MLVCNN: Multi-Loop-View Convolutional Neural Network for 3D Shape Retrieval

Jianwen Jiang, Di Bao, Ziqiang Chen, Xibin Zhao,* Yue Gao*
BNRist, KLISS, School of Software, Tsinghua University, China.
{jjw17, bd17, czq18}@mails.tsinghua.edu.cn, {zxb, gaoyue}@tsinghua.edu.cn

Abstract

3D shape retrieval has attracted much attention and become a hot topic in computer vision field recently.With the development of deep learning, 3D shape retrieval has also made great progress and many view-based methods have been introduced in recent years. However, how to represent 3D shapes better is still a challenging problem. At the same time, the intrinsic hierarchical associations among views still have not been well utilized. In order to tackle these problems, in this paper, we propose a multi-loop-view convolutional neural network (MLVCNN) framework for 3D shape retrieval. In this method, multiple groups of views are extracted from different loop directions first. Given these multiple loop views, the proposed MLVCNN framework introduces a hierarchical view-loop-shape architecture, i.e., the view level, the loop level, and the shape level, to conduct 3D shape representation from different scales. In the view-level, a convolutional neural network is first trained to extract view features. Then, the proposed Loop Normalization and LSTM are utilized for each loop of view to generate the loop-level features, which considering the intrinsic associations of the different views in the same loop. Finally, all the loop-level descriptors are combined into a shape-level descriptor for 3D shape representation, which is used for 3D shape retrieval. Our proposed method has been evaluated on the public 3D shape benchmark, i.e., ModelNet40. Experiments and comparisons with the state-of-the-art methods show that the proposed MLVCNN method can achieve significant performance improvement on 3D shape retrieval tasks. Our MLVCNN outperforms the state-of-the-art methods by the mAP of 4.84% in 3D shape retrieval task. We have also evaluated the performance of the proposed method on the 3D shape classification task where MLVCNN also achieves superior performance compared with recent methods.

Introduction

Recently, the 3D shape retrieval problem has gradually become an important issue in computer vision due to the wide use of 3D shapes in different areas such as automatic driving, 3D printing and gaming. The task of 3D shape retrieval targets on finding the most similar 3D models from the dataset given the query, and the retrieval is conducted based on the shape similarities.

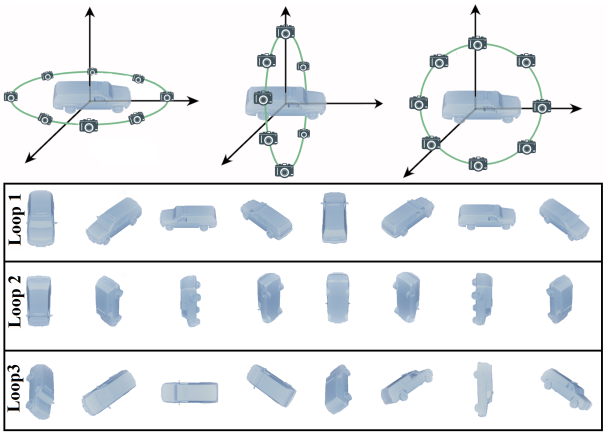

Figure 1: Demonstration of multiple loop views. In this example, 3 loops of views are provided, in which each loop is perpendicular to an axis in the three-dimensional coordinate system separately. The bottom of this figure shows the views from the 3 loops for the 'car' shape.

Generally, 3D shapes have complex geometries and variations, and thus it is very difficult to represent 3D shapes well, leading it a challenging task to retrieve 3D shapes accurately. There have been plenty of works concentrated on 3D shape retrieval in the last decade. Within the proliferation of deep learning in recent years, various deep networks have been investigated for 3D shape analysis, such as 3D ShapeNets (Wu et al. 2015), PointNet (Qi et al. 2017a) and VoxNet (Maturana and Scherer 2015). Among these methods, view-based method has shown better performance in many works. It is noted that it is not an easy task to represent 3D shapes well using multiple views. Existing methods may employ a group of views generated by a loop of virtual cameras (Su et al. 2015; Feng et al. 2018), a dense-sampled camera dome or just several predefined directions. However, existing methods have not taken the intra-view relationships into consideration, simply treating all these views independently. Under such circumstances, it is important to investigate more robust view generation and representation method for 3D shape retrieval.

In this paper, we propose a multi-loop-view convolutional

*Corresponding authors

neural network (MLVCNN) framework for 3D shape retrieval. In this method, multiple groups of views are extracted from different loop directions first. Figure 1 demonstrates 3 orthogonal loops, in which each loop contains 8 views for multiple views generation. Given these multiple loop views, the proposed MLVCNN framework introduces a hierarchical view-loop-shape architecture, i.e., the view level, the loop level, and the shape level, to conduct 3D shape representation from different scales. In the view-level, a convolutional neural network is first trained to extract view features. Then, Loop Normalization and LSTM are utilized for the views in each loop to generate the loop-level features by considering the intrinsic associations of the different views in the same loop. Finally, all the loop-level descriptors are combined into a shape-level descriptor for 3D shape representation, which is later used for 3D shape retrieval. Our proposed method has been evaluated on the public 3D shape benchmark, i.e., ModelNet40. Experiments and comparisons with the state-of-the-art methods show that the proposed MLVCNN method can achieve significant performance improvement on 3D shape retrieval tasks. We have also evaluated the performance of the proposed method on the 3D shape classification task where MLVCNN also achieves superior performance compared with recent methods.

