Evaluation of Convolution Operation Based on the Interpretation of Deep Learning on 3-D Point Cloud

Bufan Zhao, Xianghong Hua, *Member, IEEE*, Kegen Yu, *Senior Member, IEEE*, Wuyong Tao, Xiaoxing He, Shaoquan Feng, and Pengju Tian

***Abstract*—The interpretation of deep learning network is an important part in understanding the convolutional neural networks (CNNs). As an exploratory research, this article explored the interpretation method in 3-D point cloud deep learning networks, for the purpose of evaluating the performance of convolution functions in 3-D point cloud CNNs. Specifically, a 3-D point cloud classification network with two branches is used as the interpretation network in two aspects; 1) information entropy is introduced to diagnose the internal representation in the middle layer of CNN; and 2) the external consistency of convolution function is measured by per-point classification accuracy with class activation mapping technique. Four typical convolution functions are tested by the interpretation network on ModelNet40 dataset and the experimental results demonstrate that the proposed evaluation method is reliable. Feature transformation ability and feature recognition ability of convolution functions are extracted by visualization evaluation and proposed measurable metrics evaluation.**

***Index Terms*—3-D point cloud, convolution function evaluation, deep learning interpretation, external consistency, internal consistency.**

I. Introduction

THE use of deep learning for 3-D point cloud is an important topic in 3-D object detection and recognition, PointNet [1] opens up an end-to-end deep learning method which enables high-accuracy 3-D object recognition, and derives a lot of effective works in point cloud deep learning methods [2]–[5], but less work is done to explore why it works and what is learned in the network. It is an important work to open up the "Black box" in the learning process of network [6] to understand what is learned hidden inside feature map of conv-layer and which filter can guide the discernment decision of network output. At first, in order to understand the artificial neural network (ANN), the interpretation was built with fuzzy rules [7], as more attention has been paid in this region. eXplainable Artificial Intelligence (XAI) has become an extremely important job to help human understand "why," "what," "what for," and "how" neural network works and search for a direct understanding of the mechanism in the "Black box" [8]. XAI contains all techniques in AI system such as machine learning (ML), neural network (NN), deep learning (DL), etc., Explainability and interpretability of network are the cores in an XAI research, and these two concepts can be often understood as one meaning, that is the mapping of abstract concepts to areas that human can understand [9].

Recently, CNNs contribute the state-of-art models to all the areas in fundamental computer vision, from image classification [10] and object detection [11] to instance segmentation [12]. In 2-D CNNs interpretation, existing works mainly perform in two different ways: 1) first direction is mapping the output into the input image space to visualize the decision or discriminative parts which contribute most to output score [13]–[15]; and 2) the second direction focuses on diagnosing CNN internal representations to obtain insight understanding of features encoded inside CNN [16]–[18].The presentation of the interpretative methods include visualization and quantification, showing a comprehensive evaluation in 2-D CNNs works. But in 3-D point cloud CNNs, there are few works in 3-D CNNs interpretation. Since the technology of 3-D convolutional network is an extended work based on 2-D convolutional technology, and the coordinate attribute of point cloud not only denotes geometric feature, but also is the location of feature in the space of input point cloud, it is easy to establish mapping relationship between feature map and coordinate space. Therefore, this article attempts to seek for extending the techniques in 2-D CNNs interpretation to evaluate the performance of 3-D point cloud CNNs, to disentangle the hidden information contained in the conv-layers.

Different 3-D convolution, operations are proposed to solve the problem of 3-D point cloud convolution [2], [19], [20]. 3-D convolution function is designed to explore more efficient way for edge feature aggregation [21], but the effectiveness of

Manuscript received May 13, 2020; revised July 11, 2020; accepted August 17, 2020. Date of publication August 31, 2020; date of current version September 16, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 41674005 and Grant 41871373, in part by the Foundation of Key Laboratory for Digital Land and Resources of Jiangxi Province, East China University of Technology (DLLJ202015), and in part by the Natural Science Foundation of Jiangxi Province (20202BAB214029, 20202BABL214055, and 20202BABL213033). *(Corresponding author: Xianghong Hua.)*

Bufan Zhao is with the School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China, and also with the Key Laboratory for Digital Land and Resources of Jiangxi Province, East China University of Technology, Nanchang 330013, China (e-mail: bufan_zhao@whu.edu.cn).

Xianghong Hua, Wuyong Tao, Shaoquan Feng, and Pengju Tian are with the School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China (e-mail: xhhua@sgg.whu.edu.cn; wuyong_tao@whu.edu.cn; 2017202140047@whu.edu.cn; 1429857175@qq.com).

Kegen Yu is with the School of Environmental Science and Spatial Informatics, China University of Mining and Technology, Xuzhou 221116, China (e-mail: kegen.yu@ieee.org).

Xiaoxing He is with the School of Civil Engineering and Architecture, East China Jiaotong University, Nan Chang 330013, China (e-mail: hexiaoxing@whu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2020.3020321

a network depends on architecture of the framework and the design of the convolution function; it is hard to judge whether the network architecture is more efficient or the convolution function contributes more. Therefore, this article puts different convolution functions into the same network framework to evaluate the performance of convolution functions with different kernels. The framework is called interpretation network, which is based on an effective 3-D point cloud deep learning network in the existing research, it ensures that the interpretation network can perform in-depth interpretation of convolution functions with less accuracy degradation. On the other hand, in terms of evaluation metric, most of existing metrics mainly depend on the final network output, which is effective but monotonous. In order to evaluate the 3-D convolution function more comprehensively, both internal and external consistencies are considered. This article provides suitable criteria for the quantitative evaluation of convolution function in terms of internal consistency and external consistency.

To the best of our knowledge, interpretation has been widely studied for 2-D image CNNs, but little work has been done for the interpretation in 3-D CNNs, especially in the evaluation of 3-D convolution functions, a good choice of convolution functions gives the network a powerful drive to aggregate the local feature of object, generating more reliable information to lead the final classification. Focusing on the interpretation of convolution functions in 3-D CNNs, the contributions of this article are summarized as follows:

1) This article proposes two 3-D CNN evaluation criteria called internal consistency and external consistency, respectively. Internal consistency evaluation is proposed for evaluating the feature map of multiple filters, which is generated by a 3-D convolution operation. The method seeks maximum information entropy as a metric for diagnosing the representation of intermediate layer in a pretrained network, the internal network is a tool to assess the feature transformation ability of convolution functions. External consistency evaluation is proposed for evaluating the accuracy degree of network output. 3-D class activation map (CAM) is used to build up the mapping relationship between per-point and the output category. Its purpose is to assess the feature recognition ability of convolution functions in 3-D CNNs.
2) A 3-D CNN interpretation network is proposed based on the existing 3-D point cloud deep learning network (Pointnet & DGCNN). A branch architecture is adopted in the interpretation network to excavate internal and external consistency of convolution operation. Typical 3-D convolution functions are evaluated by the interpretation network, and multiple metrics are used to give them a comprehensive analysis.

The remainder of the article is organized as follows. Section II gives a brief review on the related work. Section III describes details of the proposed evaluation approach and an interpretation network. Section IV shows the experimental results and provides some analyses on the results and Section V concludes the article.

II. Related Work

A. Interpretation for 2-D CNNs

Our work builds on extending techniques of visualization and quantification of 2-D CNNs interpretation works, this article will first give a brief introduction of the existing techniques in 2-D CNNs interpretation. There are a number of recent works in the domain of interpretation and understanding 2-D CNNs; the key point is to understand which feature is the decision-making process and how much useful information is contained in internal conv-layers. Readers can refer to the review on the methods of 2-D CNNs interpretation in [22] for a comprehensive knowledge.

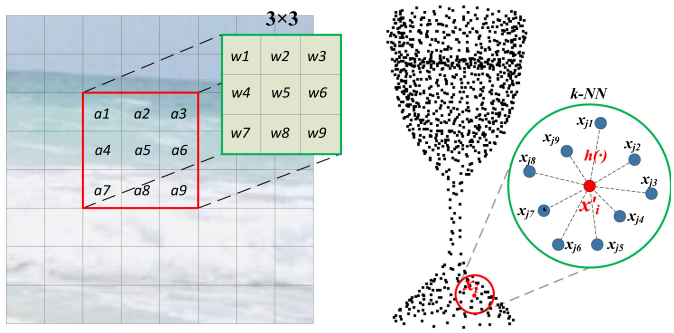
Visualization evaluation: The visualization combined feature association methods are the most commonly used in CNNs interpretation. There are two divided categories in visual interpretation, one using image synthesizer that highlights the feature with highest score [23] and the other using projection of feature response in a conv-layer back to the input space [24]. The first category method is based on the Deconvnet technique. Zeiler *et al.* performed the deconvolution for feature map of middle-layer in the network [25], [26]. The units of feature map were projected to the pixel space and observed the variety of feature map in original image. The other method is based on the CAM technique which was proposed by Zhou *et al.* [13], the original CAM method used the global average pooling (GAP) to replace the maximum pooling layer, exploiting its location ability and creating a CAM to locate the discriminative area in a 2-D object image [27]. On this basis, Selvaraju *et al.* proposed a gradient-weighted class activation mapping (Grad-CAM) to generalize the original CAM by propagating gradients of feature maps, which introduce a new way of combining feature maps using the gradient signal [28]. Kumar *et al.* proposed a class-enhanced attentive response map, which combines the dominant attentive response map and dominant class attentive map, attributing each pixel a response to the input space [29]. Both types of methods interpret 2-D CNN in different perspectives and reach a good performance, but both of them have their drawbacks. CAM methods require a modification in the network architecture, which would affect the accuracy of network; and the Deconvnet method is computationally more expensive than CAM-based methods, and does not build up the mapping relation between features and output category, so lacking of intuitional visualization to human [28].

