PPSAN: PERCEPTUAL-AWARE 3D POINT CLOUD SEGMENTATION VIA ADVERSARIAL LEARNING

Hongyan Li[1,2], *Zhengxing Sun*[1(✉)], *Yunjie Wu*[1], *Bo Li*[1]
[1]State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210046, China
[2]Institute of Computing & Software, Nanjing Vocational College of Information Technology, Nanjing 210023, China

ABSTRACT

Point cloud segmentation is a key problem of 3D multimedia signal processing. Existing methods usually use a single network structure which is trained by a per-point loss. These methods mainly focus on the geometric similarity between the prediction results and the ground truth, ignoring visual perception difference. In this paper, we present a segmentation adversarial network to overcome the drawbacks above. A discriminator is introduced to provide a perceptual loss to increase the rationality judgment of prediction and guide the further optimization of the segmentator. In order to perfectly capture the structural information of parts in the same category of objects, condition settings are employed to add a global constraint. Experimental results show the proposed methods can correct the common errors in point cloud segmentation and obtain more accurate and better segmentation of visual perceptual.

Index Terms—shape segmentation and labeling, point cloud, adversarial learning, perceptual loss

1. INTRODUCTION

3D shape segmentation aims to divide a shape into meaningful parts and to give a label for each part. In recent years, state-of-the-art methods rely on deep neural network (DNN) approaches, achieving great success. However, these networks require inputs to be highly regular data format, while in segmentation task, a 3D shape is often represented as irregular meshes or point clouds. As DNN usually requires a large amount of training data, while point cloud is easy to acquire by various 3D scanner, point cloud segmentation becomes a trend. For this reason, we focus on point clouds segmentation and labeling in this paper.

Most existing point cloud segmentation methods use a single segmentation network structure consisting of a pair of encoding module and classification module [1-4]. Encoding module is used for point feature extraction by a convolutional-like operation. Classification module is used to identify and annotate the part attributes of each point by fully connection layers. Then, these methods utilize a cross-entropy loss. Unfortunately, the common used per-point loss focuses on the average geometric similarity, but ignores the spatial consistency constraint and cannot capture the visual perceptual differences between segmentation output and ground truth. However, geometric similarity cannot measure the quality of segmentation exactly. Furthermore, two segmentation results with the same geometric similarity may have distinctly different visual perception quality. For example, Fig. 1 shows two segmentation and labeling results. Fig. 1(b) and Fig. 1(c) have the same geometric difference with ground truth, while Fig. 1(b) is better than Fig. 1(c) in the evaluation of visual perception significantly. This visual perception difference is often reflected as three common errors, such as part internal labeling noise, fuzzy part boundaries and part missing. In mesh segmentation, the consistency of labeling between adjacent facets can be constrained by conditional random fields (CRF) post-processing, which will correct the first two types of errors [5-10]. However, for point cloud, there is no clear neighborhood between points, so it is impossible or difficult to find an appropriate solution. Therefore, how to obtain accurate, reasonable and look-well segmentation and labeling of point clouds remains challenging.

(a) GT (b) SL1 (c) SL2

Fig. 1. Two results for segmentation and labeling. GT is the ground truth. SL1 and SL2 are two predicting segmentation and labeling results. They share the same geometric difference with GT. However, when the mismarking in the parts is inside or near the boundary, it leads to different visual perception. Just as (b) is obviously more reasonable and has higher visual perception quality than (c).

Thus, we hope to introduce a perceptual loss for point clouds to guarantee the overall rationality and visual quality of segmentation and labeling. However, visual perception is hard to define mathematically. Similarly, it is also difficult to define mathematically whether the generated samples are true or not in a generation task. In 2014, the generative adversarial net is proposed to generate real samples by minimax game of generator and discriminator instead of predefining generation function [11]. Subsequently, various varieties of GAN are proposed [12-16]. In addition to theoretical research, GAN is also widely used in image gene ration [17], 3D reconstruction [18], super-resolution[19], style transfer [20] and image semantic segmentation[21][22].

