

# 3DViewGraph: Learning Global Features for 3D Shapes from A Graph of Unordered Views with Attention

Zhizhong Han<sup>1,3</sup>, Xiyang Wang<sup>1,2</sup>, Chi-Man Vong<sup>4</sup>, Yu-Shen Liu<sup>1,2\*</sup>, Matthias Zwicker<sup>3</sup> and C.L. Philip Chen<sup>5</sup>

<sup>1</sup>School of Software, Tsinghua University, Beijing, China

<sup>2</sup>Beijing National Research Center for Information Science and Technology (BNRist)

<sup>3</sup>Department of Computer Science, University of Maryland, College Park, USA

<sup>4</sup>Department of Computer and Information Science, University of Macau, Macau, China

<sup>5</sup>Faculty of Science and Technology, University of Macau, Macau, China

h312h@umd.edu, wangxiya16@mails.tsinghua.edu.cn, cmvong@um.edu.mo,  
liuyushen@tsinghua.edu.cn, zwicker@cs.umd.edu, philip.chen@ieee.org

## Abstract

Learning global features by aggregating information over multiple views has been shown to be effective for 3D shape analysis. For view aggregation in deep learning models, pooling has been applied extensively. However, pooling leads to a loss of the content within views, and the spatial relationship among views, which limits the discriminability of learned features. We propose *3DViewGraph* to resolve this issue, which learns 3D global features by more effectively aggregating unordered views with attention. Specifically, unordered views taken around a shape are regarded as view nodes on a view graph. *3DViewGraph* first learns a novel latent semantic mapping to project low-level view features into meaningful latent semantic embeddings in a lower dimensional space, which is spanned by latent semantic patterns. Then, the content and spatial information of each pair of view nodes are encoded by a novel spatial pattern correlation, where the correlation is computed among latent semantic patterns. Finally, all spatial pattern correlations are integrated with attention weights learned by a novel attention mechanism. This further increases the discriminability of learned features by highlighting the unordered view nodes with distinctive characteristics and depressing the ones with appearance ambiguity. We show that *3DViewGraph* outperforms state-of-the-art methods under three large-scale benchmarks.

## 1 Introduction

Global features of 3D shapes can be learned from raw 3D representations, such as meshes, voxels, and point clouds. As an alternative, a number of works in 3D shape analysis employed multiple views [Su and others, 2015; Han *et al.*, 2019b] as raw 3D representation, exploiting the advantage that multiple

views can facilitate understanding of both manifold and non-manifold 3D shapes via computer vision techniques. Therefore, effectively and efficiently aggregating comprehensive information over multiple views, is critical for the discriminability of learned features, especially in deep learning models.

Pooling was designed as a procedure for information abstraction in deep learning models. In order to describe a 3D shape by considering features from multiple views, view aggregation is usually performed by max or mean pooling, where pooling only employs the max or mean value of each dimension across all view features [Su and others, 2015]. Although pooling is able to eliminate the rotation effect of 3D shapes, both the content information within views and the spatial relationship among views cannot be fully preserved. As a consequence, this limits the discriminability of learned features. In this work, we address the challenge to learn 3D features in a deep learning model by more effectively aggregating the content information within individual views, and the spatial relationship among multiple unordered views.

To tackle this issue, we propose a novel deep learning model called *3D View Graph* (*3DViewGraph*), which learns 3D global features from multiple unordered views. By taking multiple views around a 3D shape on a unit sphere, we represent the shape as a view graph formed by the views, where each view denotes a node, and the nodes are fully connected with each other by edges. *3DViewGraph* learns highly discriminative global 3D shape features by simultaneously encoding both the content information within the view nodes, and the spatial relationship among the view nodes.

- i) We propose a novel deep learning model called *3DViewGraph* for 3D global feature learning by effectively aggregating multiple unordered views. It not only encodes the content information within all views, but also preserves the spatial relationship among the views.
- ii) We propose an approach to learn a low-dimensional latent semantic embedding of the views by directly capturing the similarities between each view and a set of latent semantic patterns. As an advantage, *3DViewGraph*

\*Corresponding Author

avoids mining the latent semantic patterns across the whole training set explicitly.

- iii) We perform view aggregation by integrating a novel spatial pattern correlation, which encodes the content information and the spatial relationship in each pair of views.
- iv) We propose a novel attention mechanism to increase the discriminability of learned features by highlighting the unordered view nodes with distinctive characteristics and depressing the ones with appearance ambiguities.

## 2 Related Work

Deep learning models have made a big progress on learning 3D shape features from different raw representations, such as meshes [Han and others, 2018], voxels [Wu and others, 2016], point clouds [Qi and others, 2017] and views [Su and others, 2015]. Because of page limit, we focus on reviewing view-based deep learning models to highlight the novelty of our view aggregation.

### 2.1 View-based Methods

View-based methods represent a 3D shape as a set of rendered views [Kanezaki *et al.*, 2018] or panorama views [Sfikas and others, 2017]. Besides direct set-to-set comparison [Bai and others, 2017], pooling is the widely used way of aggregating multiple views in deep learning models [Su and others, 2015]. In addition to global feature learning, pooling can also be used to learn local features [Huang *et al.*, 2017; Yu *et al.*, 2018] for segmentation or correspondence by aggregating local patches.

Although pooling can aggregate views on the fly in the models, it can not encode all the content information within views and the spatial relationship among views. Thus, the strategies of concatenation [Savva and others, 2016], view pair weighting [Johns *et al.*, 2016], cluster specified pooling [Wang and others, 2017], RNN [Han and others, 2019], were employed to resolve this issue. However, these methods can not learn from unordered views or fully capture the spatial information among unordered views.

To resolve the aforementioned issues, 3DViewGraph aggregates unordered views more effectively by simultaneously encoding their content information and spatial relationship.

