Optimized PointNet for 3D Object Classification

Zhuangzhuang Li[1], Wenmei Li[1,2(✉)], Haiyan Liu[1], Yu Wang[1], and Guan Gui[1]

[1] College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
liwm@njupt.edu.cn

[2] School of Geographic and Biologic Information, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

Abstract. Three-dimensional (3D) laser scanning technology is widely used to get the 3D geometric information of the surrounding environment, which leads to a huge increase interest of point cloud. The PointNet based on neural network can directly process point clouds, and it provides a unified frame to handle the task of object classification, part segmentation and semantic segmentation. It is indicated that the PointNet is efficient for target segmentation. However, the number of neural network layers and loss function are not good enough for target classification. In this paper, we optimize the original neural network by deepen the layers of neural network. Simulation result shows that the overall accuracy increases from 89.20% to 89.35%. Meanwhile, the combination of softmax loss with center loss function is adopt to enhance the robustness of classification, and the overall accuracy is up to 89.95%.

Keywords: Point cloud · PointNet · Object classification · Center loss

1 Introduction

The three-dimensional (3D) images contain special information that allows the object to be naturally separated from the background. In recent years, the 3D coordinates of object surface can be acquired accurately and quickly with the development of 3D imaging technology [1]. Point clouds are better than images to describe the objects because they contain the largest amount of raw information. However, it is difficult to process point clouds directly. Since point clouds are not irregular and uneven distribution [2].

Deep learning has been applied in many fields, such as computer vision, natural language processing and wireless communications. Deep learning can automatically learn how to extract features from inputted raw data using deep neural networks [3–6]. Convolutional neural networks (CNN) have made great achievements in image classification and segmentation. Most researchers have changed their strategies and started to convolve with raw point clouds after transforming raw data to regular 3D voxel grids. They feed raw data to 3D CNN for feature extractions to segment point clouds [7]. Some scholars have tried to use a multi-view method to take two-dimensional

G. Gui and L. Yun (Eds.): ADHIP 2019, LNICST 301, pp. 271–278, 2019.
https://doi.org/10.1007/978-3-030-36402-1_29

images of the same object from different angles and then convolve with images to extract features [8, 9].

Charels et al. proposed a new novel type of neural network named PointNet. The main idea of PointNet is to process raw point clouds by a mini-network, feed them to the network, and finally extract global features with max pooling [7]. Ge et al. proposed the Hand PointNet network, which can directly consume 3D point clouds for hand regression. The gesture regression network can capture complex hand structures and accurately return to a low dimensional representation of the 3D hand [10].

The loss function is a way to measure the difference between the predicted output of neural network and labels. Wen et al. put forward a new loss function, namely center loss, to address the face recognition task. It is able to find a center for deep features of every class. With the center updating every time, the differences between deep features become smaller. If the distances are too far, they will be paid the penalty [11]. With the combination of the center loss and softmax loss, the highly discriminative deep features are obtained. Meanwhile, center loss is easy to realize in the CNN.

In this paper, we would optimize the network from two aspects to improve the performance of classifier model. First, the number of layers is increased in the hidden layer to get more abstract features. Second, the joint supervision of the center loss and softmax loss are applied to judge the score. The performance of model is improved with a proper parameter. The contributions of this paper can be summarized as below:

(a) Optimize the PointNet by increasing the numbers of hidden layers to get more abstract features.
(b) The joint supervision of softmax loss function and center loss have been put forward to extract deep features on the PointNet, which improves the performance of the model.

2 Related Work

In recent years, most researchers dedicated to work on driverless car technology and augmented reality. It is especially important to understand the applications of 3D scenes. 3D data has many popular representations such as point clouds, mesh, volumetric, and multi-view images. Point clouds are used more and more widely because it is close to raw sensor data and representationally simple. Volumetric CNNs are the pioneers to apply 3D CNN to process point clouds. The main idea of volumetric CNNs is to transform unordered point cloud into regular voxel grids, and apply 3D CNN to learn the features automatically. It shows good performance in classification tasks. However, it is challenging to spend more time to compute due to high space and time complexity [7]. In addition, this method results in the loss of local information, which is difficult to apply to object detection and location determination. Multi-view CNNs are different with volumetric CNNs, which apply 2D CNN to classify them. Multi-view CNNs can get more and more abstract features than a single image. However, multi-view CNNs are too constrained by 2D images due to the loss of space information to

understand scene information [12]. The k-means algorithms are applied into unsupervised classification works with point clouds, but these methods are constrained on the accuracy.