This work was supported by National High Technology Re-search and Development Program of China (No. 2007AA01Z334), National Natural Science Foundation of China (Nos. 61321491 and 61272219) and Innovation Fund of State Key Lab for Novel Software Technology (Nos. ZZKT2016A11 and ZZKT2018A09).
Email: lhynju@163.com, szx@nju.edu.cn (corresponding author), JiejiangWu@outlook.com and njumagiclibo@gmail.com

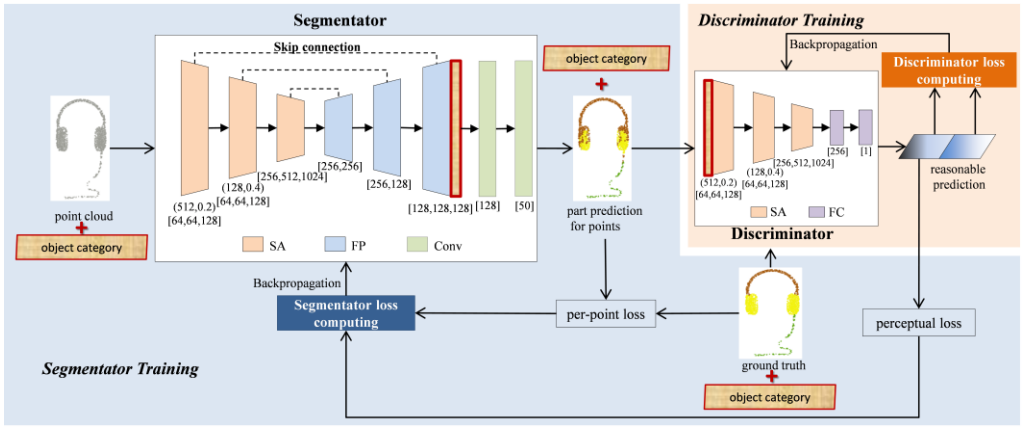

Fig. 2 Overview of the proposed adversarial learning framework for point clouds segmentation.

As for style transfer and super-resolution tasks, it has been demonstrated that GAN has the ability to understand the content and encode structure or perception loss [20][21][23].

In this paper, we introduce the adversarial thoughts into the field of point clouds segmentation and modify the traditional network structure by adding a discriminator. Thus, a point cloud segmentation framework is proposed, which is called perceptual-aware point cloud segmentation adversarial network (PPSAN). The segmentator is trained by using a hybrid loss function that is a weighted sum of per-point geometric loss and adversarial perceptual loss. It takes point cloud as input and outputs point cloud that encodes semantic part information about the object. The discriminator is trained to determine whether the input sample is segmentation prediction result or ground truth, and provides adversarial perceptual loss. In order to perfectly capture the structural information of the parts in the same category of objects, we extend our approach to use global constraint, i.e. object category label, by employing conditional PPSAN (cPPSAN).

In summary, our contributions are given as follows:

- A segmentation adversarial framework PPSAN is proposed to segmentation and labeling point cloud. With a perceptual loss provided by the discriminator, the quality of visual perception of segmentation results is concerned.
- We extend the PPSAN to the conditional setting, i.e. cPPSAN, which will further direct the segmentation and labeling process and produce more reasonable labeling results.
- Experimental results show that the proposed method can correct three common types of segmentation and labeling errors. Thus, more reasonable and better segmentation prediction results of visual perceptual quality can be obtained.

2. METHOD

2.1. Overview

The proposed framework for point clouds segmentation is illustrated in Fig. 2. The network architecture consists of segmentator and discriminator. Segmentator extracts point feature and predicts the part label for each point. Per-point loss between the prediction output of segmentator and point clouds ground truth is computed. Discriminator takes the ground truth and segmentation prediction results as inputs respectively. It will output reasonable predictions for the two inputs. The reasonable prediction can determine whether the input of discriminator is a predicting result or the ground truth. So the discriminator loss can be computed by two parts of reasonable predictions and their expectation targets, which can be used for discriminator training. The reasonable degree of part prediction results provides perceptual loss. The segmentator loss is computed by the weighted sum of per-point loss and perceptual loss and can be used for segmentator training. In the training process, segmentator and discriminator is training alternately in each iteration, while in the testing process, the inference can only adopts the trained segmentator.