The contributions of this paper are as follows:

- We introduce a multi-loop-view 3D shape representation scheme, and more specifically, it is a new hierarchical feature representation method. In this method, the intra-view relationships are taken into consideration for 3D shape description. In this way, 3D shapes can be represented better from different scales.
- We introduce Loop Normalization (LN), which can be regarded as a local normalization in the loop dimension. LN can represent information better for each loop of views by keeping local discriminativeness from being weakened by global normalization.
- We have conducted experiments on the ModelNet40 dataset. Experiments results reveal the proposed multi-loop-view structure can represent 3D shapes better compared with traditional view-based methods. Our MLVCNN outperforms the state-of-the-art methods by the mAP of 4.84% in retrieval task and also achieves superior performance with recent methods in classification task.

The rest of the paper is organized as follows. We first introduce the related work. Then we present our proposed MLVCNN method. After that, the experiments and discussions are provided. Finally, we conclude this paper in Conclusions.

Related Work

3D shape retrieval have been investigated in recent years. In this section, we briefly review recent works for 3D shape retrieval. There are plenty of handcraft 3D descriptors, and early methods can be mainly divided into two categories, i.e., model-based methods (Osada et al. 2002) and view-based methods(Chen et al. 2003).

Traditional model-based methods utilize shape distributions to measure the similarity among 3D shapes based on distance, angle, area and volume of random surface points. In (Akgül et al. 2009), Akgul et al. proposed a probabilistic generative descriptor which can use local shape properties for 3D shape retrieval. Some other methods employ voxel grid (Wu et al. 2015), polygon mesh (Bronstein et al. 2011) or local shape diameters measured at densely sampled surface points (Chaudhuri and Koltun 2010) for 3D shape retrieval. In (Gao et al. 2012), the similarity between two 3D objects was measured by the comparison between two groups of views.

In recent years, the method of deep learning has been widely used in 3D shape retrieval. Su et al. (Su et al. 2015) proposed a multi-view convolutional neural network (MVCNN). In this approach, convolutional neural networks are first used to generate the feature for each view individually and then multi-view features are fused by a pooling procedure. A low-rank Mahalanobis metric is employed in MVCNN to improve the retrieval performance. Qi et al. (Qi et al. 2016) further investigated view-based descriptor and volumetric-based descriptor in 3D shape retrieval. In (Xie et al. 2017), a progressive shape distribution encoder is introduced to generate 3D shape representation. Feng et al. (Feng et al. 2018) introduced a hierarchical view-group architecture to exploit the distinctions among views. Dai et al. (Dai, Xie, and Fang 2018) introduced BiLSTM in 3D shape retrieval. He et al. (He et al. 2018) improved center loss(Wen et al. 2016) and proposed a triplet-center loss (TCL) for 3D shape retrieval.

Although deep learning methods have shown superior performance compared with traditional methods in the task of 3D shape retrieval, it still suffers from the limitation of missing mining intra-view relationship and the hierarchical nature of 3D shape representations. In this paper, the proposed MLVCNN framework targets on handling these challenging issues by its multi-loop-view structure and hierarchical representation scheme.

The Proposed MLVCNN Method

In this section, the proposed MLVCNN method is introduced. In MLVCNN, we first provide the view-loop-shape 3D shape representation structure, which can represent 3D shape in a hierarchical way. Given the multi-loop-view data, view features are first extracted. Then a Loop Normalization (LN) method together with LSTM is introduced to generate loop-level features by exploring the relationship among views in each loop. Later, a shape-level feature is generated by combining the loop-level features. Finally, 3D shape retrieval is based on the comparison using the shape-level features.

Multi-Loop-View Hierarchical Framework

Figure 2 shows our network structure. Our network is divided into 3 stages: view-level descriptor generation, loop-level descriptor generation and shape-level descriptor generation. In the first stage, multiple groups of views are generated from different loop directions. Then a basic CNN model

