# 3D model retrieval based on deep learning approach with weighted three-view deep features

Xuemei Jiang* ,Yaqi Li*, Jiwei Hu*, Kin-Man Lam*†
*School of Information Engineering, Wuhan University of Technology
*Hubei Key Lab. of Broadband Wireless Communication and Sensor Network
†Department of Electronic and Information, Engineering, The Hong Kong Polytechnic University

## ABSTRACT

With the development of computer graphics and three-dimensional (3D) modeling technology, 3D model retrieval has been widely used in different applications, such as industrial design, virtual reality, medical diagnosis, etc. Massive data brings new opportunities and challenges to the development of the 3D model retrieval technology. However, with the emergence of complex models, traditional retrieval algorithms are not applicable to some extent. One important reason for this is that the traditional content-based retrieval methods do not take the spatial information of 3D models into account during feature extraction. Therefore, how to use the spatial information of a 3D model to obtain a more extensive feature has become a significant issue. In our proposed algorithm, we first normalize and voxelize the model, and then extract features from different views of the voxelized model. Secondly, deep features are extracted by using our proposed feature learning network. Then, a new feature weighting algorithm is applied to our 3D-view-based features, which can emphasize the more important views of the 3D models, so the retrieval performance can be improved. The experimental results on the standard 3D model dataset, Princeton ModelNet10, show that our model can achieve promising performance.

**Keywords:** 3D model retrieval, voxelization, deep features, feature weighting, deep learning

## 1. INTRODUCTION

Deep learning[1-2] has achieved excellent performance in image recognition[3-4], natural language processing, stereo vision matching[5-6] and other tasks. Convolutional neural networks[7-8] (CNNs) have been applied in the field of content-based image retrieval, and have proved the applicability of deep convolutional neural networks in the field of image retrieval. However, their applications in the field of 3D model retrieval still need to be explored and optimized. CNNs have entered into the public field of vision in the early 21st century. In 2003, the Bag of Words Model (BoW)[9] became the most effective approach for image retrieval, and then in 2004, BoW was combined with the SIFT method [10]for image classification. In the 10 years that followed, we witnessed the superiority of the BoW model, which brought improvements to the image retrieval tasks. Krizhevsky et al[11] proposed a new CNN model for image classification in 2012. Then, the features used for image retrieval gradually changed from the SIFT features to the CNN-based or deep features. Alzu'bi et al.[12] proposed a new structure based on bilinear CNN, which was directly applied to images, with different locations and scales, to perform image retrieval. Salvador et al.[13] take advantage of the object proposals learned by a Region Proposal Network (RPN) and their associated CNN features to build an instance search pipeline composed of a first filtering stage followed by a spatial reranking. They further investigate the suitability of Faster R-CNN features when the network is fine-tuned for the same objects one wants to retrieve. Li et al.[14] used CNN to extract features from multiple views of the same 3D model, and then compressed the features for each view into a 128-dimensional comprehensive features for the representation of the 3D model. Although the above models for 3D object retrieval achieved high retrieval accuracy, it is difficult to obtain the appropriate example models in practice. In addition, the spatial feature information of objects should be taken into account.

In view of the complexity of some 3D objects, it is difficult for 3D retrieval techniques to effectively describe the complicated spatial structures of the objects.Therefore, the consideration of spatial feature information of 3D models in a retrieval framework has become an important issue. In response to this problem, we propose a 3D retrieval network based on 3D-view-based weighting of deep features. In our network, we first normalize the model to have the same coordinate scale. Then, the normalized model is transformed into voxel, in order to obtain deep features of three different views of the objects, which can retain a certain amount of spatial information of the objects. This can benefit the retrieval of 3D

objects with complex spatial structures. We evaluated our framework on the Modelnet10 dataset, and the details will be illustrated in Section 3. The structure of our proposed deep model is shown in Figure 1.