2.2. Network structure

Segmentator: In order to apply the adversarial learning framework, we adopt a segmentation module similar to PointNet++ [2] as our segmentator and design a discriminator. The architecture of the segmentator network consists of three SA layers, three FP layers and two convolution layers. The set abstraction (SA) and feature propagation (FP) layers are used for point feature extraction. The convolution layers are used to identify point part category by one-dimensional convolution (conv1d). In detail, the SA layer contains sampling, grouping and MLP. As shown in Fig. 3, sampling is to select a subset of points from input points by farthest point sampling. Grouping is to search a neighborhood range for each sampled point by ball query. MLP is adopted to encode feature of each point in the local region. Then, max pooling is used to give the feature vector for the centroid sampled point. Since the SA layers make fewer and fewer points involved in the calculation, it is necessary to design the FP layer as up sampling to get the feature vector of each point. The FP layer uses the average

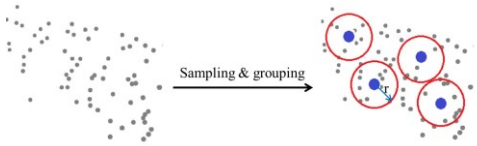

Fig. 3 Sampling and grouping process. A subset of points from input points is obtained by farthest point sampling. In the grouping process, points in a sphere with a sampled point as the center and the radius as r constitute a group.

of inverse distance weighted to compute feature vector of points that near the centered point. In this process, shallow features are added through skip connection in each FP layer. And then MLP is adopted to further encode the feature. Moreover, in order to perfectly capture the structural information of parts in the same category of objects, we add a global constraint, i.e. object category label, to point feature in the last FP layer output. Finally, two one-dimension convolution layers are used to classify points and get the part label prediction of each point. The special parameters of the entire network architecture are shown in Fig. 2.

Discriminator: The discriminator takes the object category label and point cloud with part label maps as inputs. Part label maps are from ground truth and prediction results of segmentator output respectively. The discriminator network architecture consists of three SA layers and three fully connected (FC) layers. The output of the discriminator is to identify the part label maps is ground truth or segmentation prediction, which can reflect the reasonable and visual perceptual quality of segmentation results. The definition of the SA layer is the same as the one in segmentator network. The difference is that the activation function adopts Leaky-ReLu [24]. The special parameters of the entire network architecture are shown in Fig. 2. The round brackets denote the number of points and radius of search ball in each layer, while square brackets give the output dimensions of each layer in the neural network.

2.3. Network training

Given a data set of N training point clouds x_n and a ground truth label map y_n, the discriminator and segmentator are trained alternately in each iteration. The optimization goal of the Discriminator network is to minimize L_D:

$$L_D = \sum_{n=1}^{N}\left[L_{bce}\left(D\left(x_n, y_n\right),1\right) + L_{bce}\left(D\left(x_n, \mathrm{S}\left(x_n\right)\right),0\right)\right] \quad (1)$$

where $S(\bullet$ denotes the output of the segmentation network and $D(\bullet$ is the output of the discriminator network. The L_{bce} represents the perceptual loss of input part label maps. When the input of discriminator is ground truth, L_{bce} is defined as $L_{bce}(\hat{z},1) = -\ln \hat{z}$. While when the input of discriminator is segmentation prediction label map, it is defined as $L_{bce}(\hat{z},0) = -\ln(1-\hat{z})$.

Training segmentator is based on a hybrid loss that is a weighted sum of per-point loss and perceptual loss. The per-point loss is to focus on the geometric similarity on each point independently. The perceptual loss is the output of segmentation prediction through discriminator, which reflects the reasonable of the whole prediction results in visual perception. Thus, the segmentator loss is defined as:

$$L_S = \sum_{n=1}^{N}\left[L_{mce}\left(S\left(x_n\right), y_n\right) + \lambda L_{bce}\left(D\left(x_n, S\left(x_n\right)\right),1\right)\right] \quad (2)$$

where $L_{mce}(\hat{y}, y) = -\left[y \ln \hat{y} + (1-y)\ln(1-\hat{y})\right]$ denotes the per-point loss that computed by multi-class cross-entropy loss for segmentation prediction $\hat{y}$.

