SCA-PVNet: Self-and-Cross Attention Based Aggregation of Point Cloud and Multi-View for 3D Object Retrieval

Dongyun Lin[a,*], Yi Cheng[a], Aiyuan Guo[a], Shangbo Mao[a], Yiqun Li[a]

[a]*Institute for Infocomm Research, A*STAR, Singapore, 1 Fusionopolis Way, #21-01 Connexis (South Tower), Singapore 138632*

Abstract

To address 3D object retrieval, substantial efforts have been made to generate highly discriminative descriptors for 3D objects represented by a single modality, such as voxels, point clouds, or multiview images. It is promising to leverage complementary information from multimodal representations of 3D objects to further improve retrieval performance. However, multimodal 3D object retrieval has rarely been developed or analyzed for large-scale datasets. In this paper, we propose a self-and-cross-attention-based aggregation of point clouds and multiview images (SCA-PVNet) for 3D object retrieval. With deep features extracted from point clouds and multi-view images, we design two types of feature aggregation modules, namely the in-modality aggregation module (IMAM) and the cross-modality aggregation module (CMAM), for effective feature fusion. IMAM leverages a self-attention mechanism to aggregate multiview features, whereas CMAM exploits a cross-attention mechanism to interact with point-cloud and multiview features. The final descriptor of a 3D object for object retrieval can be obtained by concatenating the aggregated feature outputs of both modules. Extensive experiments and analyses were conducted on four datasets ranging from small to large scales to demonstrate the superiority of the proposed SCA-PVNet over state-of-the-art methods. In addition to achieving state-of-the-art retrieval performance, our method is more robust in challenging scenarios when views or points are missing during inference.

Keywords: Point cloud, multi-view, self-attention, cross-attention, and multimodal 3D object retrieval

1. Introduction

Recently, large amounts of 3D data have been produced in many applications, ranging from autonomous driving to 3D printing. To manage such large-scale 3D data, an effective 3D object- retrieval system capable of producing a ranked list of objects according to their semantic similarity to a given query object is required. In computer vision and multimedia communities, 3D object retrieval has drawn significant research interest by exploiting different modalities of 3D objects, such as voxels, point clouds, and multiview images. Existing methods for 3D object retrieval are primarily deep learning models designed based on a single modality, that is, unimodal methods. For example,

*Corresponding author: Tel.: +65-84342859;
Email address: lind@i2r.a-star.edu.sg (Dongyun Lin)

VoxNet [1], DGCNN [2], and MVCNN [3] are typical models based on voxels, point clouds, and multi-view representations of 3D objects, respectively. Among the unimodal methods, voxel-based approaches acquire descriptors of 3D objects by employing dense and regular grids formed by voxels. VoxNet [1] was constructed as a 3D convolutional neural network to generate the representations of the regular grids through cascaded 3D convolution operations and pooling layers. Although voxel-based methods excel in leveraging the entirety of the objects information, they often encounter challenges related to their high computational complexity. However , point-cloud-based methods obtain descriptors of 3D objects by utilizing a set of sampled 3D points extracted from the objects. PointNet [4] was formulated to construct a deep neural network for classifying 3D objects. As an extension, PointNet++ [5] was designed to exploit the hierarchical multiscale features of PointNet. 3DCapsule [6] introduced a capsule network into the PointNet architecture for point-cloud classification. The DGCNN [2] was constructed based on edge convolution operations to learn the representations of point clouds through dynamic graph updates and hierarchical feature learning. Point-cloud-based methods can also leverage almost the complete information of objects if the sampled points are dense and can generally reduce the computational complexity compared with voxel-based methods. Multi-view-based methods for learning 3D object descriptors involve a two-step process. First, they capture multi-view images of the 3D object from various angles. Second, they combine the deep features extracted from each individual view into a single, unified descriptor. The groundbreaking work, MVCNN [3], initiated this approach by extracting visual features from each view image and then pooling them to create a global representation of the 3D object. Subsequent works have built upon this foundation. For instance, [7] replaced the cross-entropy loss in MVCNN with a triplet center loss (TCL). [8] proposed an enhanced MVCNN that incorporates group-view similarity learning. [9] treated view images as a set of visual n-grams and generated rotation-invariant visual features based on these grams. MLVCNN [10] introduced a hierarchical view-loop-shaped architecture to learn 3D shape representations at multiple scales. As well-engineered 2D deep CNNs for discriminative feature extraction [11, 12] have advanced, multiview-based methods have achieved state-of-the-art performance in 3D object retrieval.

Based on the aforementioned analysis, multiview-based methods (e.g., MVCNN) only exploit the local and partial information of 3D objects via a limited number of projected view images, whereas point-cloud-based methods (e.g., DGCNN) can efficiently learn the global representations of 3D objects by exploiting the complete information of the objects. Hence, it is promising to leverage the complementary information of multimodal representations of 3D objects to generate more discriminative descriptors, thereby improving retrieval performance. Several methods have been proposed for multi-modal 3D object retrieval. These methods primarily focus on the effective fusion of point clouds and multi-view features. Point-view network (PVNet) [13] proposes an embedding attention fusion scheme to model the correlation and discriminability of different structures of point-cloud data by employing high-level features from multiview data. This network (PVRNet) [14] effectively fuses the view and point cloud features with a proposed relation score module. Both the multimodal information fusion network (MMFN) [15] and multimodal joint network (MMJN) [16] employ correlations between different modalities for robust descriptor fusion. The feature fusion schemes of the existing methods have two limitations: (i) they focus on cross-modality feature fusion but overlook

in-modality feature fusion across multiview images and (ii) they typically fuse the cross-modality feature via simple concatenation but overlook the interactions among the fused features. Moreover, although most existing studies were evaluated using only small-scale datasets such as ModelNet40, they have rarely been analyzed on large-scale datasets.

Motivated by these findings, we propose a self-and-cross attention-based aggregation of point clouds and multiview images (SCA-PVNet) for 3D object retrieval. It comprises two branches for the point-cloud modality and multiview modality inputs. Within the multiview modality branch, we further exploit two types of representations of multiview images at two hierarchical levels, namely, object- and view-level features. Object-level features are generated by fusing the intermediate CNN features for all view images, whereas view-level features are generated by average-pooling each of the intermediate CNN features for each view image. For both object- and view-level features, to achieve single-modality and cross-modality feature aggregation, we propose two types of feature aggregation modules, namely the in-modality aggregation module (IMAM) and the cross-modality aggregation module (CMAM). Both aggregation modules were proposed to alleviate the two limitations of the feature fusion schemes of existing methods. Specifically, IMAM leverages a self-attention mechanism to aggregate multiview features, whereas CMAM exploits a cross-attention mechanism to facilitate interaction between point cloud features and the aggregation process of multiview features. Finally, we concatenate the object- and view-level aggregated feature output by IMAM and CMAM and feed the concatenated feature through a fully connected layer to obtain the final descriptors for object retrieval.