The above methods transform point clouds into 3D voxels and image grids, which result in the loss of geometric information. Charels et al. put forward a new novel type of neural network that can directly process raw point clouds [13]. The PointNet provides a unified structure for object classification, part segment, and semantic scene segment. It uses max pooling to aggregate the information from each point regardless of the input order. Although it is very simple, the PointNet shows strong performance on the test set [13]. Ge et al. improved the PointNet to capture complex hand structures and accurately regress a low dimensional representation of the 3D hand [10].

The loss function is used to estimate the differences between the predicted and labels of the model. The smaller the loss function is, the stronger the robustness of the model is. Generally, we choose cross entropy loss function, but the disadvantage of softmax is not enough focused on the classification results of boundary points. So contrastive loss was put forward in the siamese network. It can reduce dimensionality reflected in the process of feature extraction [14]. Triplet loss sets the threshold between intra-class differences and intra-class distances based on contrastive loss. The neural networks extract features automatically, which can enlarge intra-class feature distances and inter-class feature variations [15].

3 Methodology

In this section, we would introduce two approaches to optimize the PointNet. The first method is based on increasing the number of hidden layers to extract abstract features. Furthermore, inspired by the improvement of the loss function in the face recognition, the second approach is to combine softmax loss function with center loss to enhance discriminative ability.

A. Increase the number of convolution layer

A convolution layer is added into the original network. The optimized PointNet architecture is shown in Fig. 1. The input represents inputting 3D raw point clouds ($n \times 3$) and the raw point clouds are transformed by affine transformation matrix (3×3) to implement the data alignment. The aligned data is used to extract deep features by three-layer perception (64, 64, 64) with shared parameters. Each point extracts 64-dimensional features. The affine transformation matrix is predicted by the extracted features. In addition, the three-layer perception (64, 128, 1024) is continued used for feature extractions and aggregate point global features by max pooling. Finally, the fully connected layers are used to classify the features vectors. And the output represents classification scores for 40 classes.

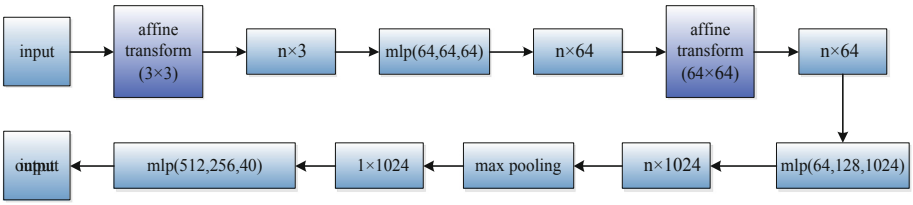

Fig. 1. Optimized PointNet.

B. The joint supervision of softmax loss and center loss

Cross entropy is an important branch of shannon's information theory. It is mainly used to measure the differences between two probability distributions. The cross entropy has two characteristics: one is non-negative, and the other is close to zero when the predicted output is close to label. The softmax converts the output of the neural network into a probability and the distances between the logits and labels are generated. The softmax loss function is formulated in Eq. 1.

$$L = -\sum_{i=1}^{M} \log \frac{e^{Wyi^T xi + byi}}{\sum_{j=1}^{n} e^{Wyj^T xi + bj}} \tag{1}$$

The $\boldsymbol{x_i}$ denotes the ith feature in the y_ith class. $\boldsymbol{W_j}$ is the jth column of the weights and $\boldsymbol{b}$ is the bias term. The deep features extracted by CNN are not only separable but also higher distinguishability. If the deep features are more discriminative, the generalization performance of model will be better. The center loss function could be applied to minimize the intra-class distances and the formula of center loss is shown in Eq. 2.

$$L_c = \frac{1}{2}\sum_{i=1}^{m} \|xi - cyi\|_2^2 \tag{2}$$

Where cyi is the yith class center of deep features. The equation effectively demonstrates the intra-class variations. It is challenging for the whole training set to update the center when the deep features change. So two necessary modifications are made to solve this problem. First, the center is updated based on mini-batch rather than the whole training. Second, a parameter α is used to control the learning rate of the centers. The formulas for the gradients of L_c and the update of cyi are as follows:

$$\frac{\partial L_c}{\partial x_i} = x_i - c_{yi} \tag{3}$$

$$\Delta c_j = \frac{\sum_{i=1}^{m} \delta(y_i = j) \cdot (c_j - x_i)}{1 + \sum_{i=1}^{m} \delta(y_i = j)} \tag{4}$$

