3D Shape Classification Using a Single View

BO DING, LEI TANG, ZHENG GAO, AND YONGJUN HE
School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China
Corresponding author: Yongjun He (holy_wit@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61673142, and in part by the Natural Science Foundation of Heilongjiang Province of China under Grant F2017013.

ABSTRACT View-based 3D shape classification is widely used in machine vision, information retrieval and other fields. However, there are two problems in current methods. First, current 3D shape classifiers fail to make good use of pose information of 3D shapes. Secondly, many views are required to obtain good classification accuracy, which leads to low efficiency. In order to solve these problems, we propose a novel 3D shape classification method based on Convolutional Neural Network (CNN). In the training stage, this method first learns a CNN to extract features, and then uses features of views from different viewpoint groups to train six 3D shape classifiers which fully mine the pose information of 3D shapes. Meanwhile, an additional class is adopted to improve the discrimination of 3D shape classifiers. In the recognition stage, the weighted fusion of image clarity evaluation functions is used to select the most representative view for the 3D shape recognition. Experiments on the ModelNet10 and ModelNet40 show that the classification accuracy of the proposed method can reach up to 91.18% and 89.01% when only using a single view and the efficiency is improved substantially.

INDEX TERMS Shape classification, convolutional neural network, 3D shape classifier, representative view.

I. INTRODUCTION
3D shape classification is a fundamental issue in the field of computer graphics and computer vision [1]. The primary problem of 3D shape classification is to effectively represent 3D shapes as feature descriptors. However, it is still a challenge. Descriptors should be representative and discriminative, i.e., can describe the most significant characteristics of a 3D shape. At the same time, the descriptors should not contain redundant information to ensure high efficiency in 3D shape classification. At present, the view-based representation is a mainstream 3D shape representation method, in which a 3D shape is projected from different locations to obtain a set of 2D views to represent global features of 3D shapes [2]. More importantly, these methods can be easily combined with deep learning to greatly improve classification accuracy.

Deep learning networks can automatically learn features and extract the inherent information of the images through operations such as multi-layer convolution and pooling [3]. Although view-based methods combined with deep learning can achieve good performance, they need to generate a large number of views. There are still two problems. First, the pose information of the 3D shape is not exploited enough. The poses and classification of 3D shapes are two tightly coupled problems. The pose information helps to determine the categories of the views. Therefore, the effective use of pose information can improve the accuracy of 3D shape classification and reduce the number of used views. Second, in the recognition stage, a large number of views are used to classify 3D shapes, which reduces the efficiency of shape classification.

In this article, we consider that the 2D views from different viewpoints are different. Six viewpoint groups are set around 3D shapes. And the views from each viewpoint group are used to train a 3D shape classifier (six classifiers in total). Each shape classifiers use the Convolutional Neural Network (CNN), which is divided into a feature extraction network and a classification network. The six 3D shape classifiers share the same feature extraction network, but have different classification networks. Meanwhile, an additional class is adopted to improve the discrimination of 3D shape classifiers. In the 3D shape recognition stage, each 3D shape classifier outputs a classification result. A classification strategy is proposed to integrate these six results to determine the final classification result. This method makes use of pose information effectively.
In order to improve the classification efficiency, we only select the most representative view for 3D shape classification. To this end, the three image clarity evaluation functions including the variance, Volath and information entropy are fused to evaluate the surface complexity of views. Since the pose information is used and the most representative view is selected, our method has achieved a high accuracy and efficiency by only using a single view.

II. RELATED WORK
The primary issue of 3D shape classification is to extract feature descriptors to effectively represent 3D shapes. The descriptors based on projective views are the most promising. This method transforms 3D shapes into a set of 2D views for shape classification and retrieval [4]. In these descriptors, the Light Field Descriptor (LFD) is the most popular one, because it is robust to transformations, noise and shape degeneracy [5]. In the LFD, a 3D shape is projected to generate 100 views. This descriptor represents 3D shapes better than other descriptors, but its time complexity is heavy because the view number used for classification is large. Recently, these methods which combine the multiple views and deep learning have achieved good performance. In these methods, the deep learning models are trained to extract features from 2D views. The 3D shape classification methods based on multiple views include Wang-MVCNN [6] and VS-MVCNN [7], which achieved accuracy of more than 90%. Wang et al. propose a view clustering and pooling layer based on dominant sets for 3D object recognition. This method uses a fast approximate learning strategy for cluster-pooling CNN, and greatly improves its training efficiency with only a slight accuracy reduction [8]. The CNN-VOTE method first classifies 2D views, and then classifies 3D shapes by voting on the recognition results of the 2D views [9]. Hegde et al. use FusionNet to combine the representation of 2D projective views and the representation of shape volume to learn new features, which yields a significantly better classifier than using either of the representations in isolation [10]. Qi et al. make a comprehensive study on the voxel-based CNNs and multi-view CNNs for 3D object classification [11]. Han et al. propose a 3D-to-Sequential Views (3D2SeqViews) method to effectively aggregate the sequential views using CNNs with a novel hierarchical attention aggregation [12].

3D shape classification and pose estimation are two tightly coupled problems. If the viewpoint of a view is known, the category of a 3D shape can be identified [13]. If the 3D shape category is known, it helps to infer the viewpoint of the view. Elhoseiny et al. explore the CNN architectures for combining object classification and pose estimation learned with multiple views [14]. Novotny et al. propose a Siamese viewpoint factorization network that robustly aligns different videos together without explicitly comparing 3D shapes, and a 3D shape completion network that can extract the full shape of an object from partial observations [15]. Kanzaki et al. improve this method by aggregating predictions from multiple images captured from different viewpoints [16].

Some researchers use panoramic views to represent 3D shapes for classification. In the DeepPano, each 3D shape is converted into a panoramic view, namely a cylinder projection around its principle axis. Then, a variant of CNN is specifically designed to learn the deep representations directly from views [17]. Sinha et al. create geometry images using authalic parametrization on a spherical domain. This spherically parameterized shape is then projected and cut to convert the original 3D shape into a flat and regular geometry image [18]. A similar method is the PANORAMA-NN [19], which also uses a panoramic view. Although the panoramic view is one view, its size is equivalent to that of multiple views, so the computational complexity is still high.

