiency of GPUs used in training for steps over time trained

by Goodfellow et al. [4]. The architecture of the SDF-GAN is inspired by that of the two dimensional deep convolutional generative adversarial network (DC-GAN) [6] and voxel based 3D-GAN [21]. Like the previous two examples, our network utilizes an full-convolutional neural network for the generation of signed distance field. Unlike the previous examples, we perform additional post-processing on the results to extract a mesh-based geometry that is more refined, more visually pleasing, and provides more detailed structures.

We use the binary sigmoid cross-entropy loss as the classification loss. The overall loss can be written as a sum of the losses from the discriminator and the generator, given as:

$$L_{GAN} = \log D(x) + \log(1 - D(G(z))) \tag{4}$$

where x is the value of the signed distance function in $64 \times 64 \times 64$ of a real sample in the training set, z is a 200-dimensional vector that is sampled i.i.d. from a uniform distribution over $[-1, 1]$. $D(x)$ is the output from the discriminator, and $G(z)$ is the output from the generator in $64 \times 64 \times 64$.

The generator takes in the vector z and projects the vector to a higher dimension using linear layer and reshapes it ($512 \times 4 \times 4 \times 4$). It then passes through four full-convolution operations with kernel size $5 \times 5 \times 5$ and strides 2, with batch normalization and followed by ReLU layer after each convolutional layer. The result is then passed through the hyperbolic tangent (tanh) activation function to comply with the output range of -1 and 1. Though not a part of GAN training process, the generated results are post-processed by passing them through a low-pass gaussian filter. A triangular-meshed surface can be further extracted from the SDF field using the marching cubes algorithm. An illustration of the above can be found in Figure 1.

The discriminator is an almost mirror image of the generator network, with the difference being that instead of using ReLu, it utilizes leaky ReLU with slope of 0.2. Leaky Relu is formulated as:

$$LReLU(x) = \max(x, \alpha x) \tag{5}$$

where $\alpha < 0$ is the slope.

4.2 Training Details

We use the Adam optimizer [22] for the training of both the generator and the discriminator. In order to prevent the discriminator from overpowering the generator, we use a combination of two approaches during the training process. First, we train the discriminator with a learning rate of 2×10^{-4} and the generator with a greater learning rate of 5×10^{-4}. Second, we adopted the approach taken by Wu et al. [21] to skip the training of the discriminator when the classification accuracy in the previous iteration exceeds 80%.

5 Results

In this section we present the results from this study. We first give a qualitative display of generated results and compare them with the previous state-of-the-art. We then show that the generator is capable of learning not only the properties of the shape, but also the physical constraints associated with the signed distance function representation (i.e. the Eikonal equations in Eqn 3). Finally, we use

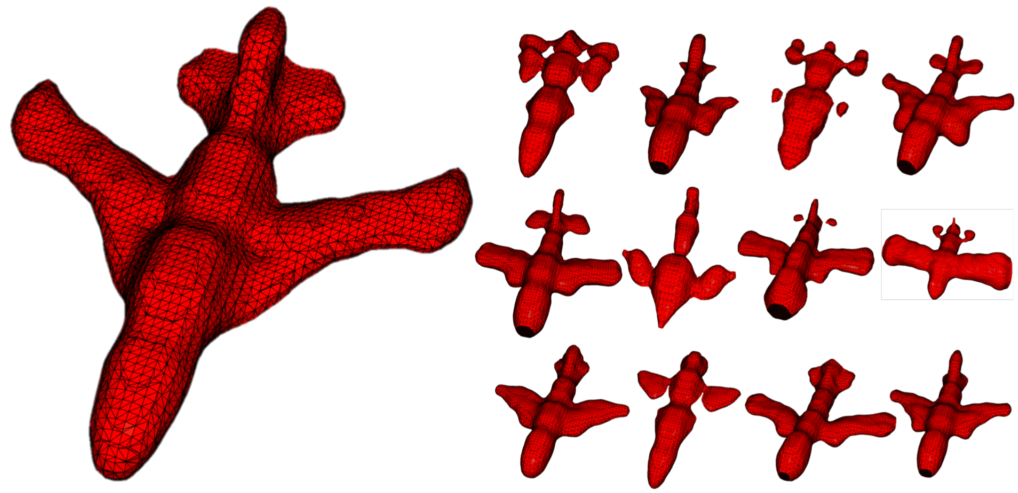

Figure 2: A sample of the 3D airplanes generated by our SDF-GAN using 200 dimensional random vectors sampled uniformly from $U[-1, 1]$. The left shows a neat zoomed-in display of a cherry-picked example. The right shows a random sample of 16 output shapes.

Figure 3: Left to right: Comparison of 3d shapes generated by SDF-GAN ($64 \times 64 \times 64$), 3D-GAN ($64 \times 64 \times 64$) [21], 3D Shapenets ($30 \times 30 \times 30$) [13]

the activation from the discriminator as features for classification tasks to illustrate the effectiveness of the latent representation learned by the SDF-GAN in an unsupervised manner.

5.1 Shape Generation

A different GAN is trained for each individual class in the ShapeNet dataset. In this study, we trained for two cases. In the first case, we trained through the 4046 examples in the airplane class of ShapeNet Dataset. In the second case, we trained through the top 7 classes in ShapeNet, namely chairs, sofas, tables, boats, airplanes, rifles and cars.

Shapes are generated based on a randomly sampled vector. A collage of random samples of generated shapes is presented in Figure 2. A 3D car generated by our neural network is compared to the results of two other previous algorithms in Figure 3.

5.2 Shape Manipulations

We performed shape manipulations by manipulating vectors in the latent space and generated resulting 3 dimensional shapes. Two examples of shape manipulations include shape arithmetic and shape interpolations. Results are displayed in Figures 4 and 5.

