IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 25, 2020, accepted August 13, 2020, date of publication August 24, 2020, date of current version September 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3018875

Semantic and Context Information Fusion Network for View-Based 3D Model Classification and Retrieval

AN-AN LIU, (Member, IEEE), FU-BIN GUO, HE-YU ZHOU, WEN-HUI LI, AND DAN SONG

School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China

Corresponding authors: Wen-Hui Li (liwenhui@tju.edu.cn) and Dan Song (dan.song@tju.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61772359, Grant 61872267, and Grant 61902277; in part by the Tianjin New Generation Artificial Intelligence Major Program under Grant 19ZXZNGX00110 and Grant 18ZXZNGX00150; and in part by the Open Project Program of the State Key Lab of CAD & CG, Zhejiang University under Grant A2005 and Grant A2012.

ABSTRACT In recent years, with the rapid development of 3D technology, view-based methods have shown excellent performance in both 3D model classification and retrieval tasks. In view-based methods, how to aggregate multi-view features is a key issue. There are two commonly used solutions in the existing methods: 1) Use pooling strategy to merge multi-view features, but it ignores the context information contained in the continuous view sequence. 2) Leverage grouping strategy or long short term memory networks (LSTM) to select representative views of the 3D model, however, it easily neglects the semantic information of individual views. In this paper, we propose a novel Semantic and Context information Fusion Network (SCFN) to compensate for these drawbacks. First, we render views from multiple perspectives of the 3D model and extract the raw feature of the individual view by 2D convolutional neural networks (CNN). Then we design the channel attention mechanism (CAM) to exploit the view-wise semantic information. By modeling the correlation among view feature channels, we can assign higher weights to useful feature attributes, while suppressing the useless. Next, we propose a context information fusion module (CFM) to fuse multiple view features to obtain a compact 3D representation. Extensive experiments are conducted on three popular datasets, *i.e.*, ModelNet10, ModelNet40, and ShapeNetCore55, which can demonstrate the superiority of the proposed method comparing to the state-of-the-arts on both 3D classification and retrieval tasks.

INDEX TERMS 3D model, semantic information, context information, CNN.

I. INTRODUCTION

In recent years, with the wide application of 3D technology in virtual reality, 3D printing, medical diagnosis, and other fields [1]–[4], the number of 3D models is proliferating, which makes the 3D model classification and retrieval tasks receive a surge of attention. The most critical step in these tasks is to learn a discriminative 3D model descriptor. The current 3D model descriptor extraction methods can be divided into two mainstreams: model-based methods and view-based methods. Model-based methods [5]–[11] describe the 3D model by the raw representation, *i.e.*, point cloud, voxel, and mesh, which can preserve more completed structure information. However, complex computation restricts its application in real scenarios. View-based methods [12]–[19] usually first place virtual cameras around the 3D model to obtain multiple views, then extract features of each view through 2D CNN, and finally, fuse those view features into a compact 3D model descriptor. Since the remarkable progress of deep learning has been achieved in the 2D image recognition field [20], [21], view-based methods have been proved more successful compared to model-based methods.

The associate editor coordinating the review of this manuscript and approving it for publication was Yongqiang Zhao.

A. MOTIVATION

Although many works focus on the 3D model classification and retrieval tasks, there still exist some issues to be solved.

- **How to exploit the view-wise semantic information contained in individual views.** Since the latest image feature extraction technique [20], [21] can be directly employed to encode multiple views of the 3D model, the current view-based 3D model analysis methods mainly focus on how to aggregate the multiple view features into a compact 3D model descriptor. For example,

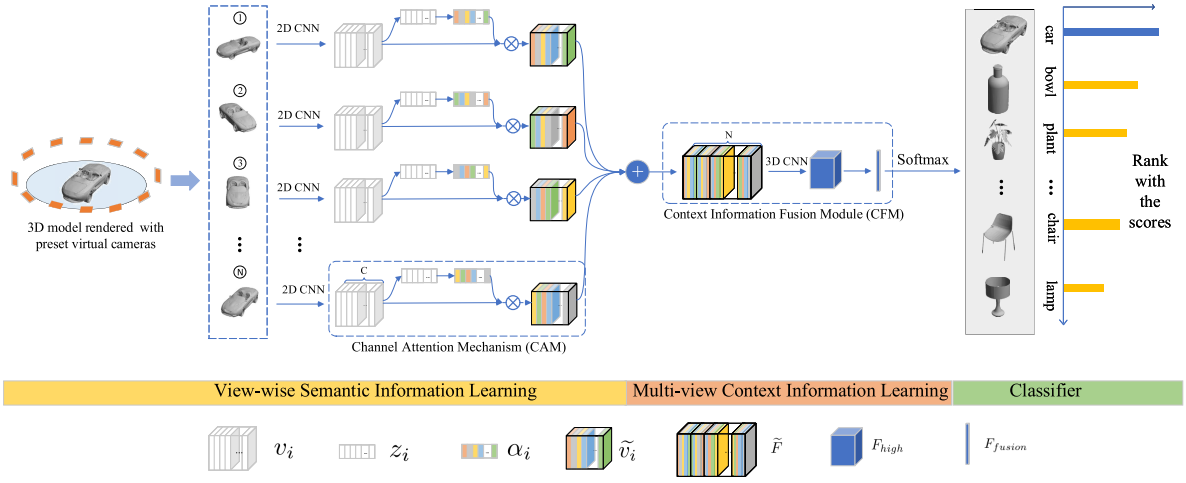


FIGURE 1. The Overview of the proposed method (SCFN).First, we place virtual cameras around the 3D model to capture multiple views from different directions. Then 2D CNN is applied to extract the raw view feature, and a channel attention mechanism (CAM) is designed to update raw view feature by channel-wise weight. Next, these updated view features are concatenated and processed by context information fusion module (CFM) to explore spatial information. At last, the fused features are extracted for classification or retrieval tasks.

some view-based methods [17] employ the pooling strategy to fuse multi-view features, while others [22], [23] attempt to select the representative views to depict the 3D model. These methods only focus on multi-view feature fusion. Nevertheless, they neglect the view-wise semantic information contained in individual views, which plays an essential role in feature representation. As introduced in [24], the different channels in 2D CNN feature focus on different regions of an image. Regions with rich semantic information contribute more to view features. For instance, a rendered view of the 3D car model typically consists of two parts: the background and the car. Some channels in this view feature pay more attention to the background, while some channels focus on the car. If we can capture more effective information from the view, the features of each view will be more distinctive, which is crucial for generating a compact 3D model descriptor. Therefore, it's necessary to exploit more effective view-wise semantic information contained in individual views.

- **How to mine multi-view context information contained in view sequence.** Although deep learning has been thoroughly studied in the field of 2D image recognition, its application in 3D models is still in infancy. Some networks and strategies [8], [17] adopted by view-based methods cannot fully consider the characteristics of 3D models. For instance, Su *et al.* [17] adopt the max-pooling layer to fuse view features. However, such an operation may discard the critical information in the informative views since it only retains the maximum value of each view feature. Besides, the fused feature does not consider the multi-view context information contained in view sequence, which has a significant influence on human observation and recognition of objects. Some previous works also use RNN or LSTM [25], [26] to aggregate multi-view features and mine context information among views. But this sequential mechanism requires much more computing resources, which substantially impair the ultimate performance. Therefore, mining the context information contained in view sequence is crucial for 3D model classification and retrieval tasks.