III. THE PROPOSED METHODOLOGY
A. THE OVERALL SCHEME
As shown in Figure 1, the whole process of the proposed 3D shape classification method can be divided into three steps: (1) 3D shape representation based on multiple views; (2) the most representative view selection; (3) 3D shape classification. In the multi-view representation of 3D shapes, we first determine the position and number of projection viewpoints, and then set up multiple viewpoint groups based on the positions. Each viewpoint group includes one main viewpoint and multiple subsidiary viewpoints. Each 3D shape generates multiple 2D projection views from each viewpoint group. In the most representative views selection, a linear regression model is built. The output of the regression model is used as the basis for the representative view selection. The image clarity evaluation functions, including the variance, Volath and information entropy, are used as the regression features. In 3D shape classification, the most representative view is selected for 3D shape classification. The most representative view is put into the six 3D shape classifiers, and each classifier outputs a result. A classification strategy is proposed to integrate these six results to get the final classification result.

B. 3D SHAPE REPRESENTATION BASED ON MULTIPLE VIEWS
The main two factors determining quality of projection views are the viewpoint arrangement and rendering method. The views under different viewpoints are different. A view rendered by different methods contains different amounts of information. The number of 2D views is the same as that of viewpoints. The steps of 3D shape representation based on multiple views are described as follows:

(1) 3D shape preprocessing. The 3D shape is zoomed and panned into a unit cube for normalization.

(2) Viewpoint group setting. We set six viewpoint groups around a 3D shape. Each viewpoint group contains one main viewpoint and eight auxiliary viewpoints. The six main viewpoints are located at the centers of the top, bottom, left, right, front and back of a unit cube. In order to increase the data
diversity and enhance the robustness of the CNN model, eight auxiliary viewpoints are set at the top, bottom, left, right, top left, top right, bottom left, and bottom right of each main viewpoint. Therefore, there are nine viewpoints for each viewpoint group, and a total of 54 viewpoints for a 3D shape. The viewpoint group setting is shown in Figure 2.

(3) 3D shape rendering. In order to increase the information quantity contained in projective views and reduce the negative impact from the shadow of the shape, we adopt the Phong Lighting Model [20] to render the shape. Firstly, an ambient light of low intensity is used, and then six fixed weak light sources at the points \((0, 0, 1), (0, 0, -1), (0, 1, 0), (0, -1, 0), (1, 0, 0)\) and \((-1, 0, 0)\) are deployed. At last, a brighter point source is set at the position of each camera, which is turned on when views are acquired. The six weak light sources and their locations are shown in Figure 3.

The 3D shape can be represented effectively through the proposed method. The reasons are: (1) this method can represent a 3D shape from all positions and angles; (2) the 2D views generated under different viewpoint groups are very different, and it is easy to use the pose information to classify the 3D shape.

C. SHAPE CLASSIFIER

It is essential to use the 2D views reasonably and effectively for feature extraction. Current feature extraction methods do not consider the poses of 3D shapes, which reduce classification performance. The pose and category of the 3D shape are tightly coupled. When the pose of the 3D shape is determined, that is, the viewpoint of the view is known, it is easy to infer the category of the shape. In other words, the 3D shape combining pose information can improve classification accuracy.

1) DEEP LEARNING NETWORK

CNNs are widely used for image classification. At present, there are a lot of CNNs, such as the VGG, GoogleNet, ResNet and DenseNet, etc. It is reported that the ResNet can achieve good performance on ImageNet. The ResNet adopts a unique “shortcut connection” which can effectively avoid gradient disappearance and ensure the training accuracy [21]. In our experiments, the ResNet50 achieved better performance than other deep neural networks, so it was used for feature extraction and classification. This network consists of 49 convolutional layers and one fully connected layer. The structure of the ResNet50 is shown in Table 1.

2) SHAPE CLASSIFIER COFIGURE

In proposed method, there are six 3D shape classifiers for six viewpoint groups. Each 3D shape classifier uses the views from the corresponding viewpoint group for training.
When the views are classified by traditional methods, all data in the same category can achieve good performance. In machine learning, the classifiers trained using similar layers of the classification network.

3) ADDITIONAL CLASS

The structure of resnet50.

| Layer name | Output size | 50-layer |
|------------|-------------|----------|
| Conv1      | 112×112     | 7×7, 64, stride 2 |
|            | 56×56       | 3×3 max pool, stride 2 |
|            |             | [1×1,64] |
| Conv2_x    | 56×56       | 3×3, 64, ×3 |
|            |             | [1×1,128] |
| Conv3_x    | 28×28       | 3×3, 128, ×4 |
|            |             | [1×1,512] |
| Conv4_x    | 14×14       | 3×3, 256, ×6 |
|            |             | [1×1,1024] |
| Conv5_x    | 7×7         | 3×3, 512, ×3 |
|            |             | [1×1,2048] |
|            | 1×1         | Average pool, 1000-d fc, softmax |

FLOPs: 3.8×10^9

The six 3D shape classifiers share the convolutional layers and the pooling layers, but have their own fully connected layers and Softmax layer. The convolution layers and pooling layers are regarded as the feature extraction layer, and the fully connected layers and the Softmax layer are regarded as 3D shape classifiers under different viewpoint groups. The reason is that the 2D views are different under different viewpoint groups. For example, the 2D view obtained from the upper viewpoint group is completely different from the front viewpoint group. It is difficult to train a good 3D shape classifier by putting such different views. Therefore, the views are set according to the six orientations of the 3D shapes, and each 3D shape classifier is according to a viewpoint group.