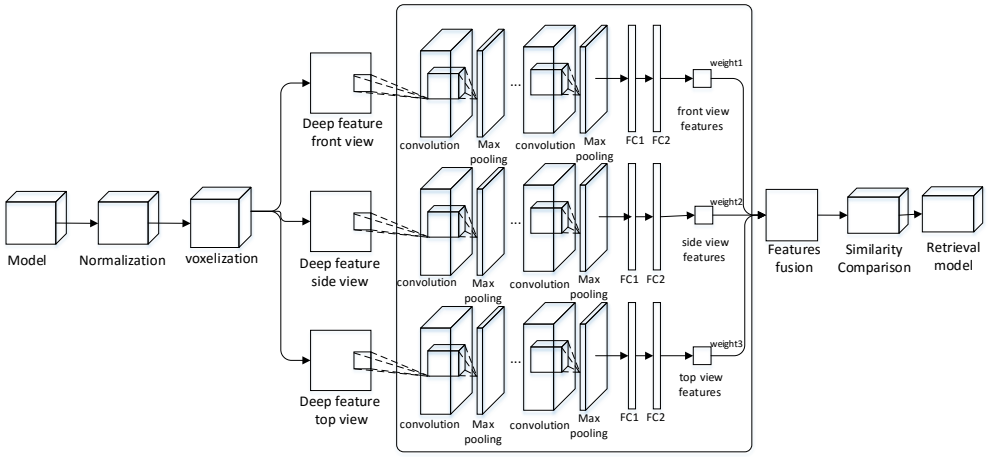



Figure 1. The flow chart of our method

Our proposed model was trained and tested on the Princeton University Standard Three-Dimensional ModelNet10 dataset. We evaluate the performance of our model for 3D object retrieval, in terms of the recall rate and accuracy. The remainder of this paper is organized as follows. We present our proposed method in detail in Section 2. The experiment set-up and results, and conclusion, are given in Sections 3 and 4, respectively.

## 2. PROPOSED METHOD

### 2.1 Feature views extraction

A 3D model consists of rich and complex data, which integrates points, lines, and surfaces. In our experiments, the ModelNet10 data set was employed, which contains a set of 3D CAD models provided by Princeton University, obtained by manual screening. The data set is composed of two subsets, namely ModelNet40 and ModelNet10, respectively. ModelNet10 was formed by selecting from the most common ten 3D object models. And in views of 3D models in ModelNet10 is more widely used in reality, ModelNet10 is selected as the data set in our work. The experiment based on modelnet40 will supplement the experimental details in the future work

The 3D model of an object may have many differences, in terms of orientation, size and position. Furthermore, the rotation, stretch and translation operations can be used to create and acquire 3D models. These geometric transformations will make the comparison between 3D models more difficult. In order to represent and compare 3D models more accurately, it is necessary to normalize the preprocessed 3D model. Normalizing a 3D model is to place the model in a unified 3D coordinate system. For this purpose, principal component analysis (PCA) is employed to normalize the model. PCA is a multivariate statistical analysis method. Initially, we denote a 3D model as $O = \{\tilde{v}, \tilde{f}\}$ where $\widetilde{v}$ and $\widetilde{f}$ represent the set of vertices and triangular facets, respectively. The vector $\widetilde{v_i} = (x_i, y_i, z_i) \in \widetilde{v}$ represents vertex coordinates, and the vector $f_j(p_{j1}, p_{j2}, p_{j3})$ represents the index sequence of the three vertices of each triangular facet. $p_{j1}, p_{j2}, p_{j3}$ represent the three vertex vectors of a triangular facet, while $m$ represents the number of triangular patches. Furthermore, $s_j$and $g_j$represent the area and the center of mass of each triangular facet, respectively. The normalization process is as follows:

(1) Translation. The centroid of the 3D model is first calculated, and the centroid is then translated to become the origin of the coordinate system. The centroid $m_{\tilde{v}}$ of the model can be expressed as follows:

$$m_{\tilde{v}} = \frac{1}{s}\sum_{j=1}^{m} s_j g_j = \frac{1}{n}\sum_{j=1}^{m} \frac{ns_j}{s}\frac{p_{j1}+p_{j2}+p_{j3}}{3} \quad (1)$$

where $n$ is number of vertices, and s is the sum of all the triangular patches of the model. Then, the model after translation is given as follows:$\widetilde{O_I} = \tilde{v} - m_{\tilde{v}}$, $p'_{jq} = (p_{jq} - m_{\tilde{v}}), j = 1, \cdots, n$, $q = 1,2,3$. Considering that the orientation of the objects in the dataset used in our experiment has been aligned, we do not perform any rotation.

(2) Scale normalization. A scaling factor $S$ is calculated, as follows:

$$S = \frac{1}{S}\sum_{j=1}^{m} s_j \left\| \frac{p'_{j1}+p'_{j2}+p'_{j3}}{3} \right\| \quad (2)$$

Therefore, the points computed, i.e. $p_{jq}'' = S^{-1}p'_{jq}, j = 1,\cdots,n,\ q = 1,2,3,$ are invariant to translation and scale normalized. In order to improve the retrieval performance, the voxelization process is needed. Then, the voxel representation can be supplemented with spatial information. In order to avoid losing the features of 3D model after voxelization, the voxel width is set at 1 in our algorithm, i.e. a cube of 1*1*1 is used as a single voxel point.And we set the resolution of voxel model to 64×64×64.

After normalization and voxelization of a 3D model, we extract the 3D-view-based deep features, which carry the spatial information about the 3D object. Specifically, we consider the top view, front view, and side view of a 3D object, which is divided into 64×64 small cubes according to each pixel block of each view along the vertical direction. Each of these voxel points which are contained in each small cube are counted. And the number of points is viewed as the gray value of this pixel block. Therefore, 2D image of each view is obtained on the basis of the gray value of each pixel. Unlike those previous projection-based algorithms, the deep features extracted from the three views can reflect the internal, spatial characteristics of the 3D voxel model, and can better supplement the information of the 3D model. Consequently, the efficiency of this 3D model recognition and retrieval method can be improved. The feature extraction process of our method is illustrated in Figure 2.

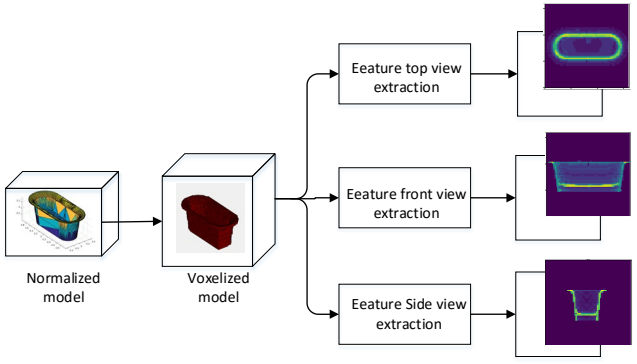



Figure 2. Voxelization and feature extraction from three different views

## 2.2 Feature learning network

VGG16 convolution neural network is a network proposed by Computer Vision Group of Oxford University. Our network uses VGG16 for reference and adjusts the parameters of convolution layer and full connection layer for general feature extraction, so as to train a multi-view deep feature map. The structure of the feature learning network with the deep feature of single view can be represented as Figure 3.