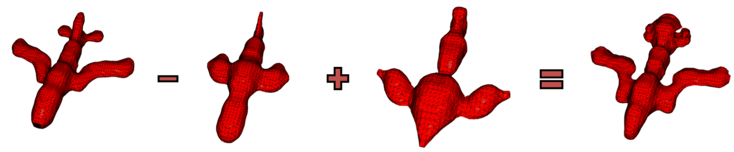

Figure 4: Shape arithmetic. Results in 3D domain show semantic arithmetic operations resulting from arithmetics in latent space.

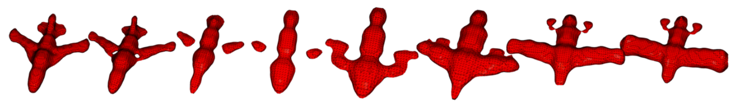

Figure 5: Shape Interpolations. Results show a smooth transition between two drastically different shapes by linear interpolation of vectors in latent space.

5.3 Unsupervised Learning of Field Properties

We also show that our generator implicitly learned about the properties of signed distance functions. Figure 6 shows the generated sign distance functions from a randomly sampled vector as well as the magnitude of the gradient across the field.

5.4 Features for Classification

The activation of the discriminator can be used as features for classification and other vision tasks that involve 3D objects. We used the features learned by our model for the classification task on the ModelNet10 dataset. The ModelNet10 dataset contains CAD models from 10 categories whose orientations have been manually aligned. The data set provides a split of training and testing sets. Using the discriminator activation and a Linear SVM model with: L2 penalty, penalty parameter C = 0.01, balanced class weights, and intercept scaling during training; we achieved an accuracy of 84.14%.

6 Conclusion

We have shown in this study that the signed distance function is a data representation that provides better resolved results for shape generation using 3D-GAN, compared to previous methods that utilize voxel representations. We have shown that generated shapes have smooth surface properties, as well as refined details. They are conceptually novel and different from examples given in the training set. We have further shown that manipulations in latent vector space can result in semantic arithmetic operations in the physical space. We have also shown that our GAN is learning not only about the geometries of the shapes, but also the properties of the signed distance functions. Finally, features learned in an unsupervised manner can be effective for transferred applications to other 3D vision tasks such as classification.

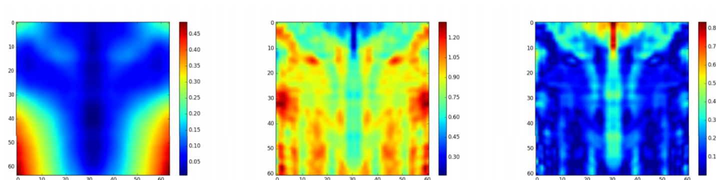

Figure 6: Left to right: a slice of the generated 3D signed distance function, the magnitude of the gradient across the field, and the error (i.e. department from unity). The average gradient error is 11.5%

Acknowledgments

We would like to acknowledge the advising and helpful input of Professor Alexei Efros. We are also grateful for Microsoft Azure for sponsoring GPU computational resources for this project.

References

[1] Wayne E Carlson. An algorithm and data structure for 3d object synthesis using surface patch intersections. *ACM SIGGRAPH Computer Graphics*, 16(3):255–263, 1982.

[2] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194. ACM Press/Addison-Wesley Publishing Co., 1999.

[3] Haibin Huang, Evangelos Kalogerakis, and Benjamin Marlin. Analysis and synthesis of 3d shape families via deep-learned generative models of surfaces. In *Computer Graphics Forum*, volume 34, pages 25–38. Wiley Online Library, 2015.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[5] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. In *Advances in neural information processing systems*, pages 1486–1494, 2015.

[6] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[7] Daniel Jiwoong Im, Chris Dongjoo Kim, Hui Jiang, and Roland Memisevic. Generating images with recurrent adversarial networks. *arXiv preprint arXiv:1602.05110*, 2016.

[8] Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European Conference on Computer Vision*, pages 702–716. Springer, 2016.

[9] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016.

[10] Rohit Girdhar, David F Fouhey, Mikel Rodriguez, and Abhinav Gupta. Learning a predictable and generative vector representation for objects. In *European Conference on Computer Vision*, pages 484–499. Springer, 2016.

[11] Yu Xiang, Wongun Choi, Yuanqing Lin, and Silvio Savarese. Data-driven 3d voxel patterns for object category recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1903–1911, 2015.

[12] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European Conference on Computer Vision*, pages 628–644. Springer, 2016.

[13] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015.

[14] Abhishek Sharma, Oliver Grau, and Mario Fritz. Vconv-dae: Deep volumetric shape learning without object labels. In *Computer Vision–ECCV 2016 Workshops*, pages 236–250. Springer, 2016.

[15] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In *Advances in Neural Information Processing Systems*, pages 1696–1704, 2016.

[16] Danilo Jimenez Rezende, SM Ali Eslami, Shakir Mohamed, Peter Battaglia, Max Jaderberg, and Nicolas Heess. Unsupervised learning of 3d structure from images. In *Advances In Neural Information Processing Systems*, pages 4997–5005, 2016.

[17] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[18] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015.

[19] Alec Jacobson, Ladislav Kavan, and Olga Sorkine-Hornung. Robust inside-outside segmentation using generalized winding numbers. *ACM Transactions on Graphics (TOG)*, 32(4):33, 2013.

[20] Alec Jacobson, Daniele Panozzo, et al. libigl: A simple C++ geometry processing library, 2016. http://libigl.github.io/libigl/.

[21] Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T Freeman, and Joshua B Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling.

[22] D Kinga and J Ba Adam. A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.