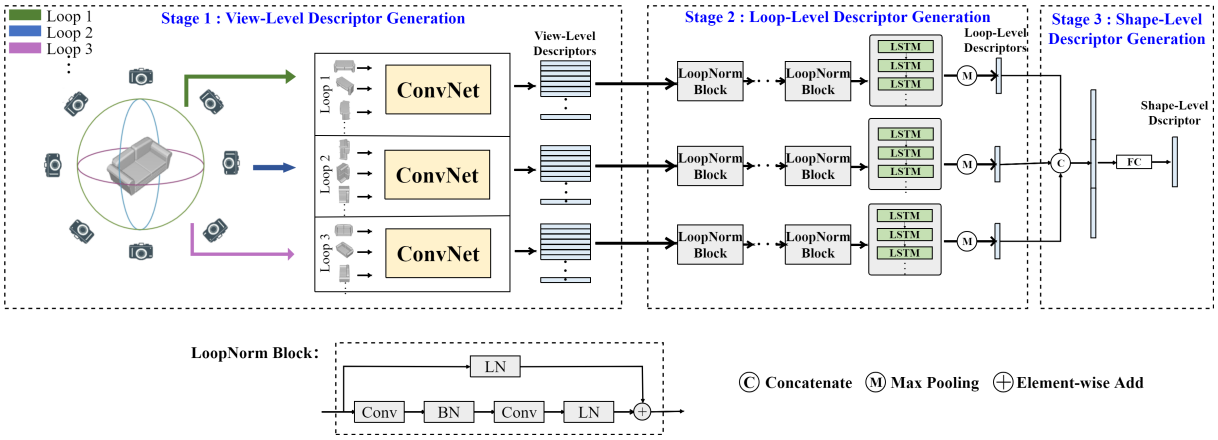

Figure 2: The Multi-Loop-View CNN framework for 3D shape retrieval. LN in LoopNorm Block denotes Loop Normalization and BN denotes Batch Normalization. The proposed MLVCNN contains 3 stages, during which it gradually obtains the descriptor from local descriptor of view-level, to more global descriptor of loop-level and shape-level.

is used to extracted features from these views to obtain view-level descriptors. In the second stage, the view-level descriptors are fed into stacked LoopNorm Blocks and LSTM block sequentially to generate loop-level descriptors. In the last stage, the loop level descriptors are aggregated to obtain a global shape-level representation.

View-level feature extraction

In order to obtain comprehensive information of a 3D shape, orthogonal loops projection are introduced for a better view-projection method. In our method, we set the number of loops to three as a default setting. Each loop of the three loops is perpendicular to an axis in a three-dimensional coordinate system as Figure 1 shows. Because of the orthogonality, the three loops projection method will increase the diversity of sequence information. Based on the three orthogonal loops, we rotate these orthogonal loops to generate more loops, which forms our multi-loop projection method. In each loop, we generate view projections like the method proposed in (Su et al. 2015).

With the help of multi-loop projection method, more discriminative views are employed, which is very important for the retrieval task. As Figure 1 shows, in loop 1, such a projection method yields eight views of the sides of the car. Nonetheless, the information from the top and bottom of the car are missing. Views in loop 2 and loop 3 provide complementary perspectives helping to capture the crucial information of the object for better 3D shape representation.

We employ a full convolutional network as our basic feature extractor. Given such views from different loops, we feed them into the basic CNN which shares the same parameters to obtain view-level descriptors. In the stage, CNN treats views separately without considering of the relationship among them.

Loop-level feature extraction

In the loop-level feature extraction, the given view-level descriptors are fed into stacked LoopNorm Blocks to obtain better view-level features which consider the statistic difference among loops. Then we employ the long short-term memory(LSTM) network on these features to generate loop-level descriptors.

Normalization methods like Batch Normalization (BN) are widely used in deep learning. In our MLVCNN framework, we propose a more suitable normalization method named Loop Normalization (LN).

In our MLVCNN, the input data is represented by a 6D tensor of (N,C,L,V,H,W), where N denotes the batch dimension, L denotes the loop dimension, V denotes the view dimension, C denotes the channel dimension, H denotes the height dimension and W denotes the width dimension. X_i is the pixel before normalization, and $\tilde{X}_i$ is the value after normalization. i is the index for each pixel in (N,C,L,V,H,W) order. μ and σ denote a mean and a standard deviation respectively. For a general feature normalization, we have

$$\tilde{X}_i = \gamma \frac{X_i - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}} + \beta \tag{1}$$

where γ is a scale, β is a shift parameter. ϵ is a small constant for numerically stability. Equation (1) shows that each value in feature map is normalized by using μ and σ, and then γ and β are applied to increase the flexibility of representation capability. For μ and σ, we have

$$\mu_i = \frac{1}{|K|} \sum_{p \in S_i} X_i \tag{2}$$

$$\sigma_i^2 = \frac{1}{|K|} \sum_{p \in S_i} (X_i - \mu_i)^2 \tag{3}$$

where $|K|$ is the number of the selected pixel. Let S_i be the set of pixels to be normalized. As shown in Figure 3, in Batch Normalization, the S_i are all pixels sharing the same channel index. Different from BN, in Loop Normalization, the S_i are all pixels sharing the same batch index and loop index.

