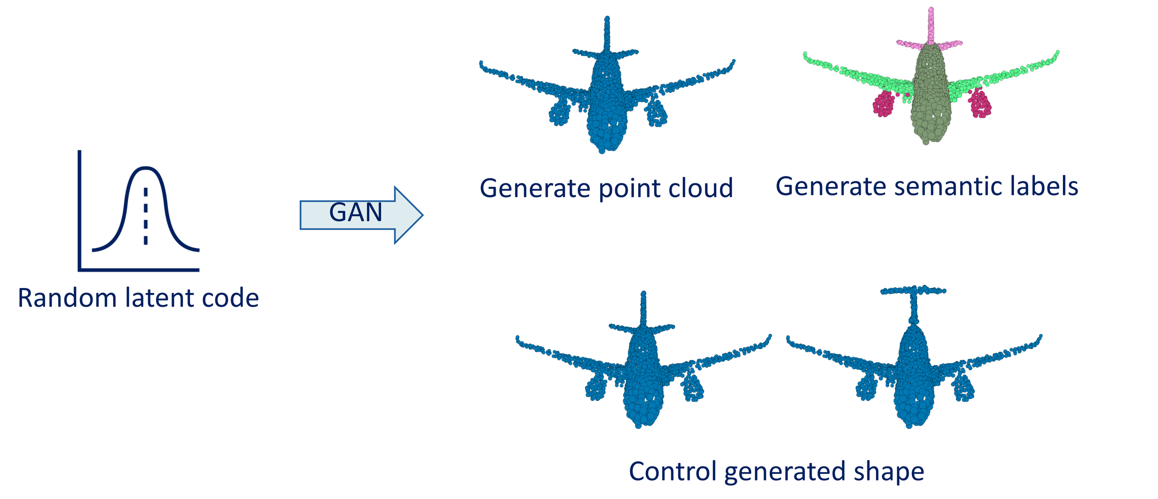
CPCGAN: A Controllable 3D Point Cloud Generative Adversarial Network with Semantic Label Generating

Ximing Yang[1], Yuan Wu[1,2], Kaiyi Zhang[1], Cheng Jin[1,2]
[1]School of Computer Science, Fudan University, Shanghai, China
[2]Peng Cheng Laboratory

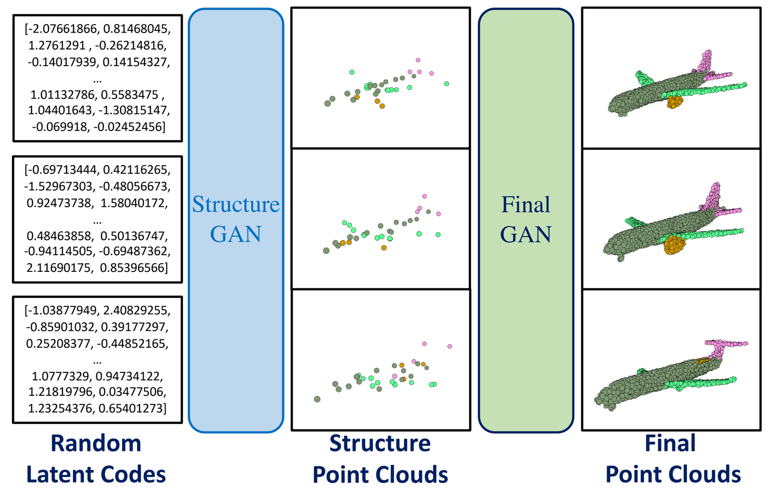
Introduction

In this paper, we present a novel method called Controllable Point Cloud Generative Adversarial Network (CPCGAN). CPCGAN can generate 3D point clouds from random latent code, generate semantic labels for points and control the generated shape.