Many experimental studies have shown that the central loss function individually does not perform well in classification tasks. The combination of softmax loss function and

center loss is used to train the neural network for discriminative feature extraction. Softmax loss function can enlarge the inter-class distances, and center loss reduce the intra-class differences. The joint supervision of softmax loss function and center loss is formulated as:

$$\begin{aligned} L &= LS + \lambda LC \\ &= -\sum_{i=1}^{M} \log \frac{e^{Wyi^T xi + byi}}{\sum_{j=1}^{n} e^{Wyj^T xi + bj}} + \frac{\lambda}{2} \sum\nolimits_{i=1}^{m} \|xi - cyi\|_2^2 \end{aligned} \tag{5}$$

The parameter λ can control the weight of softmax function and center loss. Different λ can make great differences in the deep feature distribution. A proper λ can make feature extraction yield twice the result with half the effort. The detailed information is shown in Table 1.

Table 1. Center loss.

Algorithm: Center loss
Input: Training data $\{xi\}$.Initialized parameters θc in convolution layers. Parameters W and $\{cj\|j=1,2...,n\}$ in loss layers, respectively.
Hyperparameter λ, α and learning rate μt. The number of iteration $t \leftarrow 0$.
Output: The parameters θc.
1: while not converge do
2: $t \leftarrow t+1$
3: Compute the joint loss by $L^t = L_S^t + L_S^T$
4: Compute the backpropagation error $\frac{\partial L^t}{\partial x_i^t}$ for each i by $\frac{\partial L^t}{\partial x_i^t} = \frac{\partial L_s^t}{\partial x_i^t} + \lambda \cdot \frac{\partial L_C^t}{\partial x_i^t}$
5: Update the parameters W by $W^{t+1} = W^t - \mu^t \cdot \frac{\partial L^t}{\partial W^t} = W^t - \mu^t \cdot \frac{\partial L_S^t}{\partial W^t}$
6: Update the parameters cj for each j by $c_j^{t+1} = c_j^t - \alpha \cdot \Delta c_j^t$
7: Update the parameters θc by $\theta_C^{t+1} = \theta_C^t - \mu^t \sum_i^m \frac{\partial L^t}{\partial x_i^t} \cdot \frac{\partial x_i^t}{\partial \theta_C^t}$
8: end while

4 Experiment

Experiments can be divided into two parts: (1) the performances of optimized PointNet and joint supervision of softmax loss function and center loss, and (2) the sensitivity of parameter λ.

4.1 The Performances of Improved Model

The neural network learns global feature by max pooling that can be used to classify the objects. ModelNet40 shape classification dataset is used to evaluate the performances of our optimized model, which has 40 man-made object categories such as airplanes, tables, pianos. There are 12,311 models in the dataset represented by a triangular mesh, which can be split into 9843 for training and 2468 for testing. the raw input point cloud is normalized in the data preprocessing. All layers include 1×1 convolution, RELU and batch normalization. Adam optimizer with initial learning rate 0.001, momentum 0.9 and batch size 32 are used in this paper. Dropout is the most commonly used regularization in the convolution neural network to reduce the complexity of the network, which is used on the last fully connected layer. Dropout with keep radio 0.7 is set in the experiment. It takes 6–7 h to train with TensorFlow and a GTX 1080 GPU. In Table 2, the performances of optimized neural network are compared with previous work. Especially, we use center loss based on PointNet rather than optimized neural network. The performance of our optimized model is shown better than other methods.

Table 2. The results of methods.

	Input	Views	Accuracy avg. class	Accuracy overall
SPH	Mesh	–	68.2%	–
3DShapeNets	Volume	1	77.3%	84.7%
Voxnet	Volume	12	83.0%	85.9%
Subvolume	Volume	20	86.0%	89.2%
LFD	Image	10	75.5%	–
PointNet baseline	Point cloud	–	72.6%	77.4%
PointNet	Point cloud	1	86.2%	89.2%
Ours (increasing layers)	Point cloud	1	85.7%	**89.3%**
Ours (center loss)	Point cloud	1	**86.67%**	**89.95%**

4.2 The Sensitivity of Parameter λ

If the softmax loss function or center loss is used individually, the extracted features would contain large intra-class differences or inter-class distances. The parameters of λ and α are restricted in [0, 1]. A scalar λ is used for balancing the weights for the two loss functions. A proper λ can help to extract more discriminative features, which is crucial for object classification. In the experiment, we fix α to 0.3 and vary λ from 0.0001 to 0.0007. We evaluate the average accuracy and accuracy overall of using center loss on the test set, which are up to 86.67% (accuracy average class) and 89.95% (accuracy overall) respectively. When $\lambda = 0.0005$, the verification accuracies of these parameter variations on the ModelNet40 shape dataset are shown in Fig. 2.