Our motivation for jointly leveraging local and global information from multiview and point-cloud modalities is illustrated in Fig. 1. In this paper, self-and-cross attention is proposed as a systematic method for effective feature interaction and aggregation. First, the self-attention-based IMAM is proposed to (i) aggregate the local features extracted from the multiview representations into a global descriptor and (ii) update each of the local features with the knowledge provided by the other local features through self-attention-based interaction. Second, to facilitate effective multimodality feature fusion, CMAM is proposed to update the global features extracted from the point cloud modality with the information provided by the updated local features through cross-attention-based interaction.

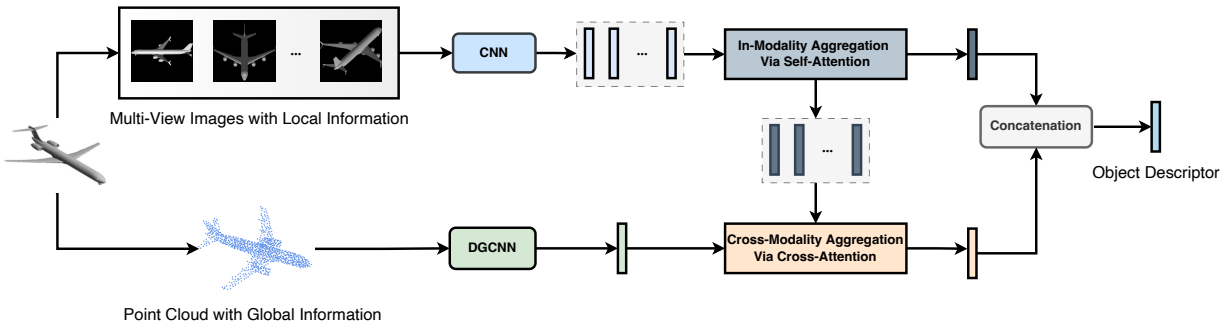

Figure 1: Illustration of our motivation for jointly leveraging the local and global information of the 3D object from its multi-view and point cloud representations via the proposed modules.

Relevant studies on multimodality 3D object retrieval and recognition, that is, PVNet [13], PVRNet [14], MMFN [15], MMJN [16] and attention-guided fusion networks [17], were only evaluated using small-scale datasets, that is, ModelNet10 and ModelNet40. In this paper, we surpass these existing works by establishing a new benchmark for multimodality 3D object retrieval performance. We evaluate our method using two larger 3D datasets: ShapeNetCore55 and MCB-A. To this end, we made substantial efforts to generate a point cloud and the corresponding multiview representations of these two large datasets. To facilitate further research on this topic, we released the generated data.

The major contributions of this paper are summarized as follows.

- We propose self-and-cross attention based aggregation of point clouds and multi-view images (SCA-PVNet) for 3D object retrieval. This network enhances the aggregation process of multi-view features under two hierarchical levels, namely object- and view-level features.
- We propose IMAM and CMAM to facilitate in-modality and cross-modality feature aggregation, respectively.
- We evaluate and analyze the proposed SCA-PVNet on four datasets (ModelNet40, ShapeNetCore55, ScanObjectNN, and MCB-A), ranging from small to large scale, to show the superiority of the proposed method over the state-of-the-art methods.

2. Related Work

In this section, we briefly review methods for 3D object retrieval, specifically focusing on unimodal and multimodal methods.

2.1. Uni-Modal Methods

Unimodel methods for learning 3D object descriptors can be categorized into three main approaches: voxel-based, point-cloud-based, and multi-view-based. Voxel-based methods utilize voxelized grids to represent 3D objects and employ 3D deep-learning models to learn their feature representations [1, 18, 19]. For instance, 3D ShapeNet [18] uses a deep belief network, while VoxNet [1] employs a 3D convolutional neural network for this purpose. Despite the advantage of using complete 3D object information, voxel-based methods often face significant computational complexity challenges. Point-cloud-based methods, on the other hand, learn object representations from uniformly sampled points on object surfaces [2, 4–6]. The seminal work, PointNet [4], constructs a 3D deep network for classifying 3D objects based on their point-cloud representations. PointNet++ [5] extends this by exploiting hierarchical multiscale features, while 3DCapsule [6] introduces a capsule network into the PointNet architecture. Point-cloud-based methods generally have lower computational complexity compared to voxel-based methods. Multi-view methods render multiple view images by projecting a 3D object from various viewpoints and learn descriptors by aggregating the visual features extracted from these images [3, 7–10, 20–32]. These methods leverage the advancements in 2D deep learning to extract discriminative features from the rendered view images. Mainstream studies represented by

MVCNN [3] and its variants [7–10, 20–23, 25, 27, 29, 30, 30, 31] applied well-engineered deep CNNs to extract features from view images and fuse these features into the descriptors of 3D objects. Another group of studies applied RNNs to aggregate the deep features extracted from each view image based on the assumption that the view images were a sequence of temporally correlated views [24, 26, 28]. Owing to the advancement of CNN/RNN networks, multiview-based methods have achieved state-of-the-art performance in 3D object retrieval. Even though unimodal methods exhibit such good performance, it is promising to explore and leverage complementary information from various data modalities to further improve retrieval performance. In this paper, we investigated the task of multimodal 3D object retrieval.