### 2.2 Graph-based Methods

To handle the irregular structure of graphs, various methods have been proposed [Hamilton and others, 2017]. Although we formulate the multiple views from a 3D shape as a view graph, existing methods proposed for graphs cannot be directly used for learning the 3D feature in our scenario. The reasons are two-fold. First, these methods mainly focus on how to locally learn meaningful representation for each node in a graph from its raw attributes rather than globally learning the feature of the whole graph. Second, these methods mainly learn how to process the nodes in a graph with firm order, while the order of views involved in 3DViewGraph are always ambiguous because of the rotation of 3D shapes.

Moreover, some methods have employed graphs to retrieve 3D shapes from multiple views [Anan *et al.*, 2015;

An-An *et al.*, 2016]. Different from these methods, 3DViewGraph employs a more efficient way of view aggregation in deep learning models, which makes the learned features useful for both classification and retrieval.

## 3 3DViewGraph

### 3.1 Overview

Fig. 1 shows an overview of 3DViewGraph, where the global feature  $F^i \in \mathbb{R}^{1 \times F}$  of a 3D shape  $m^i$  is learned from its corresponding view graph  $G^i$ . Here,  $m^i$  is the  $i$ -th shape in a training set of  $M$  3D shapes, where  $i \in [1, M]$ . Based on the  $F$ -dimensional feature  $F^i$ , 3DViewGraph classifies  $m^i$  into one of  $L$  shape classes according to the probability  $P^i = [P^i(l^i = 1|F^i), \dots, P^i(l^i = a|F^i), \dots, P^i(l^i = L|F^i)]$ , which is provided by a final softmax classifier (Fig. 1(f)), where  $l^i$  is the class label of  $m^i$ .

We first take a set of unordered views  $v^i = \{v_j^i | j \in [1, V]\}$  on a unit sphere centered at  $m^i$ , as shown in Fig. 1(a). Here, we use “unordered views” to indicate that the views cannot be organized in a sequential way. The views  $v_j^i$  are regarded as view nodes  $D_j^i$  (briefly shown by symbols) of an undirected graph  $G^i$ , where each  $D_j^i$  is fully connected with other view nodes  $D_{j'}^i$ , by edges  $E_{j,j'}^i$ , such that  $G^i = (\{D_j^i\}, \{E_{j,j'}^i\})$ .

Next, we extract low-level features  $f_j^i$  of each view  $v_j^i$  using a fine-tuned VGG19 network [Simonyan and Zisserman, 2014], as shown in Fig. 1(b), where  $f_j^i \in \mathbb{R}^{1 \times 4096}$  is extracted from the last fully connected layer. To obtain lower-dimensional, semantically more meaningful view features, we subsequently learn a latent semantic mapping  $\Phi$  (Fig. 1(c)) to project a low-level view feature  $f_j^i$  into its latent semantic embedding  $d_j^i$ .

To resolve the effect of rotation, 3DViewGraph encodes the content and spatial information of  $G^i$  by exhaustively computing our novel spatial pattern correlation between each pair of view nodes. As illustrated in Fig. 1(d), we compute the pattern correlation  $c_{j,j'}^i$  between  $D_j^i$  and each other node  $D_{j'}^i$ , and we weight it with their spatial similarity  $s_{j,j'}^i$ . In addition, for each node  $D_j^i$ , we compute its cumulative correlation  $C_j^i$  to summarize all spatial pattern correlations as the characteristics of the 3D shape from the  $j$ -th view node  $D_j^i$ .

Finally, we obtain the global feature  $F^i$  of shape  $m^i$  by integrating all cumulative correlations  $C_j^i$  with our novel attention weights  $\alpha^i$ , as shown in Fig. 1(e) and (f).  $\alpha^i$  aims to highlight the view nodes with distinctive characteristics while depressing the ones with appearance ambiguity.

### 3.2 Latent Semantic Mapping Learning

To learn global features from unordered views, 3DViewGraph encodes the content information within all views and the spatial relationship among views in a pairwise way. 3DViewGraph relies on the intuition that correlations between pairs of views can effectively represent discriminative characteristics of a 3D shape, especially considering the relative spatial position of the views. To implement this intuition, each view should be encoded in terms of a small set of common elements across all views in the training set. Unfortunately,

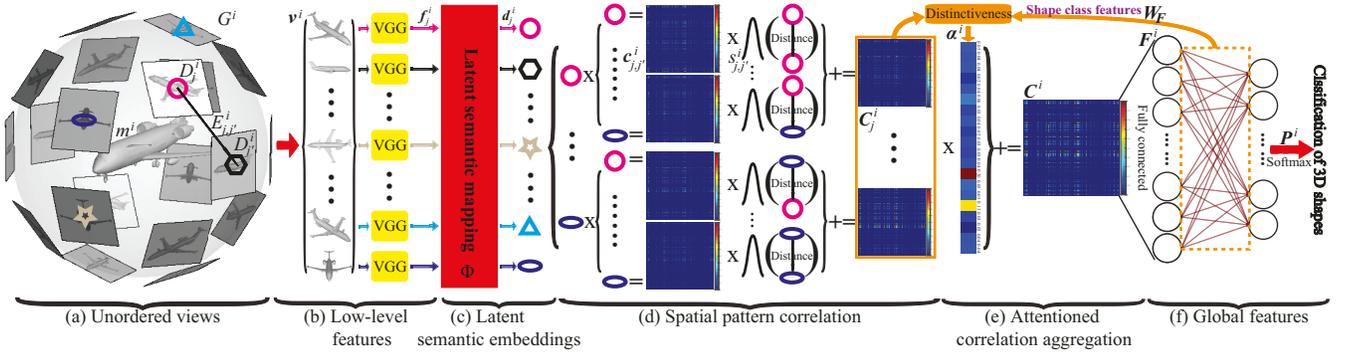
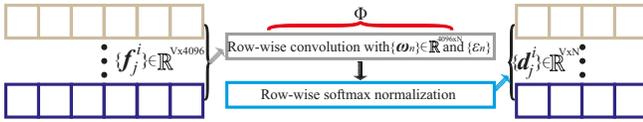


Figure 1: The demonstration of 3DViewGraph framework.

the low-level features  $\mathbf{f}_j^i$  are too high dimensional and not suitable as a representation of the views in terms of a set of common elements.