Quantification evaluation: For quantifying the interpretability of 2-D CNNs, the diagnosis work of CNN focuses on exploring representations of the black-box in the conv-layers and the most direct method is to analyze CNN features from feature map in the intermediate layers [22]. Zhang *et al.* made use of the analysis of network feature space to refine network representation, and computes the biased representations to estimate different attributes or categories [18]. Lakkaraju proposed a model-agnostic methodology, which uses feedback from an oracle to identify unknowns and intelligently guide the discovery [30]. The purpose of CNN representation diagnosis is to discover potential flaws in conv-layer, providing a guide to improve the CNNs

TABLE I
TECHNOLOGIES OF CNNS INTERPRETATION METHODS

Methods	Visualization	Quantification	Feature Mapping	Representation Diagnosis	application
CAM-based [13]	√		√		2D
Deconvnet-based [25]	√			√	2D
Network Dissection [31]	√	√	√		2D
Lakkaraju' work [30]		√		√	2D
Zhang'work [18]		√		√	2D
Internal consistency		√		√	3D
External consistency	√	√	√		3D

[22]. Bau *et al.* [31] and Zhou *et al.* [32] used a direct feature mapping approach called network dissection; units are given human interpretable labels by semantic alignment of units and input image in a given CNN, intersection over union score (IoU) is used as a metric to measure the unit interpretability score. In the same way, Net2Vec quantified the Filter-Concept overlap by IoU for each filter in a convolutional layer, in order to quantitatively demonstrate how concepts are encoded across multiple filters [33]. In essence, feature mapping approaches translate visualizations of representation into quantitative interpretations of interpretability, it is not independent but relies on external information like input image space to disentangle the internal representation. Nevertheless, it still needs to explore a suitable metric to allow for a meaningful comparison of how well a model fits the parameters from the training network [8].



(a) (b)

Fig. 1. Convolution operation of 2-D image and 3-D point cloud. (a) 2-D operation kernel. (b) 3-D operation kernel in geometric distance space.

B. Interpretation for 3-D CNNs

Although there are few systematic interpretation works for 3-D CNNs evaluation, still it can be found in some achievements contained in previous studies. An interpretation work has been done in PointNet, t-SNE was used to embed point cloud global signature into 2-D space to cluster similar shapes, and point function is visualized to gain more insights on what the learnt per-point functions detect [1]. Based on this, a C-PointNet was proposed to explain what has been learned inside the PointNet [34], it uses a Class-attentive response map to visual the activation parts in a 3-D object. The existing 3-D interpretation works mainly rely on the visualization ways and lack of measurable metric.

To this purpose, this article focuses on exploring the 3-D CNNs interpretation work based on the statistics theory, and providing an evaluation methodology for the 3-D convolution function selection. The most existed network is interpreted by visualization in the work mentioned above, but it is hard to evaluate all objects by visualization because there are multiple filters in conv-layers. Measurable metrics are needed for a comprehensive evaluation to the network. For this reason, a synthesis method is proposed for 3-D CNNs interpretation, which is constituted by internal consistency evaluation and external consistency evaluation. The internal consistency evaluation is a diagnostic internal representation work with independent metric without considering external information, while the external consistency evaluation is an extended application of CAM-based technique to 3-D CNNs interpretation, Table I lists the key technologies of reference 2-D CNNs interpretation methods and our 3-D CNNs interpretation methods.

III. 3-D CONVOLUTIONAL FUNCTION EVALUATION

Before describing the details of the interpretation network, concepts of the internal consistency and external consistency evaluation are explained as follows:

Internal consistency evaluation: Extracting the convolutional information from each filter, evaluation for the internal representation which is learned in the network, convolution functions are diagnosed by the maximum information entropy extracted from the feature maps in the conv-layers. It is a relative evaluation without contacting the outside world, only depending on the distribution of feature within and between the filters.

External consistency evaluation: Establishing a mapping relationship between each point and category, and evaluating consistency accuracy of the interpretation network output; convolution functions are evaluated by analyzing the degree of difference between ground-truth and final output in point-level. It is an absolute evaluation, which involves raw input information and the ground-truth.

A. 3-D Convolution Functions

The largest difference between 2-D convolution and 3-D convolution is that the 2-D convolution operation has a fixed pattern, as shown in Fig. 1(a), where the units are represented in regular domains and the convolution function is also regular grids such as 3×3 or 5×5. However, it cannot be analogized

to 3-D convolution directly because the locations of neighbors of central point are transformable in point cloud [35], and the location property between neighbor points and central points are treated as an important feature, which is called edge feature [21]. Many research works put forward the convolution function to aggregate edge features of point cloud, the operation of convolution function with geometric space is shown in Fig. 1(b) and can be concluded as

$$x_i^{'} = g\left(h\left(x_i, x_j\right)\right) \tag{1}$$

where the point cloud samples are denoted as $X = \{x_1, x_2, \ldots, x_n\} \in R^N$. x_i is the *i*th point in sample, x_j are the neighbors of x_i, commonly it takes *k*-nearest neighbor graph of x_i as the scope of convolution, where $j \in (1, 2, \ldots, k)$. Function $h(\cdot) : R^N \to R^K$ is a feature transformation operation and $g(\cdot)$ is a symmetric function, which aggregates all local features to $x_i^{'}$ for next conv-layer. Here, the common process activation function and bias are reduced to make the formula express more clearly.

More studies have focused on the choice of feature transformation operation $h(\cdot)$. This article enumerates four convolution functions in the existing works [1], [2], [19], [20]

$$h\left(\cdot\right) = h\left(x_i\right) \tag{2}$$

$$h\left(\cdot\right) = h\left(x_j\right) \tag{3}$$

$$h\left(\cdot\right) = h\left(x_i, x_i - x_j\right) \tag{4}$$

$$h\left(\cdot\right) = h\left(x_i, x_j, x_i - x_j\right). \tag{5}$$

Although the work in [19] has compared different functions using classification accuracy, it did not give the details about why it happened, this article will do deep study in the evaluation of convolution functions.

B. Evaluation of Internal Consistency

The interpretation work in CNNs focuses on explaining which features are activated and what degree of activation is achieved in the internal conv-layers. Feature map is defined as a list of features, which are generated by a filter in the conv-layer. Most previous interpretation studies make use of feature visualization techniques, but it is impractical for a comprehensive evaluation since so many filters need to be visualized. Hence, this article plans to measure the degree of activation of all feature maps by using a quantitative metric, and maximum information entropy is adopted to describe complexity of multiple filters in the conv-layers.

In information theory, entropy is a measure of uncertainty, the greater the uncertainty is, the larger entropy is and the greater capacity of the system to carry information. In order to accurately estimate the state of random variable, it generally adopts to maximize entropy [36]. This is because the model with largest entropy is the best model among all possible probability models. The value of feature in conv-layer feature map is more like random variable, so that the concept of entropy can be applied. According to the property of convolution, the deeper convolutional layer is, the more features are extracted and more feature reconstructions are involved [37]. There is more information inside the feature map so that the diversity of features is increased. If features in a filter are greatly activated, the value of feature in this filter will increase partially or fully. This transformation reflected in value is a numerical disorder, the entropy is larger with the variable change.