2.2. Multi-Modal Methods

Learning from multimodal data has significantly boosted the performance of computer vision models, such as vision-language models, where images and texts are employed for contrastive learning [33]. In remote sensing (RS), a growing number of studies [34, 35] have focused on leveraging multimodal RS data (e.g., hyperspectral, multispectral, and SAS) to train powerful AI models for remote sensing applications. Several methods have recently been proposed for multimodal 3D object retrieval. Because learning from voxelized dense grids is computationally expensive, most existing studies have focused on leveraging point clouds and multiview representations of 3D objects to learn discriminative descriptors for retrieval. PVNet [13] is a trailblazing initiative that combines point clouds and multiview data for 3D object recognition. It uses high-level features from multi-view data to analyze and discern the correlation and distinctiveness of structural features derived from point cloud data. PVRNet [14] was proposed to effectively fuse point clouds and multiview features using the proposed relation score module. MMJN [16] and MMFN [15] leverage the correlations among point clouds, multiple views, and panorama view modalities by introducing correlation loss. In [17], a novel attention-guided fusion network comprising point clouds and multiple views was proposed for 3D object recognition. In [36], a self-supervised cross-modal constructive-learning method called CrossPoint was proposed via enabling 3D-2D correspondence by maximizing the similarity between point clouds and the corresponding rendered 2D images. In particular, it introduced an intra-modal instance discrimination (IMID) loss to enforce the learning of geometric transformation-invariant representations of point clouds and a cross-modal instance discrimination (CMID) loss to enhance the similarity between the point cloud feature and deep CNN feature extracted for the rendered image. Compared with PVNet, PVRNet, MMJN, and MMFN, which focus on effectively fusing multimodal features, SCA-PVNet addresses the correlation and interaction of cross-modality features during the feature aggregation process. Additionally , most existing methods [13, 14, 16] were only evaluated using small-scale datasets such as ModelNet40 but rarely evaluated on large-scale datasets. In this paper, we evaluated the proposed SCA-PVNet on three datasets ranging from small to large scales to demonstrate the superiority of the proposed method. Compared with CrossPoint, which focuses on learning cross-modality correspondence representations via contrastive loss, our SCA-PVNet focuses on learning discriminative feature representations via classification loss. Our proposed IMAM and CMAM are different from IMID and CMID in CrossPoint. Specifically, we proposed a

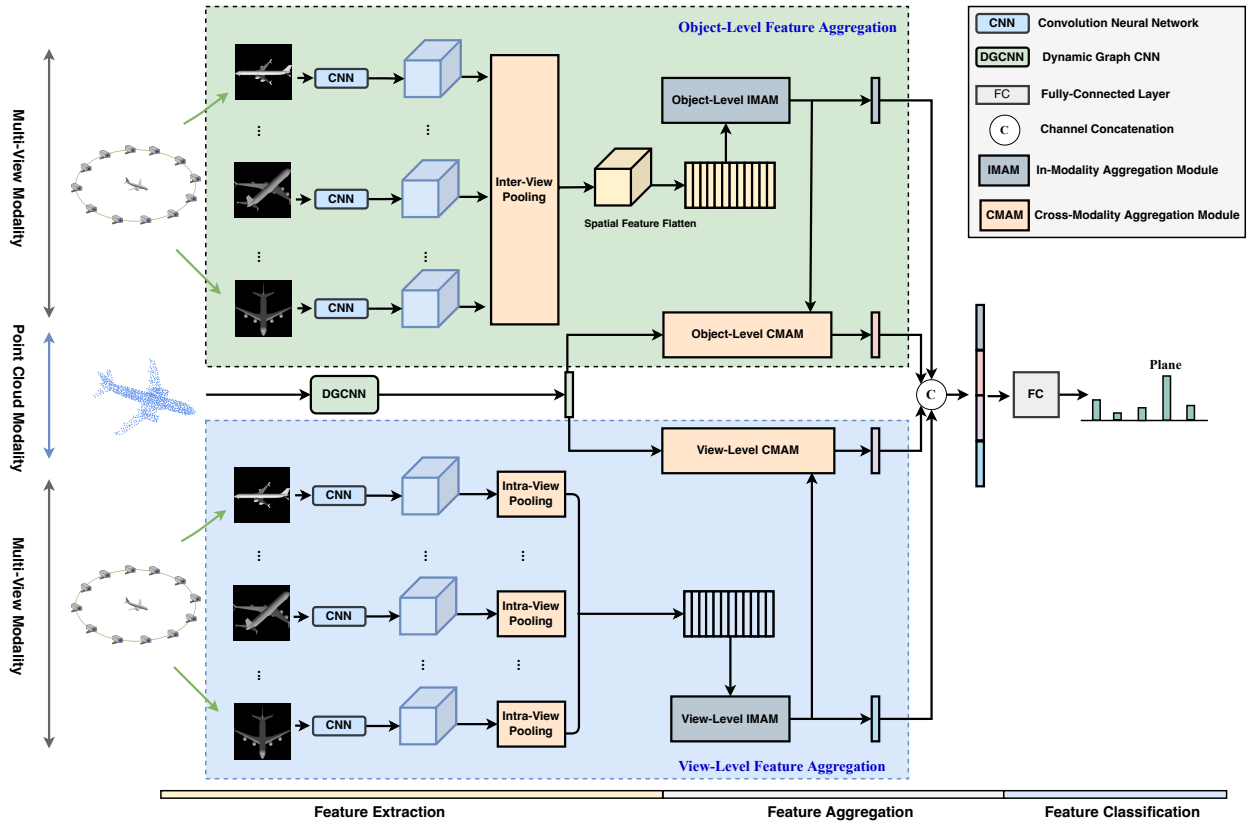

Figure 2: Overall architecture of the proposed SCA-PVNet which comprises three stages: (i) Feature Extraction; (ii) Feature Aggregation and (iii) Feature Classification.

self-and-cross attention mechanism to facilitate feature interaction and fusion within and across modalities instead of enforcing feature agreement across different modalities.

3. Proposed Method

In this section, we introduce the proposed network for multi-modal 3D object retrieval. The overall architecture of the proposed network is shown in Fig. 2. Generally, it comprises three stages: (i) feature extraction, (ii) feature aggregation, and (iii) feature classification. The network uses two types of modalities as inputs: point clouds and multi-view images). A given 3D object is first represented as a point cloud and a set of view images captured from multiple viewpoints. In the feature extraction stage, the point cloud is fed through a DGCNN [2] to obtain the point cloud feature, whereas the view images are fed into a CNN to generate the deep feature representations. For view feature extraction, we exploited two hierarchical levels of representation: object- and view-level features. In the feature aggregation stage, the object- and view-level features were aggregated via a self-attention process to generate object- and view-level global features, respectively. Simultaneously, we propose leveraging the cross-attention mechanism to correlate point cloud features with object- and view-level features to obtain object- and view-level cross-modality

global features, respectively. Finally, in the feature classification stage, we concatenate all global features and exploit a fully connected layer to map the concatenated features to the final descriptor. The details of each stage are presented in the following subsections.