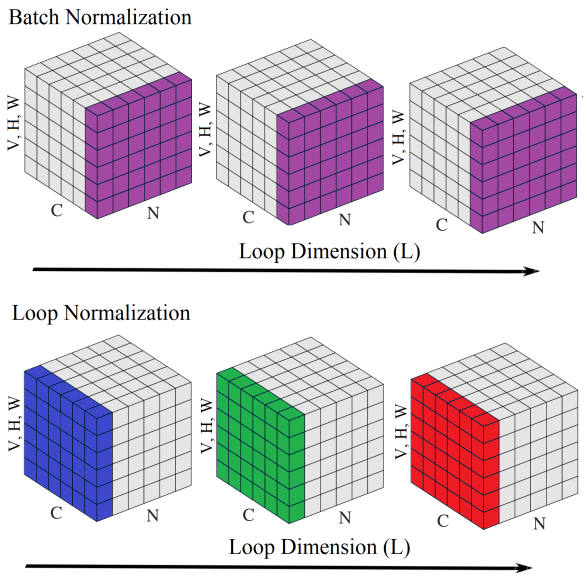

Figure 3: Batch Normalization(Top) and Loop Normalization(Bottom). In each row, cubes represent the input 6D tensor of (N,C,L,V,H,W) and each cube represents the sub-tensor sharing the same loop index in the input feature map tensor. In BN, pixels sharing the same channel(marked in purple) are normalized together. In LN, only pixels sharing the same channel and loop(marked in blue, green and red, respectively) are normalized together.

As shown in Figure 2, based on Loop Normalization, We design our LoopNorm Block (LN Block) like residual blocks in ResNet18 (He et al. 2016). In our LN Block, we add Loop Normalization(LN) to the block after last convolutional layer and the identity path. There are two reasons to do so. Firstly, in residual path, we add LN after last convolutional layer in order to maintain the content discrimination in loop dimension. Secondly, we add LN in identity path in order to avoid misalignment between residual path and identity path.

After employing LN Block, we treat the features of each view in the same loop as each element in the input sequence. Then we feed the sequence to a LSTM network. After that, we apply maxpool to the outputs from all hidden layers in the LSTM to obtain loop-level descriptors. By employing LSTM, our proposed method considers the relationship among views in the same loop. For each view in the loop, LSTM is used to obtain its relationship among all previous views. LN block maintains the difference in statistics among loops and LSTM is applied in each loop separately, which strengthens the relationship of the features within the same loop and is helpful for enhancing the discrimination of loop-level features.

Shape-level feature extraction and retrieval

In shape-level, we concat features from each loop to form a global feature. Then we feed these features to a fully connected layer to obtain compact shape-level representation which is more suitable for retrieval task. The process can be represented by

$$feat_g = Concat(feat_{l1}, ..., feat_{li}, ..., feat_{ln}) \quad (4)$$

where $feat_{li}$ denotes the feature of loop i, $feat_g$ denotes the global feature of the object.

Then, we use the global feature to obtain the shape-level descriptor by

$$embedding = W \cdot feat_g + \sigma \quad (5)$$

where W denotes weight matrix, σ denotes a bias term and $embedding$ denotes the shape-level descriptor.

In MLVCNN, we directly use shape-level descriptor for 3D shape retrieval and use L2 distance between two 3D shapes as similarity measure in retrieval task. The distance metric formula is defined in Equation 6.

$$d(f_a, f_b) = ||f_a - f_b||_2 \quad (6)$$

Based on the L2 distance, we can rank the distance between query object and object in our dataset to generate the retrieval result.

Experiments

In this section, we first provide the experiments on 3D shape retrieval. Then we discuss the results and comparisons with the state-of-the-art methods. After that, the impact of our proposed Loop Normalization method and the influence of the number of views and loops on the 3D shape retrieval performance are investigated. In the last part, we evaluate the proposed method on 3D shape classification task.

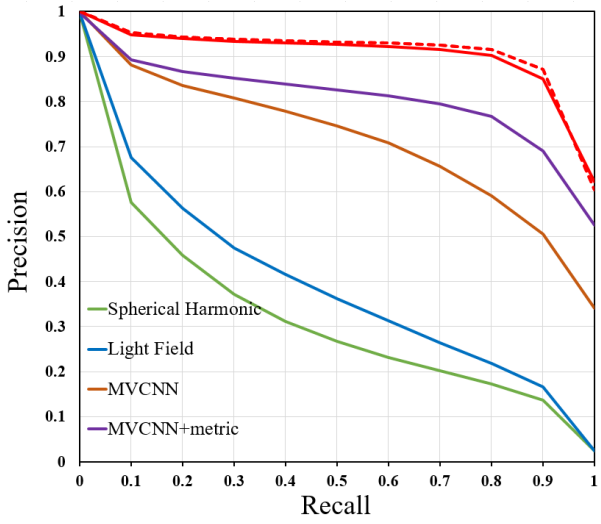

Figure 4: Precision-recall curves for our MLVCNN and some other methods on the task of shape retrieval on the ModelNet40 dataset. Our MLVCNN, without metric learning methods including low-rank Mahalanobis metric learning and metric learning loss, still outperforms the state-of-the-art methods.