Assume a point cloud of object *l* contains *n* points and *m* dimensions in the feature map of conv-layer, and let α_{ij} denote the activated value of the *i*th position of points in the *j*-th filter in the feature map, where $i = 1, \ldots, n, j = 1, \ldots, m$. To obtain the information entropy of each filter, first α_{ij} is normalized

$$\bar{\alpha}_{ij} = \frac{\alpha_{ij} - \min\left(\alpha_{1j}, \alpha_{2j}, \ldots, \alpha_{nj}\right)}{\max\left(\alpha_{1j}, \alpha_{2j}, \ldots, \alpha_{nj}\right) - \min\left(\alpha_{1j}, \alpha_{2j}, \ldots, \alpha_{nj}\right)}. \tag{6}$$

Second, in the *j*th filter, taking the percentage of activation value between the *i*th position $\bar{\alpha}_{ij}$ and the sum of all points $\sum_{i=1}^{n} \bar{\alpha}_{ij}$ as variable probability

$$P_{ij} = \frac{\bar{\alpha}_{ij}}{\sum_{i=1}^{n} \bar{\alpha}_{ij}}, \; i = 1, 2, \ldots, n, \; j = 1, 2, \ldots, m \tag{7}$$

where P_{ij} denotes the probability value of normalized activation value $\bar{\alpha}_{ij}$. Then, the entropy of the *j*th filter is obtained as

$$e_j = -\gamma \sum_{i=1}^{n} P_{ij} \ln\left(P_{ij}\right), \; j = 1, 2, \ldots, m \tag{8}$$

where $\gamma = 1/\ln(n)$ to keep $e_j > 0$. As a consequence, $E^t(l) = \{e_j | j = 1 \ldots m\}$ denote the entropy of *m* filters in the *t*th conv-layer for object *l*, which is used as a measurable metric to analyze the feature transformation ability of convolution function in the internal network, more detail will be discussed in Section IV-B.

C. Evaluation of External Consistency

In the common classification network, the fully connected layers are used to recombine features which are extracted by convolution layers, it can establish the linear relationship between the features and categories, but the location of feature would be disordered by the operation of fully connected layers [38] and it hard to map the location of feature into input space. In order to observe which part of points dominate the classification output score most, class activation technique was developed to map 2-D image into 3-D points in this article; a 3-D CAM is implemented in the classification network, the fully connected operation is replaced with the shared multilayer perceptron (shared MLP). Also, global max pooling layer is replaced with GAP layer, the use of average pooling layer focuses on the complete extent of the feature maps, and better fits to identify the discriminative points of object [13]. In the external network, let α_k^n denote the active value of feature in the *k*th filter of the *i*th position of points in last conv-layer, where $k = 1, \ldots, m$ and $i = 1, \ldots, n$, $A_k(l) = \frac{1}{n} \sum_i \alpha_k^i(l)$ denotes the value of global average feature in the *k*th filter of object *l*, the class score S_c can be obtained as [13]

$$S_c\left(l\right) = \sum_k w_k^c A_k\left(l\right) = \frac{1}{n} \sum_k w_k^c \sum_i \alpha_k^i\left(l\right)$$

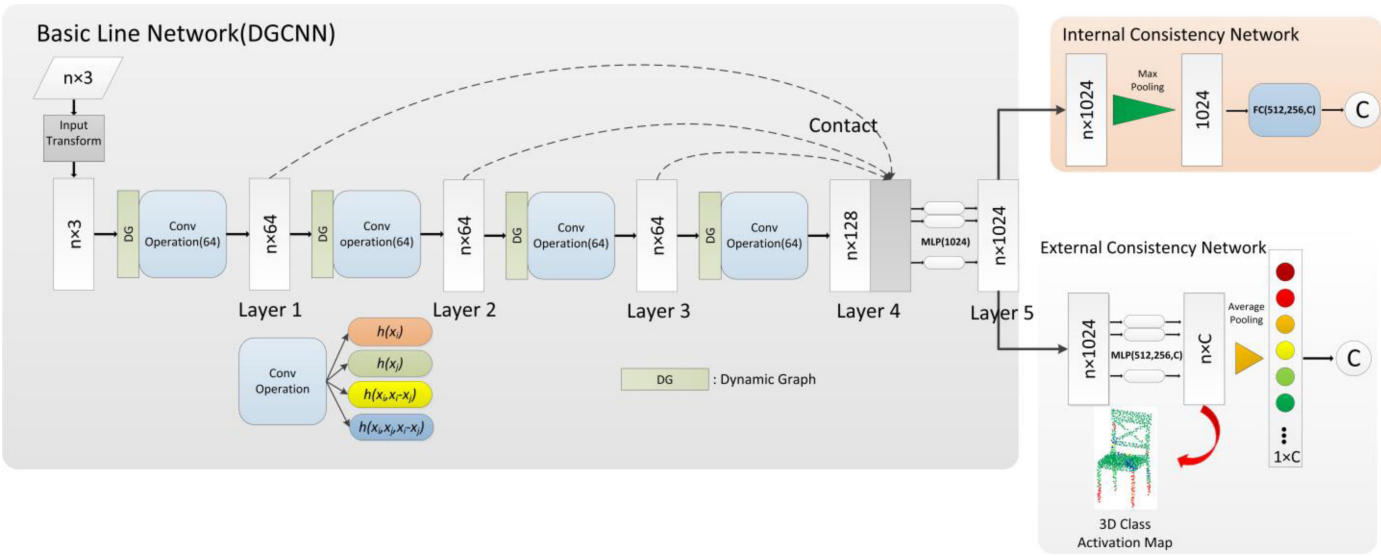


Fig. 2 Framework of the interpretation network.

$$= \frac{1}{n} \sum_{i} \sum_{k} w_k^c \alpha_k^i (l) \tag{9}$$

where w_k^c denotes the weight of feature in the *k*th filter for the class c, in essence, w_k^c emphasizes the influence of $A_k(l)$ for class c. Then, the 3-D class active map can be defined as

$$Cls_c^i (l) = \sum_{k} w_k^c \alpha_k^i (l) \tag{10}$$

which denotes the contribution of the *n*th points to the category c; that is, $Cls_c^i(l)$ can be used to get the mapping relationship between each point and a category c. Taking the maximum value of $Cls_c^i(l)$ in tunnel c as the classification result of each point, the class mapping score for points to each class is given by

$$\mathrm{Map}^i (l) = \max_c Cls_c^i (l) \tag{11}$$

where $\mathrm{Map}^i(l)$ donates the classification result of the *i*th points in object l. To evaluate the classification results, counting the number of correctly classified points in $\mathrm{Map}^i(l)$, and the external consistency accuracy (ECA) is defined as

$$\mathrm{ECA} = \frac{N}{n} \tag{12}$$

where N is the number of correctly classified points in $\mathrm{Map}^i(l)$ and n is the number of points in $\mathrm{Map}^i(l)$. The ECA can be used as an evaluation metric to describe the degree of consistency between the category c and input points.

D. 3-D Interpretation Network

As mentioned above, when a 3-D CNNs are working, it is hard to judge whether the network framework or the convolution function works better. To evaluate the 3-D convolution function, this article uses a common framework as the interpretation network, as shown in Fig. 2. It consists of three parts: the basic line network is based on DGCNN [19]; a branch divides the backbone network into two classification networks, named as internal consistency network and external consistency network; and the inputs of conv-operation are the different convolution functions to be tested. More details are explained as follows.

Basic line network: The basic line network is designed for feature transformation with 3D convolution function. There are three reasons to adopt the DGCNN framework as the basic line network:

1) DGCNN considers graph-structure in feature space distance instead of geometric space distance when the local graph of central point is created, DGCNN proposed a dynamic graph which rebuilds the local graphs using the distance of neighbor units in the feature space in each conv-layers; the points with similar features are aggregated layer by layer, which is beneficial to extract the outstanding feature in conv-layers. Then if a convolution function has a stronger feature transformation ability, the similar outstanding features can be clustered and show an obvious homogeneity [19]; and the feature cluster can be visualized in input space and make it more intuitive for human to understand the extracted features. In other words, feature graph is more convenient to disassemble "the black box," which can contribute our interpretation work to diagnosing feature transformation of convolution function.
2) Based on the feature graph in former conv-layers, the next convolutional layer continues to extract the feature by rebuilding graph. Similar features are further explored [19], the deeper information of feature can be found and the feature transformation ability of convolution function is accumulated as convolutional layers become deeper. The distinction of feature transformation ability between different convolution functions is amplified as convolution layer goes deeper, which provides the basic in convolution function evaluation.
3) DGCNN is a highly accurate 3-D point cloud deep learning method and it shows the feature graph can be used in

TABLE II
DETAILS OF THE INTERPRETATION NETWORK

Conv-function	Input Data	Classification Accuracy		Training Time	
		Internal consistency network	External consistency network	Internal consistency network	External consistency network
$h(x_i)$	1024 xyz	89.4%	86.5%	3h18m17s	5h8m47s
$h(x_j)$	1024 xyz	90.4%	87.7%	9h14m35s	11h54m39s
$h(x_i, x_i - x_j)$	1024 xyz	**91.6%**	**88.8%**	6h41m37s	10h10m3s
$h(x_i, x_j, x_i - x_j)$	1024 xyz	91.4%	88.5%	5h26m48s	9h53m27s

3-D CNN scenarios, although its accuracy may not be the highest. Interpretation work results in a decrease in accuracy [22], but the use of feature graph enables a higher ceiling to mitigate the loss of accuracy, to ensure the performance of the interpretation network.