3.1. Feature Extraction

In this paper, we exploited two modalities to represent a 3D object: point clouds and multiview images, as shown in Fig. 2. For the point cloud modality, the object O_i is represented by n sampled points via the farthest point sampling (FPS) algorithm, denoted by $P_i = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_n\}$, where $\boldsymbol{p}_j \in \mathbb{R}^3$ contains the 3D coordinates of the j^{th} point. A DGCNN was used to extract the vectorized representation for the point cloud modality. Following the model outlined in [2], a DGCNN is built using a 3D spatial transform network, several EdgeConv layers, and a max-pooling operation. $\boldsymbol{P}_i$ is fed through DGCNN to generate the point-cloud modality features,

$$\boldsymbol{f}_{\text{point},i} = \text{DGCNN}(\boldsymbol{P}_i; \theta_{\text{point}}), \tag{1}$$

where $\boldsymbol{f}_{\text{point},i}$ denotes the point cloud modality feature for O_i and θ_{point} denotes the learnable parameters for the DGCNN.

For the multiview representation of object O_i, multiple view images $\{\boldsymbol{I}_i^1, \boldsymbol{I}_i^2, \ldots, \boldsymbol{I}_i^M\}$ are rendered by setting up M virtual cameras around the object with a uniform interval angle of 30 along the Z-axis. Multi-view modality features were generated by feeding these view images through a CNN without the pooling operation and classification head:

$$\boldsymbol{F}_{\text{view},i}^j = \text{CNN}(\boldsymbol{I}_i^j; \theta_{\text{view}}),\ j = 1, 2, \ldots, M, \tag{2}$$

where $\boldsymbol{F}_{\text{view},i}^j \in \mathbb{R}^{C\times H\times W}$ denotes the feature map of the j_{th} view of O_i and θ_{view} denotes the CNN parameters. With these intermediate CNN feature maps corresponding to the view images, we introduce two hierarchical levels of representations for the multiview modality, that is, object- and view-level representations. To generate object-level representation $\boldsymbol{F}_{\text{object,i}}$, the intermediate feature maps $\{\boldsymbol{F}_{\text{view},i}^1, \boldsymbol{F}_{\text{view},i}^2, \ldots, \boldsymbol{F}_{\text{view},i}^M\}$ are fed through an Inter-View Pooling module where the feature maps are fused via average pooling across the views:

$$\boldsymbol{F}_{\text{object},i} = \text{AP}_{\text{Inter}}(\boldsymbol{F}_{\text{view},i}^1, \boldsymbol{F}_{\text{view},i}^2, \ldots, \boldsymbol{F}_{\text{view},i}^M), \tag{3}$$

where AP_{Inter} denotes inter-view average pooling operation. $\boldsymbol{F}_{\text{object},i} \in \mathbb{R}^{C\times H\times W}$ summarizes the multi-view images and it comprises HW C-dimensional vectorized features $\{\boldsymbol{f}_{\text{object},i}^l\}(l = 1, 2, \ldots, HW)$. $\boldsymbol{f}_{\text{object},i}^l$ contains the global object-level semantics of O_i at the location l. In addition to object-level view representation, we introduce view-level representation by feeding each intermediate feature map through an intra-view pooling module via an average pooling operation:

$$\boldsymbol{f}_{\text{view},i}^j = \text{AP}_{\text{Intra}}(\boldsymbol{F}_{\text{view},i}^j)\ j = 1, 2, \ldots, M, \tag{4}$$

where AP_{Intra} denotes intra-view average pooling. $\boldsymbol{f}_{\text{view},i}^j$ contains the local view-level semantics of O_i reflected in view image $\boldsymbol{I}^j$.

3.2. Feature Aggregation

After the feature extraction stage, given O_i for the point cloud modality, the global feature $\boldsymbol{f}_{\text{point},i}$ is generated. For the multi-view modality, we obtain a set of object-level global features $\{\boldsymbol{f}^{l}_{\text{object},i}\}(l = 1, 2, \ldots, HW)$ and a set of view-level local features $\{\boldsymbol{f}^{j}_{\text{view},i}\}(j = 1, 2, \ldots, M)$, respectively.

In the feature aggregation stage, we designed two types of feature aggregation modules: the in-modality aggregation module (**IMAM**) and cross-modality aggregation module (**CMAM**). IMAM leverages the multihead **self-attention** mechanism inspired by the vision transformer (ViT) [37] to aggregate object-level global and view-level local features. By contrast, CMAM exploits a **cross-attention** mechanism inspired by the cross-attention vision transformer (CrossVit) [38] to interact with the point cloud global feature with the object- and view-level features in the aggregation process, thereby generating cross-modality global representations of the object. For implementation simplicity, both object-level IMAM/CMAM and view-level IMAM/CMAM adopt similar self- and cross-attention architectures that vary with different input dimensions. None of the modules shared any parameters. Specifically, object-level IMAM/CMAM aggregates object-level features with different spatial information, whereas view-level IMAM/CMAM aggregates view-level features with different viewpoint information. Therefore, they were proposed to facilitate the interaction and fusion of local multi-view features from different perspectives.

3.2.1. *In-Modality Aggregation Module*

As shown in Fig. 2, the two IMAMs correspond to object-level global features and view-level local features, that is, the object- and view-level IMAMs. For simplicity, we denote the two levels of input features for O_i by $\boldsymbol{F}_{\text{T},i}$, where $\text{T} \in \{\text{object}, \text{view}\}$. Given $\boldsymbol{F}_{\text{T},i}$ as the input, a class token is prepended to $\boldsymbol{F}_{\text{T},i}$ which is learnable and serves as the summarized feature representation of the input features. Subsequently , a set of learnable position embeddings is added to the input features (together with the class-token feature) to generate position-relevant feature embeddings $\boldsymbol{Z}_{\text{T},i}$. To aggregate these features as shown in Fig. 3, $\boldsymbol{Z}_{\text{T},i}$ is fed through a ViT encoder, as follows:

$$\boldsymbol{Z}^{'}_{\text{T},i} = \text{MSA}(\text{LN}(\boldsymbol{Z}_{\text{T},i})) + \boldsymbol{Z}_{\text{T},i}, \tag{5}$$

$$\boldsymbol{Z}^{*}_{\text{T},i} = \text{LN}(\text{MLP}(\text{LN}(\boldsymbol{Z}^{'}_{\text{T},i})) + \boldsymbol{Z}^{'}_{\text{T},i}). \tag{6}$$

As shown in Fig. 3, $\boldsymbol{Z}^{'}_{\text{T},i}$ denote the intermediate features of the multi-head attention module. $\boldsymbol{Z}^{*}_{\text{T},i}$ denote the output features from the ViT encoder, LN denotes LayerNorm operation, MSA denotes multihead self-attention (MSA), and MLP refers to multilayer perceptron (MLP) blocks. After the feature aggregation process, the object- and view-level self-attended global features, denoted by $\boldsymbol{f}^{S}_{\text{object},i}$ and $\boldsymbol{f}^{S}_{\text{view},i}$ are the features associated with the class token.

$$\boldsymbol{f}^{S}_{\text{T},i} = \boldsymbol{Z}^{*}_{\text{T},i}[\text{class token}] \quad \text{T} \in \{\text{object}, \text{view}\}. \tag{7}$$

3.2.2. *Cross-Modality Aggregation Module*

Similar to the IMAM, as shown in Fig. 4, two CMAMs are proposed to implement cross-modality aggregation for object-level global features and view-level local features, respectively, that is, object- and view-level IMAMs.