 Figure 2: The demonstration of latent semantic mapping  $\Phi$ .

To resolve this issue, 3DViewGraph introduces a latent semantic mapping  $\Phi$  by learning a kernel  $K$  to directly capture the similarities between  $V$  views  $v_j^i$  and  $N$  latent semantic patterns  $\{\phi_n\}$ . Our approach avoids additionally and explicitly mining  $\{\phi_n\}$  across the whole training set as the common elements.  $\Phi$  projects low-level view features  $\mathbf{f}_j^i$  into latent semantic space spanned by  $\{\phi_n\}$  as latent semantic embeddings  $\mathbf{d}_j^i$ .  $\mathbf{d}_j^i$  represents view nodes  $D_j^i$  with more semantic meaning but lower dimension than  $\mathbf{f}_j^i$ . Specifically, predicted by kernel  $K$ , the  $n$ -th dimension of  $\mathbf{d}_j^i$  characterizes the similarity between  $\mathbf{f}_j^i$  and the  $n$ -th semantic pattern  $\phi_n$ , such that  $\mathbf{d}_j^i = [K(\mathbf{f}_j^i, \phi_1), \dots, K(\mathbf{f}_j^i, \phi_n), \dots, K(\mathbf{f}_j^i, \phi_N)] \in \mathbb{R}^{1 \times N}$ . We define the kernel  $K$  as

$$K(\mathbf{f}_j^i, \phi_n) = \frac{\exp(-\beta \|\mathbf{f}_j^i - \phi_n\|_2^2)}{\sum_{n'=1}^N \exp(-\beta \|\mathbf{f}_j^i - \phi_{n'}\|_2^2)}, \quad (1)$$

where the similarity  $K(\mathbf{f}_j^i, \phi_n)$  is inversely proportional to the distance between  $\mathbf{f}_j^i$  and  $\phi_n$  through  $\exp()$ , and gets normalized across the similarities between  $\mathbf{f}_j^i$  and all  $\phi_n$ . Parameter  $\beta$  controls the decay of the response with the distance. This equation can be further simplified by cancelling the nor-

m of  $\mathbf{f}_j^i$  from the numerator and the denominator as follows,

$$\begin{aligned} K(\mathbf{f}_j^i, \phi_n) &= \frac{\exp(-\beta \|\mathbf{f}_j^i\|_2^2 + 2\beta \mathbf{f}_j^i \phi_n^T - \beta \|\phi_n\|_2^2)}{\sum_{n'=1}^N \exp(-\beta \|\mathbf{f}_j^i\|_2^2 + 2\beta \mathbf{f}_j^i \phi_{n'}^T - \beta \|\phi_{n'}\|_2^2)} \\ &= \frac{\exp(2\beta \mathbf{f}_j^i \phi_n^T - \beta \|\phi_n\|_2^2)}{\sum_{n'=1}^N \exp(2\beta \mathbf{f}_j^i \phi_{n'}^T - \beta \|\phi_{n'}\|_2^2)} \\ &= \frac{\exp(\mathbf{f}_j^i \omega_n + \varepsilon_n)}{\sum_{n'=1}^N \exp(\mathbf{f}_j^i \omega_{n'} + \varepsilon_{n'})}, \end{aligned} \quad (2)$$

where in the last step, we substituted  $2\beta \phi_n^T$  and  $-\beta \|\phi_n\|_2^2$  by  $\omega_n$  and  $\varepsilon_n$ , respectively. Here,  $\{\omega_n\}$ ,  $\{\varepsilon_n\}$  and  $\{\phi_n\}$  are sets of learnable parameters, in addition, both  $\{\omega_n\}$  and  $\{\varepsilon_n\}$  depend on  $\{\phi_n\}$ . However, to obtain more flexible training by following the viewpoint in [Arandjelovic and others, 2016], we employ two independent sets of  $\{\omega_n\}$ ,  $\{\varepsilon_n\}$ , decoupling  $\{\omega_n\}$  and  $\{\varepsilon_n\}$  from  $\{\phi_n\}$ . This decoupling enables 3D-ViewGraph to directly predict the similarity between  $\mathbf{f}_j^i$  and  $\phi_n$  by the kernel  $K$  without explicitly mining  $\phi_n$  across all low-level view features in the training set.

Based on the last line in Eq. 2, we implement the latent semantic mapping  $\Phi$  as a row-wise convolution with each pair of  $\{\omega_n\}$  and  $\{\varepsilon_n\}$  corresponding to a filter and a row-wise softmax normalization, as shown in Fig. 2.

### 3.3 Spatial Pattern Correlation

The pattern correlation  $\mathbf{c}_{j,j'}^i$  aims to encode the content of view nodes  $D_j^i$  and  $D_{j'}^i$ .  $\mathbf{c}_{j,j'}^i$  makes the semantic patterns that co-occur in both views more prominent while the non-co-occurring ones more subtle. More precisely, we use the latent semantic embeddings  $\mathbf{d}_j^i$  and  $\mathbf{d}_{j'}^i$  to compute  $\mathbf{c}_{j,j'}^i$  as follows,

$$\mathbf{c}_{j,j'}^i = (\mathbf{d}_j^i)^T \times \mathbf{d}_{j'}^i, \quad (3)$$

where  $\mathbf{c}_{j,j'}^i$  is a  $N \times N$  dimensional matrix whose entry  $\mathbf{c}_{j,j'}^i(n, n')$  measures the correlation between the semantic pattern  $\phi_n$  contributing to  $\mathbf{d}_j^i$  and  $\phi_{n'}$  contributing to  $\mathbf{d}_{j'}^i$ .