Method	Training Config.		Data Representation.	Retrieval (mAP)
	Pre train	Fine tune		
(1)SPH(Kazhdan, Funkhouser, and Rusinkiewicz 2003)	-	-	-	33.3%
(2)LFD(Chen et al. 2003)	-	-	-	40.9%
(3)MVCNN (Su et al. 2015)	ImageNet1K	ModelNet40	View	80.2%
(4)GIFT(Bai et al. 2016)	-	ModelNet40	View	81.9%
(5)DeepPano(Shi et al. 2015)	-	ModelNet40	View	76.8%
(6)GVCNN(Feng et al. 2018)	-	ModelNet40	View	85.7%
(7)MVCNN with TCL(He et al. 2018)	-	ModelNet40	View	88.0%
(8)3D ShapeNets(Wu et al. 2015)	ModelNet40	ModelNet40	Voxel	49.2%
(9)DLAN(Furuya and Ohbuchi 2016)	-	ModelNet40	-	85.0%
(10)RED(Bai et al. 2017)	-	ModelNet40	Multi-Modality	86.3%
(11)MLVCNN(without LN), 3x8	ImageNet1K	ModelNet40	View	88.55%
(12)MLVCNN(without LN), 3x12	ImageNet1K	ModelNet40	View	89.05%
(13)MLVCNN(with LN), 3x8	ImageNet1K	ModelNet40	View	91.07%
(14)MLVCNN(with LN), 3x12	ImageNet1K	ModelNet40	View	91.15%
(15)MLVCNN(with LN) + Center Loss, 3x8	ImageNet1K	ModelNet40	View	92.22%
(16)MLVCNN(with LN) + Center Loss, 3x12	ImageNet1K	ModelNet40	View	**92.84**%

Table 1: Retrieval results on the ModelNet40 dataset. In experiments, our proposed MLVCNN method is compared with the state-of-the-art methods that use different representations of 3D shapes. Noted that, '3x8' indicates the number of loops is 3 and the number of views in each loop is 8. So is '3x12'.

3D Shape Retrieval Results

To evaluate the effectiveness of the proposed MLVCNN method, we have conducted 3D shape retrieval experiments on the Princeton ModelNet dataset. ModelNet dataset contains 127,915 3D CAD models from 622 object categories. A subset of 40-common classes (ModelNet40) including 12311 3D shapes is used in our experiments. We follow the same training and testing split setting in (Su et al. 2015). To evaluate the performance of 3D shape retrieval, the widely used retrieval mAP is employed here. Different from original MVCNN, We adopt ResNet18, a CNN model with similar performance to VGG on imagenet classification task, as our backbone due to the limitation of GPU memory. In the next sub-section, in order to demonstrate the effectiveness of MLVCNN, ResNet18 is chosen to be base model for MVCNN and MLVCNN to do ablation studies.

To evaluate the performance of the proposed method, the following methods are selected for comparison, including hand-craft descriptors, deep learning models and ensemble methods. More specifically, in hand-craft descriptors, Rotation Invariant Spherical Harmonic Representation (SPH) (Kazhdan, Funkhouser, and Rusinkiewicz 2003) and Lighting Field Descriptor (LFD) (Chen et al. 2003) are employed. In deep learning models, Multi-View CNN (MVCNN) (Su et al. 2015), GIFT (Bai et al. 2016), DeepPano (Shi et al. 2015), Group-View CNN (Feng et al. 2018), MVCNN with Triplet Center Loss (He et al. 2018)), 3D shapeNets (Wu et al. 2015) and Deep Local feature Aggregation Network(DLAN) (Furuya and Ohbuchi 2016) are chosen. A typical ensemble method Regularized Ensemble Diffusion (RED) (Bai et al. 2017) is also selected for comparison.

The experimental results and comparisons among all methods are demonstrated in Table 1. The precision-recall curves are provided in Fig.4. As shown in these results, our proposed MLVCNN outperforms all other compared methods with an mAP of 91.15%. Compared to original MVCNN, we obtain the gains of 10.95%, simply with softmax loss.

Compared with the recent state-of-the-art 3D shape retrieval methods, i.e., MVCNN with TCL, the proposed method without metric learning loss achieves a gain of 3.15% on mAP. With metric learning loss like Center Loss, our method can further achieve a gain of 4.84% on mAP.