To address the aforementioned problems, we propose a novel semantic and context information fusion network (SCFN) for 3D model classification and retrieval. As shown in Fig. 1, SCFN mainly consists of two modules: view-wise semantic information learning module and multi-view context information learning module. In the first module, we employ CNN to encode individual views of the 3D model to obtain the multiple raw view features. Then we propose a channel attention mechanism (CAM) to exploit the useful semantic information of the raw view features. Specifically, by explicitly modeling the correlation among feature channels, we can enhance the useful attributes according to the channel-wise weight and suppress the attributes that are not useful for the current task. In the second part, we propose a context information fusion module (CFM) to acquire a compact 3D representation. Since the rendered views of the 3D model are sequential, there exists context information between the adjacent 2D view features. To effectively synthesize spatial and context information, a 3D CNN followed by global average pooling is proposed in CFM. By employing 3D CNN, the motion and appearance among multiple consecutive views are well modeled simultaneously. Finally, we use this unified 3D descriptor to complete the classification and retrieval tasks. Extensive experiments on three benchmarks, ModelNet10, ModelNet40, and ShapeNetCore55, can verify the superiority of the proposed method comparing against state-of-the-art approaches on both 3D model classification and retrieval tasks.

B. CONTRIBUTION

The main contributions of this paper are summarized in the following three aspects:

- Unlike previous view-based 3D model analysis methods, which only focus on the multi-view feature aggregation but neglect the feature representation capability of individual views, we propose a channel attention mechanism (CAM) to enhance the useful semantic information contained in individual views.
- We employ a context information fusion module (CFM) to aggregate multi-view feature by a 3D CNN and a global average pooling layer, which can fully explore the context information and spatial information among view sequences.
- We conduct extensive experiments on ModelNet10, ModelNet40 and ShapeNetCore55 [27], respectively. The experimental results can validate the superiority and effectiveness of the proposed method compared to the state-of-the-art methods.

This paper is organized as follows. In Section II, we introduce some related works. The specific details of the method are explained in Section III. The relevant experimental settings are presented in Section IV. Experimental results and discussion are provided in Section V. Finally, Section VI draws a conclusion of the paper.

II. RELATED WORK

In this section, we briefly review the impressive methods proposed in the 3D model classification and retrieval tasks. According to the diverse data formats used in these 3D models depicting methods, we can divide them into two categories: model-based methods and view-based methods.

- **Model-based methods**: The model-based methods take the raw representation of the 3D model as input, such as mesh, volume and point cloud. Since handcrafted features are widely used in the many fields of computer vision [28]–[31], which can well reflect the characteristics of the data. In previous work, researchers tended to design handcrafted features, like point feature histograms [32], and local surface feature descriptions [33] to recognize these raw representations. Mian *et al.* [34] propose a fully automatic 3D model-based free-form object recognition and segmentation algorithm. It is a multiview correspondence algorithm that automatically registers unordered views of an object with $O(n)$ complexity. Fang *et al.* [35] propose a temperature distribution descriptor. TD descriptor is capable of exploring the intrinsic geometric features on the shape. Andy *et al.* [36] propose 3DMatch. The method learns a local volumetric patch descriptor to match partial 3D data. However, the handcrafted features have to be redesigned when the data source changes, which limited its application scenarios. In recent years, more and more researchers have investigated deep learning methods to process 3D models and achieved great success. Wu *et al.* [27] propose 3D ShapeNet, which represents the 3D model as the probability distribution of the 3D voxel grid binary variable. Gernot *et al.* [37] present OctNet, where a set of unbalanced octrees are employed to divide the sparse input data space, and the leaf nodes are used to store the set features, which greatly reduces the cost of computing and memory. Brock *et al.* [38] design a voxel-based variable autoencoder to explore the latent space of 3D shapes, and a deep convolutional neural network architecture for object classification. They address the unique challenges of voxel-based representations. Despite that VRN Ensemble preforms well, it requires more computation cost. Wu *et al.* [11] propose a disordered graph convolutional neural network (DGCNN). This architecture can dynamically update the graph by EdgeConv, which can capture local geometric information while ensuring the invariance of permutation. Qi *et al.* [9] propose PointNet, this structure can extract the key points to represent the 3D model. This capability makes the PointNet robust to noise and data loss. However, this method cannot capture the local structure of the 3D model due to the correlation between local points is not learned. To solve this problem, Qi *et al.* propose PointNet++ [10]. This method employs the hierarchical neural network to recursively apply PointNet on the point cloud, which has achieved satisfactory results. Zhi *et al.* [39] propose LightNet. It uses multi-task learning to solve real-time 3D object recognition problems. Extensive experiments have proven LightNet's superior object recognition accuracy and computational efficiency in real-time tasks. This method provides a fairly basic structure and the recognition performance can be further improved by adding more effective modules. Through the above introduction, it can be seen that the model-based method implicitly requires model information. According to the model information, these model-based methods can be further divided into mesh-based methods, volume-based methods, point-cloud-based methods and so on.
- **View-based methods**: The view-based methods leverage a set of views to depict the 3D model, and researchers construct a discriminative 3D descriptor from these views. In previous work, Chen *et al.* [13] propose a visual 3D model retrieval system. The system employs the Zernike moment and Fourier descriptor encoding views for the 3D model retrieval task. Liu *et al.* [40] propose a multi-view latent variable model (MLVM). This method designs an undirected graph structure to discover the potential spatial context information of a given 3D model. Su *et al.* [17] propose a multi-view convolutional neural network (MVCNN). This method employs 2D CNN to extract the feature of each view and fuses the multi-view features by max-pooling strategy, which has achieved great success in 3D model classification and retrieval tasks. Bai *et al.* [12] propose a 3D shape search engine (GIFT). This engine employs 2D CNN to extract view features and

matches them to calculate the similarity between 3D models. Seout *et al.* [41] propose a stereographic projection neural network (SPNet). This method learns the feature representation of a 3D model by transforming the input 3D model into a 2D planar image using stereo projection. Sfikas *et al.* [15] propose an ensemble of PANORAMA-based 2D CNN (PANORAMA-NN). This method takes the panoramic view of the 3D model as the input to the convolutional neural network and employs the SYMPAN method to normalize the pose. Jiang *et al.* [42] propose a multi-loop-view convolutional neural network (MLVCNN). It introduces a hierarchical view-loop-shape architecture to deal with multiple groups of views. It makes better use of the local features of a view in a loop, while taking into account the global feature representation. However, the view rendering setting and network structure is more complicated than most view-based methods. Kanezaki *et al.* [43] propose RotationNet. It considers view-point labels that are ignored by other methods as latent variables and jointly estimates object category and view point from each single-view image. It achieves excellent results in the 3D classification task. However, it also has the limitation that each image should be observed from one of the pre-defined viewpoints. These view-based methods have achieved promising progress in the 3D model classification and retrieval tasks, but there are still some problems to be solved. For example, MVCNN [17] propose the max-pooling layer to fuse multi-view features, which does not consider view-wise semantic information and ignores the context information contained in the consecutive view sequence.

In recent years, some researchers propose multimodal fusion methods [8], [8], [44], [45]. For example, Vishakh *et al.* [44] propose FusionNet. This method combines volumetric representation and pixel representation to learn new features. Qi *et al.* [8] design a new method to improve the existing volumetric CNN and multi-view CNN. These methods make full use of the characteristics of different data representations, but are also limited by the gap between different data representations.