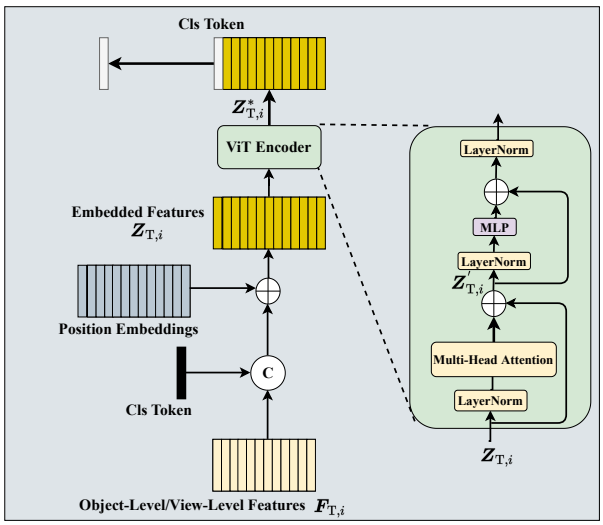

Figure 3: Architecture of IMAM.

The key motivation of CMAM is to correlate the point cloud modality global feature with the aggregation process of multi-view modality features to enhance the feature discriminative capability. To this end, we leveraged the cross-attention mechanism introduced in cross-attention ViT to design CMAMs. Specifically, given $\boldsymbol{F}_{\mathrm{T},i}$ for O_i where $\mathrm{T} \in \{\text{object}, \text{view}\}$ is the input, the input features are first fed into an IMAM module to generate aggregated features, as introduced in Section 3.2.1:

$$\boldsymbol{Z}^S_{\mathrm{T},i} = \mathrm{IMAM}(\boldsymbol{F}_{\mathrm{T},i}). \tag{8}$$

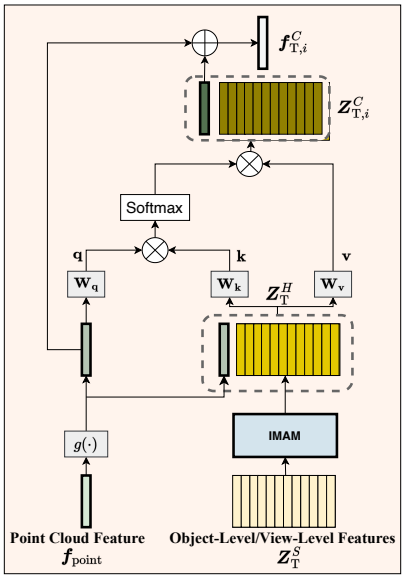

Figure 4: Architecture of CMAM.

Subsequently, the global feature for the point cloud modality $\boldsymbol{f}_{\text{point}}$ generated from Eq. (1) is concatenated with the aggregated multiview features $\boldsymbol{Z}_{\text{T}}^{S}$ to construct the hybrid aggregated features $\boldsymbol{Z}_{\text{T}}^{H}$.

$$\boldsymbol{Z}_{\text{T},i}^{H} = g(\boldsymbol{f}_{\text{point},i}) \oplus \boldsymbol{Z}_{\text{T},i}^{S}, \tag{9}$$

where function $g(\cdot)$ performs feature dimension alignment and $\oplus$ denotes feature concatenation. Finally, to obtain the global cross-modality feature $\boldsymbol{f}_{\text{T},i}^{C}$, the point cloud feature $g(\boldsymbol{f}_{\text{point},i})$ serves as the query vector that interacts with the hybrid aggregated features $\boldsymbol{Z}_{\text{T},i}^{H}$ via a cross-attention mechanism:

$$\boldsymbol{q} = g(\boldsymbol{f}_{\text{point},i})\boldsymbol{W}_q,\ \boldsymbol{k} = \boldsymbol{Z}_{\text{T},i}^{H}\boldsymbol{W}_k,\ \boldsymbol{v} = \boldsymbol{Z}_{\text{T},i}^{H}\boldsymbol{W}_v, \tag{10}$$

$$\boldsymbol{A} = \text{softmax}(\boldsymbol{q}\boldsymbol{k}^{T} / \sqrt{D/h}), \tag{11}$$

$$\boldsymbol{Z}_{\text{T},i}^{C} = \boldsymbol{A}\boldsymbol{v}, \tag{12}$$

where $\boldsymbol{W}_q$, $\boldsymbol{W}_k$ and $\boldsymbol{W}_v$ refer to learnable transformation matrices for generating query, key, and value features for attention computation. D denotes the feature dimension, and h denotes the number of heads for the multihead self-attention module. As shown in Fig. 4, the object- and view-level cross-modality global features $\boldsymbol{f}_{\text{object},i}^{C}$ and $\boldsymbol{f}_{\text{view},i}^{C}$ are calculated by adding $g(\boldsymbol{f}_{\text{point},i})$ with the class token feature in $\boldsymbol{Z}_{\text{T},i}^{C}$:

$$\boldsymbol{f}_{\text{T},i}^{C} = g(\boldsymbol{f}_{\text{point},i}) + \boldsymbol{Z}_{\text{T},i}^{C}[\text{class token}] \quad \text{T} \in \{\text{object, view}\}. \tag{13}$$

3.3. Feature Classification

After the feature aggregation stage, for O_i, four aggregated features are obtained, namely object-level and view-level self-attended global features $\boldsymbol{f}_{\text{object},i}^{S}$ and $\boldsymbol{f}_{\text{view},i}^{S}$, together with object-level and view-level cross-modality global features $\boldsymbol{f}_{\text{object},i}^{C}$ and $\boldsymbol{f}_{\text{view},i}^{C}$. The final descriptor $\boldsymbol{f}_{\text{global},i}$ is obtained by feeding forward the concatenation of all four aggregated features through an MLP layer as follows:

$$\boldsymbol{f}_{\text{global},i} = \text{MLP}(\boldsymbol{f}_{\text{object},i}^{S} \oplus \boldsymbol{f}_{\text{view},i}^{S} \oplus \boldsymbol{f}_{\text{object},i}^{C} \oplus \boldsymbol{f}_{\text{view},i}^{C}). \tag{14}$$