In order to further improve the labeling quality, we add the global constraint to guide segmentation and labeling. Thus, the PPSAN is implemented in condition settings, becoming cPPSAN. The L_D and L_S in the training process becomes:

$$L_D = \sum_{n=1}^{N}\left[L_{bce}\left(D\left(x_n, y_n, l\right),1\right) + L_{bce}\left(D\left(x_n, \mathrm{S}\left(x_n\right), l\right),0\right)\right] \quad (3)$$

$$L_S = \sum_{n=1}^{N}\left[L_{mce}\left(S\left(x_n\right), y_n, l\right) + \lambda L_{bce}\left(D\left(x_n, S\left(x_n\right), l\right),1\right)\right] \quad (4)$$

where l is the object category label, encoding in the one-hot vector. Generally, the discriminator network often converges quickly in practice. Therefore, when discriminator accuracy achieves a certain threshold, the updating of the discriminator network is terminated, and only the segmentator parameters are still updated.

3. EXPERIMENTS

3.1. Experimental setup

The proposed method is evaluated on ShapeNet part dataset from [25], which including 16,880 shapes from 16 object categories, annotated with 50 parts with 2 to 6 parts on per category. We mix the 16 categories of shapes and train our network for all the categories. Following existing works[2], the official train/test split is used. As evaluation metrics, mIoU is emplyed, which is an average of point Intersection over Union across all part classes.

We implement our approach using TensorFlow and train the network on GeForce GTX 1080 GPU. To train the proposed network, Adam optimizer is adopted, where the momentum is set to 0.9, the initial learning rate is set to 0.001, and the exponential decay is set to 0.7. For discriminator training, updating procedure is terminated the training accuracy reaches 0.85. For segmentator training, the weight of the adversarial item is set to 0.06.

3.2. Experimental results

We evaluate our framework in two settings, with or without global constraint, and compare the results with state-of-the

Table 1 Segmentation results compared with state-of-the-art methods on the ShapeNet part dataset.

Methods	aero	bag	cap	car	chair	ear phone	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	Skate board	table	mean	Mean (all shapes)
Yi [25]	80.96	78.37	77.68	75.67	87.64	61.89	91.79	85.36	80.59	95.58	70.59	91.85	85.94	53.13	69.81	75.33	**78.89**	**81.43**
PointNet[1]	83.40	78.70	82.50	74.90	89.60	73.00	91.50	85.90	80.80	95.30	65.20	93.00	81.20	57.90	72.80	80.60	**80.39**	**83.70**
RSNet[3]	82.70	**86.40**	84.10	78.20	90.40	69.30	91.40	87.00	83.50	95.40	66.00	92.60	81.80	56.10	**75.80**	82.20	**81.43**	**84.90**
Dynamic GCN[4]	**84.20**	83.70	84.40	77.10	90.90	78.50	**91.50**	**87.30**	82.90	96.00	67.80	93.30	82.60	**59.70**	75.50	82.00	**82.34**	**85.10**
PointNet++ (SSG)[2]	82.60	81.72	**86.46**	77.71	**90.75**	68.25	90.92	85.62	82.92	95.47	71.01	94.77	**82.77**	55.90	75.26	81.80	**81.50**	**84.74**
Ours1 (PPSAN)	82.89	80.26	83.99	**78.99**	90.56	74.07	90.70	85.70	83.26	95.36	73.62	95.08	81.16	56.52	75.45	81.20	**81.80**	**85.07**
Ours2 (cPPSAN)	82.90	82.67	86.36	78.90	90.56	**76.50**	91.02	85.69	**84.31**	**96.10**	**74.42**	**95.09**	81.82	58.25	75.46	**82.53**	**82.66**	**85.18**