III. METHODS

In this section, we firstly give the problem definition and overview. In the remaining part, we introduce the SCFN network architecture and explain the main modules of SCFN in detail.

A. PROBLEM FORMULATION AND OVERVIEW

1) PROBLEM FORMULATION

In the 3D model classification and retrieval tasks, the view-based methods usually use a view set of the 3D model to represent the 3D model. Given a 3D model, we render N views from different perspectives. In the training process, we train a feature extraction network to extract the feature of each view. Then we select an appropriate strategy to aggregate the features of all views to generate a compact 3D descriptor. In the 3D classification task, we use the 3D descriptor as the input of the softmax function to predict the category label of the 3D model. In the 3D retrieval task, we adopt Euclidean distance between descriptors to measure the similarity of two models. Based on the similarity, we select the candidate set that meets the query criteria as the retrieval result.

2) OVERVIEW

As shown in Fig.1, in our method, we designed the following two modules to complete these two tasks:

- **View-wise Semantic Information Learning.** In order to exploit the useful semantic information of the raw view features, this module consists of two parts: raw view feature extraction and channel attention mechanism. Firstly, we leverage virtual cameras to extract a set of views of the 3D model and use 2D CNN to extract the raw features of each view. Secondly, we enhance the raw view features by channel-wise attention modeling.
- **Multi-view Context Information Learning.** The purpose of this module is to fuse multiple enhanced view features and generate a compact 3D description. Inspired by the relevant methods in the field of video processing [46]–[48], we use 3D CNN as the essential component of CFM to mine context information.

B. VIEW-WISE SEMANTIC INFORMATION LEARNING

1) RAW VIEW FEATURE EXTRACTION

Given a 3D model, we extract multiple 2D views from different viewpoints, as in Fig 1. In the view capturing process, we fix an upright orientation axis as the rotating axis. Then we place virtual cameras point to the centroid of the 3D model with intervals of θ [17]. We use AlexNet [20] as the backbone CNN for 2D view feature extraction. The AlexNet is composed of five convolutional layers (*conv*1-*conv*5) and three fully connected layers (*fc*6-*fc*8). We acquire the output of *conv*5 as the raw view feature. Therefore, the raw multi-view feature set representation can be written as:

$$F = \{v_1, v_2, \ldots, v_i, \ldots, v_N\}, \quad v_i \in R^{H \times W \times C} \tag{1}$$

where v_i represents the raw view feature of i-th view, the width, height, and number of channels of the feature are denoted by W, H and C respectively.

2) CHANNEL ATTENTION MECHANISM

Previous methods [8], [17], [23] usually directly employ the raw view feature v_i for the subsequent multi-view feature fusion procedure. However, we notice that each channel of the v_i contains certain channel-wise statistics. In fact, the view feature channels are the response to the different convolutional filters. By explicitly modeling the correlation among channels, the network can increase the sensitivity responded to the useful statistics.

Therefore, we seek to design a novel channel attention mechanism (CAM) that can calculate the channel-wise importance according to the statistics contained in each channel. Based on the channel-wise importance, we can re-weight the channels to get a more informative view feature. Then, we

apply the average pooling operation on each channel of v_i to obtain a statistics $z_i \in R^{C\times 1}$. During this procedure, the c-th element of z_i is formulated as follow:

$$z_i^c = \frac{1}{H_i^c \times W_i^c} \sum_{h=1}^{H_i^c} \sum_{w=1}^{W_i^c} v_i^c(h, w) \tag{2}$$

where v_i^c represents the c-th channel of v_i. H_i^c and W_i^c refer to the height and width of v_i^c, respectively.

Since the statistics z_i don't have a non-linear learning ability to learn more complex correlations among channels, we adopt a gate mechanism with a sigmoid activation function to endow z_i a good non-linear learning ability.

$$\alpha_i = \phi\left(T_2\psi\left(T_1 z_i\right)\right) \tag{3}$$

where the sigmoid function and ReLU function are denoted by ϕ and ψ, respectively. The transformation T_1, T_2 are performed by two fully connected layers, which act as a bottleneck with a dimension reduction ratio to reduce the training cost.

Then we use $\alpha_i \in R^{C\times 1}$ to update the raw view feature v_i and get the updated view feature $\widetilde{v_i}$. The c-th channel of $\widetilde{v_i}$ is obtained by:

$$\widetilde{v_i^c} = \alpha_i^c v_i^c \tag{4}$$

where α_i^c represents the c-th element of α_i. In the end, the enhanced multi-view feature set is defined as:

$$\widetilde{F} = \{\widetilde{v_1}, \widetilde{v_2}, \ldots, \widetilde{v_i}, \ldots, \widetilde{v_N}\}, \quad \widetilde{v_i} \in R^{H\times W\times C} \tag{5}$$

C. MULTI-VIEW CONTEXT INFORMATION LEARNING

After obtaining the enhanced multi-view feature set $\widetilde{F}$, how to fuse the multi-view features into a unified 3D feature vector is vital to the final performance. Previous works [8], [17] usually employ the pooling strategy to fuse the multi-view features, which cannot discover the multi-view context information. We introduce the context information fusion module (CFM) to tackle this challenge. The module is inspired by the success in the video processing [46], [47], [49]. In this field, plenty of approaches divide the whole video into some frames and explore the context information among frames by 3D CNN.

CFM consists of two 3D convolution modules and a global average pooling operation. The 3D convolution module contains three parts: a 3D convolution layer with 1024 filters, a batch normalization layer, and a ReLU layer.

In the first 3D convolution module, the kernel size is set to $1 \times 1\times 1$ to increase network nonlinearity. The kernel size of the second 3D convolution module is $3 \times 3 \times 3$ to explore the context information among multi-view features. After the 3D convolution, the output is a high-dimensional feature $F_{high} \epsilon \mathcal{R}^{1024\times N'\times H'\times W'}$.

We use the high-dimensional feature F_{high} as the input of global average pooling layer to obtain the final fused feature:

$$F_{fusion} = \frac{1}{N'}\frac{1}{H'}\frac{1}{W'} \sum_{n=1}^{N'} \sum_{h=1}^{H'} \sum_{w=1}^{W'} F_{high} \tag{6}$$

The context information fusion module (CFM) proposed in this paper aims to explore the spatial and context information contained in view sequence and enhance the sensitivity of the network to input changes. Since there are numerous similar objects in the 3D model database, such as vase and bottle, these objects often lead to the wrong classification results due to their high similarity. Given some variations to the input, our 3D CNN module can notice the change and feed it back to the output. On the contrary, the single max-pooling operation will produce the same output, unless the input variable is obvious enough to alter the maximum value. The ability to capture subtle changes in the 3D CNN space helps to enhance the discrimination ability of the network. Besides, the final global average pooling facilitates SCFN resist against local region noise.

We use F_{fusion} as the final 3D descriptor, which is combined with the softmax function for the 3D model classification. In the 3D model retrieval, we define the Euclidean distance between two 3D model descriptors as the similarity of the two models. According to the similarity, we select the candidate set that meets the query criteria as the retrieval result.

IV. EXPERIMENTAL SETTINGS

A. DATASETS

Our method is evaluated on three popular 3D datasets: ModelNet10, ModelNet40 and ShapeNetCore55 [27].