We further enhance the pattern correlation  $\mathbf{c}_{j,j'}^i$  between the view nodes  $D_j^i$  and  $D_{j'}^i$  by their spatial similarity  $s_{j,j'}^i$ , which forms the spatial pattern correlation  $s_{j,j'}^i \mathbf{c}_{j,j'}^i$ .

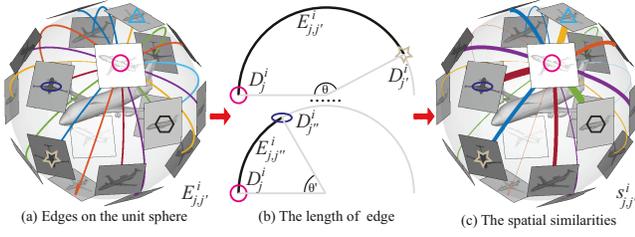

 Figure 3: The illustration of spatial similarity  $s_{j,j'}^i$ .

Fig. 3 visualizes how we compute the spatial similarity  $s_{j,j'}^i$ . In Fig. 3(a), we show all edges  $E_{j,j'}^i$  connecting  $D_j^i$  to all other view nodes  $D_{j'}^i$ , in different colors, where  $D_j^i$  is briefly shown by symbols. The length of  $E_{j,j'}^i$  is measured by the length of the shortest arc connecting the two view nodes  $D_j^i$  and  $D_{j'}^i$  on the unit sphere. Thus,  $E_{j,j'}^i = 2\pi \times 1 \times (\theta/2\pi) = \theta$  as illustrated in Fig. 3(b), where  $\theta$  is the central angle of the arc and the factor 1 corresponds to the radius of the unit sphere. To reduce the high variance of  $\{E_{j,j'}^i\}$ , we employ  $E_{j,j'}^i = 0.5(1 - \cos \theta)$  instead of  $E_{j,j'}^i = \theta$ , which normalizes  $E_{j,j'}^i$  into the range of  $[0, 1]$ . Finally,  $s_{j,j'}^i$  is inversely proportional to  $E_{j,j'}^i$  as follows,

$$s_{j,j'}^i = \exp(-\sigma E_{j,j'}^i), \quad (4)$$

where  $\sigma$  is a parameter to control the decay of the response with the edge length. In Fig. 3(c), we visualize  $s_{j,j'}^i$  by mapping the value of  $s_{j,j'}^i$  to the width of edges  $E_{j,j'}^i$ .

To represent the characteristics of 3D shape  $m^i$  from the  $j$ -th view node  $D_j^i$  on  $G^i$ , we finally introduce the cumulative correlation  $C_j^i$ , which encodes all spatial pattern correlations starting from  $D_j^i$  as follows,

$$C_j^i = \sum_{j'=1}^V s_{j,j'}^i C_{j,j'}^i. \quad (5)$$

### 3.4 Attentioned Correlation Aggregation

Intuitively, more views will provide more information to any deep learning model, which should allow it to produce more discriminative 3D features. However, additional views may also introduce appearance ambiguities that negatively affect the discriminability of learned features, as shown in Fig. 4.

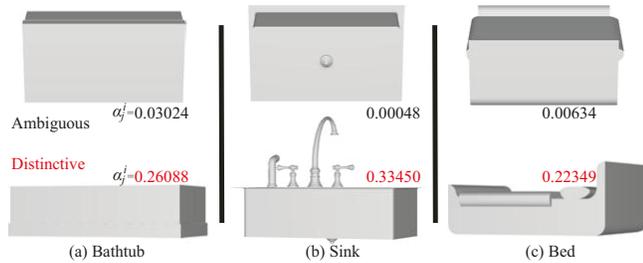


Figure 4: The ambiguous views and distinctive views.

To resolve this issue, 3DViewGraph employs a novel attention mechanism in the aggregation of the 3D shape characteristics from all unordered view nodes of a shape, as illustrated in Fig. 1(e). 3DViewGraph learns attention weights  $\alpha^i = \{\alpha_j^i | j \in [1, V]\}$  for all view nodes  $D_j^i$  on  $G^i$ , where  $\alpha_j^i$  would be a large value (the second row in Fig. 4) if the view  $v_j^i$  has distinctive characteristics, while  $\alpha_j^i$  would be a small value (the first row in Fig. 4) if  $v_j^i$  exhibits appearance ambiguity with views from other shapes. Note that  $\sum_{j=1}^V \alpha_j^i = 1$ .