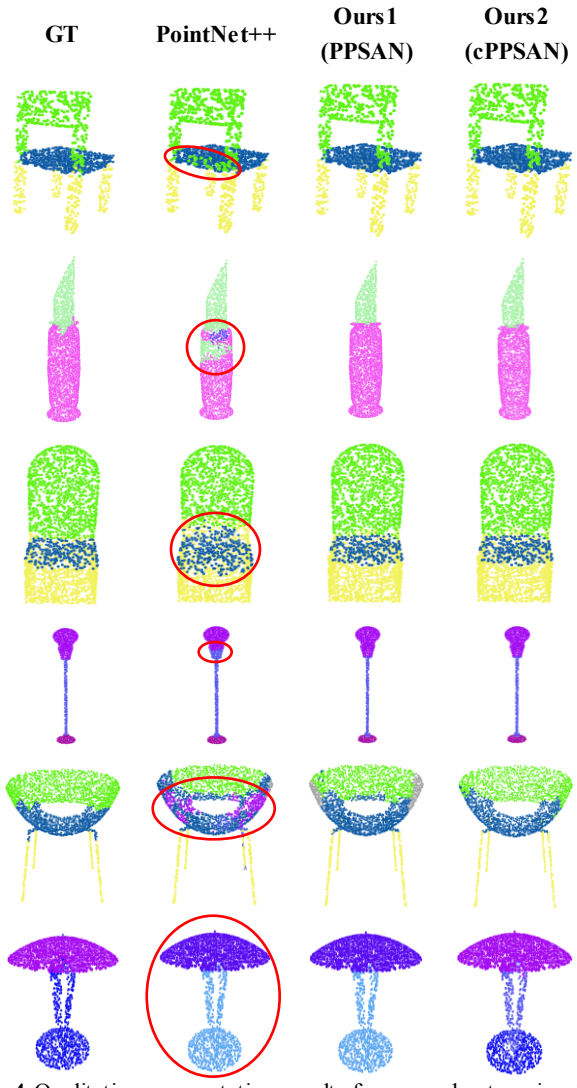

Fig. 4 Qualitative segmentation results for several categories. The segmentation and labeling results of ground truth, PointNet++[2], ours1 and ours2 are given in the first to four columns. Ours1 represents to integrate PointNet++ as a baseline into our adversarial learning framework, i.e. the PPSAN. Ours2 corresponds to our full pipeline i.e. the proposed cPPSAN, which extends the PPSAN to conditional PPSAN by adding global constraints. (Different parts are shown in different colors.)

-art methods in quantitative and qualitative. Table 1 shows the quantitative comparison results. All values are reported on the same point cloud dataset, by mIoU. On average, our approach achieves the best performance. Qualitative segmentation and labeling results of PointNet++ [2] and our approach for several categories are shown in Fig. 4**.** As shown in Fig. 4, the design of the segmentation adversarial network framework makes the proposed method pay more attention to the overall visual perception quality. While the added global constraint further improves the effect of label assignment. As a result, our approach can correct three types of segmentation and labeling errors. For example, rows 1 and 2 show that PPSAN can correct the internal labeling noise of one part. Rows 3 and 4 show that PPSAN smoothes the segmentation boundaries of two parts. As can be seen from rows 5 and 6, PPSAN can optimize the segmentation results for part missing error. As shown in the 5th row, PPSAN can correct errors that certain points are incorrectly labeled as parts in another object category. While cPPSAN can further optimize the segmentation and labeling results. However, as seen from the 6th row when the object is more similar to another category in overall shape, PPSAN can segment the parts, but often cannot give the correct labels. At this point, cPPSAN with global constraint can effectively correct the labeling error.

4. CONCLUSION

In this paper, we propose a framework PPSAN for shape segmentation. The framework contains two sub-networks: segmentator network and discriminator network. The segmentation network is used to segment point cloud. Then the prediction label or ground truth will feed to the discriminator network. The discriminator network is used to distinguish the segmentation prediction and ground truth. The whole framework introduces a hybrid loss to train the segmentator, which not only pays attention to a per-point loss but also a perceptual loss provided by the discriminator. The proposed framework can be integrated into some advanced segmentation network to improve the results further. Experimental results show the effectiveness of the proposed approach.