- **ModelNet10:** ModelNet10 consists of 4,899 3D CAD models in 10 classes. Like [27], we select 3,991 3D models for training and 908 3D models for testing.
- **ModelNet40**[1]**:** ModelNet40 is composed of 12,311 3D CAD models of 40 classes. In ModelNet40, 9,843 3D models are used as the training set, and the remaining 3D models are used as the test set following the splits by Su *et al.* [17].
- **ShapeNetCore55**[2]**:** ShapeNetCore55 contains 51,190 3D models from 55 model categories. We set the training set, verification set, and test set ratio to 7:2:1 according to [27].

B. EVALUATION CRITERIA

On ModelNet10 and ModelNet40, we adopt the eight most representative and widely used criteria to validate the performance of our method, which are defined as follows [23], [50]:

- Nearest Neighbor (NN): It represents the percentage of the closest 3D models that matching the query.
- First Tier (FT): It represents the recall for the first top Q matching models, where Q is the number of query categories.
- Second Tier (ST): It represents the recall for the first top $2Q$ matching models, where Q is the number of query categories.
- The Mean Average Precision (mAP): It is a ranking measurement that can solve the problem of the single point value limitation of accuracies and recalls rate.

[1]http://modelnet.cs.princeton.edu/
[2]https://shapenet.cs.stanford.edu/shrec17/

TABLE 1. Performance comparison of different methods on ModelNet10 and ModelNet40.

Method	ModelNet10		ModelNet40	
	Accuracy	mAP	Accuracy	mAP
(1)3D ShapeNets [27]	91.0%	68.2%	77.3%	49.2%
(2)VoxNet [7]	92.0%	-	83.0%	-
(3)VRN [38]	-	-	91.3%	-
(4)MVCNN-MultiRes [8]	-	-	91.4%	-
(5)PointNet [9]	77.6%	-	-	-
(6)PointNet++ [10]	-	-	89.2%	-
(7)KD-Network [5]	-	-	91.8%	-
(8)PointCNN [6]	-	-	91.8%	-
(9)DGCNN [11]	-	-	92.2%	-
(10)LightNet [39]	93.9%	-	88.9%	-
(11)VRN Ensemble [38]	97.1%	-	95.5%	-
(12)SPH [14]	-	-	68.2%	33.3%
(13)LFD [13]	-	-	75.5%	40.9%
(14)Multiple Depth Maps [18]	-	-	87.8%	-
(15)GIFT [12]	92.3%	91.1%	83.1%	81.9%
(16)Pairwise [52]	92.8%	-	90.7%	-
(17)MVCNN [17]	-	-	90.1%	79.5%
(18)DeepPano [16]	84.5%	84.1%	77.6%	76.8%
(19)PANORAMA-NN [15]	91.1%	85.4%	90.7%	83.4%
(20)MLVCNN [42]	-	-	94.2%	**92.8%**
(21)RotationNet [43]	**98.5%**	-	**97.4%**	-
(22)Our	94.1%	**91.9%**	93.1%	85.1%

TABLE 2. Performance comparison of different methods on ShapeNetCore55.

Method	micro					macro				
	P@N	R@N	F1@N	mAP	NDCG	P@N	R@N	F1@N	mAP	NDCG
Kanezaki_RotationNet	81.0%	80.1%	**79.8%**	77.2%	86.5%	60.2%	63.9%	**59.0%**	58.3%	65.6%
Zhou_Improved_GIFT	78.6%	77.3%	76.7%	72.2%	82.7%	59.2%	65.4%	58.1%	57.5%	65.7%
Tatsuma_ReVGG	76.5%	80.3%	77.2%	74.9%	82.8%	51.8%	60.1%	51.9%	49.6%	55.9%
Furuya_DLAN	**81.8%**	68.9%	71.2%	66.3%	76.2%	**61.8%**	53.3%	50.5%	47.7%	56.3%
Thermos_MVFusionNet	74.3%	67.7%	69.2%	62.2%	73.2%	52.3%	49.4%	48.4%	41.8%	50.2%
Deng_CM-VGG5-6DB	41.8%	71.7%	47.9%	54.0%	65.4%	12.2%	66.7%	16.6%	33.9%	40.4%
Li_ZFDR	53.5%	25.6%	28.2%	19.9%	33.0%	21.9%	40.9%	19.7%	25.5%	37.7%
DMk_DeepVoxNet	79.3%	21.1%	25.3%	19.2%	27.7%	59.8%	28.3%	25.8%	23.2%	33.7%
SHREC16-Bai_GIFT	70.6%	69.5%	68.9%	64.0%	76.5%	44.4%	53.1%	45.4%	44.7%	54.8%
SHREC16-Su_MVCNN	77.0%	77.0%	76.4%	73.5%	81.5%	57.1%	62.5%	57.5%	56.6%	64.0%
Our	52.6%	**82.9%**	59.2%	**80.1%**	**88.2%**	20.1%	**76.4%**	21.3%	**62.5%**	**79.3%**

- F_measure: It is a comprehensive measurement considering both accuracy and recall for the top 20 retrieval results.
- Average Normalized Modified Retrieval Rank (ANMRR): It considers the ranking information of the relevant 3D models to measure the performance of the ranking list.
- Discounted Cumulative Gain (DCG): It is a statistical measurement that assigns higher weights to related 3D models while assigns lower weights to unrelated 3D models.
- Precision-Recall curve (PR curve): It is a key plot that visualizes the correlation between the precision and the recall.

On ShapenetCore55 [27], we adopt the evaluation code and indicators provided in shrec17 [3]. The indicators include Precision (P@N), Recall (R@N), F-score(F1@N), Mean Average Precision (mAP), and Normalized Discounted Cumulative Gain (NDCG). N is the length of the retrieval list. Besides, the micro averaged versions of these indicators give a weighted mean according to the size of each category, while the macro averaged versions give the unweighted mean regardless of the size of each category.

C. IMPLEMENTATION DETAILS

We employ a standard backpropagation strategy to train the entire network in an end-to-end manner. During the training stage, we utilize some strategies like dropout, weight decay, *etc.*, to prevent over-fitting. In our experiments, the learning rate is fixed at 0.0001. Our method is implemented based on the PyTorch framework.[4] All experiments are conducted on a server with two GeForce GTX1080 GPUs equipped with 12G memory, one Intel (R) Xeon (R) CPU, and 32G RAM.

V. EXPERIMENTAL RESULTS

A. COMPARISON WITH STATE-OF-THE-ART METHODS

As shown in Table.1 and Table.2, we compare SCFN with the state-of-the-art methods on ModelNet10, ModelNet40, and ShapeNetCore55. The experimental results on ShapeNet Core55 are all from the shrec17 competition. We chose

[3]https://shapenet.cs.stanford.edu/shrec17/

[4]https://pytorch.org/

accuracy and mAP as the main indicators to analyze the 3D model classification and retrieval performance.