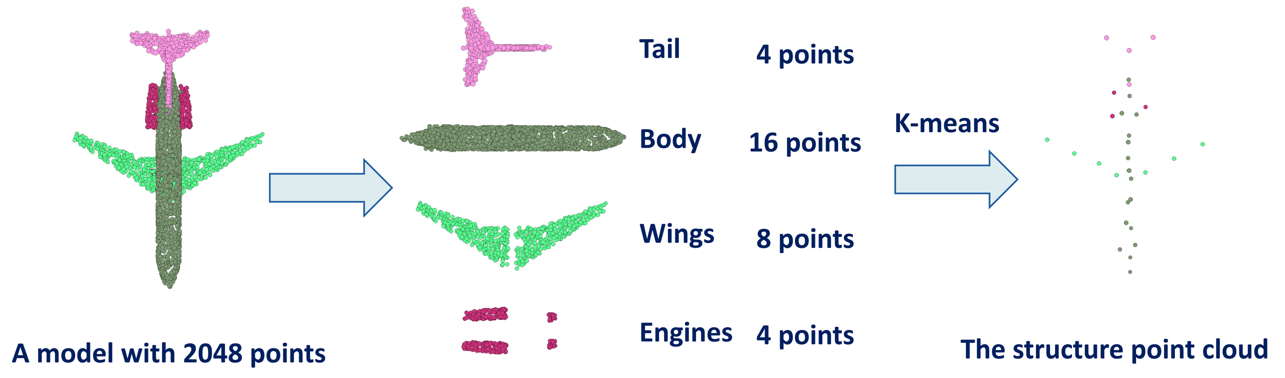


Method

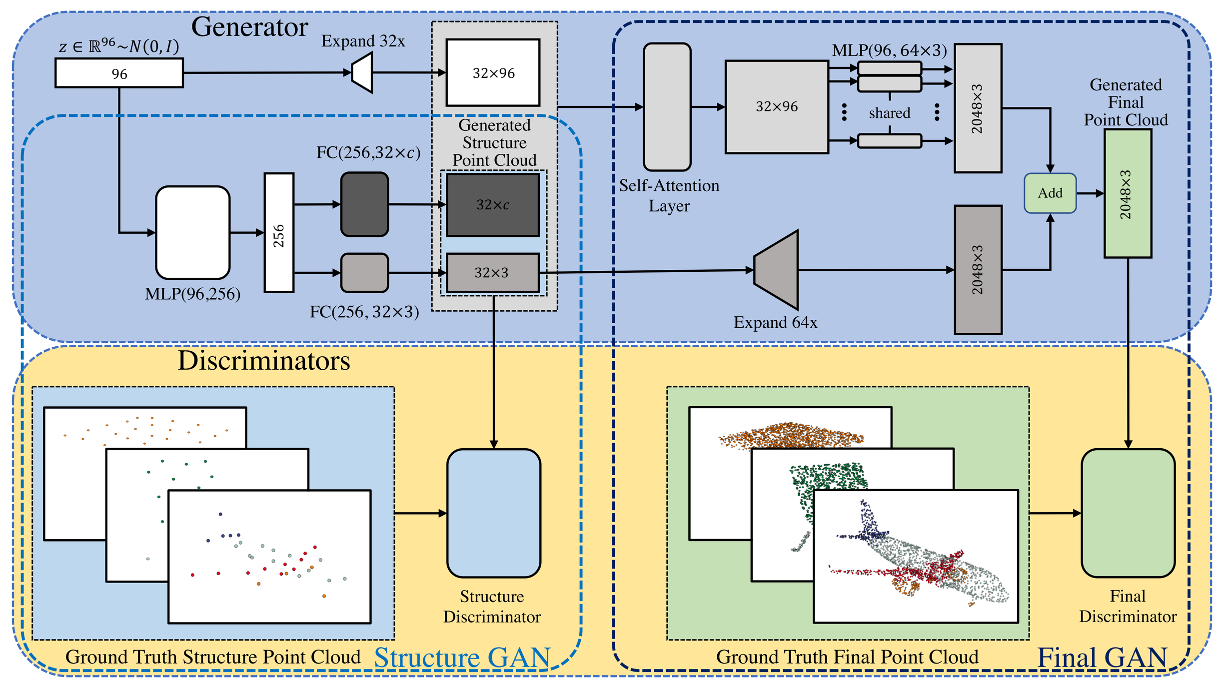
A two-stage GAN framework is proposed. The first stage GAN generates a structure point cloud as a middle-level representation. The second stage GAN generates final point cloud from the structure point cloud.