Inspired by the work [22, 25], which applied contrastive losses to learn discriminative features in the angular feature space to generate a highly discriminative final descriptor for object retrieval, we exploited the additive angular margin loss (ArcFace) [39] to train the network. Given a minibatch of N objects belonging to K classes, we first generate the final descriptors $\boldsymbol{f}_{\text{global},i}$ $(i = 1, 2, \ldots, N)$. These final descriptors are normalized using their L2 norm, that is, $\hat{\boldsymbol{f}}_{\text{global},i} = \frac{\boldsymbol{f}_{\text{global},i}}{\|\boldsymbol{f}_{\text{global},i}\|}$. We then introduce a weight matrix $\boldsymbol{W} \in \mathbb{R}^{D\times K}$ where the k^{th} column of $\boldsymbol{W}$ denoted by $\boldsymbol{W}_k$ is related to the k^{th} category and is also L2 normalized. To enhance the discriminative capability of the features, the ArcFace loss function incorporates an additional angular margin to the angle between the feature and the weight associated with the ground truth class:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s\cos(\theta_{y_i}+m)}}{e^{s\cos(\theta_{y_i}+m)} + \sum_{k=1,k\neq y_i}^{K} e^{s\cos\theta_k}}, \tag{15}$$

where y_i is the ground truth class of the i^{th} training sample. $\theta_k \triangleq \arccos(\boldsymbol{W}_k^{\mathrm{T}} \hat{\boldsymbol{f}}_{\mathrm{global},i})$ denotes the angle between $\boldsymbol{W}_k$ and $\hat{\boldsymbol{f}}_{\mathrm{global},i}$. m represents the value of the additive angular margin, and s represents the scaling factor. The overall network was trained by minimizing the ArcFace loss $\mathcal{L}$. After training, the final descriptor for each object is adopted for 3D object retrieval.

4. Experiments

In this section, experiments are conducted on three benchmark datasets, ranging from small to large scales, to demonstrate the effectiveness of the proposed SCA-PVNet.

4.1. Datasets and Evaluation Metrics

We adopted four benchmark datasets to evaluate the retrieval performance: ModelNet40, ShapeNetCore55, ScanObjectNN, and mechanical components benchmark (MCB). The ModelNet40 dataset comprises 12,311 3D CAD models derived from 40 distinct categories, partitioned into a training set with 9,843 objects and a testing set with 2,468 objects. ShapeNetCore55 includes 51,162 3D objects categorized into 55 primary classes and 203 subordinate classes. Following the official setting of the SHREC2017 competition, the training/validation/testing splitting was 70%/10%/20%, respectively . We leveraged the aligned version of the ShapeNetCore55 dataset, which comprises organized 3D objects. ScanObjectNN is a point-cloud dataset comprising 2902 3D objects from 15 categories. This dataset was challenging because the samples were from real-world scenes with occlusions, background noise, and deformation. We conducted experiments on the PB-T50-RS splits of ScanObjectNN, which represent the most challenging cases with 11,416 training samples and 2,882 testing samples. The MCB dataset comprised 58,696 objects across 68 categories. For our experiments, we utilized the MCB dataset version A (MCB-A), wherein the objects were uniformly aligned, adhering to the training and testing split configurations established in the original study [40].

We generated a point-cloud representation of each 3D object by sampling 1024 points using the FPS method. Additionally, we rendered 12 view of the object using the Blender software and Phong shading. This was accomplished by positioning 12 virtual cameras around the object, each at a 30-degree angle along the z-axis. For ScanObjectNN, we used the point-cloud representations of 2040 points for each object and generated 12 224 224 depth images by projecting the point clouds from different perspectives as the input for the multiview branch.

To evaluate the metrics on ModelNet40, we used mean average precision (MAP). This involves calculating the average precision for each 3D object in the test set and using it as a query sample. A ranked list of all test objects was then generated based on pairwise cosine distances, and the mean of these average precisions was calculated. For ShapeNetCore55, we use the SHREC2017 competition released evaluator to compute the retrieval performance in terms of F1@N, mAP, and normalized discounted cumulative gain (NDCG) under “micro” and “macro” settings. For ScanObjectNN dataset, the overall classification accuracy (OA) was adopted as a performance metric. For the MCB dataset, we report F1@N, mAP, and NDCG@N under “micro, “macro, and “micro + macro” settings, respectively.

4.2. Implementation and Training Details

In the feature extraction stage, the DGCNN [2] was adopted for the point cloud modality on ModelNet40, ShapeNetCore55, and MCB-A. Conversely, PointNext [41] was used for ScanObjectNN. For all datasets, the MVCNN was adopted for the multiview modality. We used AlexNet [42] as the backbone network of the MVCNN for ModelNet40 and ResNet-34 [43] for ShapeNetCore55, ScanObjectNN, and MCB-A. Following [14], both the DGCNN and MVCNN were pretrained on the training set using cross-entropy loss. During the feature aggregation phase for the IMAM model, we set the multihead self-attention count to four . In addition, the dimensions of the hidden features were set to 128, and the dimensions of the aggregated features were set to 512. For CMAM, the number of attention heads was set to four, and the dimension of the aggregated feature was set to 512. In the feature classification stage, the fully connected layer maps the concatenated aggregated feature $\boldsymbol{f}_{\text{global},i}$ onto a vector whose dimension is the same as the number of object categories. In the pre-training phase, the batch size, learning rate, weight decay, and number of epochs were set to 400, 0.01, 0.001, and 50, respectively. For fine-tuning, we followed the settings in PVRNet [14]. During the initial 10 epochs, the feature extraction models remained unchanged, while the remaining components, namely IMAM, CMAM, and the FC layers, were fine-tuned. The learning rate for the first 10 epochs was set at 0.01, and 0.001 after the subsequent 10 epochs. The batch size was set to 28 per GPU, the weight decay to $1e^{-5}$ and the number of epochs to 50. For ArcFace loss, the angular margin was set to 0.5, and the scale was set to 64.