Our novel attention mechanism evaluates how distinctive each view is to the views that 3DViewGraph has processed. To comprehensively represent the characteristics of the views that 3DViewGraph has processed, the attention mechanism employs the fully connected weights  $W_F$  in the final softmax classifier which accumulates the information of all views, as shown in Fig. 1(f). The attention mechanism projects the characteristics  $C_j^i$  of 3D shape  $m^i$  from the  $j$ -th view node  $D_j^i$  and the characteristics  $W_F$  of the views that 3DViewGraph has processed into a common space to calculate the distinctiveness of view  $v_j^i$ , as defined below,

$$\begin{aligned} \alpha_j^i &= \omega(W_C C_j^i \omega_C + W_F \omega_F + \mathbf{b}), \\ \alpha^i &= \text{softmax}(\alpha^i), \end{aligned} \quad (6)$$

where  $W_C$ ,  $\omega_C$ ,  $\omega_F$ ,  $\mathbf{b}$  and  $\omega$  are learnable parameters in the attention mechanism,  $W_F \in \mathbb{R}^{L \times F}$ , where  $F$  is the dimension of the learned global feature  $F^i$ , and  $L$  is the number of shape classes. With  $W_C \in \mathbb{R}^{L \times V}$  and  $\omega_C \in \mathbb{R}^{V \times 1}$ ,  $C_j^i$  is projected into a  $L \times 1$  dimensional space, where  $\mathbf{b} \in \mathbb{R}^{L \times 1}$  is a bias in that space. In addition,  $W_F$  is projected into the same space by  $\omega_F \in \mathbb{R}^{K \times 1}$  to compute the similarities between  $C_j^i$  and  $W_F$  along all  $L$  dimensions. Subsequently, the attention weight  $\alpha_j^i$  is calculated by comprehensively summarizing all similarities along all the  $L$  dimensions with a linear mapping  $\omega \in \mathbb{R}^{1 \times L}$ . Finally, the  $\alpha_j^i$  in  $\alpha^i$  for all views of  $i$ -th shape are normalized by softmax normalization.

Based on  $\alpha^i$ , the characteristics  $C_j^i$  of 3D shape  $m^i$  from all view nodes are aggregated with weighting  $\alpha^i$  into attentioned correlation aggregation  $C^i$ , as defined below,

$$C^i = \sum_{j=1}^V \alpha_j^i C_j^i, \quad (7)$$

where  $C^i$  represents 3D shape  $m^i$  as a  $N \times N$  matrix, as shown in Fig. 1(e). Finally, the global feature  $F^i$  of 3D shape  $m^i$  is learned by a fully connected layer with attentioned correlation aggregation  $C^i$  as input, as shown in Fig. 1(f), where the fully connected layer is followed by a sigmoid function.

Using  $F^i$ , the final softmax classifier computes the probabilities  $P^i$  to classify the 3D shape  $m^i$  into one of  $L$  shape classes as

$$P^i = \text{softmax}(W_F F^i + \mathbf{b}_F), \quad (8)$$

where  $W_F \in \mathbb{R}^{L \times K}$  and  $\mathbf{b}_F \in \mathbb{R}^{K \times 1}$  are learnable parameters for the computation of  $P^i$ .  $W_F$  is used to represent all the characteristics of views that 3DViewGraph has processed, as employed to calculate  $\alpha^i$  in Eq. 6.

### 3.5 Learning Inference

The parameters involved in 3DViewGraph are optimized by minimizing the log-likelihood  $O$  over  $M$  3D shapes in the training set, where  $Q^i$  is the truth label,

$$O = -\frac{1}{M} \sum_{i=1}^M \sum_{a=1}^L Q^i(l^i = a) \log P^i(l^i = a). \quad (9)$$

The parameter optimization is conducted by back propagation of classification errors of 3D shapes. Noteworthy,  $\mathbf{W}_F$  is updated by two elements with the learning rate  $\varepsilon$  as follows,

$$\mathbf{W}_F \leftarrow \mathbf{W}_F - \varepsilon \left( \frac{\partial O}{\partial \mathbf{W}_F} + \sum_{j=1}^V \frac{\partial \alpha_j^i}{\partial \mathbf{W}_F} \right). \quad (10)$$

The advantage of Eq. (10) is that  $\mathbf{W}_F$  can be learned more flexibly for optimization convergence.  $\mathbf{W}_F$  also enables  $\alpha^i$  to simultaneously observe the characteristics of shape  $m^i$  from each view node  $D_j^i$  and take all views that have been processed from different shapes as reference.

## 4 Results and Analysis

We evaluate 3DViewGraph by comparing it with the state-of-the-art methods in shape classification and retrieval under ModelNet40 [Wu and others, 2015], ModelNet10 and ShapeNetCore55 [Savva and others, 2017]. We also show ablation studies to justify the effectiveness of novel elements.

$F$	64	128	256	512	1024
Acc %	93.44	93.03	<b>93.80</b>	93.07	93.19

Table 1:  $F$  comparison,  $\varepsilon = 0.009$ ,  $\sigma = 10$ ,  $N = 128$ .

### 4.1 Parameters

We first explore how the important parameters  $F$ ,  $N$  and  $\sigma$  affect the performance of 3DViewGraph under ModelNet40. The comparison in Table. 1, 2, and 3 shows that their effects are slight in a proper range.

$N$	32	64	128	256	512
Acc %	90.84	92.91	<b>93.80</b>	93.44	93.40

Table 2:  $N$  comparison,  $\varepsilon = 0.009$ ,  $\sigma = 10$ ,  $F = 256$ .

### 4.2 Classification

As compared under ModelNet in Table 4, 3DViewGraph outperforms all the other methods under the same condition<sup>1</sup>. In addition, we show the single view classification accuracy in VGG fine-tuning (“VGG(ModelNet)”). To highlight the contribution of VGG fine-tuning, spatial similarity, and attention, we remove fine-tuning (“Ours(No finetune)”) or set all spatial

<sup>1</sup>We use the same modality of views from the same camera system for the comparison, where the results of RotationNet are from Fig.4 (d) and (e) in <https://arxiv.org/pdf/1603.06208.pdf>. Moreover, the benchmarks are with the standard training and test split.

similarity (“Ours(No spatiality)”) and attention (“Ours(No attention)”) to 1. The degenerated results indicate these elements are important for 3DViewGraph to achieve high accuracy. Similar phenomena is observed when we justify the effect of  $C_j^i$  and  $\mathbf{W}_F$  in Eq. 6 by setting them to 1 (“Ours(No attention- $C_j^i$ )”), respectively. We also justify the latent semantic embedding and spatial pattern correlation by replacing them by single view features (“Ours(No latent)”) and summation (“Ours(No correlation)”), the degenerated results also show that they are important elements. Finally, we compare our proposed view aggregation with mean (“Ours(MeanPool)”) and max pooling (“Ours(MaxPool)”) by directly pooling all single view features together. Due to the loss of content information in each view and spatial information among multiple views, pooling performs worse.