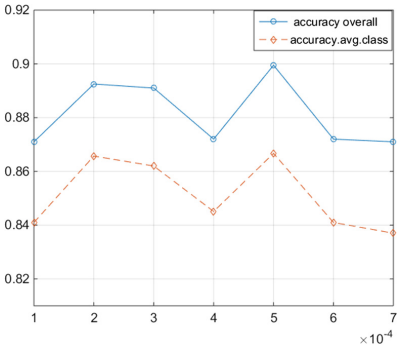

Fig. 2. The performances are influenced by different λ.

5 Concluding Remarks

In this paper, we have proposed two approaches to improve the accuracy of PointNet based 3D classification model. The first method increases the number of hidden layers to extract more abstract features. The second approach is to combine softmax loss function with center loss to obtain the discriminative features. Both of the two methods obtained better performance compared to the original PointNet. The overall accuracy is up to 89.35% and 89.95%, respectively. There are still some problems to be studied further, such as network optimization with lightweight network or deep volume layer network (ResNet, VGG). And a wide range of λ, α may enhance the discrimination of target.

References

1. Yanaka, K., Yamanouchi, T.: 3D image display courses for information media students. IEEE Comput. Graph. Appl. **36**(2), 68–73 (2016)
2. Grubisic, I., Gjenero, L., Lipic, T., Sovic, I., Skala, T.: Active 3D scanning based 3D thermography system and medical applications. In: Proceedings of the 34th International Convention MIPRO, pp. 269–273 (2011)
3. Ravi, D., et al.: Deep learning for health informatics. IEEE J. Biomed. Heal. Inform. **21**(1), 4–21 (2016)
4. Huang, H., Yang, J., Song, Y., Huang, H., Gui, G.: Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system. IEEE Trans. Veh. Technol. **67**(9), 8549–8560 (2018)
5. Zhang, L., Jia, J., Gui, G., Hao, X., Gao, W., Wang, M.: Deep learning based improved classification system for designing tomato harvesting robot. IEEE Access **6**, 67940–67950 (2018)
6. Li, W., Liu, H., Wang, Y., Li, Z., Jia, Y., Gui, G.: Deep learning-based classification methods for remote sensing images in urban built-up areas. IEEE Access **7**, 36274–36284 (2019)

7. Garcia-Garcia, A., Gomez-Donoso, F., Garcia-Rodriguez, J., Orts-Escolano, S., Cazorla, M., Azorin-Lopez, J.: PointNet: a 3D convolutional neural network for real-time object class recognition. In: Proceedings of the International Joint Conference on Neural Networks, vol. 2016, pp. 1578–1584, October 2016
8. Wang, X., Liu, M.: Multi-view deep metric learning for volumetric image recognition. In: Proceedings - IEEE International Conference on Multimedia and Expo, Mvdml, 2018, vol. 2018, pp. 1–6, July 2018
9. Gao, Y., Radha, H.: Multi-view image coding using 3-D voxel models. In: Proceedings - International Conference on Image Processing, ICIP, vol. 2, pp. 257–260 (2005)
10. Ge, L., Cai, Y., Weng, J., Yuan, J.: Hand PointNet: 3D hand pose estimation using point sets. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 8417–8426 (2018)
11. Li, Z.: A discriminative learning convolutional neural network for facial expression recognition. In: 2017 3rd IEEE International Conference on Computer and Communications, pp. 1641–1646 (2017)
12. Ge, L., Liang, H., Yuan, J., Thalmann, D.: Robust 3D hand pose estimation in single depth images: from single-view cnn to multi-view CNNs. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016, pp. 3593–3601 (2016)
13. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: deep learning on point sets for 3D classification and segmentation. In: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, vol. 2017, pp. 77–85, January 2017
14. Chen, Y., Chen, Y., Wang, X., Tang, X.: Deep learning face representation by joint identification-verification. In: International Conference on Neural Information Processing Systems, pp. 1988–1996 (2014)
15. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 07–12 June, pp. 815–823 (2015)