On the ModelNet10 and ModelNet40 dataset, our method outperforms most of the methods, except VRN Ensemble [38], MLVCNN [17] and RotationNet [43]. Compared to these methods, on ModelNet10, SCFN gains 1.4% to 21.2% on accuracy and 0.4% to 34.1% on mAP. On ModelNet40, SCFN improves accuracy and mAP by 0.9% to 36.5% and 2.7% to 155.6%, respectively. In summary, we can draw the following conclusions:

- SCFN is superior to other representative view-based methods. Previous methods [8], [17] usually employ the pooling strategy to fuse multi-view features to obtain a unified 3D model representation. Although this operation is robust to view input order, the pooling strategy is too simple to capture the multi-view spatial context information. In contrast, our method can well handle this challenge with the help of 3D CNN.
- The performance of SCFN is better than most model-based methods, where the raw representation such as point cloud, voxel, and volume, are directly utilized to capture the structure information of the 3D model. However, these methods [6], [9], [10] cannot capture the latent visual semantic information of the 3D model. Comparatively, SCFN can capture that information by channel attention mechanism (CAM).
- Compared with SCFN, VRN Ensemble [38], MLVCNN [42] and RotationNet [43] achieve relatively better performance. VRN Ensemble designs a method for training voxel-based autoencoder to solve the unique challenges of voxel-based representation. But it is not scalable for real application since the voxel data is difficult to capture. Comparatively, 2D images are more available and it further boost the demand of view-based 3D shape retrieval methods. MLVCNN designs a hierarchical view-loop-shape structure, taking into account the local features of the view in loop and global feature representation. However, the multiple loop views (3 loops $\times$ 8 views) require more computation cost and memory storage. Comparatively, SCFN can achieve satisfactory performance with less views, and the experiments in Section.V-D can further validate the effectiveness of the proposed method. RotationNet considers view-point labels that are ignored by other methods as latent variables and jointly estimates object category and view point from each single-view image. However, it strongly depends on specific camera array settings, where the input view sequence should be in a fixed order, which restrict its application in real scenarios. Comparatively, the proposed SCFN is of the camera-constraint-free setting, where both the view number and view order can be arbitrary, and the details are presented in Section.V-C and Section.V-D.

As can be seen from Table.2, on ShapeNetCore55, compared with other state-of-the-art algorithms, the proposed SCFN obtains better accuracies in six indicators, such as R@N, mAP, and NDCG (both micro and macro), but relatively poor performance in other four indicators. Intuitively, the high precision and recall denotes a well performance, but in fact these two evaluation criteria performance in contrary. In other words, a well precision usually means a passable but not well recall. Therefore, the F1 score and mAP have been introduced to comprehensively measure them. The F1 score can be computed by the harmonic mean of precision and recall, it, however, may cause a single value limitation problem and thus will be easily affected by the sample distribution. Comparatively, this problem can be solved by the mAP since it is decided by the area circled by the precision and recall. mAP, hence, is usually regarded as the most important criterion for measuring the retrieval performance. This explains the lower P@N, F1@N but higher R@N and mAP achieved by the proposed method. Besides, the NDCG takes into account the position of the retrieval result in the retrieval list and thus has been regarded as an important evaluation criterion as well. Compared with other state-of-the-art algorithms, SCFN can achieve a gain of 3.7%-317.2%, 7.2%-169.4%, 1.9%-218.4%, and 20.9%-137.0% in terms of micro mAP, macro mAP, micro NDCG, and macro NDCG, respectively. This verified the proposed SCFN.

Fig.2 presents some samples of the retrieval results. The overall search results are satisfactory. Based on the query model, our method can obtain correct results in most categories, such as "car", "chair" and "guitar". Only a few categories contain errors, such as "cup" and "nightstand". We can find the categories with poor retrieval performance have great similarity in appearance. It is also very difficult to distinguish manually.

B. ABLATION STUDY

We conduct an ablation study to show the effect of different parts in our model. Fig.3 presents the PR curves of four architectures: MVCNN, MVCNN + CAM, MVCNN + CFM and SCFN (MVCNN + CAM + CFM) on ModelNet40 dataset. The performance of different architectures on ModelNet10 and ModelNet40 dataset are reported in Table.3 and Table.4, respectively, which demonstrates that each part is crucial to the final performance. The four architectures are detailed as follows:

- MVCNN: It renders the 3D model from different perspectives to get multiple views, extract the view features by 2D CNN, and aggregate these view features to a unified 3D descriptor by the max-pooling operation.
- MVCNN + CAM: It renders views from multiple perspectives of the 3D model and extract the raw view feature by 2D CNN, then the channel attention mechanism (CAM) is employed to exploit the channel-wise statistics and update the raw view features by reweighting the channels. Finally, the learned view features are taken to the max-pooling process to get a 3D descriptor.
- MVCNN + CFM: On the basis of MVCNN, it replaces the max-pooling strategy by a context information

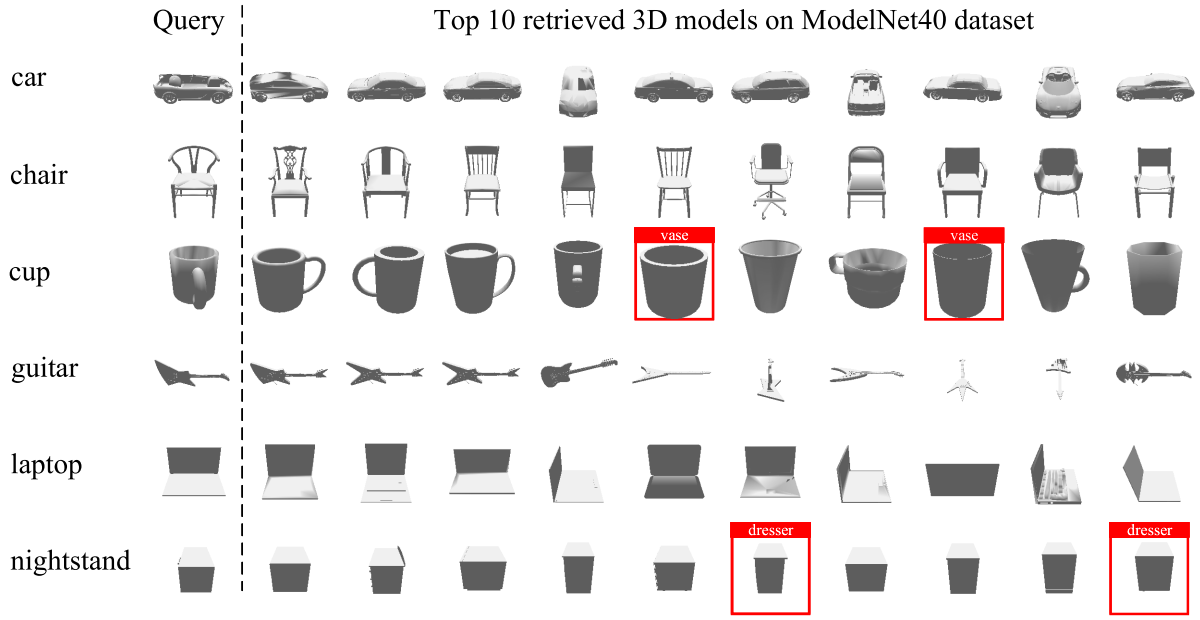


FIGURE 2. Retrieval examples on ModelNet40. The given query 3D models are located on the left side of the dotted line, and the right side of the dotted line is the top 10 3D models retrieved. The incorrect retrieval results are marked with red boxes.