The middle-level representation is a structure point cloud with semantic labels. The ground truth of structure point clouds are extracted from the original dataset.



Extraction of the ground truth structure point cloud



Structure GAN:

- Use MLP to generate a unified vector
- Use two FC layers to generate structure point cloud and semantic labels separately
- The generated semantic labels are supposed to be a one-hot vector
- The generated information is treated as a (3+c) dimensional point cloud
- A PointNet with average-pooling is used as the discriminator

Final GAN:

- The input of the Final GAN is the combination of random latent code and generated structure point cloud
- A self-attention layer is used to transform features between points
- Shared MLPs are used to generate 64 position offsets for each structure point
- The final point cloud is generated by adding the offsets to structure points
- No semantic information will be fed into the discriminator. The final semantic labels are inherited from structure
- Another PointNet with max-pooling is used as the discriminator

Loss Function:

$$L_{G_s} = -\mathbb{E}_{z\sim Z}[D_s(G_s(z))]$$

$$L_{D_s} = \mathbb{E}_{z\sim Z}[D_s(G_s(z))] - \mathbb{E}_{x_s\sim R_s}[D_s(x_s)] + \lambda_{gp}\mathbb{E}_{\hat{x_s}}[(||\nabla_{\hat{x_s}} D_s(\hat{x_s})||_2 - 1)^2]$$

$$L_{G_f} = -\mathbb{E}_{z\sim Z}[D_f(G_f(z))]$$

$$L_{D_f} = \mathbb{E}_{z\sim Z}[D_f(G_f(z))] - \mathbb{E}_{x_f\sim R_f}[D_f(x_f)] + \lambda_{gp}\mathbb{E}_{\hat{x_f}}[(||\nabla_{\hat{x_f}} D_s(\hat{x_f})||_2 - 1)^2]$$

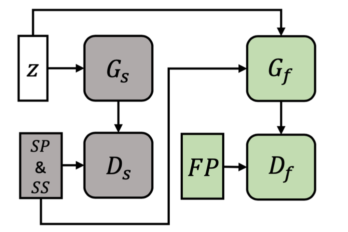
Wasserstein loss with gradient penalty

$$L_G = L_{G_f} + \lambda_s L_{G_s}$$

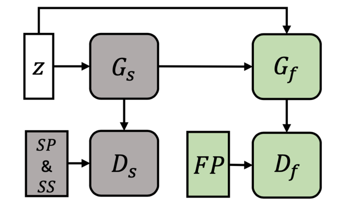
$$L_D = L_{D_f} + \lambda_s L_{D_s}$$

The final loss is a weighted sum of the losses from two GANs
$\lambda_s = 0.5$

Training Trick:



Structure GAN and Final GAN are trained separately for 500 epochs first



Then the GANs are trained together for 1500 epochs

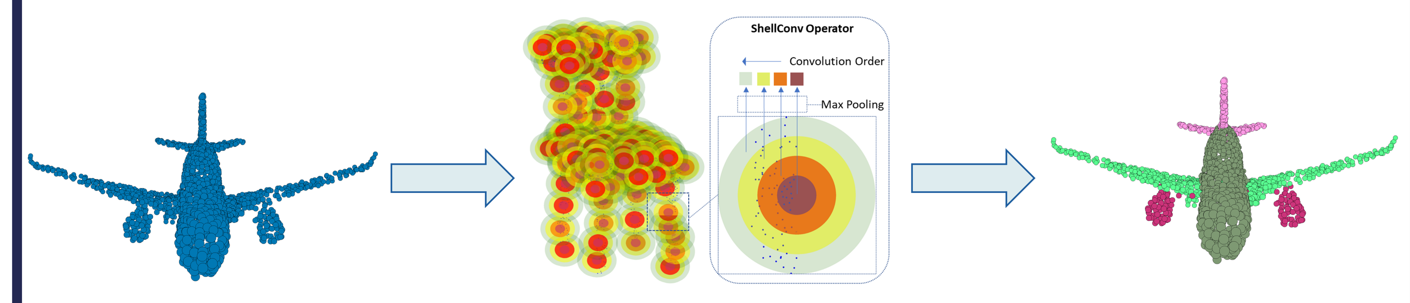
Point Cloud Generation

Class	Model	FPD↓	JSD↓	MMD-CD↓	MMD-EMD↓	COV-CD↑	COV-EMD↑
Chair	r-GAN*	1.860	0.238	0.0029	0.136	33	13
	Valsesia et al.*	-	0.100	0.0029	0.097	30	26
	tree-GAN	1.114	0.0725	0.00191	0.0900	60.21	33.92
	CPCGAN(ours)	0.877	0.0433	0.00186	0.0753	62.33	50.41
Airplane	r-GAN*	1.016	0.182	0.0009	0.094	31	9
	Valsesia et al.*	-	0.083	0.0008	0.071	31	14
	tree-GAN	0.549	0.0854	0.00039	0.0584	58.40	23.13
	CPCGAN(ours)	0.522	0.0296	0.00038	0.0417	59.63	45.36

In Earth-Move-Distance(EMD) based metrics, our CPCGAN can gain more benefit by generating more evenly in space

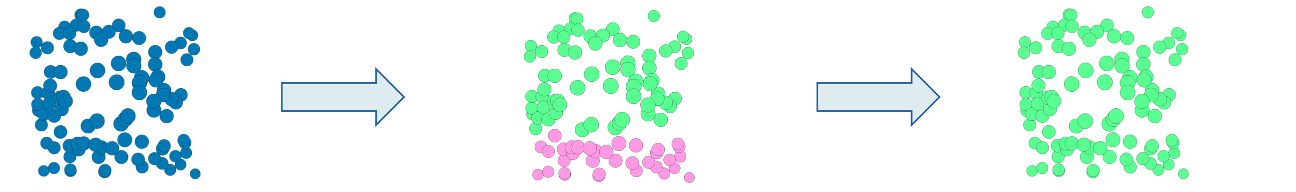
Semantic Label Generation

Generated point clouds have no semantic ground truth. Thus, we use a pretrained semantic segmentation model to generate a ground truth.



A generated point cloud
ShellNet pre-trained on the ShapeNet
Semantic ground truth for the generated point cloud

For methods like Tree-GAN which can generate clustered points but cannot generate semantic labels. We set the most common ground truth label in a cluster as the fake generated label of this cluster. Metrics based on this are marked as NL