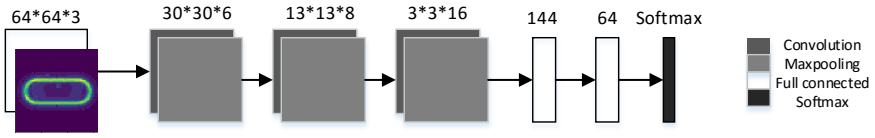



Figure 3. Feature extraction network structure of a single deep feature view

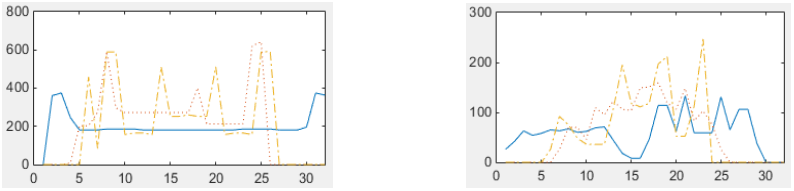
In this paper, the area of a 2D view is normalized to form a 64×64 input. The deep network is trained with stochastic gradient descent (SGD). Unlike VGG16, which sets the convolutional kernels to the size of 3×3, our method uses convolutional kernels of the optimal size of 5×5 by constantly adjusting network parameters. The convolution layer output is linearly deformed, followed by the non-linear ReLU activation function, which can reduce the tendency of over-fitting by the increase the sparsity of the network. At the end of the convolution layer, a pooling layer based on 2×2 convolution kernels is performed. In the design of the fully connected layer, our method generates features with dimensions of 144 and 64. Considering that the deep feature maps contain less information than a natural image, the three-layer fully connected layer is reduced to two layers only to improve the convergence speed. The features from the three views are fused to form

the final feature vector for 3D model classification and retrieval. For image retrieval, the distance between the feature vector of the input 3D model and that for each 3D model in the database is computed, and the twelve with the shortest distance are taken as the matched or retrieved 3D object.

# 3. EXPERIMENTS AND RESULTS

## 3.1 Feature Extraction Based on Modelnet10

Firstly, we normalize and voxelize the 3D models in ModelNet10, and obtain the three views for each model, which reflect the spatial structure of the model from multiple perspectives. In this process, we construct the feature view according to the voxel point distribution, so we can observe the voxel distribution from the extracted view. In order to improve the retrieval performance, the corresponding weights, which are set according to the distribution, can be adjusted. The voxel distributions of different models are shown in Figure 4. Furthermore, convolutional neural networks are employed to extract the deep features of the three views, which are then used for matching and retrieval.




a. The voxel distributions of sofa_0552.off, for the three different views b. Voxel distributions of toilet_0198.off , for the three different views

Figure 4. Voxel distributions of the features in three different views.

In the voxel distribution curves, the abscissa represents the resolution of a voxel, the ordinate represents the number of voxel points, and '. ' , '. -' and '-' represent the voxel distribution of the emmetropic view, the side view, and the top view, respectively. In order to improve the rendering speed, the resolution of the voxelized model is set to 32×32×32, so the maximum abscissa coordinate is 32.

In the training of a convolutional neural network, the learning rate will affect the loss convergence speed of the network. In order to avoid using a higher learning rate, which will make the accuracy oscillate and be difficult to converge, we firstly select parameters by setting different learning rates. In addition, we can observe the change of loss curve to find the optimal learning rate. In our experiments, we found the optimal learning rate to be set at 0.07.

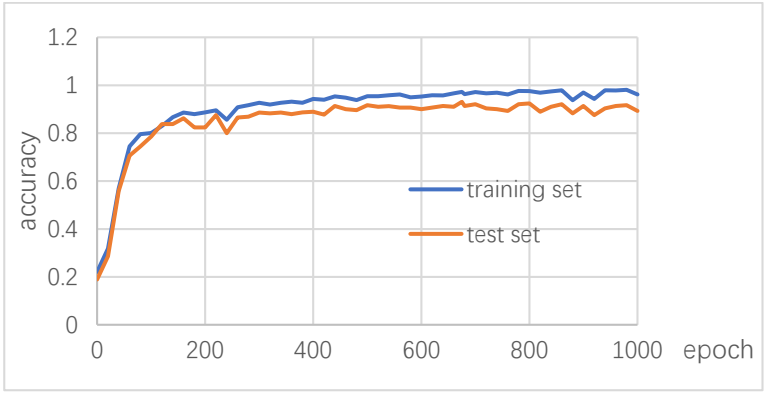



Figure 5. The accuracy curves on the training and testing samples from ModelNet10.