5. REFERENCES

[1] Charles Ruizhongtai Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," *Conference on Computer Vision and Pattern Recognition* ., pp. 77–85, 2017.

[2] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," *Conference on Neural Information Processing Systems*., pp. 5105–5114, 2017.

[3] Q. Huang, W. Wang, and U. Neumann, "Recurrent Slice Networks for 3D Segmentation of Point Clouds," *Conference on Computer Vision and Pattern Recognition*, no. 1, pp. 2626–2635, 2018.

[4] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic Graph CNN for Learning on Point Clouds," *eprint arXiv:1801.07829*, 2018.

[5] T. Le, G. Bui, and Y. Duan, "A multi-view recurrent neural network for 3D mesh segmentation," *Computers &Graphs.*, vol. 66, pp. 103–112, 2017.

[6] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri, "3D Shape Segmentation with Projective Convolutional Networks," *Conference on Computer Vision and Pattern Recognition*., pp. 3779–3788, 2017.

[7] K. Guo, D. Zou, and X. Chen, "3D Mesh Labeling via Deep Convolutional Neural Networks," *ACM Transactions on Graphics.*, vol. 35, no. 1, pp. 1–12, 2015.

[8] H. Xu, M. Dong, and Z. Zhong, "Directionally Convolutional Networks for 3D Shape Segmentation," *IEEE International Conference on Computer Vision.*, no. July, pp. 2717–2726, 2017.

[9] P. Wang *et al.*, "3D shape segmentation via shape fully convolutional networks," *Computers & Graphics.*, vol. 70, pp. 128–139, 2018.

[10] E. Kalogerakis, A. Hertzmann, and K. Singh, "Learning 3D mesh segmentation and labeling," *ACM Transactions on Graphics.*, vol. 29, no. 4, p. 1, 2010.

[11] I. Goodfellow *et al.*, "Generative Adversarial Nets," *Conference on Neural Information Processing Systems.*, pp. 2672–2680, 2014.

[12] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," arXiv:1411.1784, pp. 1–7, 2014.

[13] E. Denton, S. Chintala, A. Szlam, and R. Fergus, "Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks," *Conference on Neural Information Processing Systems*, pp. 1–9, 2015.

[14] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *International Conference on Learning Representations*, pp. 1–16, 2016.

[15] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least Squares Generative Adversarial Networks," *IEEE International Conference on Computer Vision*., pp. 1–16, 2016.

[16] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," *International Conference on Machine Learning*., 2017.

[17] E. Denton, S. Chintala, A. Szlam, and R. Fergus, "Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks," *Conference on Neural Information Processing Systems*, pp. 1–9, 2015.

[18] J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum, "Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling," *Conference on Neural Information Processing Systems*, pp. 82–90, 2016.

[19] C. Ledig *et al.*, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," *Conference on Computer Vision and Pattern Recognition* ., pp. 4681–4690, 2017.

[20] Y. Wang, L. Xie, S. Qiao, Y. Zhang, W. Zhang, and A. L. Yuille, "Perceptual Losses for Real-Time Style Transfer," *European Conference on Computer Vision*, pp. 694–711, 2018.

[21] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic Segmentation using Adversarial Networks," *Conference on Neural Information Processing Systems*, 2016.

[22] N. Souly, C. Spampinato, and M. Shah, "Semi and Weakly Supervised Semantic Segmentation Using Generative Adversarial Network," *International Conference on Computer Vision*, 2017.

[23] P. Isola, J.-Y. Zhu, T. Zhou, A. A. Efros, and B. A. Research, "Image-to-Image Translation with Conditional Adversarial Networks," *Conference on Computer Vision and Pattern Recognition* ., pp. 5967–5976, 2017.

[24] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier Nonlinearities Improve Neural Network Acoustic Models," *International Conference on Machine Learning.*, vol. 28, p. 6, 2013.

[25] L. Yi *et al.*, "A scalable active framework for region annotation in 3D shape collections," *ACM Transactions on Graphics.*, vol. 35, no. 6, pp. 1–12, 2016.