We first train a baseline system using all 2D views from all viewpoints, and then use the parameters of the baseline system to initialize the parameters of six 3D shape classifiers. Therefore, the training of the 3D shape classifiers is divided into two steps: (1) baseline system training. We take all 2D views as input and use the weight parameters obtained by training ImageNet for initialization. (2) 3D shape classifier training. Each 3D shape classifier corresponds to a viewpoint group. Therefore, the input for training each 3D shape classifier is the views from the corresponding viewpoint group, and finally the pose-based 3D shape classification under different viewpoints is realized. In classifier training, we fix the parameters of the convolutional and pooling layer of the feature extraction network, and only train the fully connected layer of the classification network.

3) ADDITIONAL CLASS

In machine learning, the classifiers trained using similar data in the same category can achieve good performance. When the views are classified by traditional methods, all views of a 3D shape are considered to belong to the same category. In fact, the views of a 3D shape from different viewpoints are different. If we put these views into one classifier, it is difficult to learn a good classifier. In this article, we think that the views from the same viewpoint group belong to a category, the views from other viewpoint groups do not belong to this category, and they also do not belong to other categories on this classifier. Therefore, we construct a new class for these data, namely the additional class. This can effectively improve the discrimination of classifiers. Taking ModelNet10 as an example, in addition to the existing 10 classes, each 3D shape classifier adds an additional class, namely class 11. When a view is input into the non-corresponding 3D shape classifier, the view is classified into the 11th class.

We set six viewpoint groups with each one having nine viewpoints, and each viewpoint corresponds to a projection view. For a 3D shape classifier, the views in the first 10 classes consist of all views of each shape under the corresponding viewpoint groups. The dataset of the additional class needs to cover the 2D views of all 3D shapes under the non-corresponding viewpoint groups, so the views of the additional class are taken from the views of all 3D shapes in the remaining five viewpoint groups. Since all the views from the remaining five viewpoint groups are selected, the amount of data is large, a sampling selection is adopted. At the same time, because most 3D shapes in the ModelNet10 are symmetrical, we only select the asymmetric views from the remaining four viewpoint groups for the additional class.

The structure of resnet50.

D. REPRESENTATION VIEW SELECTION BASED ON SURFACE COMPLEXITY

It is important to enhance the representation of the views within the same class and the discrimination of the views between classes for 3D shapes. The former affects accuracy, and the latter affects efficiency. At present, view-based 3D shape representation methods mostly use uniform projection, which do not consider the surface complexity difference of 2D views from different viewpoints. And these methods use all projection views to classify 3D shapes. Although it can obtain high accuracy, it seriously reduces the efficiency. A view with higher surface complexity means that it contains more information. If we can find one view with the highest complexity and use it for 3D shape classification, not only the accuracy can be guaranteed, but also the classification efficiency can be greatly improved. Therefore, we propose a representative view selection method based on surface complexity, which is aimed to selecting the most representative view.

1) SURFACE COMPLEXITY CALCULATION METHOD

2D views from different viewpoints describe different parts of 3D shapes. High-complexity views contain a large amount of detailed information of shapes, and are helpful to judge the shape category. We have tried Brenner gradient, Laplacian gradient, SMD (gray-scale variance), SMD2 (gray-
scale variance product), variance, energy gradient, Vollath, information entropy and other image clarity evaluation functions. Finally, we select the variance, Vollath and information entropy to calculate the surface complexity of views. The formula of the variance function is:

\[ D(f) = \sum_{x} \sum_{y} (f(x, y) - \mu)^2 \]  

(1)

where \( f(x, y) \) is the gray value of the corresponding pixel \((x, y)\), \( \mu \) is the average gray value. This function is sensitive to noise. The smaller the image, the smaller the function value it is. The formula of the Vollath function is:

\[ D(f) = \sum_{y} \sum_{x} f(x, y)(x + 1, y) - MN \mu^2 \]  

(2)

where \( M \) and \( N \) are the width and height of a view respectively. The formula of the information entropy function is:

\[ D(f) = - \sum_{i=0}^{L-1} p_i \log_2(p_i) \]  

(3)

where \( p_i \) is the probability of the pixel with the gray value \( i \) in the image, \( L \) is the total number of gray levels (usually 256).

The surface complexity of views is calculated by fusing the above three clarity evaluation functions, and then these views are sorted according to the complexity. Let \( V_i \) be the views, \( T_i \) the viewpoints, \( i = \{1, 2, 3, ...54\} \). Each view \( V_i \) corresponds to a viewpoint \( T_i \), \( C(V_i) \) is the surface complexity of the \( i \)-th view, \( X_i, Y_i, Z_i \) are respectively the variance, Vollath and information entropy. The surface complexity is calculated by

\[ C(V_i) = \theta_1 X_i + \theta_2 Y_i + \theta_3 Z_i \]  

(4)

where \( \theta \) is the weight learned with training data.

2) LINEAR REGRESSION MODEL LEARNING
The higher the view surface complexity is, the larger the amount of information of the view is, and the more representative and discriminative are. When a view with a high surface complexity is used to classify a 3D shape, the classification accuracy is high. Therefore, we replace complexity with classification accuracy to train the linear regression model.

In this article, the number of viewpoints is 54. The projection views of all 3D shapes under each viewpoint are divided into one group, the projection views can be divided into 54 groups. We use these 54 view groups for 3D shape classification to obtain classification accuracy of each 3D shape under each viewpoint.

3) SELECTION OF THE MOST REPRESENTATIVE VIEW
The surface complexity of 54 views for each 3D shape is calculated according to formula (4), and the view with the maximum value is the most representative view. The most representative view is determined by

\[ V_{\text{max}} = \arg \max \limits_{V_i} C(V_i) \]  

(5)

where \( V_{\text{max}} \) is the most representative view.

IV. 3D SHAPE CLASSIFICATION BASED ON A SINGLE VIEW
Since the viewpoint of the most representative view is unknown, it is input into the six shape classifiers. The proposed method is summarized in Algorithm 1.