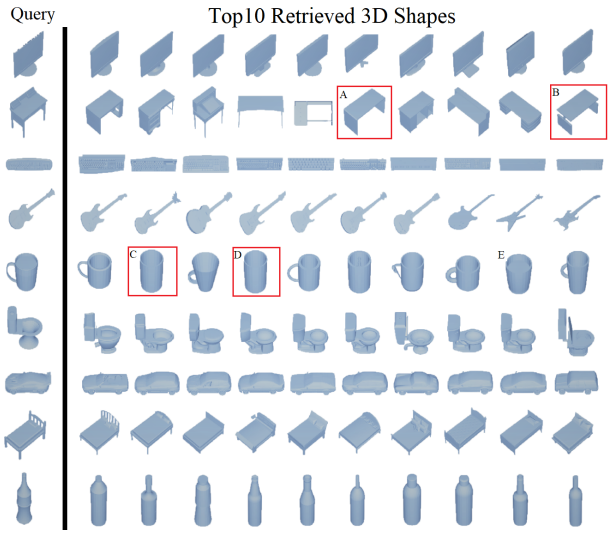

Figure 5: Retrieval examples on ModelNet40 dataset. Top10 retrieval results are shown for each query. Incorrect retrieved objects are marked with red box

Our better performance can be dedicated to the following

Models	Number of Views	mAP
MVCNN	8	80.52%
MVCNN	24	74.96%
MLVCNN	8(1 loop)	85.89%
MLVCNN	24(3 loop)	91.07%

Table 2: The performance comparison with MVCNN

reasons.

First, The 3-tier architecture can represents 3D shape effectively, which forms a hierarchical representation of view, loop, and shape, providing a compact representation of the 3D shape. Second, compared with the previous single loop or dense sampling methods, because the relationship between views is taken into consideration in the feature extraction process, MLVCNN has more space-preserving performance. Last, Loop Normalization (LN) added in our network normalizes features in loop dimension. LN exploits the loop independence and maintains the difference in statistics among different loops, which further enhances the representation capacity of our MLVCNN.

Fig.5 shows some retrieved examples of our method on the ModelNet40 dataset. The query shapes are listed at the left column, which are from 8 categories including monitor, desk, keyboard, guitar, cup, toilet, car, bed, and bottle. The top 10 retrieved results are listed on the right side. In these results, the incorrect retrieval results are marked with red box. As shown in these results, most of mistakes come from 3D shapes with very similar appearance, which are challenging even for human. For example, in the retrieval task of a desk, shape A and shape B is from class 'table'. It is difficult to identify a desk from a table by using just the model itself. Another example is from the cup and vase. Shape C and D are from class 'vase', but shape E, which looks the same as C and D, is from class 'cup'.

Ablation Studies

In MLVCNN, the proposed Multi-Loop-View framework plays an important role. In this sub-section, we further investigate the multi-loop framework in details. In this part, we set the number of the views in each loop to 8 as a default setting which is common in multi-view based methods.

In multi-view based method, rendering 3D shape along horizontal plane is a typical method, as shown in Figure 1(like loop 1). Our proposed rendering method is also demonstrated in Figure 1. It is obvious that our proposed multi-loop can observe more distinguishing views, which is helpful for 3D shape representation.

We then evaluate the performance of our proposed method on different rendering settings and demonstrate the results in Table 2. To conduct fair comparison and evaluation, in this part of experiments, the MVCNN in Table 2 denotes our implementation according to (Su et al. 2015) and MLVCNN denotes our proposed method. Noted that, all the views are regarded as the same contribution in MVCNN and the view relationship has not been taken into consideration, the number of views in MVCNN only means the difference of input scale. For MLVCNN, views are generated from different loop structures. For example, 24 views in Table 2 are equivalent to that there are 3 loops and each loop contains 8 views. As shown in the comparison, we can observe that MVCNN with 8 views outperforms MVCNN with 24 views. It means that it is hard for MVCNN to integrate information from views when the number of views increases. But for MLVCNN, with the increasing of the number of views, the performance raises accordingly. We also show the results of MVCNN and MLVCNN under the same views setting in Table 2. We can find that under one loop setting, MLVCNN can still outperform MVCNN by 5.37% on mAP, which means the sequence modeling of views in one loop is beneficial for retrieval. Under the three loops setting, our MLVCNN outperforms MVCNN by 16.11% in terms of mAP. The results indicate that such hierarchical representation structure can exploit information of views and describe 3D shapes better compared with traditional multi-view methods.

Compared with original MVCNN, when extracting features from loops, MLVCNN assumes the independence among loops and finally fuses feature of all loops to obtain global representation. In this way, the view-loop-shape structure leads to a local-to-global 3D shape representation approach. Through this from local to global approach, MLVCNN makes it possible to employ the local features of views in one loop better and take global feature representation into account meanwhile.

On Loop Normalization

To demonstrate effectiveness of Loop Normalization(LN) better, we further investigate LN in this sub-section. We evaluate the performance of the proposed method with LN and without LN in Table 3. We also apply LN to the model with different numbers of loops and demonstrate them in Table 3. We fix the number of views to 8 and vary the number of loops from 1 to 6 in this part experiments.

As shown in the results, we can find that the model with LN outperforms the model without LN in the case of the same number of loops. When the number of loop is 5, the performance of using LN on the model can achieve a gain of 2.04% on mAP.