Internal consistency network: This branch is similar to the classification network of PointNet&DGCNN, via max pooling and fully connection layer to output the classified score. The purpose of this operation is to obtain the feature map of each filter in conv-layer, and evaluate the performance of different convolution functions by exploring the internal representation contained in the feature map of the network.

External consistency network: This branch is inspired by a 2-D image class activation mapping technique. 3-D class attention map is built by the modification of original classification network. The dimension of the global feature is reduced to the number of the category through replacing the MLP layer with the fully connection layer; each dimension is associated with a particular class, and average pooling is performed to obtain the class score to replace the maximum pool; the response between the per-point and the class is then established. The purpose of this operation is to explore the contribution of each point to the classification output.

IV. EXPERIMENT AND ANALYZE

In this section, the performances of different convolution functions are evaluated with the methods described above. First, the implementation and training accuracy of the interpretation network are illustrated to study the feasibility of the network. Second, different convolution functions are trained and tested through the interpretation network, to evaluate their classification ability by visualizing and quantifying analysis from internal consistency and external consistency, respectively.

A. Implement Details

The data adopted for classification and evaluation come from ModelNet40 [39], and the network to be implemented is the same as DGCNN. Stochastic gradient descent (SGD) is used with learning rate 0.1, and the rate is reduced until 0.001 using cosine annealing. The momentum for batch normalization is 0.9, the batch size is 32, the delay rat is 0.7, and the max epochs is set as 250. All models in this article are trained on a single NVIDIA GPU with 8 GB GTX 1080Ti, 8 GB RAM computer with platform TensorFlow, and language python 3.6.

The internal consistency network and external consistency network in the interpretation network are conducted, respectively, and convolution functions used to evaluate are $h(x_i)$, $h(x_j)$, $h(x_i, x_i - x_j)$, $h(x_i, x_j, x_i - x_j)$, which are described in Section III-A. The classification accuracy of these convolution functions in two branch networks are shown in Table II.

As shown in Table II, it can be found that four convolution functions have high classification accuracy in internal consistency network. Comparatively, the accuracy of the external consistency network degrades slightly due to the absence of fully connection layer. As an exploratory study, the design of the network for interpretation may influence the accuracy of classification of original network a bit [40], but it still retains an acceptable accuracy for evaluation in our research.

B. Internal Consistency Evaluation

Section III has introduced the max information entropy theory derived in convolutional feature map. This part will first demonstrate the role of entropy e_j in representation of features, then e_j is taken as a comparable metric to evaluate the propagation of feature information in the interpretation process.

1) Demonstration of Entropy: In the process of CNNs, conv-layer decides which features are activated and propagated to next conv-layer, the key of convolution function evaluation is to figure out which and how many features are activated. However, it is impracticable to observe the contribution of each filter by visualizing all features maps. So primary concern in this article is whether a filter is fully activated, or the filter is partially activated. The examples of the two situations in terms of fully activated filter and partially activated filter are shown in Fig. 3. It can be observed that fully activated filter activates most points in object, and the partially activated filter activates a specific part of points in object, these useful filters in conv-layer generate more meaningful feature maps.

As the valuable information in conv-layer, these two kinds of feature maps are needed in propagation and transformation of the network. Full activation denotes that all points are activated in this filter, illustrating that the tested point clouds response to all the parameters of the filter. On the other hand, partial activation denotes that some particular features of object are activated, which gives more varieties for feature transformation and recombination to next layer. In a feature map, the activated representations can be summarized by adding up all values of the feature in each filter [41], so our work explored the mean and standard deviation (std) of activated values α in each filter to denote the full activation and partial activation.

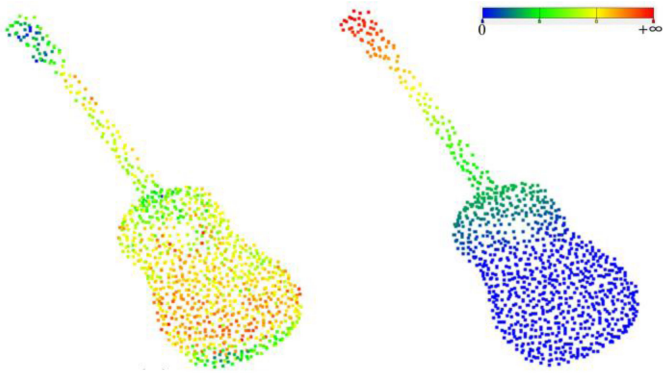


Fig. 3. Points in input data response to features from the filters, which are fully activated and partially activated, respectively. (a) Visualization of fully activated feature map in guitar sample. (b) Visualization of partially activated feature map in guitar sample. As activated values go from small to large, colors of points in figure go from blue to red. The same expression applies to illustrate the point cloud feature map in the following figures.

Fig. 4 shows visualization of the responding points for the feature map, which has different mean and standard deviation of activated value. From second column in Fig. 4, it can be seen that the filter with larger mean(α) is more likely to be an fully activated feature map, most of points in the object are activated and colors trend to red, illustrating that the feature map is responsive to the points and more features for identifying the classification of sample can be extracted in this filter. The third column in Fig. 4 shows the filters with large std(α); extracted features response part of points in sample and some special features of sample are activated, feature maps are more likely to be generated by partially activated filters. In the fourth column of Fig. 4, it can be seen that the filters with large mean(α) and large std(α) also own useful information, the main skeleton of sample is responded, so these filters are also meaningful for feature transformation. The last column of Fig. 4 shows the filters with small value of mean(α) and std(α), it shows that fewer and irregular points are activated, illustrating that the parameters of the filters are not suitable for this category of sample, so these filters should be avoided in the conv-layer because less useful information can transmit to next layer.

The first three kinds of filters above are all beneficial for feature propagating in the conv-layer, but as the measurements, there is an intersection between mean value and standard deviation, both indexes have limits. Observing the value of entropy e in each sample in Fig. 4, it can be found that the entropies e of the meaningful filters are larger, illustrating that the information contained in feature map can connect to the entropy e of features. Therefore, this article attempts to represent the mean value and standard deviation value by the entropy e. Finding the relationship between e and mean(α) and std(α), the correlation coefficients between entropy value and mean\std value of the corresponding activation value are shown in Table III. Table III picks the correlation coefficients of last convolution conv-layer (layer 4 in Fig. 2), the reason for the choice of the last layer is that it contains higher dimensions which own more diverse features. From Table III, it can be concluded that the entropy of the activation value is linearly positively correlated with the corresponding mean and standard deviation, especially for mean value, it indicates that entropy can reflect the activation for feature map. So, this article adopts the maximum entropy to measure the degree of feature activation instead of using the mean value and standard deviation. Fig. 5 shows the response points in the filters with large entropy for different samples; it can be found that most points in major structure are activated.

TABLE III
CORRELATION COEFFICIENTS BETWEEN MEAN AND STD TO THE ENTROPY IN DIFFERENT CONVOLUTION FUNCTIONS

Statistics	$h(x_i)$	$h(x_j)$	$h(x_i, x_i - x_j)$	$h(x_i, x_j, x_i - x_j)$
R(Mean, Entropy)	0.761	0.801	0.733	0.806
R(Std, Entropy)	0.379	0.451	0.399	0.315

Then it can be concluded that the entropy e relates to the information contained in the feature map of conv-layer, the larger the value is, the more useful information it contains. It can be used as a measurable metric to evaluate the quality of convolution function, and the performance of convolution functions is diagnosed by entropy in the next caption.

2) Convolution Operation Diagnosing: The ability of feature transformation is an important property for the convolution function. In this article, feature transformation ability is simply expressed as the amount of useful feature information extracted by the convolution function. It is inferred from Section III-B that the entropy e can measure the useful information contained in a feature map. For the comprehensive evaluation, taking the mean value of $E^t(l) = \{e_j | j = 1, 2, \ldots, m\}$ to represent the activation level of all filter in one conv-layer as

$$\varepsilon_l^t = \frac{1}{m} \sum_{j=1}^{m} e_j \tag{13}$$

where m is the size of filters in the tth conv-layer, ε_l^t denotes the average entropy of all feature maps in the tth conv-layer for object l, which called convolutional activation entropy in this article because it can measure the activation of feature in conv-layer.

Feature transformation ability of different convolution functions is evaluated according to the distribution of convolutional activation entropy ε. The test models in ModelNet40 are used for the evaluation in internal consistency network. Fig. 6 shows ε of all test objects in each conv-layer with convolution functions $h(x_i, x_i - x_j)$, $h(x_i, x_j, x_i - x_j)$, $h(x_j)$, and $h(x_i)$, and displays their distribution between conv-layers in boxplot. It can be seen that ε has different activation level in different conv-layers. The layer 5 has lowest value of entropy since it contacts the features, which are generated by the first four layers, and it is handled directly by MLP, not transformed by the test convolution function, so it is not exactly what our work concern about, but it is displayed in Fig. 6 for referring; the main objects of our work concern about are 1–4 conv-layers.