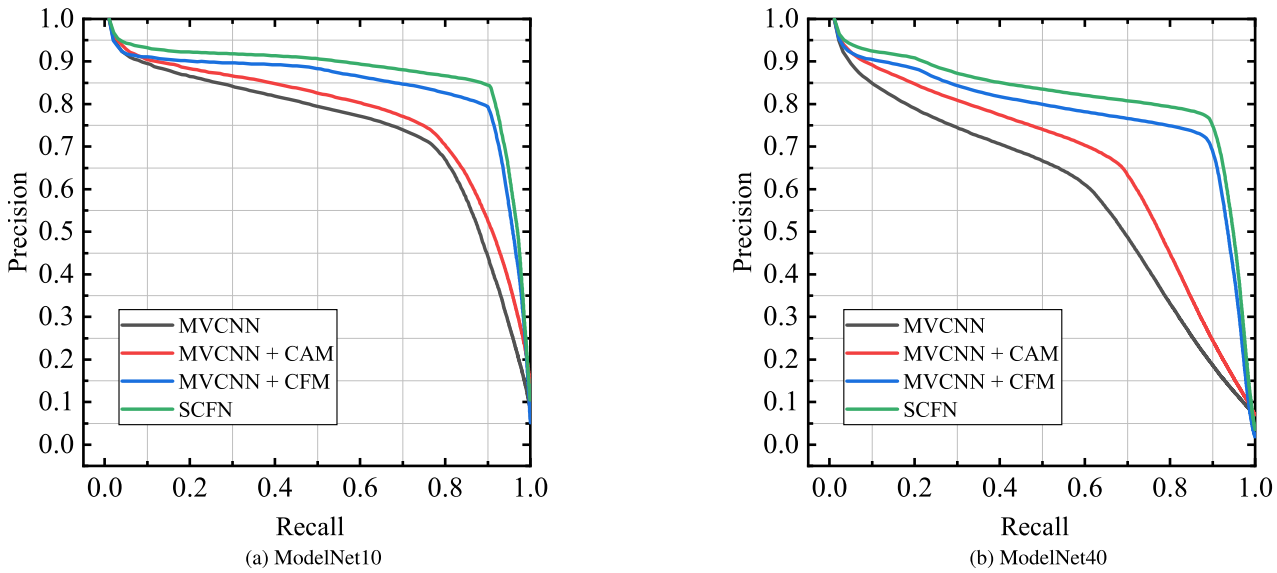


FIGURE 3. PR curves by models with different components on ModelNet10 and ModelNet40.

TABLE 3. Classification and Retrieval performance results by models with different components on ModelNet10.

Method	NN	FT	ST	F_measure	DCG	ANMRR	mAP	ACC
MVCNN	92.2%	74.4%	88.8%	31.4%	79.1%	22.3%	82.4%	92.9%
MVCNN + CAM	92.4%	80.1%	90.0%	31.1%	82.3%	20.1%	84.2%	93.3%
MVCNN + CFM	93.2%	86.2%	95.1%	32.3%	88.4%	11.6%	89.6%	93.6%
SCFN	**93.6%**	**86.9%**	**95.6 %**	**32.5%**	**89.1%**	**11.2%**	**91.9%**	**94.1%**

fusion module (CFM) to acquire a compact 3D representation.

- SCFN (MVCNN + CAM + CFM): It first places the virtual cameras around the 3D model to capture multiple views from different directions. Then 2D CNN is applied to extract the raw view feature, and a channel attention mechanism (CAM) is designed to update raw view feature by channel-wise weight. Next, these updated view features are concatenated and processed by context information fusion module (CFM) to explore spatial information.

It can be observed that the method with channel attention mechanism (MVCNN + CAM) outperforms the basic method (MVCNN) since it can learn effective deep visual

TABLE 4. Classification and Retrieval performance results by models with different components on ModelNet40.

Method	NN	FT	ST	F_measure	DCG	ANMRR	mAP	ACC
MVCNN	89.2%	55.9%	69.4%	28.4%	62.8%	39.7%	70.2%	90.7%
MVCNN + CAM	90.9%	63.8%	75.4%	30.1%	69.3%	32.9%	75.3%	92.4%
MVCNN + CFM	92.2%	83.5%	91.7%	33.8%	86.1%	13.3%	83.7%	92.7%
SCFN	**92.8%**	**84.6%**	**92.2 %**	**34.8%**	**87.4%**	**12.8%**	**85.1%**	**93.1%**

TABLE 5. Classification and Retrieval performance vs. the view order on ModelNet10 and ModelNet40.

Criteria	ModelNet10		ModelNet40	
	Disordered View	Ordered View	Disordered View	Ordered View
NN	93.2±0.4%	93.6%	92.4±0.4%	92.8%
FT	86.4±0.5%	86.9%	84.2±0.4%	84.6%
ST	95.6±0.3%	95.8%	92.1±0.1%	92.2%
F_measure	32.1±0.6%	32.5%	34.3±0.5%	34.8%
DCG	88.0±1.0%	89.1%	87.1±0.3%	87.4%
ANMRR	12.0±0.6%	11.2%	12.7±0.1%	12.8%
mAP	91.4±0.5%	91.9%	84.7±0.4%	85.1%
ACC	93.9±0.2%	94.1%	93.0±0.1%	93.1%

TABLE 6. Classification and retrieval performance vs. the view numbers on ModelNet10.

Views	NN	FT	ST	F_measure	DCG	ANMRR	mAP	ACC
2	85.1%	76.9%	91.5%	30.3%	79.8%	20.6%	81.6%	88.8%
4	88.8%	80.9%	93.4%	31.4%	83.8%	16.7%	83.0%	90.9%
6	89.9%	77.8%	90.4%	31.1%	81.3%	19.9%	85.7%	91.3%
8	91.1%	83.2%	93.7%	32.7%	85.5%	14.8%	87.2%	92.3%
10	92.7%	86.3%	95.7%	32.4%	88.5%	11.9%	90.2%	93.2%
12	**93.6%**	**86.9%**	**95.6 %**	**32.5%**	**89.1%**	**11.2%**	**91.9%**	**94.1%**

TABLE 7. Classification and retrieval performance vs. the view numbers on ModelNet40.

Views	NN	FT	ST	F_measure	DCG	ANMRR	mAP	ACC
2	85.4%	63.5%	73.8%	30.2%	69.2%	32.2%	77.5%	89.5%
4	88.7%	68.9%	78.3%	31.4%	74.3%	27.2%	79.7%	90.6%
6	89.9%	73.2%	82.9%	32.1%	77.9%	23.3%	80.5%	91.7%
8	90.3%	81.4%	86.2%	33.2%	82.9%	16.7%	83.9%	92.2%
10	92.5%	83.2%	91.5%	34.3%	85.7%	14.1%	84.3%	92.7%
12	**92.8%**	**84.6%**	**92.2 %**	**34.8%**	**87.4%**	**12.8%**	**85.1%**	**93.1%**

features by re-weighting the feature channels, which benefit our model greatly. It can be seen that the use of context information fusion module (MVCNN + CFM) is also better than MVCNN. Given some variations to the input, the 3D CNN module in CFM can capture the variance. Instead, unless the input variations change the maximum value, the single max-pooling operation will produce the same output. The ability to capture subtle change helps to enhance the discrimination ability of our model. As we expected, the combination of CAM and CFM can further significantly improve the performance consistently.

C. SENSITIVITY ANALYSIS ON VIEW ORDER

To test whether our model is affected by the input order of views, we conduct 50 experiments on ModelNet10 and ModelNet40. Each experiment randomly disturbs the input order of views and employed a series of indicators to measure classification and retrieval performance. The result is reported in Table.5. Counting the results of 50 experiments, we observe that, compared to sequentially ordered input (ordered view), the out-of-order setting (disordered view) does not have much effect on the experimental results. Our method introduces 3D CNN in the fusion part, which can enhance the sensitivity of our network to input changes and capture the spatial and context information of the input data. Due to the characteristics of 3D CNN, our method is affected by the order of view input to some degree, but combined with the global average pooling in the fusion part, this impact is limited.