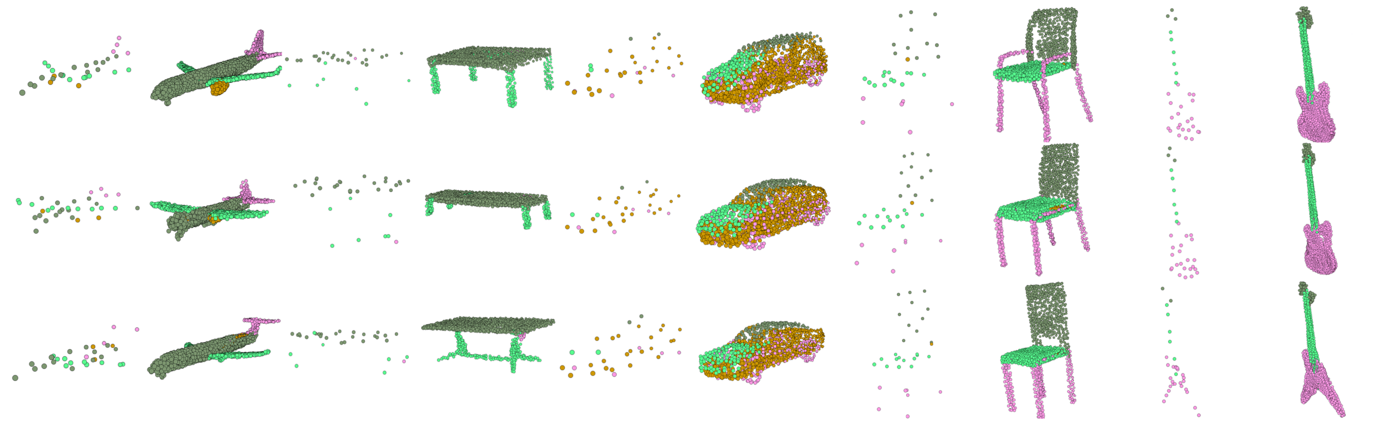


Generated cluster
Ground truth semantic label
Fake generated semantic label

Class	Model	mIoU%	fwIoU%	mAcc%
Chair	CPCGAN	59.003	76.269	84.954
Airplane	CPCGAN	69.677	76.436	85.865

Class	Model	$mIoU_{NL}$%	$mAcc_{NL}$%
Chair	tree-GAN	85.342	91.605
	CPCGAN	89.814	94.925
Airplane	tree-GAN	54.263	79.072
	CPCGAN	80.216	92.355

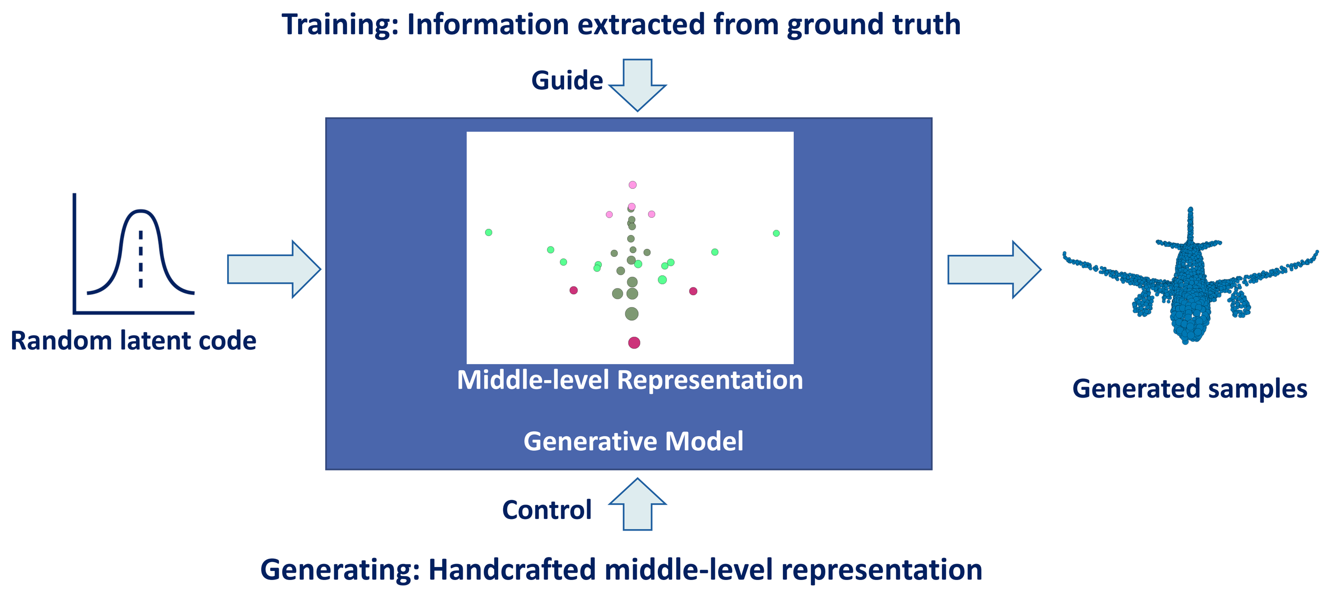
Semantic label generation performance



Generated samples, different colors denote different semantic values

Controllability

Most earlier controllable GAN using property-based method to control the generation. In CPCGAN, we proposed a novel method to control the generation.



