Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling

Nagesh TR
nagesh.ramamoorthy@rwth-aachen.de
Supervisor: Prof. Dr. Jürgen Gall
gall@iai.uni-bonn.de

Media Informatics,
RWTH Aachen
Computer Vision Group,Institute of Computer Science III
University of Bonn

Three-dimensional(3D) Reconstruction is a vital and challenging research topic in advanced computer graphics and computer vision due to the intrinsic complexity and computation cost. Modelling the space of 3D shapes are difficult when compared to space of 3D images due to high dimensionality. 3D shapenets[1] introduced by 3D deep learning for modellingshapes as volumetrically discritized, showed that intuitive 3D features can be learned directly in 3D. There are many artifcats that exists in generated objects(eg. fragments or holes). The problem of near-perfect image generation was smashed by the DCGAN,an application of GAN[2] in 2015 and taking inspiration from the same MIT CSAIL came up with 3D-GAN (published at NIPS '16) which generated near perfect voxel mappings. The idea was to combine Genearative Adverserial Networks and Variational Auto encoder for modelling volumetric objects, to generate objects that are novel and realistic. Modelling 3D objects in a generative-adversarial way not only makes it possible to sample novel 3D objects from a probabilistic latent space such as a Gaussian or uniform distribution but also the discriminator in the generative-adversarial approach carries informative features for 3D object recognition.

In the initial experiment the authors train a GAN using a 3D model dataset. The motivation behind using the GAN is that it will force the 3D model to look original (without holes and blurs). Generative Adversarial Network (GAN) consists of a generator and a discriminator, where the discriminator tries to classify real objects and objects synthesized by the generator, and the generator attempts to confuse the discriminator. The generator maps a 200-dimensional latent vector z randomly sampled from a probabilistic latent space to a 64 x 64 x 64 cube as G(z) in 3D voxel space. The generator upsamples the 3D voxel reshaped from the latent vector z using dense 3D up-convolution, the discriminator mirrors the generator using dense 3D convolution . 3D GAN also uses the binary cross entropy criterion as the classification loss, and hence the overall loss function for the generative adversarial network is as follows:

$$L_{3D-GAN} = \; logD(x) + \log(1 - D(G(z)))........(1)$$

where x is real 3D voxel of an object rendered from the engine. Here z is randomly sampled from an i.i.d. Gaussian Distribution p(z) between 0 to 1. The generator network structure used in the presented paper consists of five volumetric fully convolutional layers of kernel sizes 4 x 4 x 4 and strides 2, with batch normalization and ReLU layers added in between and a Sigmoid layer at the end. ADAM optimizer was used for training setting the initial learning rate to 2 x 10 - 4 and delay to be 5 x 10 - 4. The problem in training using GAN structure is that the discriminator usually learns much faster than the generator as generating objects is more difficult than differentiating it. 3D Generative Adeversarial Network (3D-GAN) Network structure is shown below:

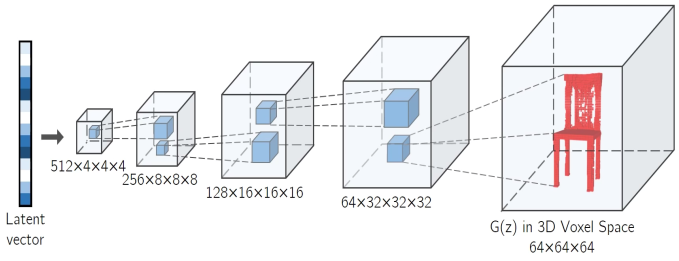

Figure 1: The generator in 3D-GAN. The discriminator mostly mirrors the generator.Image src: *Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling*

Later 3D VAE GAN was used as an extension to 3D GAN. Image encoder E was replaced in place of randomly sampling a latent vector z .It takes a 2D image x as input and outputs the latent representation vector z. The image encoder consists of five spatial convolution layers with kernel size {11, 5, 5, 5, 8} and strides {4, 2, 2, 2, 1}, respectively . As aresult of additional encoder there are different loss functions :
The overall loss used for training is as follows :

$$L = L_{3D-GAN} + \alpha_1 L_{KL} + \alpha_2 L_{recon}........(2)$$

where α_1 and α_2 are weights of the KL divergence loss and the reconstruction loss. We have

$$L_{3D-GAN} = \; logD(x) + \log(1 - D(G(z)))........(3)$$

This loss ensures that the 3D model generated looks realistic.

$$L_{KL} = D_{KL}(q(z|y)||p(z))........(4)$$

This is the variational distribution. This loss ensures that this distribution is close to the latent vector distribution.

$$L_{recon} = ||G(E(y)) - x||_2........(5)$$

This ensures that the 3D model generated represents the 2D image and not a random 3D model.

The evaluation of the model is done using 3 different perspectives . The model is tested for 3D object generation, 3d object claasification and Single image reconstruction and compared with previous works. 3D GAN is able to synthesize high-resolution 3D objects with detailed geometries. Similarly in 3D classification 3D-GAN outperforms other unsupervised learning methods by a large margin, and is comparable to some recent supervised learning frameworks . For single image 3D Reconstruction, 3D VAE GAN model tested on IKEA dataset produced better results compared to previous works. During 3D object generation, the biggest concern is to ensure that it is not overparameterized. The presented paper compare synthesized objects with their nearest neighbor in the training set. The generated objects are similar, but not identical, to examples in the training set.

The representations learned by both the generator and the discriminator of 3D-GAN are analysed. For Generative Representations three methods are used.Namely "Visualizing the object vector" - showing semantic meaning of each dimension in a vector, "Interpolation " - results of interpolating between two object vectors and "Arithmetic" - Vector space arithmetic in latent space to perform semantic operations are used. Starting with the 200-dimensional object vector, generator produces various objects. As part of the Discriminative Representation neurons in the discriminatorare analysed and demonstrate that these units capture informative semantic knowledge of the objects, which justifies its good performance on object classification. The focus of this part is to show what input objects, and which part of them produce the highest intensity values for each neuron.

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. *Generative adversarial nets*. NIPS, 2014.

[2] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. *3d shapenets: A deep representation for volumetric shapes.* CVPR, 2015.