Number of Loops	mAP (w/o LN)	mAP (w/t LN)
1	84.38%	85.89%
2	88.34%	88.85%
3	88.55%	91.07%
4	89.68%	90.91%
5	89.19%	91.26%
6	89.90%	91.33%

Table 3: The comparison of different numbers of loops for Loop Normalization.

Clearly, when the number of loops is quite small, such as 1 or 2, there is no much difference in the distribution among loops. Therefore, normalization along the loop dimension does not play a significant role in the model. As the number of loops increases, the distribution differences among loops

become apparent, and global batch normalization will pay less attention to local differences and even weaken the discriminative information. LN provides local normalization, which leads to performance improvement.

On the number of Views and Loops

In this sub-section, we focus on a critical issue about the robustness of our framework, i.e., the number of views and loops. We quantitatively investigate the influence of different numbers of views and loops on retrieval performance.

Num. of Loop	Num. of View	mAP	mAP(+CL)
3	4	88.85%	91.49%
3	6	90.18%	92.03%
3	8	91.07%	92.22%
3	12	91.15%	92.84%
2	8	88.85%	90.59%
3	8	91.07%	92.22%
4	8	90.91%	92.26%
5	8	91.26%	92.32%
6	8	91.33%	92.36%

Table 4: The comparison of different numbers of views and loops. mAP(+CL) denotes MLVCNN with Center Loss.

First, we fix the number of loops to 3 and vary the number of views. The number of employed views for each loop is selected as 4, 6, 8 and 12, respectively. The retrieval results are shown in Table 4. As shown in these results, we can observe that the increase of views can improve the retrieval performance. It is worth noted that even with 4 views in each loop (12 views in total), which is a common number of views in MVCNN based method, our MLVCNN still achieves the best performance compared with other methods.

Second, we fix the number of views to 8 and vary the number of loops. The number of employed loops is selected as 2, 3, 4, 5 and 6, respectively. The retrieval results are shown in Table 4. The increase of loops brings performance improvements, which can be attributed to the more effective information provided by more loops. Because MLVCNN ensures the independence of each loop, the local-global fusion approach makes the feature representation more compact than MVCNN. Therefore, the increasing of redundant information by using more views has little influence on the proposed MLVCNN method. We can find from the results that when the number of loop is 6, MLVCNN achieves an mAP of 91.33%, which further indicates the validity of MLVCNN.

We also apply Center Loss to MLVCNN in the retrieval task. The results show that MLVCNN with Center Loss can obtain a little gain. When the number of loops increases, the gain of center loss does increase simultaneously. However, it is obvious that the gain from added views are more than that from added loop, which reveals that better use of multi-view sequence information is more important than simply adding views.

3D Shape Classification

In this sub-section, we further investigate our method in the classification task, and the experimental results are demonstrated in Table 5. We compare our method with the state-of-the-art methods, including MVCNN (Su et al. 2015), VoxNet (Maturana and Scherer 2015), MVCNN-MultiRes (Qi et al. 2016), 3D shapeNets (Wu et al. 2015), VRN (Brock et al. 2016),PointNet (Qi et al. 2017a), PointNet++ (Qi et al. 2017b), KD-Network (Klokov and Lempitsky 2017), GVCNN (Feng et al. 2018), PointCNN (Li et al. 2018), DGCNN (Wang et al. 2018) and SO-Net (Li, Chen, and Lee 2018).

Method	Modality	Classification
(1)3D ShapeNets	Voxel	77.3%
(2)VoxNet	Voxel	83.0%
(3)VRN	Voxel	91.3%
(4)MVCNN	View	89.9%
(5)MVCNN-MultiRes	View	91.4%
(6)GVCNN	View	93.1%
(7)FusionNet	Voxel and View	90.8%
(8)PointNet	Point Cloud	89.2%
(9)PointNet++	Point Cloud	90.7%
(10)KD-Network	Point Cloud	91.8%
(11)PointCNN	Point Cloud	91.8%
(12)DGCNN	Point Cloud	92.2%
(13)SO-Net	Point Cloud	93.4%
(14)MLVCNN	View	**94.16**%

Table 5: Classification results on the ModelNet40 dataset.

In the results, we can find that our proposed MLVCNN also performs better compared with other methods. MLVCNN obtains the competitive performance on classification with an accuracy of 94.16%. This shows that our model has competitive representation ability and has strong ability of the robustness and generalization, which make it available for being applied to different tasks.

Conclusions

In this paper, we present the MLVCNN framework for 3D shape retrieval.

In this method, a hierarchical view-loop-shape architecture is introduced to obtain better 3D shape representation. In view-level, we propose a multi-loop projection method. In loop-level, we propose Loop Normalization (LN). LN together with LSTM utilizes intrinsic associations in each loop to generate loop-level descriptors. Last, all loop-level descriptors are aggregated to obtain the shape-level representation. We evaluate MLVCNN on 3D shape benchmark, i.e., ModelNet40. The results show that MLVCNN can achieve significant performance improvement on 3D shape retrieval and classification.