D. SENSITIVITY ANALYSIS ON THE NUMBER OF VIEWS

Because the number of views rendered from the 3D model may affect the performance of 3D model classification and retrieval, we conduct several comparative experiments to explore the impact of view numbers on classification and retrieval performance. As shown in Section III-B, we set the interval angle θ of virtual cameras to 30°, 36°, 45°, 60°, 90°, and 180°, to generate 12, 10, 8, 6, 4, and 2 views for each 3D model, respectively. The experimental results are shown in Table.6, Table.7, and Fig.4. From the experimental

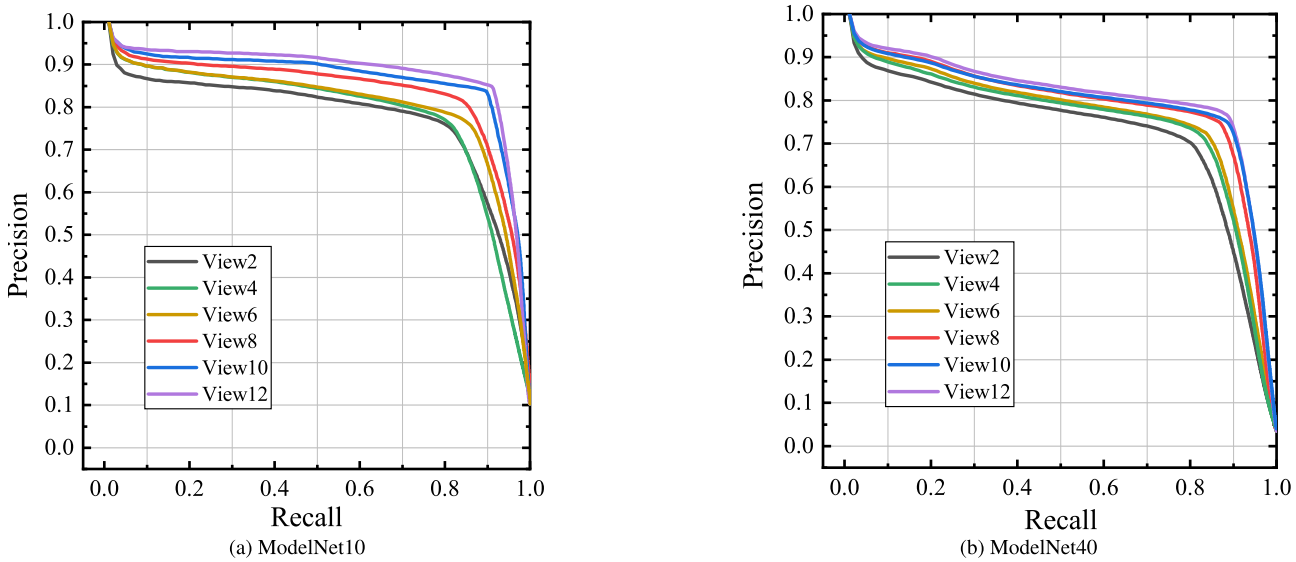


FIGURE 4. PR curves with different number of the views on ModelNet10 and ModelNet40.

results, we can get the following conclusion: with the view numbers increasing, the amount of information carried by the multi-view increases and the performance of classification and retrieval improves. When the number of views approaches 12, the PR curves are gradually approach each other, and the extent of indicators improvemetn becomes smaller. When the number of views reaches 12, SCFN has the best performance.

VI. CONCLUSION

In this paper, we propose a novel network (SCFN) to learn a discriminative 3D descriptors for 3D model classification and retrieval tasks. We design a channel attention mechanism (CAM) to enhance the useful semantic information in view features and suppress the useless information to acquire a more effective visual feature. For multi-view feature fusion, we propose the context information fusion module (CFM) to replace the traditional pooling strategy. Compared with other methods, the CFM can exploit multi-view context information better. We compare SCFN with some state-of-the-art methods on three challenging datasets: ModelNet10, ModelNet40, and ShapeNetCore55. The experimental results verify the superiority and effectiveness of SCFN in the 3D model classification and retrieval tasks. In the related work, we find that the existing view-based methods have done a lot of work on the correlation between views and the fusion of multi-view features, and achieved good results. However they rarely consider the semantic information contained in the view itself. SCFN has done some work in this area, but there is still some room for improvement. In the follow-up work, we will continue to pay attention to this aspect. Besides, SCFN we proposed is designed to model the multi-view information of the 3D model, but it does not combine the other modality information, such as point cloud, voxels, grids information. In the future, we will also try to combine the view information of the 3D model and other modal information to generate a more discriminative 3D descriptor.

REFERENCES

[1] Z. Cheng, X. Chang, L. Zhu, R. C. Kanjirathinkal, and M. Kankanhalli, "MMALFM: Explainable recommendation by leveraging reviews and images," *ACM Trans. Inf. Syst.*, vol. 37, no. 2, pp. 1–28, Mar. 2019.
[2] X. Liu, W. Liu, T. Mei, and H. Ma, "PROVID: Progressive and multimodal vehicle reidentification for large-scale urban surveillance," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 645–658, Mar. 2018.
[3] A. Godil, "Applications of 3D shape analysis and retrieval," in *Proc. IEEE Appl. Imag. Pattern Recognit. Workshop (AIPR)*, Oct. 2009, pp. 1–4.
[4] A.-A. Liu, S. Xiang, W.-Z. Nie, and D. Song, "End-to-End visual domain adaptation network for cross-domain 3D CPS data retrieval," *IEEE Access*, vol. 7, pp. 118630–118638, 2019.
[5] R. Klokov and V. Lempitsky, "Escape from cells: Deep kd-networks for the recognition of 3D point cloud models," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 863–872.
[6] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "Point CNN: Convolution on x-transformed points," in *Proc. NeurIPS*, 2018, pp. 828–838.
[7] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.
[8] C. R. Qi, H. Su, M. NieBner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view CNNs for object classification on 3D data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5648–5656.
[9] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85.
[10] 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. NeurIPS*, 2017, pp. 5099–5108.
[11] B. Wu, Y. Liu, B. Lang, and L. Huang, "DGCNN: Disordered graph convolutional neural network based on the Gaussian mixture model," *Neurocomputing*, vol. 321, pp. 346–356, Dec. 2018.
[12] S. Bai, X. Bai, Z. Zhou, Z. Zhang, and L. J. Latecki, "GIFT: A real-time and scalable 3D shape search engine," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5023–5032.
[13] D. Chen, X. Tian, Y. Shen, and M. Ouhyoung, "On visual similarity based 3d model retrieval.," *Comput. Graph. Forum*, vol. 22, no. 3, pp. 223–232, 2003.
[14] M. M. Kazhdan, T. A. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3d shape descriptors," in *Proc. SGP*, 2003, pp. 156–164.
[15] K. Sfikas, I. Pratikakis, and T. Theoharis, "Ensemble of PANORAMA-based convolutional neural networks for 3D model classification and retrieval," *Comput. Graph.*, vol. 71, pp. 208–218, Apr. 2018.
[16] B. Shi, S. Bai, Z. Zhou, and X. Bai, "DeepPano: Deep panoramic representation for 3-D shape recognition," *IEEE Signal Process. Lett.*, vol. 22, no. 12, pp. 2339–2343, Dec. 2015.