Algorithm 1 gives the procedure of 3D shape classification, where \( \text{max}() \) represents computing the maximum value of the input and returning the row and column of the maximum value. Firstly, each 3D shape is projected to generate 54 views. Secondly, surface complexity of the views is calculated to select the most representative view, and then the features of the most representative view are extracted and put into the 3D shape classifiers. Finally, the classification result is obtained.

Since the viewpoint of the input view is unknown, the results on the six 3D shape classifiers are different. We need to integrate the results of these classifiers. The classification strategy is described as follows:

(1) If the view is classified into the additional classes, the probability values corresponding to all additional classes are set to 0. And then the maximum value of these probabilities is found. The 3D shape is classified into the class corresponding to the maximum value.

(2) If the view is classified into the first 10 classes for no more than three times, the probability values of the same class are averaged, and the class corresponding to the maximum value is the prediction result of the 3D shape.

(3) If the view is classified into the first 10 classes for more than three times, the probability values of additional classes in \( P \) are set as 0, and then find the maximum probability value, and the corresponding class is the class of the 3D shape.

V. EXPERIMENTS AND ANALYSIS
The experiments are conducted on a PC with an Intel i5 8400 CPU and GTX 2080TI GPU. The proposed method is implemented based on the MXNET framework. In this section, we compare the proposed method with the state-of-the-art methods on ModelNet10 and ModelNet40 [22] datasets. We follow the training and testing splitting of ModelNet10 and ModelNet40. ModelNet10 consists of 4899 shapes in 10 categories, and 3991 shapes are used as the training dataset and 908 shapes are used as the test dataset. ModelNet40 consists of 12311 shapes in 40 categories, and 9843 shapes are used as the training dataset and 2468 shapes are used as the test dataset.

A. VIEWS FOR TRAINING AND TESTING ON MODELNET10
All 3D shapes are projected to generate views for training and testing. The views for training and testing on ModelNet10 are shown in Table 2.

(1) Views for baseline system. In experiments, we set the ResNet50 as the baseline system. And it is also used as the feature extraction network of the proposed method. During training, the viewpoint difference of views is not considered. The last layer of the ResNet50 uses the Sigmoid function as the classifier. The output of the last second layer...
Algorithm 3D Shape Classification

Input: M represents a 3D shape; $f(x)$ represents the feature extraction function; $V_i$ represents the i-th view, $i \in \{1, 2, \cdots, 54\}$; $V_{\text{max}}$ represents the most representative view; $C_i$ represents the surface complexity of views; $F$ represents features of the most representative view; $P$ represents the probability matrix generated by the six classifiers, $P \in \mathbb{R}^{6 \times (N+1)}$ ($N$ is the number of categories); indicator is a vector indicating whether the 3D shape is classified into the additional class. If the classification result is the additional class, the corresponding element of indicator is 0, otherwise it is 1; $\text{count}$ represents the numbers of classifiers that the view is classified into the first 10 classes; $R$ represents the average classification results.

Output: final category $L_{\text{final}}$

(1) $M$ is projected to generated $V_i$, $i \in \{1, 2, \cdots, 54\}$;
(2) The surface complexity of views is calculated: $C_i \leftarrow C(V_i)$;
(3) The most representative view is selected by the surface complexity: $V_{\text{max}} \leftarrow \text{argmax}_i(C_i)$;
(4) The features of $V_{\text{max}}$ is extracted: $F \leftarrow f(V_{\text{max}})$;
(5) Input $F$ into six 3D shape classifiers to get the classification result matrix $P$;
(6) for $i = 1:6$
(7) $\text{indicator}[i] \leftarrow 1$;
(8) $[i, j] = \max P[i][1:11]$;
(9) if ($j == 11$)$\text{indicator}[i] \leftarrow 0$;
(10) endfor
(11) $\text{count} \leftarrow \text{sum}(\text{indicator})$;
(12) switch($\text{count}$){
(13) case 0;
(14) for $i = 1:6$
(15) $P[i][N + 1] \leftarrow 0$;
(16) endfor
(17) $[i, j] = \max P$;
(18) $L_{\text{final}} \leftarrow i$;
(19) break;
(20) case 1, 2, 3:
(21) $R \leftarrow 0$;
(22) for $i = 1:6$
(23) if($\text{indicator}[i] == 1$)
(24) $R \leftarrow R + P[i][1:11]$;
(25) endfor
(26) $L_{\text{final}} \leftarrow i$;
(27) break;
(28) case 4, 5, 6:
(29) for $i = 1:6$
(30) if($\text{indicator}[i] == 0$)
(31) $P[i][1:11] \leftarrow 0$;
(32) endfor
(33) $\text{count} \leftarrow \text{sum}(\text{indicator})$;
(34) $[i, j] = \max P$;
(35) $L_{\text{final}} \leftarrow i$;
(36) break;
}
TABLE 2. Modelnet10 dataset.

| Classifier    | Test set | Training set |
|---------------|----------|--------------|
|               | First 10 classes | Average view number | Class 11 | First 10 classes | Average view number | Class 11 |
| Shape classifiers | 8172 | 817 | 3632 | 35919 | 3592 | 15964 |
| Baseline system | 49032 | 4903 | - | 215514 | 21551 | - |

TABLE 3. View classification accuracy(%).

| Shape classifier | Viewpoint group 1 | Viewpoint group 2 | Viewpoint group 3 | Viewpoint group 4 | Viewpoint group 5 | Viewpoint group 6 |
|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| 10 classes       | Training         | 94.95            | 95.71            | 96.28            | 95.49            | 95.95            | 94.62            |
|                  | Test             | 85.88            | 87.21            | 89.72            | 86.73            | 89.53            | 83.92            |
| 11 classes       | Training         | 95.03            | 94.14            | 95.53            | 93.21            | 95.84            | 95.16            |
|                  | Test             | **89.61**        | **87.25**        | **89.74**        | **87.03**        | **88.91**        | **88.41**        |

TABLE 4. 3D Shape classification accuracy under different viewpoints (%).