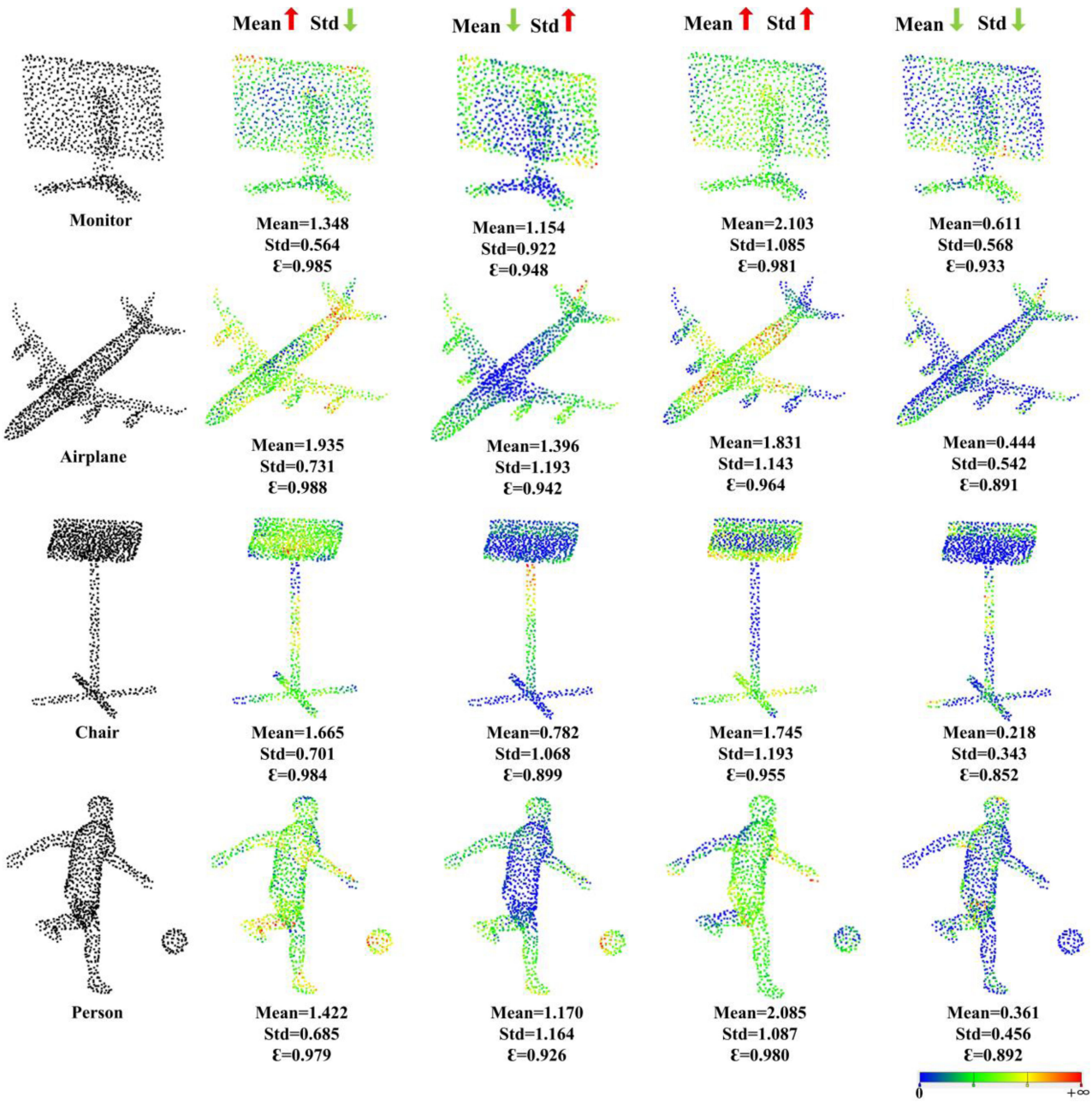


Fig. 4. Feature maps which have different values of mean and standard deviation projecting to the position of input point cloud space. The red arrow on top of the figures means the larger value and the blue one means smaller value.

Given a vertical comparison by observing the distribution of ε between 1–4 conv-layers, the convolution functions $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$ have the similar variation pattern in 1–4 conv-layers, the overall trend of ε is rising. The values of ε in the initial layer are lower, as convolutional layers deeper, the value of ε increases gradually and reaches the peak value in the fourth conv-layer, as the number of layers increases, the amount of information in the feature map increases. But in the convolution functions $h(x_i)$ and $h(x_j)$, the overall trend of ε is downward, the peak value turns up at layer 2 in $h(x_i)$ and layer 3 in $h(x_j)$, both convolution functions show irregular variation pattern of ε.

Consider analyzing the phenomenon in Fig. 6 with the convolutional property. As the conv-layer goes deeper and the number of filter increases, the extracted features become rich and the information contained increases [37]; therefore, the value of ε becomes larger with conv-layer going deeper. It can be found that the same property reflected in the distribution of ε for convolution functions $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$, which shows these convolution functions have good performance in feature transformation; on the contrary, the behaviors of $h(x_i)$ and $h(x_j)$ in the distribution of ε illustrate that the ability of feature transformation is weaker than the first two. More details, it can be founded that in $h(x_i)$, $h(x_i, x_i - x_j)$, and

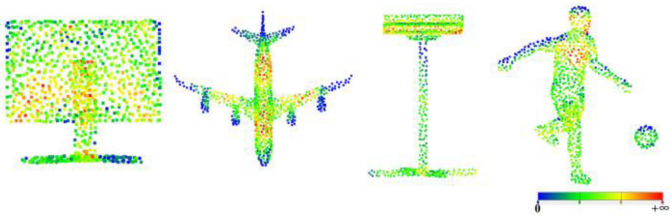


Fig. 5. Points response to the feature map which with larger entropy is denoted by e.

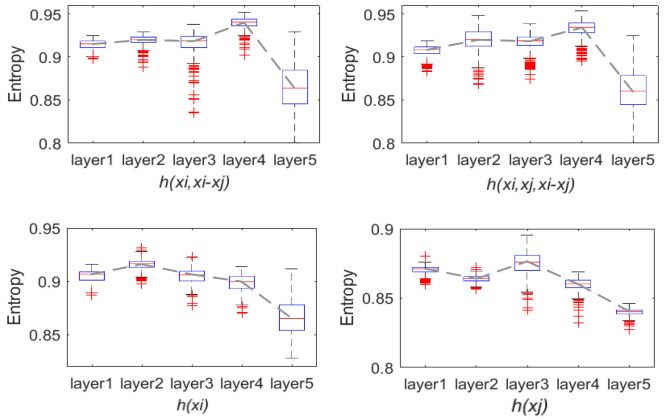


Fig. 6. Boxplot of the distribution of ε for different convolution functions.

$h(x_i, x_j, x_i - x_j)$, from layer 1 to layer 2 the trend is the ascent, while in $h(x_j)$, the entropy of layer 2 is lower than layer 1, illustrating the edge feature x_i plays a key role in the first convolution layer. In [37], the feature of original layer is mostly low-level feature, it can be inferred that edge feature x_i is useful for transforming the low-level feature, but as the layers deepen, the transformation ability of $h(x_i)$ is reduced when handling the high-level feature. $h(x_j)$ is able to extract more information from the second layer than $h(x_i)$, illustrating that the edge feature x_i and x_j work in different places, x_i helps the low-level information transform, x_j prefers to high-level information extraction. But comparing them to $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$, their feature transformation abilities are obviously inferior to the latter two. It can be said that single edge feature is weaker than the multiple edge feature in feature transformation ability.

Regarding the ability of feature transformation, $h(x_i)$ directly convolved all the points without considering the local features, so it cannot gain the graph attributes in geomatic or feature space. In the follow-up study [2], [19], it shows that the local relationship between points is an important feature; that is the reason $h(x_i)$ performs weak in feature transformation ability. $h(x_j)$ considers the neighbor points and gathers them in the central point x_i, but it lacks relative relationship between central point x_i and neighbors x_j, the level of ε in $h(x_j)$ is less than $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$. $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$ have made use of relative information between points and both achieve better results, but according the discrete points in box plots, the distribution of ε in $h(x_i, x_j, x_i - x_j)$ is more concentrated than $h(x_i, x_i - x_j)$, it means although the classification accuracy of $h(x_i, x_i - x_j)$ in Table II is a bit higher than $h(x_i, x_j, x_i - x_j)$, the stability of $h(x_i, x_j, x_i - x_j)$ is better $h(x_i, x_i - x_j)$ in feature transformation.