As can be seen from Figure 5, when the number of training epochs reaches about 1000, the accuracy of the training set achieves 98.91%, while the accuracy of the set is 92.41%. Compared with other feature extraction algorithms, our approach has outstanding advantages, in terms of accuracy, as shown in Table 1.

Table 1. Comparisons of accuracy between our net and traditional feature learning networks

| Net | Orthographic Net | PANORAMA-NN | Geometry Image | DeepPano | 3Dshapenet | Ours |
|---|---|---|---|---|---|---|
| Accuracy | 88.56% | 91.1% | 88.4% | 85.45% | 83.5% | 92.41% |

### 3.2 3D Model Retrieval

Generally, recall rate R and accuracy P are the commonly used evaluation criteria for text and visual information retrieval. In the evaluation of the recall rate and the precision rate, R actually reflects the overall ability of the retrieval algorithm, that is, the ability to retrieve all relevant models in the database, while P reflects the system accuracy, i.e., the ability to retrieve only the relevant category model. In order to obtain the overall performance of the network, we use $mAP$ to evaluate the network. The limitation of P and R can be solved by $mAP$, which reflects the average of all kinds of average precision $AP$,as shown in Formula 3.

$$AP = \int_0^1 P(R)\, dR \tag{3}$$

In our framework, we first calculate the similarity between the feature of the model to be retrieved and the other models in the database. Then, the retrieved models can be ranked according to the measured similarity between the input and the feature vector from the database. Then, the recall and accuracy can be calculated based on the ranking. Finally, the Precision-Recall (PR) curve is acquired. In the training process, our network fuses multi-view features by selecting different weights. w1, w2 and w3 are the weights of deep features of the emmetropic view, the side view, and the top view, respectively. Figure 6 shows the PR curve generated with the selected optimum weight parameters, which are w1=0.4, w2=0.6, w3=0.6. We compare the performance of our proposed method with current 3D model retrieval algorithms, and the results are tabulated in Table 2. We can see that our method outperforms the other methods, in terms of in terms of *mAP*.

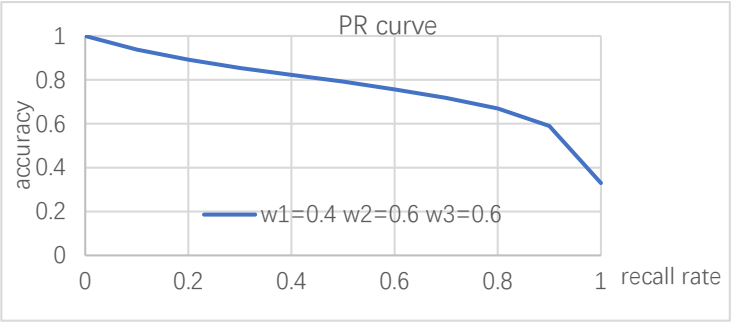



Figure 6. PR curves with optimal feature weights.

Table 2. $mAP$ comparison between our network and traditional three-dimensional retrieval network.

| Net | LFD | SPH | 3Dshapenet | Geometry Image | Ours |
|---|---|---|---|---|---|
| $mAP$ | 49.82% | 44.05% | 68.3% | 74.9% | 76.02% |

In the retrieval experiments, we have developed a visual interface. By choosing the model to be retrieved, we can directly observe the feature distribution curve of the three different views in the interface. In order to optimize the retrieval, we can modify the weight value according to the feature distribution curve. The retrieval model firstly identifies the type of the object to be retrieved. Then, the surface shows 12 models which are most similar to the target. As shown in Figure 7.