4.3. Comparison with State-of-the-art Methods

4.3.1. ***Performance Comparison on ModelNet40 Dataset***

Table 1 lists the retrieval performance of ModelNet40 on the mAP. The table indicates that our SCA-PVNet outperforms all the other methods. Particularly, our method outperforms all methods (PVNet, PVRNet, MMFN, and Attention-Guided Fusion Network) that use point clouds and multiview modalities for 3D object retrieval. Compared with PVRNet, which uses the same feature extraction networks (DGCNN and MVCNN with AlexNet) as our method, our method produced a 2.0% performance gain in mAP, reflecting the effectiveness of the proposed IMAM and CMAM in feature aggregation. Compared with MMFN, which uses three data modalities, including point cloud, multi-view, and panorama views, our method can still achieve better performance (with a 2.2% performance gain), even though we only exploit two data modalities. Additionally, our method outperformed the methods based on a single modality (point cloud or multiview). In particular, compared to the vanilla MVCNN [3], we achieved a 12.3% performance gain. Compared with advanced MVCNN variants, such as MVCNN with cube loss [44], we still achieved a 3.6% performance gain. These observations demonstrate the superiority of effectively incorporating point-cloud representations and interacting with multi-view representations. Finally, it is evident that our model, trained using the ArcFace loss, outperforms models trained with other metric learning losses, such as TCL [7, 8], ATCL [25], and cube loss [44].

We compared the proposed SCA-PVNet with multimodal methods for 3D object classification in full-set and few-shot settings. In this experiment, full-set setting refers to the case in which all training samples were used to train

Table 1: Retrieval performance comparison on ModelNet40 (in %). The best results are in bold.

Methods	mAP
SPH [45]	33.3
LFD [46]	40.9
3DShapeNet [18]	49.2
DLAN [47]	85.0
DeepPano [48]	76.8
GIFT [49]	81.9
MVCNN (AlexNet) [3]	80.2
MVCNN (ViT-B-16) [3]	83.3
RED [50]	86.3
GVCNN [51]	85.7
TCL [7]	88.0
ATCL [25]	86.1
SeqViews2SeqLabels [28]	89.1
VDN [52]	86.6
Batch-wise [53]	83.8
MVCNN (Cube loss) [44]	88.9
VNN [9]	88.9
NCENet [23]	87.1
3DViewGraph [54]	90.5
GSL (ATCL) [8]	90.1
DAN [21]	90.4
DAGL+Attention-NetVlad [55]	91.1
PVNet [13]	89.5
MMJN [16]	89.8
PVRNet (AlexNet) [14]	90.5
PVRNet (ViT-B-16) [14]	90.6
MMFN [15]	90.3
Attention-Guided Fusion Network [17]	91.4
Our SCA-PVNet	**92.5**

Table 2: Classification performance comparison on ModelNet40 under the full-set setting (in %)

Method	Classification Acc
PVNet [13]	93.3
MMJN [16]	93.8
PVRNet [14]	93.6
MMFN [15]	92.9
Attention-Guided Fusion Network [17]	93.5
Our SCA-PVNet	**94.1**

the model. The few-shot setting refers to the case in which only k samples (k shots") from each category are adopted to train the model. For both settings, the trained models were evaluated using all the testing samples in the testing set of ModelNet40. Table 2 lists the classification performance of ModelNet40 in a full-set setting. Our method also outperforms these multimodal methods in 3D object classification, demonstrating the effectiveness of the proposed modules in producing discriminative feature representations for 3D objects. Fig. 5 shows the few-shot classification performance with shot numbers selected from {8, 16, 32, 64, 128}. We compared our method to PointCLIP [56] which was evaluated using a similar few-shot setting protocol. PointCLIP adopts the recently released contrastive language-

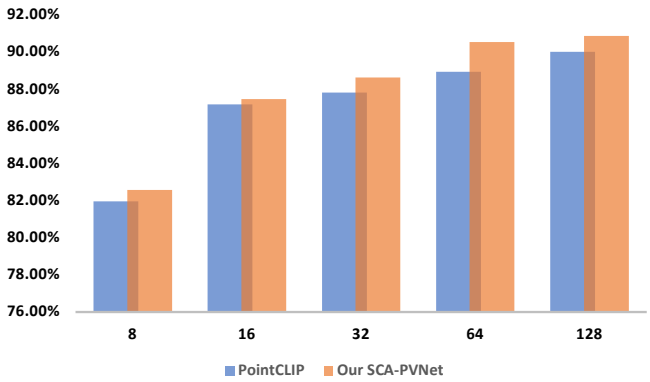

Figure 5: Classification performance comparison on ModelNet40 under few-shot setting

image pretrain (CLIP) model to extract features from projected point-cloud views for 3D model classification. From the figure, it can be observed that our SCA-PVNet outperforms PointCLIP for all shot number settings.

*4.3.2. **Performance Comparison on ShapeNetCore55 Dataset***

Table 3 lists the retrieval performance of the ShapeNetCore55. Based on the table, our SCA-PVNet demonstrated superior performance in five of the six performance metrics, excluding F1@N. Although RotationNet emerged as the winning solution in the SHREC2017 competition for 3D object retrieval, our method outperformed it in terms of mAP and NDCG in both micro and macro settings. For F1@N, our method achieved the same performance as RotationNet under the micro setting and was only 0.8% lower under the macro setting. Compared with MVCNN, our method achieves a significant performance gain in all six metrics, demonstrating the effectiveness of incorporating an additional point cloud modality.

Table 3: Performance comparison on ShapeNetCore55 (in %)

Methods	micro			macro		
	F1@N	mAP	NDCG	F1@N	mAP	NDCG
RotationNet [29]	**79.8**	77.2	86.5	**59.0**	58.3	65.6
Improved_GIFT [57]	76.7	72.2	82.7	58.1	57.5	65.7
ReVGG [58]	77.2	74.9	82.8	51.9	49.6	55.9
DLAN [47]	71.2	66.3	76.2	50.5	47.7	56.3
MVFusionNet [59]	69.2	62.2	73.2	48.4	41.8	50.2
CM-VGG5-6DB [60]	47.9	54.0	65.4	16.6	33.9	40.4
ZFDR [61]	28.2	19.9	33.0	19.7	25.5	37.7
DeepVoxNet [1]	25.3	19.2	27.7	25.8	23.2	33.7
GIFT [57]	68.9	64.0	76.5	45.4	44.7	54.8
MVCNN [3]	76.4	73.5	81.5	57.5	56.6	64.0
Our SCA-PVNet	**79.8**	**78.6**	**86.6**	58.2	**59.3**	**66.6**

4.3.3. *Performance Comparison on ScanObjectNN Dataset*

Table 4 lists the overall classification accuracy for the ScanObjectNN dataset. We compared our SCA-PVNet with two multimodal methods: PVNet [13] and PVRNet [14]. To establish a fair comparison, PointNext [41] and MVCNN [3] were used as the backbone networks to extract features for the point-cloud and multi-view modalities, respectively. From the table, SCA-PVNet outperformed both methods.