[17] H. Su, S. Maji, E. Kalogerakis, and E. G. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 945–953.

[18] P. Zanuttigh and L. Minto, "Deep learning for 3D shape classification from multiple depth maps," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3615–3619.

[19] H. Zeng, Q. Wang, and J. Liu, "Multi-feature fusion based on multi-view feature and 3D shape feature for non-rigid 3D model retrieval," *IEEE Access*, vol. 7, pp. 41584–41595, 2019.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[21] A. Verma, H. Qassim, and D. Feinzimer, "Residual squeeze CNDS deep learning CNN model for very large scale places image recognition," in *Proc. IEEE 8th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, Oct. 2017, pp. 463–469.

[22] Y. Feng, Z. Zhang, X. Zhao, R. Ji, and Y. Gao, "GVCNN: Group-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1–8.

[23] A.-A. Liu, N. Hu, D. Song, F.-B. Guo, H.-Y. Zhou, and T. Hao, "Multi-view hierarchical fusion network for 3D object retrieval and classification," *IEEE Access*, vol. 7, pp. 153021–153030, 2019.

[24] X.-S. Wei, J.-H. Luo, J. Wu, and Z.-H. Zhou, "Selective convolutional descriptor aggregation for fine-grained image retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2868–2881, Jun. 2017.

[25] C. Ma, Y. Guo, J. Yang, and W. An, "Learning multi-view representation with LSTM for 3-D shape recognition and retrieval," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1169–1182, May 2019.

[26] A. Liu, H. Zhou, M. J. Li, and W. Nie, "3D model retrieval based on multi-view attentional convolutional neural network," *Multim. Tools Appl.*, vol. 79, nos. 7–8, pp. 4699–4711, 2020.

[27] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1912–1920.

[28] N. Alpaslan and K. Hanbay, "Multi-scale shape index-based local binary patterns for texture classification," *IEEE Signal Process. Lett.*, vol. 27, pp. 660–664, Jun. 2020.

[29] N. Alpaslan and K. Hanbay, "Multi-resolution intrinsic texture geometry-based local binary pattern for texture classification," *IEEE Access*, vol. 8, pp. 54415–54430, 2020.

[30] A. M. Hasan, H. A. Jalab, F. Meziane, H. Kahtan, and A. S. Al-Ahmad, "Combining deep and handcrafted image features for MRI brain scan classification," *IEEE Access*, vol. 7, pp. 79959–79967, 2019.

[31] J. He, C. Zhang, X. He, and R. Dong, "Visual recognition of traffic police gestures with convolutional pose machine and handcrafted features," *Neurocomputing*, vol. 390, pp. 248–259, May 2020.

[32] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 3212–3217.

[33] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, and J. Wan, "3D object recognition in cluttered scenes with local surface features: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2270–2287, Nov. 2014.

[34] A. S. Mian, M. Bennamoun, and R. Owens, "Three-dimensional model-based object recognition and segmentation in cluttered scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1584–1601, Oct. 2006.

[35] Y. Fang, M. Sun, and K. Ramani, "Temperature distribution descriptor for robust 3D shape retrieval," in *Proc. CVPR WORKSHOPS*, Jun. 2011, pp. 9–16.

[36] A. Zeng, S. Song, M. Niessner, M. Fisher, J. Xiao, and T. Funkhouser, "3DMatch: Learning local geometric descriptors from RGB-D reconstructions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 199–208.

[37] G. Riegler, A. O. Ulusoy, and A. Geiger, "OctNet: Learning deep 3D representations at high resolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6620–6629.

[38] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Generative and discriminative voxel modeling with convolutional neural networks," in *Proc. CoRR*, 2016, pp. 1–9.

[39] S. Zhi, Y. Liu, X. Li, and Y. Guo, "Toward real-time 3D object recognition: A lightweight volumetric CNN framework using multitask learning," *Comput. Graph.*, vol. 71, pp. 199–207, Apr. 2018.

[40] A.-A. Liu, W.-Z. Nie, and Y.-T. Su, "3D object retrieval based on multi-view latent variable model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 3, pp. 868–880, Mar. 2019.

[41] M. Yavartanoo, E. Kim, and K. M. Lee, "Spnet: Deep 3D object classification and retrieval using stereographic projection," in *Proc. ACCV*, 2018, pp. 691–706.

[42] J. Jiang, D. Bao, Z. Chen, X. Zhao, and Y. Gao, "MLVCNN: Multi-loop-view convolutional neural network for 3d shape retrieval," in *Proc. AAAI*, 2019, pp. 8513–8520.

[43] A. Kanezaki, Y. Matsushita, and Y. Nishida, "RotationNet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5010–5019.

[44] V. Hegde and R. Zadeh, "Fusionnet: 3D object classification using multiple data representations," in *Proc. CoRR*, 2016, pp. 1–8.

[45] H. You, Y. Feng, R. Ji, and Y. Gao, "PVNet: A joint convolutional network of point cloud and multi-view for 3D shape recognition," in *Proc. ACM Multimedia Conf. Multimedia Conf.*, 2018, pp. 1–4.

[46] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.

[47] Z. Jiang, V. Rozgic, and S. Adali, "Learning spatiotemporal features for infrared action recognition with 3D convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 309–317.

[48] H. Xie, Z. Mao, Y. Zhang, H. Deng, C. Yan, and Z. Chen, "Double-bit quantization and index hashing for nearest neighbor search," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1248–1260, May 2019.

[49] L. Wang, W. Li, W. Li, and L. Van Gool, "Appearance-and-Relation networks for video classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1430–1439.

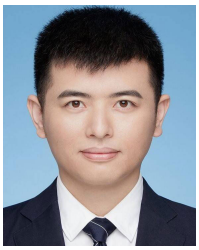
[50] A.-A. Liu, W.-Z. Nie, Y. Gao, and Y.-T. Su, "View-based 3-D model retrieval: A benchmark," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 916–928, Mar. 2018.

[51] E. Johns, S. Leutenegger, and A. J. Davison, "Pairwise decomposition of image sequences for active multi-view recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3813–3822.

AN-AN LIU (Member, IEEE) received the B.Eng. and Ph.D. degrees from Tianjin University, Tianjin, China. He is currently a Professor with the School of Electronic Information Engineering, Tianjin University. He was a Visiting Scholar with the Robotics Institute, Carnegie Mellon University, where he worked with Prof. T. Kanade. His research interests include computer vision and machine learning.

FU-BIN GUO is currently pursuing the master's degree with Tianjin University, Tianjin, China. His research interests include 3D object retrieval and classification.

HE-YU ZHOU is currently pursuing the Ph.D. degree with Tianjin University, Tianjin, China. His research interests include 3D object analysis, few-shot learning, and transfer learning.



WEN-HUI LI received the M.S. and Ph.D. degrees from the School of Electrical and Information Engineering, Tianjin University. He was an Intern Student with the SeSaMe Center, National University of Singapore. His research interests are in the field of computer vision, machine learning, and 3D model retrieval.

DAN SONG received the Ph.D. degree in computer science and technology from the Zhejiang University of China. Her research interests include computer graphics, computer vision, 3D human body reconstruction, and virtual fitting.