The optimizations of edge features have been already known by existed qualitative research works, here, this article illustrates these optimizations in the quantitative analysis by comparing the distribution of ε between edge features. According to the quantitative results in Fig. 6, it can diagnose that the first two convolution functions $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$ are "healthy" in feature transformation ability, while the latter two convolution functions $h(x_i)$ and $h(x_j)$ seem "sick" in feature transformation ability. And different edge features play different roles in convolution layer, some features may work well in low-level information extraction, some may be good for the stability.

Parameters selection: In the internal network, the only selected parameter γ is a parameter to control the size of max information entropy and keep the value greater than zero. In this article, it set as $\gamma = 1/\ln(n)$, n is the number of points in sample, γ is set as self-adapting in order to keep the value of entropy within acceptable limits in any case. Comparing with another choice that set the value of γ as a constant, different γ only decides the scale of value. Fig. 7 compares two selected situations when $\gamma = 1/\ln(n)$ and $\gamma =$ constant for different number of input point clouds, where $n = 1024, 768, 512, 256$, respectively. For the choice of constant, γ is set as 0.15 to keep the value of entropy close to $\gamma = 1/\ln(n)$ when $n = 1024$ for an intuitive comparison.

It can be observed that when $\gamma = 1/\ln(n)$, the value of entropy does not change a lot when n changes, values are always concentrated within a range. But when γ is a constant, the value of entropy decreases as n reduces. It can be concluded that the selection of $\gamma = 1/\ln(n)$ is able to unify dimensions with different size of input data, it is more flexible than using a constant when dealing with different types of data.

C. External Consistency Evaluation

In 2-D CNNs, 2-D CAM can locate the units labeled category in the image. In a similar way, per-points in the input point cloud are allocated to a particular category by extending the CAM technique to 3-D point cloud. Based on this, one can observe which part of object is correctly classified and which features in object mislead the classification output by comparing the output from the 3-D CAM with the ground-truth of category. In this article, convolution functions are evaluated from two aspects by the 3-D class activation mapping technique: 1) the CAMs of the network are visualized to observe what is the decision-making part in object and evaluate what is the difference in these convolution functions. 2) The ECA is used to measure the per-point classification accuracy, which reflects the degree of consistency between the output of the category in per-points and the ground-truth.

1) Visualization: Through the visualization of 3-D CAM, one can figure out which part of feature in object determines the classification decision, which is so-called feature recognition;

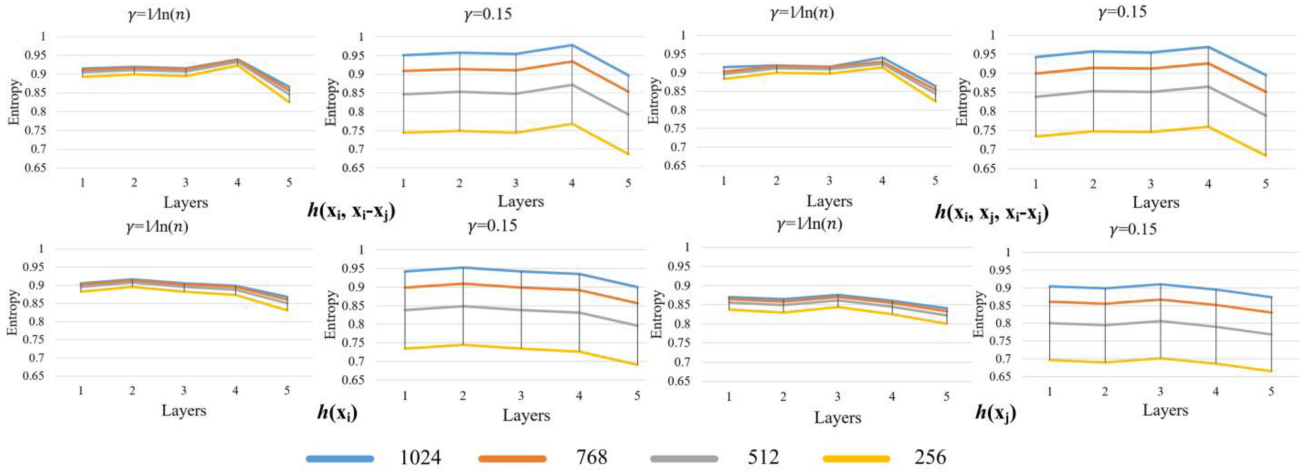


Fig. 7. Comparison for the parameter selection of γ between $\gamma = 1/\ln(n)$ and $\gamma =$ constant in different number of input point clouds.

Fig. 8. Visualization of per-point classification in different objects, the upper left corner, upper right corner, lower left corner, and lower right corner, respectively, correspond to the results of $h(x_i)$, $h(x_j)$, $h(x_i, x_i - x_j)$, and $h(x_i, x_j, x_i - x_j)$.

more features are detected, more points are allocated to corresponding category. Then, the class with most points makes the greatest contribution to final category in the network, it is desired that the correctly classified points are in the majority, and the misclassified points are as fewer as possible. The visualization of per-point classification of different objects is shown in Fig. 8.

First, feature recognition ability of convolution function is discussed. Fig. 8(a) displays the results in per-point classification for airplane model; both convolution functions have good performance in detecting airplane model because features in airplane model are easy to recognize. Although almost all points are correctly classified in this case, there are differences in

TABLE IV
AVERAGE ECA OF CONVOLUTION FUNCTIONS

	$h(x_i)$	$h(x_j)$	$h(x_i, x_i - x_j)$	$h(x_i, x_j, x_i - x_j)$
Average ECA	63.9%	86.2%	94.1%	**94.8%**

other models. In Fig. 8(b) which shows the 3-D CAM of grass model, discrepancy points can be seen in different degrees for test functions. The misclassified points are mostly at the leaf of grass for $h(x_i)$ and $h(x_j)$, while the misleading part is located at the inner of grass in $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$. It can be inferred that the related relationship between x_i and x_j can help to deal with the sharp features like leaf of grass model, but inner scattered point clouds are hard to distinguish from each other. Fig. 8(c) and (d) shows that the discernment of $h(x_i)$ is obviously weaker than others, because lacking of neighbor features makes $h(x_i)$ not sensitive to planes and curves sharps. In conclusion, the per-point classification of $h(x_j)$, $h(x_i, x_i - x_j)$, and $h(x_i, x_j, x_i - x_j)$ has better performance than $h(x_i)$, illustrating $h(x_i)$ is weaker than other functions in feature recognition ability.

Next, the recognizable parts of the objects are discussed according to the CAM. In the case of grass model in Fig. 8(b), it can be seen that the leaf is an outstanding part to enhance the ability to recognize grass model when comparing the performance of four convolution functions. Fig. 8(d) indicates that the decision part in the bottle is the punt, not the bottleneck. Fig. 8(e) and (f) shows a category of chairs of different types; it can be seen that the decision parts of chairs are gathered in the main skeleton, the feet of the chair are confusable parts, and the pole sharp is hard to distinguish between other categories with the same sharps. From the visualization of different objects, it can infer that the common sharp parts of structure will have certain effect in the recognition ability of convolution function, it is easily confusing when just utilize single local feature, so that is why the global feature of the context should be taken into account [42].

Visualization gives a direct observation of these conclusions. For a comprehensive evaluation, the ECA proposed in Section III-C is used as a metric of the per-point classification accuracy and show more discovery.

2) Measurement: First, the average ECA of all samples with four convolution functions is counted in Table IV for overall observation. $h(x_i, x_j, x_i - x_j)$ receives the highest ECA and ECA of $h(x_i, x_i - x_j)$ gets close to $h(x_i, x_j, x_i - x_j)$, indicating that $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$ have majority of correctly classified points and good ability to distinguish features of these models. The ECA of $h(x_j)$ is lower than $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$, but it still got 86.2% recognition rate. The performance of $h(x_i)$ is worse than others and it only got ECA of 63.9%, illustrating that the ability of feature recognition of $h(x_i)$ is weak, the conclusion obtained from visualization is verified quantitatively in here. By the average EAC, feature recognition ability can be initially ranked from $h(x_i, x_j, x_i - x_j) \geq h(x_i, x_i - x_j) > h(x_j) \gg h(x_i)$.

Next, information extracted from the ECA of each category is discussed. Fig. 9 shows the average EAC of the samples, which belong to the same class for 40 categories. Histogram is used to compare the recognition of each category between four convolution functions. It can be seen that the convolution function $h(x_i, x_j, x_i - x_j)$ achieves the highest ECA in most categories, and EAC of each category are in a higher level. $h(x_i, x_i - x_j)$ is second, and its performance is better than $h(x_i, x_j, x_i - x_j)$ in some particular categories. Although in Table I the classification accuracy of external network shows that $h(x_i, x_i - x_j)$ is more efficient than $h(x_i, x_j, x_i - x_j)$, but in terms of the ability of feature recognition, $h(x_i, x_j, x_i - x_j)$ is better than $h(x_i, x_j, x_i - x_j)$; and $h(x_j)$ is in the third, but in some categories, it is much weaker than the first two functions; $h(x_i)$ produces unsatisfactory results, although mostly category with a enough level of EAC can guide the correct classification, but lower EAC means the convolution function are vulnerable by the points which are wrongly classified, resulting in the lower classification accuracy. From Fig. 9, it can be concluded that if the ECA of convolution function is in a high level, more points inside the objects are correctly classified, and final output is more stable, illustrating more features are correctly recognized and the feature recognition ability of convolution function is better.