| Location | Viewpoint group 1 | Viewpoint group 2 | Viewpoint group 3 | Viewpoint group 4 | Viewpoint group 5 | Viewpoint group 6 |
|----------|------------------|------------------|------------------|------------------|------------------|------------------|
| 1        | 76.76            | 81.50            | 86.23            | 81.50            | 83.81            | 70.82            |
| 2        | 83.81            | 86.89            | 88.55            | 86.89            | 88.33            | 83.15            |
| 3        | 84.14            | 86.78            | 89.43            | 85.24            | 88.55            | 83.26            |
| 4        | 87.33            | 87.00            | 88.55            | 85.13            | 88.00            | 85.35            |
| 5        | 86.12            | 84.69            | 89.98            | 84.58            | 89.54            | 86.78            |
| 6        | 90.09            | 89.98            | 91.41            | 90.31            | 91.63            | 88.99            |
| 7        | 90.31            | 90.09            | 90.75            | 89.65            | 90.75            | 87.11            |
| 8        | 87.22            | 90.31            | 92.07            | 89.32            | 90.31            | 88.66            |
| 9        | 88.55            | 87.44            | 91.63            | 90.53            | 91.52            | 88.99            |

viewpoint group 6 correspond to the top and bottom of 3D shapes with less discriminative view, so the classification accuracies are low.

The overall classification accuracy of the test set of the 3D shape classifiers with additional classes is improved. Especially for the 3D shape classifier corresponding to viewpoint group 3, the classification accuracy reaches 89.74%. The reason is that the additional class is added. It can make the view more likely to choose the correct 3D shape classifier. It is predictable that our method can have higher performance when the 3D shape surface is more complicated and the views are different under different viewpoints.

Among the six 3D shape classifiers, once the accuracy of view classification of one classifier is higher than the accuracy of the baseline system, the information of these classifiers can be combined to achieve a higher accuracy than the baseline system.

C. REPRESENTATIVE VIEWS SELECTION METHOD

The three clarity evaluation functions, namely the variance, Volath and information entropy, are used for selecting the most representative view. The linear regression model is a mapping from the three clarity evaluation functions to surface complexity of a view. The surface complexity of the most representative view is the highest. In training, the surface complexity is replaced by the view classification accuracy under different viewpoints. The 3D shape classification accuracy under different viewpoints is shown in Table 4.

In the proposed method, all views are divided into 54 groups according to viewpoints, so each group has 908 views. These 54 view groups used for 3D shape classification are equivalent to 54 single-view shape classification experiments. That is, views are selected based on different viewpoints, and then the selected view is used for 3D shape classification.

We can see that the lowest classification accuracy is 70.82%, the highest is 92.07%, and the average is 87.34%. The view classification accuracy under position 8 of viewpoint group 3 is the highest (92.07%), which means that the view is the most discriminative. The accuracy of view classification achieved by the proposed method can be close to the highest value in Table 3. The pose of the 3D shape cannot
| VOLUME 8, 2020 | 200819 |