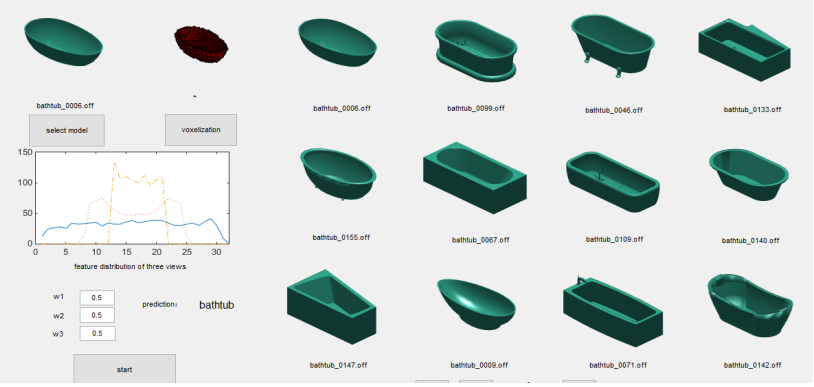



Figure 7. Retrieval results based on three-dimensional models to be retrieved

## 4. CONCLUSIONS

In this paper, a new three-dimensional (3D) model retrieval scheme, based on multi-view feature weighting, is proposed. Based on the respective voxel distribution curve of the three views after voxelization, we can observe that each model has different spatial feature information. Our features extracted from three different views can effectively supplement the missed information. In addition, the retrieval efficiency can be improved by setting different weights, through the feature distributions among the three views. The proposed method can achieve excellent performance on the Princeton University data set ModelNet10. Compared with the traditional 3D retrieval methods, experimental results shows that our framework achieves the best performance, in terms of different performance indices. More importantly, our method also achieves promising efficiency and accuracy.

## REFERENCES

[1] Schmidhuber, Jürgen., "Deep learning in neural networks: An overview". Neural Networks, 61:85-117(2015).
[2] SUN Z Y, LU C X, SHI Z Z, et al., "Research and progress of deep Learning, "Computer Science, 43(2):1-8(2016).
[3] LUO W, SCHWING A G, URTASUN R, "Efficient Deep Learning for Stereo Matching, "IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 5695-5703(2016).
[4] ZHANG J, BAI C, NEZAN J F, et al., "Joint motion model for local stereo video-matching method, "Optical Engineering, 54(12):123108(2015).
[5] Shi M , Xie F , Zi Y , et al., "Cloud detection of remote sensing images by deep learning, "IGARSS 2016 - 2016 IEEE International Geoscience and Remote Sensing Symposium. IEEE(2016).
[6] Zhang J , Shang J , Zhang G ., "Verification for Different Contrail Parameterizations Based on Integrated Satellite Observation and ECMWF Reanalysis Data, "Advances in Meteorology, 2017:1-11(2017).
[7] He K , Zhang X , Ren S , et al., "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition, "IEEE Transactions on Pattern Analysis & Machine Intelligence, 37(9):1904-16(2014).
[8] Tao W, Wu D J, Coates A, et al., " End-to-End Text Recognition with Convolutional Neural Networks, " International Conference on Pattern Recognition (2012).
[9] Sivic J and Zisserman A., "Video google: A text retrieval approach to object matching in videos, "IEEE 9th International Conference on Computer Vision (ICCV) ,1470-1477(2003).
[10] Lowe D G ., "Distinctive Image Features from Scale-Invariant Keypoints, "International Journal of Computer Vision, 60(2):91-110(2004).
[11] Krizhevsky A, Sutskever I, Hinton G E., "Imagenet classification with deep convolutional neural networks"Advances in neural information processing systems, 1097-1105(2012).
[12] ALZU'BI A, AMIRA A, RAMZAN N., "Content Based Image Retrieval with Compact Deep Convolutional Features,, "Neurocomputing, 249(2):95-105(2017).
[13] SALVADOR A, GIROINIETO X, MARQUES F, et al., "Faster R-CNN Features for Instance Search, "IEEE Conference on Computer Vision and Pattern Recognition Workshops. IEEE Computer Society, 394-401(2016).
[14] LI X X, CAO Q, WEI S., "3D object retrieval based on multiview convolutional neural networks, "Multimedia Tools & Applications, 76(19):20111-201214(2017).