Finally, the feature recognition ability of convolution function for different category is disused as follows. The average EAC of samples which belong to the same category for every convolution function is shown in Fig. 10, respectively; four convolution functions get higher ECA in the category of airplane, chair, and keyboard indicating that these categories have obvious features that are easy to distinguish. But the EAC of wardrobe, radio, and xbox are lower than other categories obviously, because these categories have common attributes like square shape which can be easily confused, making the convolution function not sensitive to them. These objects belong to the categories which are lack of the particular recognizable feature. It can be concluded that convolution functions have different ability to recognize different categories, which depends on the geometry character of them, that is why the edge feature is needed in $h(\cdot)$.

Through the visualization and measurable metric, the feature recognition ability of the convolution function is evaluated from the external consistency network. It can be inferred that ECA of convolution function $h(x_i, x_j, x_i - x_j) \geq h(x_i, x_i - x_j) > h(x_j) \gg h(x_i)$, which indicates that the feature recognition ability are affected by uses of different edge features, and through the difference of ECA of category, it can be concluded that the shape of object is also an influence factor in object recognition.

D. Results of Evaluation

To sum up, this article evaluates the feature transformation ability and feature recognition ability of four typical convolution functions from internal consistency evaluation and external consistency evaluation, the overall results are shown in Table V. The performance of convolution functions is compared with quantification and visualization evaluation. It can

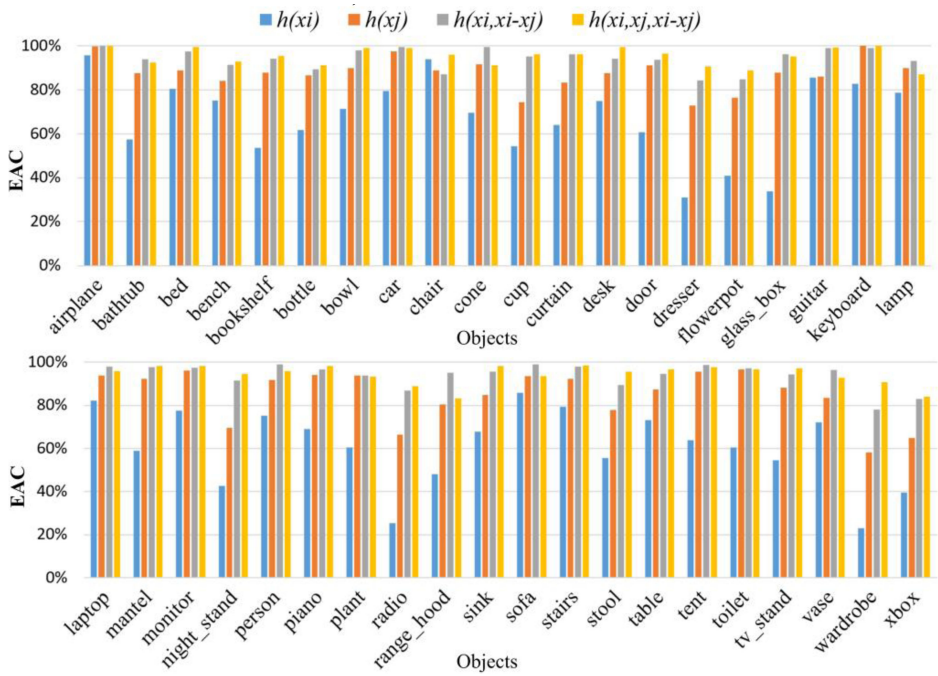


Fig. 9. Average EAC of four convolution functions for 40 categories.

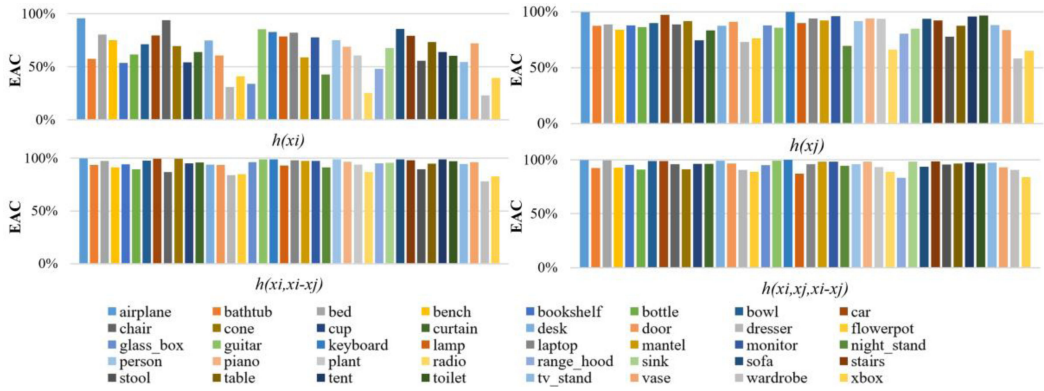


Fig. 10. Average EAC of 40 categories under every convolution function, respectively.

TABLE V
OVERALL RESULTS OF CONVOLUTION FUNCTION EVALUATION

	Feature transformation ability	Feature recognition ability
$h(x_i)$	low	low
$h(x_j)$	low	high
$h(x_i, x_i - x_j)$	higher	higher
$h(x_i, x_j, x_i - x_j)$	higher	higher

be seen that the edge features are of great significance for feature transformation and recognition, and the effects vary with different form of edge feature, such as $h(x_j)$ does not perform as well as $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$. Regarding $h(x_i, x_i - x_j)$ and $h(x_i, x_j, x_i - x_j)$, they have their own characteristics in terms of feature transformation ability and feature recognition ability, and they achieve better performance in the 3-D classification network as evidenced by experimental results.

Limitations: 1) Our work only test four typical convolution functions, more convolution functions from state-of-the-art 3-D CNNs work are expected to involve. 2) In the external consistency network, in order to obtain 3-D CAM, the network structure is modified to reinforce interpretation, incurring reduced accuracy of final output. Although this is a common problem in interpretation work [40], it is desired to study a method to build up the projection between the category output and per-point without accuracy degradation. 3) At present, the interpretation work of 3-D CNNs is limited to only 3-D convolution function evaluation in our work, so it is necessary to explore more aspects in the 3-D CNNs interpretation work.

V. Conclusion

The purpose of this article is to study the interpretation of 3-D point cloud deep learning network and evaluate the convolution operation performance of the typical edge convolution functions. The key work of this article can be included in three points: 1) An interpretation network with branch structure is studied for a comprehensive evaluation of 3-D convolution functions. It provides the materials for the assessment of the internal consistency and external consistency; it can retain an acceptable classification accuracy as evidenced by the experiment to support the follow-up evaluation. 2) In the convolutional layer, max information entropy is introduced to represent the activation of the dimensional feature; it proved that entropy has a positive correlation with effective activation feature, which provides the basis to take entropy as a metric of feature transformation ability of convolution function. The "health state" of the convolution function can be diagnosed by comparing the entropy distribution between different conv-layers; and their feature transformation ability can be evaluated by comparing the entropy values in corresponding convolutional layer. 3) The 3-D CAM technique was used to establish the response between per-point and the classification category. The ECA is used to measure the per-point classification accuracy, competing the feature recognition ability of different convolution functions and exploring the identifiability of category by quantitative analysis. Finally, the evaluation between different convolution functions summarizes based on their performance in the interpretation work.

The research work presented in this article is a heuristic work, and it has a prospect in the application of the measurable metric, for example, entropy value can be used to select the dimensions with rich information, so as to optimize the network frame and save the consumption of parameter storage. And multisource data could be considered, like RGB-data or 4-D point cloud [43], [44], more work could be extended in 3-D CNNs interpretation works in future work.

References

[1] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 652–660.

[2] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[3] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on x-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 828–838.

[4] W. Wu, Z. Qi, and L. Fuxin, "PointConv: Deep convolutional networks on 3D point clouds," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 9621–9630.

[5] Y. Xie, J. TIAN, and X. Zhu, "Linking points with labels in 3D: A review of point cloud semantic segmentation,"*IEEE Geosci. Remote Sens. Mag.*, to be published, doi: 10.1109/MGRS.2019.2937630.