## TABLE 5. Classification accuracy of different 3D shapes in different positions.

|   | bathtub | bed | chair | desk | dresser | monitor | night_stand | sofa | table | toilet |
|---|---------|-----|-------|------|---------|---------|-------------|------|-------|--------|
| V1-L1 | 0.74 | 0.88 | 0.93 | 0.48 | 0.74 | 0.93 | 0.51 | 0.92 | 0.15 | 0.99 |
| V1-L2 | 0.80 | 0.92 | 0.96 | 0.62 | 0.80 | 0.97 | 0.69 | 0.94 | 0.43 | 1.00 |
| V1-L3 | 0.84 | 0.97 | 0.92 | 0.65 | 0.81 | 1.00 | 0.70 | 0.97 | 0.42 | 0.98 |
| V1-L4 | 0.92 | 0.96 | 0.98 | 0.80 | 0.87 | 0.91 | 0.66 | 0.97 | 0.57 | 0.97 |
| V1-L5 | 0.88 | 0.99 | 0.98 | 0.69 | 0.81 | 0.96 | 0.76 | 0.96 | 0.52 | 1.00 |
| V1-L6 | 0.82 | 1.00 | 0.97 | 0.83 | 0.91 | 0.98 | 0.79 | 0.96 | 0.74 | 0.96 |
| V1-L7 | 0.72 | 0.97 | 0.99 | 0.88 | 0.90 | 0.96 | 0.80 | 0.97 | 0.71 | 0.97 |
| V1-L8 | 0.82 | 1.00 | 0.98 | 0.79 | 0.90 | 0.97 | 0.65 | 0.95 | 0.72 | 0.98 |
| V1-L9 | 0.78 | 1.00 | 0.97 | 0.73 | 0.91 | 0.93 | 0.70 | 0.97 | 0.74 | 0.97 |
| V2-L1 | 0.74 | 0.85 | 0.97 | 0.63 | 0.77 | 1.00 | 0.65 | 0.85 | 0.77 | 0.99 |
| V2-L2 | 0.66 | 0.93 | 1.00 | 0.78 | 0.83 | 0.99 | 0.76 | 0.93 | 0.82 | 0.98 |
| V2-L3 | 0.68 | 0.91 | 0.99 | 0.80 | 0.78 | 1.00 | 0.71 | 0.92 | 0.81 | 0.97 |
| V2-L4 | 0.80 | 0.91 | 0.99 | 0.77 | 0.72 | 0.99 | 0.76 | 0.89 | 0.74 | 0.99 |
| V2-L5 | 0.56 | 0.89 | 0.97 | 0.73 | 0.78 | 0.98 | 0.69 | 0.82 | 0.76 | 0.97 |
| V2-L6 | 0.78 | 0.97 | 1.00 | 0.78 | 0.80 | 0.98 | 0.79 | 0.94 | 0.81 | 0.99 |
| V2-L7 | 0.84 | 0.94 | 1.00 | 0.86 | 0.84 | 0.99 | 0.76 | 0.93 | 0.82 | 0.98 |
| V2-L8 | 0.62 | 0.92 | 1.00 | 0.85 | 0.84 | 0.98 | 0.81 | 0.85 | 0.85 | 0.97 |
| V2-L9 | 0.70 | 0.95 | 1.00 | 0.80 | 0.77 | 1.00 | 0.74 | 0.88 | 0.85 | 0.97 |
| V3-L1 | 0.68 | 0.94 | 0.99 | 0.62 | 0.69 | 0.98 | 0.63 | 0.89 | 0.92 | 0.97 |
| V3-L2 | 0.72 | 0.96 | 1.00 | 0.70 | 0.84 | 0.96 | 0.67 | 0.91 | 0.83 | 0.98 |
| V3-L3 | 0.74 | 0.97 | 0.99 | 0.69 | 0.88 | 0.96 | 0.73 | 0.92 | 0.87 | 0.98 |
| V3-L4 | 0.80 | 0.97 | 0.99 | 0.72 | 0.86 | 0.93 | 0.77 | 0.92 | 0.82 | 1.00 |
| V3-L5 | 0.76 | 0.96 | 1.00 | 0.80 | 0.84 | 0.97 | 0.78 | 0.87 | 0.86 | 0.98 |
| V3-L6 | 0.82 | 0.99 | 1.00 | 0.78 | 0.84 | 0.99 | 0.80 | 0.93 | 0.86 | 0.99 |
| V3-L7 | 0.86 | 0.97 | 1.00 | 0.78 | 0.88 | 0.98 | 0.80 | 0.94 | 0.85 | 0.99 |
| V3-L8 | 0.82 | 0.98 | 0.99 | 0.83 | 0.91 | 0.96 | 0.78 | 0.88 | 0.90 | 0.98 |
| V3-L9 | 0.80 | 0.96 | 1.00 | 0.83 | 0.92 | 0.98 | 0.78 | 0.90 | 0.88 | 0.98 |
| V4-L1 | 0.68 | 0.73 | 0.97 | 0.60 | 0.77 | 0.98 | 0.66 | 0.84 | 0.77 | 0.95 |
| V4-L2 | 0.74 | 0.86 | 0.99 | 0.77 | 0.76 | 1.00 | 0.70 | 0.91 | 0.79 | 0.96 |
| V4-L3 | 0.64 | 0.90 | 1.00 | 0.76 | 0.79 | 0.98 | 0.77 | 0.92 | 0.82 | 0.97 |
| V4-L4 | 0.80 | 0.77 | 0.97 | 0.80 | 0.72 | 1.00 | 0.73 | 0.85 | 0.75 | 0.96 |
| V4-L5 | 0.54 | 0.83 | 1.00 | 0.77 | 0.77 | 0.97 | 0.70 | 0.86 | 0.79 | 0.93 |
| V4-L6 | 0.84 | 0.88 | 1.00 | 0.87 | 0.77 | 0.98 | 0.76 | 0.91 | 0.82 | 0.96 |
| V4-L7 | 0.86 | 0.93 | 1.00 | 0.80 | 0.87 | 0.99 | 0.77 | 0.92 | 0.83 | 0.97 |
| V4-L8 | 0.68 | 0.91 | 1.00 | 0.85 | 0.79 | 0.99 | 0.78 | 0.90 | 0.84 | 0.95 |
| V4-L9 | 0.68 | 0.94 | 1.00 | 0.85 | 0.84 | 0.99 | 0.81 | 0.88 | 0.84 | 0.96 |
| V5-L1 | 0.70 | 0.98 | 0.99 | 0.67 | 0.65 | 0.98 | 0.58 | 0.93 | 0.92 | 0.98 |
| V5-L2 | 0.68 | 0.95 | 1.00 | 0.71 | 0.76 | 0.96 | 0.69 | 0.87 | 0.83 | 0.97 |
| V5-L3 | 0.76 | 0.97 | 0.99 | 0.74 | 0.76 | 0.98 | 0.72 | 0.90 | 0.83 | 0.97 |
| V5-L4 | 0.74 | 0.96 | 0.99 | 0.87 | 0.80 | 0.97 | 0.71 | 0.91 | 0.89 | 1.00 |
| V5-L5 | 0.78 | 0.96 | 0.99 | 0.85 | 0.84 | 0.94 | 0.65 | 0.91 | 0.85 | 0.99 |
| V5-L6 | 0.86 | 0.98 | 1.00 | 0.84 | 0.85 | 0.98 | 0.79 | 0.93 | 0.87 | 1.00 |
TABLE 5. (Continued.) Classification accuracy of different 3D shapes in different positions.

| Shape        | V5-L7 | V5-L8 | V5-L9 | V6-L1 | V6-L2 | V6-L3 | V6-L4 | V6-L5 | V6-L6 | V6-L7 | V6-L8 | V6-L9 |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
|              | 0.88  | 0.97  | 1.00  | 0.86  | 0.83  | 0.98  | 0.79  | 0.94  | 0.85  | 1.00  | 0.82  | 0.98  |
|              | 0.82  | 0.98  | 0.99  | 0.86  | 0.78  | 0.98  | 0.72  | 0.94  | 0.86  | 0.97  | 0.76  | 0.99  |
|              | 0.76  | 0.99  | 1.00  | 0.87  | 0.80  | 0.97  | 0.74  | 0.91  | 0.85  | 0.99  | 0.64  | 0.52  |
|              | 0.64  | 0.52  | 0.84  | 0.53  | 0.77  | 0.89  | 0.60  | 0.68  | 0.18  | 0.97  | 0.68  | 0.79  |
|              | 0.68  | 0.79  | 0.95  | 0.83  | 0.81  | 0.94  | 0.63  | 0.85  | 0.58  | 0.98  | 0.64  | 0.81  |
|              | 0.64  | 0.81  | 0.95  | 0.76  | 0.77  | 0.93  | 0.67  | 0.86  | 0.61  | 0.95  | 0.76  | 0.96  |
|              | 0.76  | 0.96  | 1.00  | 0.79  | 0.77  | 0.91  | 0.73  | 0.89  | 0.68  | 0.94  | 0.74  | 0.91  |
|              | 0.74  | 0.91  | 0.99  | 0.79  | 0.78  | 0.92  | 0.69  | 0.83  | 0.68  | 0.97  | 0.76  | 0.94  |
|              | 0.76  | 0.94  | 0.99  | 0.81  | 0.86  | 0.93  | 0.73  | 0.92  | 0.75  | 0.99  | 0.74  | 0.91  |
|              | 0.74  | 0.91  | 0.94  | 0.84  | 0.85  | 0.95  | 0.77  | 0.96  | 0.69  | 0.95  | 0.76  | 0.93  |
|              | 0.76  | 0.93  | 0.98  | 0.80  | 0.92  | 0.96  | 0.79  | 0.89  | 0.70  | 1.00  | 0.74  | 0.92  |

