1P-02

Automatic 3D Shape Generation from Category Information

Xu Cao† Katashi Nagao‡
Graduate School of Informatics, Nagoya University†

1. Introduction

3D shape generation is a promising application of generative adversarial network (GAN). While GANs have been used in this task, previous work focused on mapping random latent vectors to 3D shapes, that is, the generated 3D shapes are uncontrolled, which means if we want to synthesize 3D shapes from one particular category, multiple models have to be trained.

We address the issue of generating 3D shapes of our interest by providing additional category information. A novel approach for synthesizing 3D shapes given its category information by employing a conditional GAN is proposed, and it successfully generates discriminable 3D shapes from multiple categories.

2. 3D Shape Generation

2.1. 3D Shape Representation

In this work, a 3D object is represented as a volume. The space in which a 3D object lies is split into multiple grids called voxel, which is either occupied by the 3D object or not. Thus, a 3D object turns into a binary 3-dimensional tensor, and only its shape information is kept.

2.2. Generative Adversarial Network (GAN)

Among all kind of generative models, recent developments of GAN demonstrate its powerful capability of generating novel realistic samples [1]. A GAN consist of two networks, namely, a generator and a discriminator. The discriminator tries to decide whether an input sample is from the real world or the generator. On the contrast, the generator attempts to generated samples that can fool the discriminator into determining the synthesized samples as real samples. After adversarial training of the two networks, the generator becomes able to mimic the real sample distribution.

The input of the generator is a randomly sampled vector **z** from a specified distribution, usually called a latent vector.

2.3. 3D Shape Generation via GANs

Several work has been done to generate 3D shapes from a random vector by GANs, such as 3D-GAN and 3D-IWGAN [2]. Some work considers utilizing prior information. 3D-VAE-GAN convert a photo of an object into its corresponding 3D shape, while 3D-RecGAN reconstruct a complete 3D shape from a single depth view.

In this research, the category information is taken into account when generating 3D shapes.

3. 3D Shape Generation using Category Info.

3.1. Architecture

In this work, auxiliary classifier generative adversarial network (ACGAN), a variant of conditional GAN, is utilized to generate 3D shapes from category information. There are two improvements in the architecture of ACGAN compared to the vanilla GAN. First, besides the latent vector **z**, the generator of ACGAN takes a one-hot vector **c** representing category information as input. Second, an additional classifier network, of which the function is to determinate which category the input 3D shape belongs to, is stacked on top of the discriminator. Benefiting from the improvements, the generator of ACGAN is forced to generate 3D shapes matching the given category information.

The generator is a 3D transposed convolutional neural network which takes a vector concatenated by the latent vector and the one hot vector as input. Two fully connected layers and three transposed convolutional layers are stacked consecutively, mapping a low-dimensional vector to a 3D tensor gradually, as shown in Figure 1. Batch normalization is

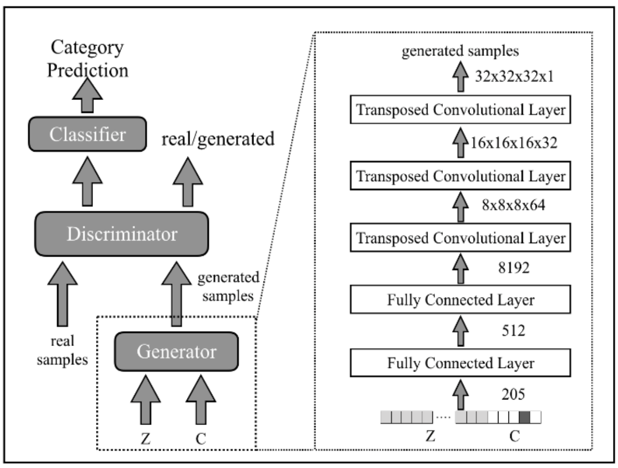

Figure1.Architecture of our model

カテゴリー情報からの三次元形状の自動生成
†曹旭（sou@nagao.nuie.nagoya-u.ac.jp）
‡長尾確（nagao@i.nagoya-u.ac.jp）
†名古屋大学　大学院情報学研究科

applied after all hidden layers. The activation function of hidden layers is leaky ReLU with leaky rate 0.2 while that of the output layer is tanh.