[6] R. C. Fong and A. Vedaldi, "Interpretable explanations of black boxes by meaningful perturbation," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV), Venice*, pp. 3449–3457, 2017, doi: 10.1109/ICCV.2017.371.

[7] J. M. Benítez, J. L. Castro, and I. Requena, "Are artificial neural networks black boxes," *IEEE Trans. Neural Netw.*, vol. 8, no. 5, pp. 1156–64, Sep. 1997.

[8] B. A. Alejandro *et al.*, "Explainable artificial intelligence (XAI) concepts, taxonomies, opportunities and challenges toward responsible AI," *Inf. Fusion*, vol. 58, pp. 82–115, 2019.

[9] F. K. Dosilovic, M. Brcic, and N. Hlupic, "Explainable artificial intelligence: A survey," in *Proc. MIPRO 41st Int. Convention Inf. Commun. Technol., Electron. Microelectron.*, 2018, pp. 0210–0215.

[10] Q. Q. Yuan, Q. Zhang, J. Li, H. She, and L. Zhan, "Hyperspectral image denoising employing a spatial-spectral deep residual convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1205–1218, Feb. 2019.

[11] X. Yang, W. Yan, W. Ni, X. Pu, H. Zhang, and M. Zhang, "Object-guided remote sensing image scene classification based on joint use of deep-learning classifier and detector," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2673–2684, May 2020.

[12] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6690–6709, Sep. 2019.

[13] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 2921–2929.

[14] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Proc. Int. Conf. Comput. Vision*, Barcelona, Spain, 2011, pp. 2018–2025, doi: 10.1109/ICCV.2011.6126474.

[15] M. D. Zeiler and R. Fergus. "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vision, Cham*, Switzerland, 2014, vol. 8689, pp. 818–833, doi: 10.1007/978-3-319-10590-1_53.

[16] C. Szegedy W. Zaremba, I. Sutskever *et al.*, "Intriguing properties of neural networks," *Comput. Vision and Pattern Recognition*, pp. 1–9, 2013, *arXiv:1312.6199*.

[17] J. W. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 828–841. Oct. 2019.

[18] Q. S. Zhang, W. G. Wang, and S. C. Zhu, "Examining CNN representations with respect to dataset bias," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 4464–4473.

[19] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 146:1–146:12, 2019.

[20] Y. Liu, B. Fan, S. Xiang, and C. Pan, "Relation-shape convolutional neural network for point cloud analysis," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit. (CVPR)*, Long Beach, CA, USA, 2019, pp. 8887–8896, doi: 10.1109/CVPR.2019.00910.

[21] M. Simonovsky and N. Komodakis, "Dynamic edge-conditioned filters in convolutional neural networks on graphs," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 29–38, doi: 10.1109/CVPR.2017.11.

[22] Q. S. Zhang and S. C. Zhu, "Visual interpretability for deep learning: A survey," *Frontiers Inf. Technol. Electron. Eng.*, vol. 19, no. 01, pp. 30–42, 2018.

[23] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013.

[24] A. Dosovitskiy and T. Brox, "Inverting visual representations with convolutional networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 4829–4837, doi: 10.1109/CVPR.2016.522.

[25] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vision (ECCV)*, 2014, vol. 8689, pp. 818–833.

[26] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," in *IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit.*, San Francisco, CA, USA, 2010, pp. 2528–2535, doi: 10.1109/CVPR.2010.5539957.

[27] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free?—Weakly-supervised learning with convolutional neural networks," in *Proc. Comput. Vision Pattern Recognit.*, 2015.

[28] R. R. Selvaraju *et al.*, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," *Int. J. Comput. Vision*, vol. 128, no. 2, pp. 336–359, 2020.

[29] D. Kumar, A. Wong, and G. W. Taylor, "Explaining the unexplained: A class-enhanced attentive response (CLEAR) approach to understanding deep neural networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, 2017, pp. 1686–1694, doi: 10.1109/CVPRW.2017.215.

[30] H. Lakkaraju, E. Kamar, R. Caruana, and E. Horvitz, "Identifying unknown unknowns in the open world: Representations and policies for guided exploration," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 2124–2132.

[31] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, "Network dissection: Quantifying interpretability of deep visual representations," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 6541–6549.
[32] B. Zhou *et al.*, "Interpreting deep visual representations via network dissection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2131–2145, Sep. 2019.
[33] R. Fong and A. Vedaldi, "Net2Vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 8730–8738.
[34] B. Zhang *et al.*, "Explaining the pointNet what has been learned inside the pointNet," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 71–74.
[35] Y. Y. Li *et al.*, "PointCNN convolution on X-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018.
[36] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 4, pp. 623–656, 1948.
[37] J. Yosinski *et al.*, "How transferable are features in deep neural networks?," *Mach. Learn.*, 2014, *arXiv:1411.1792*.
[38] M. Oquab *et al.*, "Is object localization for free?—Weakly-supervised learning with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2015, pp. 685–694.
[39] Z. Wu *et al.*, "3D shapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 1912–1920.
[40] Q. Zhang, Y. N. Wu, and S. C. Zhu, "Interpretable convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 8827–8836, doi: 10.1109/CVPR.2018.00920.
[41] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Object detectors emerge in deep scene CNNs," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2014.
[42] S. Xia, D. Chen, R. Wang, J. Li, and X. Zhang, "Geometric primitives in LiDAR point clouds: A review," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 685–707, 2020.
[43] M. Ioannides *et al.*, "Online 4D reconstruction using multi-images available under open access," *ISPRS Photogr. Remote. Sens. Spat. Inf. Sc.*, vol. 2, pp. 169–174, 2013.
[44] A. Mustafa *et al.*, "Temporally coherent 4D reconstruction of complex dynamic scenes," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 4660–4669.

Bufan Zhao received the B.S. and M.S. degrees from the School of Geodesy and Geomatics, Wuhan University, Wuhan, China, where he is currently working toward the Ph.D. degree.

His research interests include 3-D laser scanning data processing and quality evaluation.

Xianghong Hua (Member, IEEE) was born in Tai Xian, Jiangsu Province, China, in 1963. He received the Ph.D. degree in engineering from Wuhan University, Wuhan, China, in 2006.

He is currently a Professor of Geodesy and surveying engineering, the Director of the Wuhan University Hazard Monitoring and Prevention Research Center. His research interests include engineering survey and thematic GIS, 3D laser scanning data processing and quality evaluation, engineering and disaster monitoring and forecasting, multisensor information fusion, and seamless positioning and navigation technology.

Kegen Yu (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Sydney, Sydney, NSW, Australia, in 2003.

He was with Jiangxi Geological and Mineral Bureau, Nanchang, China; Nanchang University, Nanchang, China; the University of Oulu, Oulu, Finland; the CSIRO ICT Center, Sydney, NSW, Australia; Macquarie University, Sydney, NSW, Australia; the University of New South Wales, Sydney, NSW, Australia; and Wuhan University, Wuhan, China. He is currently a Professor with the School of Environmental Science and Spatial Informatics, China University of Mining and Technology, Xuzhou, China. He has coauthored the book *Ground-Based Wireless Positioning* (Wiley and IEEE Press, 2009, a Chinese version of the book is also available) and another book *Wireless Positioning: Principles and Practice* (Springer, 2018), and has authored or coauthored more than 80 refereed journal papers and more than 60 conference papers. He edited the book *Positioning and Navigation in Complex Environments* (IGI Global, 2018) and another book *Indoor Positioning and Navigation* (Science Press, 2019). His research interests include global-navigation-satellite-system reflectometry, ground-based and satellite-based positioning, and remote sensing.

Wuyong Tao received the master's degree from the East China Institute of Technology, Nanchang, China, in 2015. He is currently working toward the Ph.D. degree with the School of Geodesy and Geomatics, Wuhan University, Wuhan, China.

His research interests include laser scanning and data processing.

Xiaoxing He received the Ph.D. degree in geodesy and surveying engineering from the School of Geodesy and Geomatics, Wuhan University, Wuhan, China, in 2016.

He has been a Postdoctoral Researcher with the GNSS Research Center, Wuhan University, since 2017. He is currently a Lecturer with the School of Civil Engineering and Architecture, East China JiaoTong University, Nanchang, China. His research interests include the theory of satellite Geodesy and its applications, assessment of noise characteristics, and analysis of geodetic time series.

Shaoquan Feng received the bachelor's degree from Wuhan University of Technology, Wuhan, China, in 2017. He is currently working toward the master's degree with the School of Geodesy and Geomatics, Wuhan University, China.

His research interests include laser scanning and simultaneous localization and mapping.

Pengju Tian received the B.S. and M.S. degrees from Wuhan University, Wuhan, China, in 2014 and 2018, respectively. He is currently working toward the Ph.D. degree with the School of Geodesy and Geomatics, Wuhan University, Wuhan, China.

His research interests include laser scanning and data processing.