be determined in real applications; therefore, the viewpoint information is unknown.

We can get the classification accuracy of each category in different viewpoints, and use the accuracy as surface complexity for training of the linear regression model. As shown in Table 5, V1-L1 represents position 1 of viewpoint group 1.

The training of linear regression model requires labeled data. In this article, the clarity evaluation functions are used to calculate the features, and the average classification accuracy at each viewpoint is used as the regression value to train the regression model. In order to obtain a reasonable set of model parameters, we learn a linear regression model on each 3D shape category, compare their performance, and select a set of model parameters with the best performance. The classification accuracy of a single view is shown in Table 6.

TABLE 6. Classification accuracy of a single view(\%).

| Shape        | Classification accuracy |
|--------------|-------------------------|
|              | Classifiers with 11 classes | Classifiers with 10 classes | Baseline system |
| bathtub      | 90.85                   | 90.19                   | 89.75               |
| bed          | 91.07                   | 90.63                   | 90.08               |
| chair        | 90.96                   | 90.96                   | 90.52               |
| desk         | 90.08                   | 90.85                   | 90.41               |
| dresser      | 90.63                   | 90.63                   | 90.08               |
| monitor      | 90.96                   | 90.96                   | 90.41               |
| night_stand  | 91.18                   | 90.41                   | 89.97               |
| sofa         | 91.18                   | 90.52                   | 90.08               |
| table        | 84.80                   | 84.69                   | 84.80               |
| toilet       | 91.18                   | 90.52                   | 90.30               |

It is shown in Table 6 that, except for the table class, the results of the linear regression models of the other classes are similar; the accuracy rate is about 90%, and the highest is 91.18%, which is higher than average value, and close to highest value in Table 3. It also can be seen that the results of the 3D shape classifiers with the additional class is better than that without the additional class.

It can be seen from Table 7 that our method has achieved the best performance in ModelNet10, with a classification accuracy of 91.18%. And it is only 1.69% lower than that of PANORAMA-NN in ModelNet40. Compared with PANORAMA-NN, our method uses a single view, rather than the panoramic view obtained by splicing multiple views. The panoramic view is equivalent to 54 views in the proposed method. That is to say, our method is simpler in view acquisition, and achieves high classification accuracy.

TABLE 7. Classification accuracy based on a single view compared with other methods(\%).

| Methods          | View number | Classification accuracy | ModelNet10 | ModelNet40 |
|------------------|-------------|-------------------------|------------|------------|
| DeepPanel\cite{17} | 1           | 85.45                   | 77.63      |
| Geometry image\cite{18} | 1           | 88.40                   | 83.90      |
| RotationNet\cite{19} | 1           | 91.10                   | 84.90      |
| PANORAMA-NN\cite{19} | 1           | 91.12                   | 90.70      |
| ours             | 1           | \textbf{91.18}          | 89.01      |

We can see from Table 8 that the classification accuracy of our method with 36 views is the best on both ModelNet10 and ModelNet40, with classification accuracies of 95.70% and 93.47 respectively.

The confusion matrix of the proposed method based on a single view and multiple views in ModelNet10 is shown in Figure 4 and Figure 5. We can see in Figure 4 that most shapes can be correctly classified given only one view, especially the classes of bed, chair, monitor, sofa and toilet. The reason is that: (1) our method makes full use of the pose information of 3D shapes; (2) these classes are complex,
TABLE 8. Classification accuracy based on multiple view compared with other methods(%).

| Methods      | View number | Classification accuracy | ModelNet10 | ModelNet40 |
|--------------|-------------|-------------------------|------------|------------|
| CNN-VOTE[9]  | 12          | 93.18                   | 91.82      |            |
| RotationNet[16]| 12          | 93.84                   | 90.65      |            |
| 3D2SeqViews[12]| 12          | 94.68                   | 91.64      |            |
| FusionNet [10]| 60          | 93.11                   | 90.80      |            |
| VS-MVCNN[7]  | 80          | 93.50                   | 90.90      |            |
| ours         | 36          | 95.70                   | 93.47      |            |

FIGURE 4. Classification results based on a single view.

FIGURE 5. Classification results based on multiple views.

so the selected single view contains enough information for classification.

The misclassified 3D shapes are distributed in different classes. The reason is that the single view can only describe the 3D shape at one viewpoint. This view may be similar to the view of other shapes at a certain viewpoint. And if the 3D shapes are too similar, for example, table and desk, night_stand and dresser, it is easy to misclassify them. Compared with 3D shape classification based on a single view, the shapes misclassified are significantly reduced through multiple views in Figure 5. That is because multiple views provide more information about 3D shapes.

D. CLASSIFICATION EFFICIENCY ANALYSIS

One purpose of the proposed method is to improve the classification efficiency of 3D shapes. Compared with the panoramic view used in DeepPano and PANORAMA-NN, our method is simple in obtaining the view and need small number of views for 3D shape classification. Compared with Geometry image, our method does not require geometric transformation of views. Our method outperforms RotationNet which uses only one view to classify 3D shapes. Therefore, the proposed method can not only improve the classification accuracy, but also improve the classification efficiency.