$\sigma$	0	1	5	10	11
Acc %	92.91	93.48	93.72	<b>93.80</b>	93.48

Table 3:  $\sigma$  comparison,  $\varepsilon = 0.009$ ,  $N = 128$ ,  $F = 256$ .

Methods	MN40(%)	MN10(%)
3DGAN[Wu and others, 2016]	83.3	91.0
PointNet++[Qi and others, 2017]	91.9	-
FoldingNet[Yang <i>et al.</i> , 2018]	88.4	94.4
PANO[Sfikas and others, 2017]	90.7	91.1
Pairwise[Johns <i>et al.</i> , 2016]	90.7	92.8
GIFT[Bai and others, 2017]	89.5	91.5
Domi[Wang and others, 2017]	92.2	-
MVCNN[Su and others, 2015]	90.1	-
Spherical[Cao <i>et al.</i> , 2017]	93.31	-
Rotation[Kanezaki <i>et al.</i> , 2018]	92.37	94.39
SO-Net[Li and others, 2018]	90.9	94.1
SVSL[Han and others, 2019]	93.31	94.82
VIPGAN[Han <i>et al.</i> , 2019a]	91.98	94.05
VGG(ModelNet40)	87.27	-
VGG(ModelNet10)	-	88.63
Ours	<b>93.80</b>	<b>94.82</b>
Ours( $\sigma = 5$ )	<b>93.72</b>	<b>95.04</b>
Ours(No finetune)	90.40	-
Ours(No spatiality)	92.91	94.16
Ours(No attention)	93.07	93.72
Ours(No attention- $C_j^i$ )	91.82	93.39
Ours(No attention- $\mathbf{W}_F$ )	91.57	93.28
Ours(No latent)	92.34	92.95
Ours(No correlation)	89.30	93.83
Ours(MeanPool)	92.38	93.06
Ours(MaxPool)	91.89	92.84

Table 4: Classification comparison under ModelNet with  $\varepsilon = 0.009$ ,  $\sigma = 10$ ,  $F = 256$ ,  $N = 128$ , unless noted otherwise.

3DViewGraph also achieves the best under the more challenging benchmark ShapeNetCore55, based on the fine-tuned VGG (“VGG(ShapeNetCore55)”), as shown in Table 5. We also find that different parameters do not significantly affect the performance, such as  $N$  and  $\sigma$ .

Methods	Views	Accuracy(%)
VIPGAN[Han <i>et al.</i> , 2019a]	12	82.97
SVSL[Han and others, 2019]	12	85.47
VGG(ShapeNetCore55)	1	81.33
Ours	20	<b>86.87</b>
Ours( $N = 256$ )	20	<b>86.36</b>
Ours( $\sigma = 5$ )	20	<b>86.56</b>
Ours( $\sigma = 5, N = 256$ )	20	<b>86.71</b>

Table 5: Classification comparison under ShapeNetCore55 with  $\epsilon = 0.009$ ,  $\sigma = 10$ ,  $F = 256$ ,  $N = 128$ , unless noted otherwise.

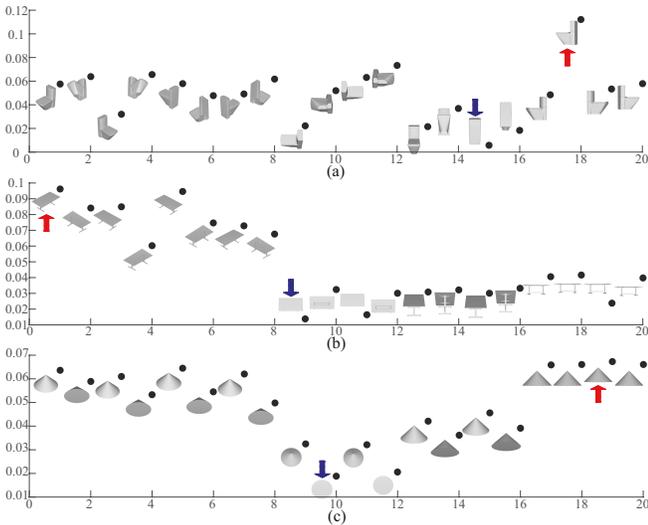


Figure 5: The visualization of attention weights (black nodes) learned for views of (a) a toilet, (b) a table, and (c) a cone. The highest and lowest attention weights are indicated by red upward arrow and blue downward arrow, respectively.

### 4.3 Attention Visualization

We visualize the attention learned by 3DViewGraph under ModelNet40, which demonstrates how 3DViewGraph understands 3D shapes by analyzing views on a view graph. In Fig. 5, attention weights  $\alpha^i$  on view nodes  $D_j^i$  of  $G^i$  are visualized as a vector which is represented by scattered black nodes, where the corresponding views are also shown nearby, such as the views of a toilet in Fig. 5(a), a table in Fig. 5(b) and a cone in Fig. 5(c). The coordinates of black nodes along the y-axis indicate how much attention 3DViewGraph pays to the corresponding view nodes. In addition, the views that is paid the most and least attention to are highlighted by the red upward and blue downward arrow, respectively.