Compare to property-based controllable generation, middle-level representation based methods can use richer ways to control the generation.

We think a good middle-level representation needs to be:

- Intuitive

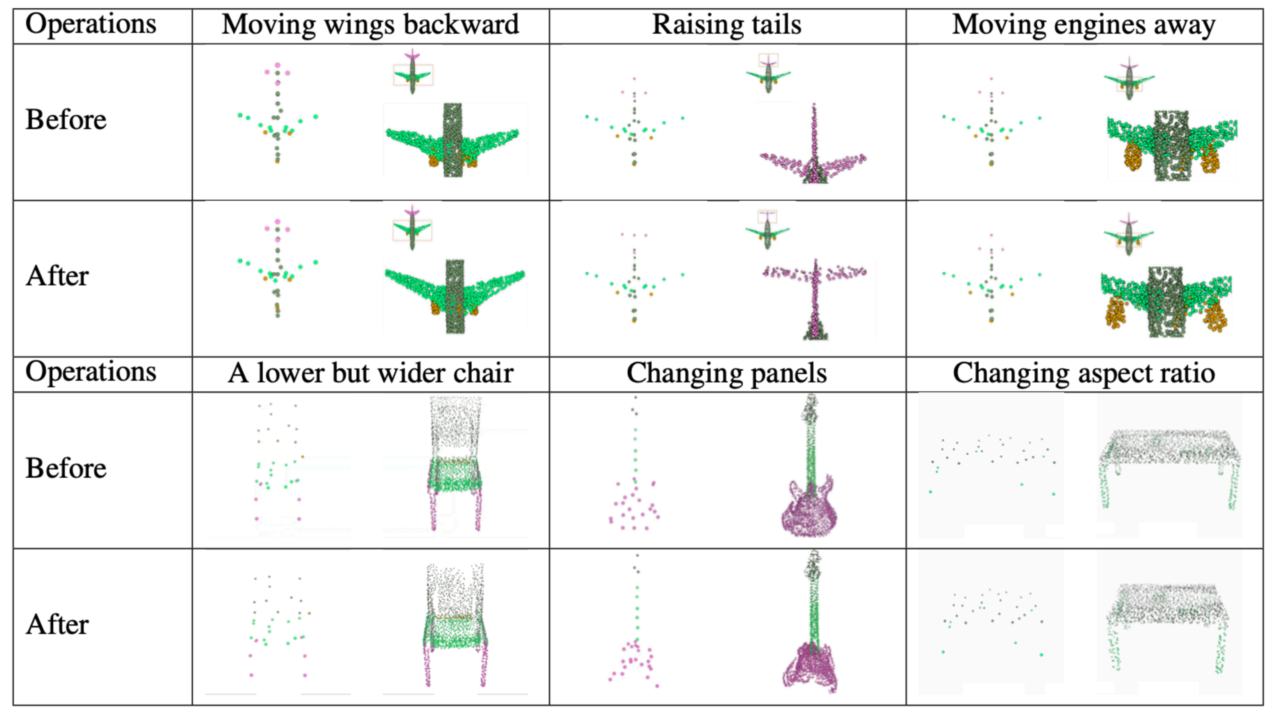
Or we can barely control the generation as our wish
In our case: A spatial and semantically intuitive structure point cloud

- Informative

Or the model cannot generate a good sample
In our case: 32 positions and 32 semantic labels in structure point cloud

- Easy to get

Or no enough data for training the model
In our case: Structure point cloud can be extracted by a simple K-means



Examples for controllable generation

Supported by National Natural Science Fund of China (61672165), Zhejiang Lab (2019KD0AB06)