Acknowledgments

This work was supported by National Key R&D Program of China (Grant No. 2017YFC0113000), National Natural Science Funds of China (U1701262, 61671267),

National Science and Technology Major Project (No. 2016ZX01038101), MIIT IT funds (Research and application of TCN key technologies) of China, and The National Key Technology R&D Program (No. 2015BAG14B01-02).

References

Akgül, C. B.; Sankur, B.; Yemez, Y.; and Schmitt, F. 2009. 3D model retrieval using probability density-based shape descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31(6):1117–1133.

Bai, S.; Bai, X.; Zhou, Z.; Zhang, Z.; and Jan Latecki, L. 2016. Gift: A real-time and scalable 3D shape search engine. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5023–5032.

Bai, S.; Zhou, Z.; Wang, J.; Bai, X.; Latecki, L. J.; and Tian, Q. 2017. Ensemble diffusion for retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 774–783.

Brock, A.; Lim, T.; Ritchie, J. M.; and Weston, N. 2016. Generative and discriminative voxel modeling with convolutional neural networks. corr abs/1608.04236 (2016).

Bronstein, A. M.; Bronstein, M. M.; Guibas, L. J.; and Ovsjanikov, M. 2011. Shape google: Geometric words and expressions for invariant shape retrieval. *ACM Transactions on Graphics* 30(1):1.

Chaudhuri, S., and Koltun, V. 2010. Data-driven suggestions for creativity support in 3D modeling. *ACM Transactions on Graphics* 29(6):183.

Chen, D.-Y.; Tian, X.-P.; Shen, Y.-T.; and Ouhyoung, M. 2003. On visual similarity based 3D model retrieval. In *Computer Graphics Forum*, volume 22, 223–232. Wiley Online Library.

Dai, G.; Xie, J.; and Fang, Y. 2018. Siamese CNN-BiLSTM Architecture for 3D Shape Representation Learning. In *IJCAI*, 670–676.

Feng, Y.; Zhang, Z.; Zhao, X.; Ji, R.; and Gao, Y. 2018. GVCNN: Group-View Convolutional Neural Networks for 3D Shape Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 264–272.

Furuya, T., and Ohbuchi, R. 2016. Deep Aggregation of Local 3D Geometric Features for 3D Model Retrieval. In *BMVC*.

Gao, Y.; Tang, J.; Hong, R.; Yan, S.; Dai, Q.; Zhang, N.; and Chua, T.-S. 2012. Camera constraint-free view-based 3D object retrieval. *IEEE Transactions on Image Processing* 21(4):2269–2281.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

He, X.; Zhou, Y.; Zhou, Z.; Bai, S.; and Bai, X. 2018. Triplet-Center Loss for Multi-View 3D Object Retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Kazhdan, M.; Funkhouser, T.; and Rusinkiewicz, S. 2003. Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Symposium on Geometry Processing*, volume 6, 156–164.

Klokov, R., and Lempitsky, V. 2017. Escape from cells: Deep kd-networks for the recognition of 3D point cloud models. In *Proceedings of the IEEE International Conference on Computer Vision*, 863–872. IEEE.

Li, Y.; Bu, R.; Sun, M.; and Chen, B. 2018. PointCNN: Convolution On X-Transformed Points. *arXiv preprint arXiv:1801.07791*.

Li, J.; Chen, B. M.; and Lee, G. H. 2018. SO-Net: Self-Organizing Network for Point Cloud Analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9397–9406.

Maturana, D., and Scherer, S. 2015. Voxnet: A 3D convolutional neural network for real-time object recognition. In *IEEE International Conference on Intelligent Robots and Systems*, 922–928. IEEE.

Osada, R.; Funkhouser, T.; Chazelle, B.; and Dobkin, D. 2002. Shape distributions. *ACM Transactions on Graphics* 21(4):807–832.

Qi, C. R.; Su, H.; Nießner, M.; Dai, A.; Yan, M.; and Guibas, L. J. 2016. Volumetric and multi-view cnns for object classification on 3D data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5648–5656.

Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 77–85.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, 5105–5114.

Shi, B.; Bai, S.; Zhou, Z.; and Bai, X. 2015. Deeppano: Deep panoramic representation for 3-d shape recognition. *IEEE Signal Processing Letters* 22(12):2339–2343.

Su, H.; Maji, S.; Kalogerakis, E.; and Learned-Miller, E. 2015. Multi-view convolutional neural networks for 3D shape recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 945–953.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2018. Dynamic Graph CNN for Learning on Point Clouds. *arXiv preprint arXiv:1801.07829*.

Wen, Y.; Zhang, K.; Li, Z.; and Qiao, Y. 2016. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, 499–515. Springer.

Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; and Xiao, J. 2015. 3D shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1912–1920.

Xie, J.; Zhu, F.; Dai, G.; Shao, L.; and Fang, Y. 2017. Progressive shape-distribution-encoder for learning 3D shape representation. *IEEE Transactions on Image Processing* 26(3):1231–1242.