Fig. 5 demonstrates that 3DViewGraph is able to understand each view, since the view with the most ambiguous appearance in a view graph is depressed while the view with the most distinctive appearance is highlighted. For example, the most ambiguous views of toilet, table and cone merely show some basic shapes that provide little useful information for classification, such as the rectangles of the toilet and table, and the circle of the cone. In contrast, the most distinctive views of toilet, table and cone exhibit more unique and distinctive characteristics.

Methods	Range	MN40	MN10
SHD	Test-Test	33.26	44.05
LFD	Test-Test	40.91	49.82
3DNets[Wu and others, 2015]	Test-Test	49.23	68.26
GImage[Sinha <i>et al.</i> , 2016]	Test-Test	51.30	74.90
DPano[Shi and others, 2015]	Test-Test	76.81	84.18
MVCNN[Su and others, 2015]	Test-Test	79.50	-
PANO[Sfikas and others, 2017]	Test-Test	83.45	87.39
GIFT[Bai and others, 2017]	Random	81.94	91.12
Triplet[He <i>et al.</i> , 2018]	Test-Test	88.0	-
Ours	Test-Test	<b>90.54</b>	<b>92.40</b>
Ours	Test-Train	<b>93.49</b>	<b>95.17</b>
Ours	Train-Train	<b>98.75</b>	<b>99.79</b>
Ours	All-All	<b>96.95</b>	<b>98.52</b>

Table 7: Retrieval comparison (mAP) under ModelNet,  $\epsilon = 0.009$ ,  $\sigma = 10$ ,  $F = 256$ ,  $N = 128$ .

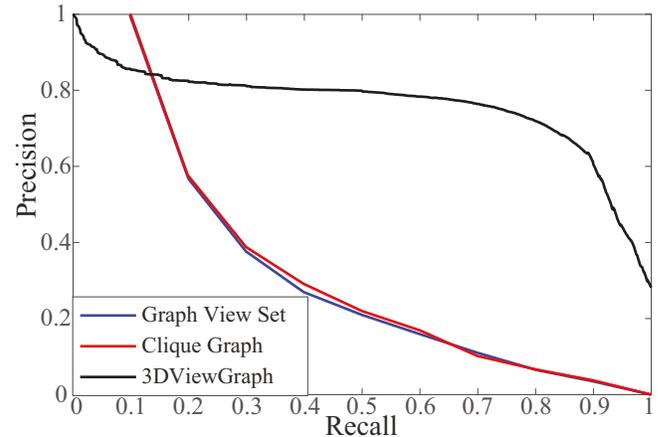


Figure 6: The precision and recall comparison with graph-based multi-view learning methods under PSB.

### 4.4 Retrieval

We evaluate the retrieval performance of 3DViewGraph under ModelNet in Table 7. We outperform the state-of-the-art methods, where the retrieval range is also shown. We further detail the precision and recall curves of these results in Fig. 7. In addition, 3DViewGraph also achieve the best results under ShapeNetCore55 in Table 6. We compare 10 state-of-the-art methods under testing set in the SHREC2017 retrieval contest [Savva and others, 2017] and Taco [Cohen *et al.*, 2018], where we summarize all the 10 methods (“All”) by presenting the best result of each metric due to page limit. Finally, we demonstrate that 3DViewGraph is also superior to other graph-based multi-view learning methods [Anan *et al.*, 2015; An-An *et al.*, 2016] under Princeton Shape Benchmark (PS-B) in Fig. 6.

## 5 Conclusion

In view-based deep learning models for 3D shape analysis, view aggregation via widely used pooling, leads to information loss about content and spatial relationship of views. We propose 3DViewGraph to address this issue for 3D global feature learning by more effectively aggregating unordered

Methods	micro					macro				
	P@N	R@N	F1@N	mAP@N	NDCG@N	P@N	R@N	F1@N	mAP@N	NDCG@N
All	<b>0.818</b>	0.803	<b>0.798</b>	0.772	0.865	<b>0.618</b>	0.667	<b>0.590</b>	0.583	0.657
Taco	0.701	0.711	0.699	0.676	0.756	-	-	-	-	-
Ours	0.6090	<b>0.8034</b>	0.6164	<b>0.8492</b>	<b>0.9054</b>	0.1929	<b>0.8301</b>	0.2446	<b>0.7019</b>	<b>0.8461</b>

Table 6: Retrieval comparison under ShapeNetCore55,  $\epsilon = 0.009$ ,  $\sigma = 10$ ,  $F = 256$ ,  $N = 128$ .

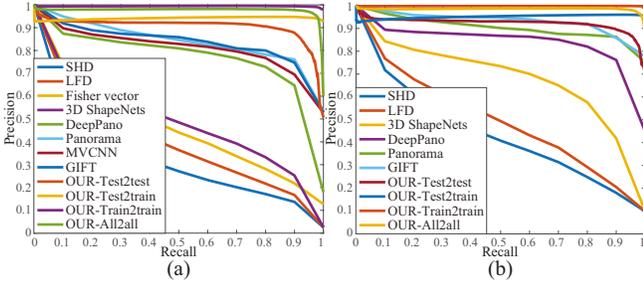


Figure 7: The precision and recall curve comparison among different methods under (a) ModelNet40 and (b) ModelNet10.

views with attention. By organizing unordered views taken around a 3D shape into a view graph, 3DViewGraph learns global features of the 3D shape by simultaneously encoding both the content information within view nodes and the spatial relationship among the view nodes. Through a novel latent semantic mapping, low-level view features are projected into a meaningful, lower-dimensional latent semantic embedding using a learned kernel function, which directly captures the similarities between low-level view features and latent semantic patterns. The latent semantic mapping successfully facilitates 3DViewGraph to encode the content information and the spatial relationship in each pair of view nodes using a novel spatial pattern correlation. Further, our novel attention mechanism effectively increases the discriminability of learned features by efficiently highlighting the unordered view nodes with distinctive characteristics and depressing the ones with appearance ambiguity. Our results in classification and retrieval under three large-scale benchmarks show that 3DViewGraph can learn better global features than the state-of-the-art methods due to its more effective view aggregation.