We use the ModelNet10 as an example. It consists of 2468 shapes. The total classification time and the average classification time of the 3D shape using the different view number is shown in Table 9.

TABLE 9. 3D shape classification time.

| Shape number | View number | Total view number | Total classification time (s) | Average classification time (ms) |
|--------------|-------------|-------------------|-------------------------------|---------------------------------|
| 2468         | 1           | 2468              | 23                            | 9                               |
| 2468         | 2           | 4936              | 35                            | 14                              |
| 2468         | 3           | 7404              | 53                            | 21                              |
| 2468         | 4           | 9872              | 82                            | 33                              |
| 2468         | 5           | 12340             | 90                            | 36                              |

One can see that the total and average classification time increase significantly as the view number increasing. The reason is that the 3D shape classification is based on the view classification. The more views each shape selects, the more time it takes. Therefore, 3D shape classification using a single view can greatly improve efficiency.

VI. CONCLUSION

With the increasing of 3D shapes, it is an urgent need to improve the accuracy and efficiency of 3D shape classification. To solve these problems, we propose a 3D shape classification method which makes use of pose information of 3D shapes, and selects the most representative view for classification. In the proposed method, six 3D shape classifiers for different viewpoint groups are trained. These classifiers can make full use of the pose information of 3D shapes. Furthermore, additional classes are learned to improve the discrimination of classifiers. In order to improve efficiency, the representative view selection method is proposed based on surface complexity of 3D shapes. This method only selects one representative view with the highest surface complexity to realize 3D shape recognition. Experiments show that the proposed method outperforms the state-of-the-art methods in both accuracy and efficiency.

REFERENCES

[1] S. Chen, L. Zheng, Y. Zhang, Z. Sun, and K. Xu, “VERAM: View-enhanced recurrent attention model for 3D shape classification,” IEEE Trans. Vis. Comput. Graphics, vol. 25, no. 12, pp. 3244–3257, Dec. 2019.

[2] Z. Han, M. Shang, Z. Liu, C.-M. Vong, Y.-S. Liu, M. Zwicker, J. Han, and C. L. P. Chen, “SeqViews2SeqLabels: Learning 3D global features via aggregating sequential views by RNN with attention,” IEEE Trans. Image Process., vol. 28, no. 2, pp. 658–672, Feb. 2019.
A.-A. Liu, H.-Y. Zhou, M.-J. Li, and W.-Z.Nie, “3D model retrieval based on multi-view attentional convolutional neural network,” Multimedia Tools Appl., vol. 79, nos. 7–8, pp. 4699–4711, Feb. 2020.

Z. Wen and J. Jin-Yuan, “Web3D lightweight for sketch-based shape retrieval Using SVM learning algorithm,” Acta Electronica Sinica, vol. 47, no. 1, pp. 92–99, 2019.

D. Y. Chen, X. P. Tian, and Y. T. Shen, “On visual similarity based on 3D model retrieval,” Comput. Graph. Forum, vol. 22, no. 3, pp. 223–232, 2003.

C. Wang, M. Pelillo, and K. Siddiqi, “Dominant set clustering and pooling for multi-view 3D object recognition,” in Proc. Brit. Mach. Vis. Conf. Berlin, Germany: Springer, 2017, pp. 1–11.

Y. X. Ma, B. Zheng, and Y. L. Guo, “Boosting multi-view convolutional neural networks for 3D object recognition via view saliency,” in Proc. Chin. Conf. Image Graph. Technol. Berlin, Germany: Springer, 2017, pp. 199–209.

C. Wang, M. Pelillo, and K. Siddiqi, “Dominant set clustering and pooling for multi-view 3D object recognition,” 2019, arXiv:1906.01592. [Online]. Available: http://arxiv.org/abs/1906.01592.

B. Jing, S. Qinglong, and Q. Feiwei, “3D model classification and retrieval based on CNN and voting scheme,” J. Comput.-Aided Des. Comput. Graph., vol. 31, no. 2, pp. 303–314, 2019.

V. Hegde and R. Zadeh, “Fusionnet: 3D object classification using multiple data representations,” in Proc. 6th Int. Conf. Learn. Represent. (ICLR), 2018, pp. 1–10.

C. R. Qi, H. Su, and M. Niessner, “Volumetric and multi-view CNNs for object classification on 3D data,” in Proc. 29th IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 5648–5656.

Z. Han, H. Lu, Z. Liu, C.-M. Vong, Y.-S. Liu, M. Zwicker, J. Han, and C. L. P. Chen, “3D2SeqViews: Aggregating sequential views for 3D global feature learning by CNN with hierarchical attention aggregation,” IEEE Trans. Image Process., vol. 28, no. 8, pp. 3986–3999, Aug. 2019.

H. Su, C. R. Qi, Y. Li, and L. J. Guibas, “Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3D model views,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 2686–2694.

M. Elhoseiny, T. El-Gaaly, and A. Bakry, “A comparative analysis and study of multi-view CNN models for joint object categorization and pose estimation,” in Proc. Int. Conf. Mach. Learn. (ICML), vol. 2, 2016, pp. 1402–1422.

D. Novotny, D. Larlus, and A. Vedaldi, “Learning 3D object categories by looking around them,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 5228–5237.

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.

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.

A. Sinha, J. Bai, and K. Ramani, “Deep learning 3D shape surfaces using geometry images,” in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2016, pp. 223–240.

K. Sifakis, T. Theoharis, and I. Pratikakis, “Exploiting the PANORAMA representation for convolutional neural network classification and retrieval,” in Proc. 10th Eurograph. Workshop 3D Object Retr., 2017, pp. 1–7.

S.-M. Woo, S.-H. Lee, J.-S. Yoo, and J.-O. Kim, “Improving color constancy in an ambient light environment using the phong reflection model,” IEEE Trans. Image Process., vol. 27, no. 4, pp. 1862–1877, Apr. 2018.

K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, “The princeton shape benchmark,” in Proc. Shape Modeling Appl., 2004, pp. 167–178.