The architecture of discriminator almost mirrors that of the generator except that transposed convolutional layers are replaced by convolutional layers. The classifier is a neural network consisting of two fully connected layers.

3.2. Loss Function

The loss function we used to train the model in this work consist of three parts as follows:

$$L_D = -E[\log D(x)] - E\left[\log\left(1 - D\big(G(z,c)\big)\right)\right]$$

$$L_G = -E\left[\log\left(D\big(G(z,c)\big)\right)\right]$$

$$L_c = -E[\log(P(C=c|x))] - E[\log P(C=c|G(z,c))]$$

where $L_D/L_G/L_c$ represents the loss of discriminator/generator/classifier, respectively. x/ $G(z,c)$ represents real/generated 3D shapes, and $D(\cdot)$ indicates the probability of the input 3D shape being real.

Similar to the training strategy of vanilla GAN, the generator and the discriminator has different training goals, and the parameters of generator and discriminator are updated alternately. The generator tries to minimize $L_G + L_c$ while the discriminator attempts to minimize $L_D + L_c$.

4. Experiments and Evaluation

4.1. Generation Experiment

ShapeNetCore, an open 3D shape dataset, is used as training samples to train the model. ShapeNetCore contains roughly 55000 3D shapes over 55 object categories. In experiments, we choose five categories, namely, chair, table, sofa, airplane, and car to train the model because they contain the most 3D shapes.

In training procedure, we use ADAM optimizer with the default setting. The learning rate is set to 0.001 for the generator and 0.0002 for the discriminator. In consideration of computation cost, the resolution of 3D shape volume is set to 32, and a mini-batch of training samples contains 32 3D shapes. To stabilize training, 10% of real 3D shapes are labeled as generated 3D shapes, which will prevent the discriminator from converging too fast, resulting in the generator learned nothing.

The generated samples are shown in Figure 2. Each column represents 3D shapes that are generated from the same category. We find that although they are not real, they have sharp shapes discriminable to human.

4.2. Quantitative Evaluation of Quality

To evaluate the quality of samples generated by GAN quantitatively, we mainly consider how well the generated samples match the prior information, that is, to what extent the generated 3D shape will be correctly recognized as the category used to produce them. To

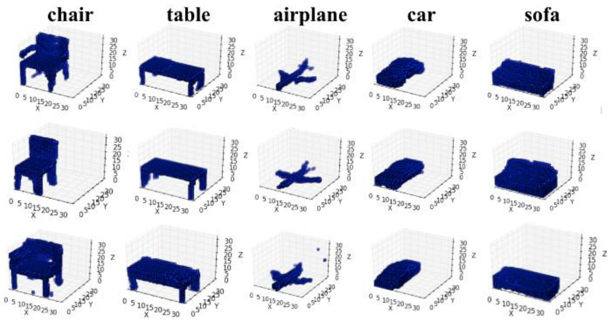

Figure2.Generated 3D shapes.

this end, we train an additional classification network to predict the category of the generated samples. We suppose that the category predicted by the network will have a high probability of being the same as that used to generate the 3D shape if the generated 3D shape has good quality. The network is trained on ShapeNetCore which is split into train/validation/test set with a ratio 7:2:1. The category used is the same as that of ACGAN and the network score 96.85% test accuracy.

We take into account the F score for each category and assume that the improvement of the quality of generated 3D shapes will increase of F score. Since during the training procedure, the classification network didn't experience the test samples and the samples generated by ACGAN, it is a reasonable comparison between them.

4.3. Results

For comparison, we randomly generate 1280 3D shapes of each category and feed them into the classification network. The F score of 3D shapes from real test set and the generator is shown in table 1.

Although generated samples are discriminable to human, they struggle to acquire high F score partly because not each latent vector can be mapped to a 3D shape with high quality.

Table 1. Comparison of F score of each category

	sofa	car	chair	table	airplane
Real	0.9080	0.9745	0.9535	0.9738	0.9756
Fake	0.1533	0.1420	0.2412	0.2723	0.1188

5. Conclusion and Future Work

We proposed a novel 3D shape generation method given its category information by employing ACGAN. The generation results show that our model can generate discriminable 3D shapes. Future work will focus on improving the quality of generated 3D shapes and introducing more prior information.

Reference

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. In Proc. of NIPS, 2014.

[2] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In Proc. of NIPS, 2016.