### Acknowledgments

This work was supported by National Key R&D Program of China (2018YFB0505400), NSF (1813583), University of Macau (MYRG2018-00138-FST), and FDCT (273/2017/A). We thank all anonymous reviewers for their constructive comments.

### References

[An-An *et al.*, 2016] Liu An-An, Nie Wei-Zhi, and Su Yu-Ting. Multi-modal clique-graph matching for view-based 3D model retrieval. *IEEE Transactions on Image Processing*, 25(5):2103–2115, 2016.

[Anan *et al.*, 2015] Liu Anan, Wang Zhongyang, Nie Weizhi, and Su Yuting. Graph-based characteristic view

set extraction and matching for 3D model retrieval. *Information Sciences*, 320:429–442, 2015.

[Arandjelovic and others, 2016] Relja Arandjelovic et al. NetVLAD: CNN architecture for weakly supervised place recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5297–5307, 2016.

[Bai and others, 2017] Song Bai et al. GIFT: Towards scalable 3D shape retrieval. *IEEE Transaction on Multimedia*, 19(6):1257–1271, 2017.

[Cao *et al.*, 2017] Zhangjie Cao, Qixing Huang, and Karthik Ramani. 3D object classification via spherical projections. In *International Conference on 3D Vision*. 2017.

[Cohen *et al.*, 2018] Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *International Conference on Learning Representations*, 2018.

[Hamilton and others, 2017] William L. Hamilton et al. Representation learning on graphs: Methods and applications. *IEEE Data Engineering Bulletin*, 40(3):52–74, 2017.

[Han and others, 2018] Zhizhong Han et al. Deep spatiality: Unsupervised learning of spatially-enhanced global and local 3D features by deep neural network with coupled softmax. *IEEE Transactions on Image Processing*, 27(6):3049–3063, 2018.

[Han and others, 2019] Zhizhong Han et al. Seqviews 2seqlabels: Learning 3D global features via aggregating sequential views by rnn with attention. *IEEE Transactions on Image Processing*, 28(2):1941–0042, 2019.

[Han *et al.*, 2019a] Zhizhong Han, Mingyang Shang, Yu-Shen Liu, and Matthias Zwicker. View inter-prediction gan: Unsupervised representation learning for 3D shapes by learning global shape memories to support local view predictions. In *AAAI*, 2019.

[Han *et al.*, 2019b] Zhizhong Han, Mingyang Shang, Xiyang Wang, Yu-Shen Liu, and Matthias Zwicker. Y2seq2seq: Cross-modal representation learning for 3D shape and text by joint reconstruction and prediction of view and word sequences. In *AAAI*, 2019.

[He *et al.*, 2018] Xinwei He, Yang Zhou, Zhichao Zhou, Song Bai, and Xiang Bai. Triplet-center loss for multi-view 3D object retrieval. In *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[Huang *et al.*, 2017] H. Huang, E. Kalerakakis, S. Chaudhuri, D. Ceylan, V. Kim, and E. Yumer. Learning local shape descriptors with view-based convolutional neural networks. *ACM Transactions on Graphics*, 2017.

- [Johns *et al.*, 2016] Edward Johns, Stefan Leutenegger, and Andrew J. Davison. Pairwise decomposition of image sequences for active multi-view recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3813–3822, 2016.
- [Kanezaki *et al.*, 2018] Asako Kanezaki, Yasuyuki Matsushita, and Yoshifumi Nishida. Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.
- [Li and others, 2018] Jiaxin Li et al. SO-Net: Self organizing network for point cloud analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018.
- [Qi and others, 2017] Charles Qi et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, pages 5105–5114, 2017.
- [Savva and others, 2016] M. Savva et al. Shrec’16 track large-scale 3D shape retrieval from shapeNet core55. In *EG 2016 workshop on 3D Object Recognition*, 2016.
- [Savva and others, 2017] Manolis Savva et al. SHREC’17 Large-Scale 3D Shape Retrieval from ShapeNet Core55. In *Eurographics Workshop on 3D Object Retrieval*, 2017.
- [Sfikas and others, 2017] Konstantinos Sfikas et al. Exploiting the PANORAMA Representation for Convolutional Neural Network Classification and Retrieval. In *EG Workshop on 3D Object Retrieval*, pages 1–7, 2017.
- [Shi and others, 2015] B. Shi et al. Deeppano: Deep panoramic representation for 3D shape recognition. *IEEE Signal Processing Letters*, 22(12):2339–2343, 2015.
- [Simonyan and Zisserman, 2014] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.
- [Sinha *et al.*, 2016] Ayan Sinha, Jing Bai, and Karthik Ramani. Deep learning 3D shape surfaces using geometry images. In *European Conference on Computer Vision*, pages 223–240, 2016.
- [Su and others, 2015] Hang Su et al. Multi-view convolutional neural networks for 3D shape recognition. In *International Conference on Computer Vision*, pages 945–953, 2015.
- [Wang and others, 2017] Chu Wang et al. Dominant set clustering and pooling for multi-view 3D object recognition. In *British Machine Vision Conference*, 2017.
- [Wu and others, 2015] Zhirong Wu et al. 3D ShapeNets: A deep representation for volumetric shapes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015.
- [Wu and others, 2016] Jiajun Wu et al. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In *Advances in Neural Information Processing Systems*, pages 82–90, 2016.
- [Yang *et al.*, 2018] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.
- [Yu *et al.*, 2018] Tan Yu, Jingjing Meng, and Junsong Yuan. Multi-view harmonized bilinear network for 3D object recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018.