Table 4: Performance comparison on ScanObjectNN dataset (in %)

Method	OA
PVNet [13]	88.1
PVRNet [14]	88.0
Our SCA-PVNet	**89.1**

4.3.4. *Performance Comparison on MCB-A Dataset*

Table 5 presents the retrieval performance for the MCB-A dataset. We compared our method with that reported in [40]. Among the nine performance metrics, our method secured the first rank in eight metrics. In particular, compared with PointCNN which achieves the best performance among those using only the point-cloud modality, our method produces 10.4%, 7.8%, and 10.3% in F1@N, mAP, and NDCG@N under the Micro+Macro settings, respectively. However, compared with RotationNet which achieves the best performance among those using only multi-view modality data, under the Micro+Macro settings, our method achieves scores of 30.6% in F1@N, 15.1% in mAP, and 27.0% in NDCG@N. Additionally, we present the 2D t-SNE embeddings of the testing samples in the MCB-A dataset for all compared methods in Fig. 6. This figure illustrates that our method generated the most distinguishable embeddings for the test samples, leading to the highest retrieval performance.

Table 5: Performance comparison on MCB-A dataset (in %). The best results are in bold.

Method	Micro			Macro			Micro + Macro		
	F1@N	mAP	NDCG@N	F1@N	MAP	NDCG@N	F1@N	mAP	NDCG@N
PointCNN [62]	69.0	88.9	89.8	**83.3**	88.6	85.4	76.2	88.3	87.6
PointNet++ [5]	61.3	79.4	75.4	71.2	80.3	74.6	66.3	79.9	75.0
SpiderCNN [63]	66.9	86.7	79.3	77.6	87.7	81.2	72.3	87.2	80.3
MVCNN [3]	48.8	65.7	48.7	58.5	73.5	64.1	53.7	69.6	56.4
RotationNet [64]	50.8	80.5	68.3	68.3	81.5	73.5	56.0	81.0	70.9
DLAN [65]	56.8	87.9	82.8	82.0	88.0	84.5	69.4	88.0	83.7
VRN [19]	40.2	65.3	51.9	50.7	66.4	57.6	45.5	65.9	54.8
Our SCA-PVNet	**92.8**	**94.0**	**99.1**	80.4	**98.1**	**96.7**	**86.6**	**96.1**	**97.9**

4.4. *Discussion*

In this section, we describe several ablation experiments conducted to analyze the proposed SCA-PVNet.

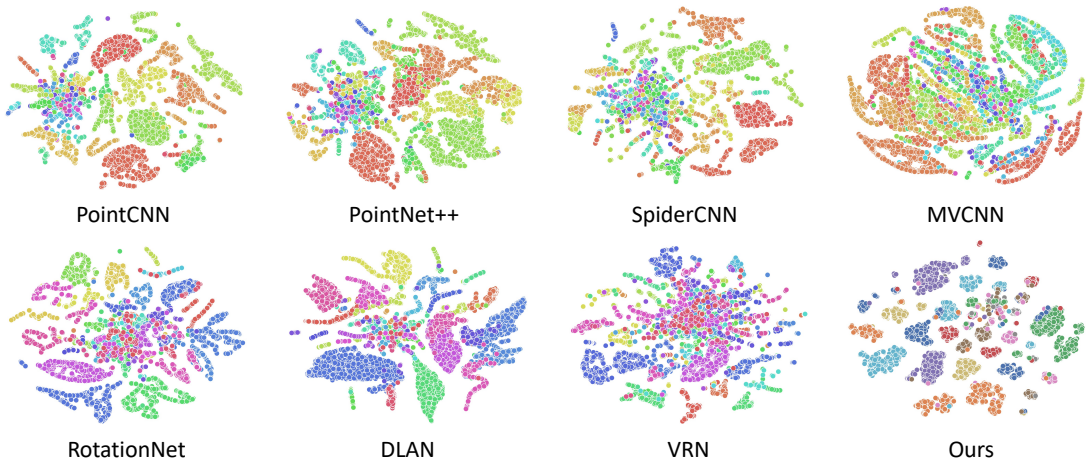

Figure 6: 2D t-SNE embeddings of all the testing samples in the MCB-A dataset.

4.4.1. ***On the proposed modules***

First, we analyze the effects of the proposed modules. We consider eight ablational models: (i) DGCNN using only the point cloud modality; (ii) MVCNN using only multi-view modality; (iii) the model exploiting a direct concatenation of feature representations of (i) and (ii); (iv) the proposed model without view-level feature aggregation branch; (v) the proposed model without an object-level feature aggregation branch; (vi) the proposed model without IMAM (for both branches); (vii) the proposed model without CMAM (for both branches); and (viii) the proposed SCA-PVNet. Table 6 lists the retrieval mAP for ModelNet40 produced by each ablation model. The models incorporating both modalities significantly outperform those incorporating a single modality by approximately 10% in mAP. Second, the model incorporating the proposed view-level or object-level feature aggregation branches outperformed the model that directly concatenated the features generated by DGCNN and MVCNN. It was demonstrated that removing either the view- or object-level feature aggregation branches caused performance degradations of 1.3% and 1.1%, respectively. Regarding the aggregation modules, removing IMAM or CMAM reduced the retrieval performance by 1.2% and 1.1%, respectively. To summarize, our SCA-PVNet incorporating both aggregation branches and aggregation modules, achieved the best retrieval performance among all ablation methods, demonstrating the effectiveness of the proposed modules.

4.4.2. ***On Loss Function***

In this section, we compare the retrieval performance of the proposed SCA-PVNet trained using cross-entropy loss and ArcFace loss. We then conducted ablation studies on the key hyperparameters of ArcFace loss, namely, the scaling factor s and margin m.

Table 7 presents the mAP on ModelNet40 produced by the SCA-PVNet trained with cross-entropy loss and that with ArcFace Loss. From the table, both ablation models could produce comparable performances, and SCA-PVNet

Table 6: mAP for the ablation models for the proposed modules (in %)

Model	mAP
DGCNN (Point Cloud Modality)	81.6
MVCNN (Multi-View Modality)	80.2
Direct Concatenation	91.0
Ours w/o View-Level Feature Aggregation	91.2 (-1.3)
Ours w/o Object-Level Feature Aggregation	91.4 (-1.1)
Ours w/o IMAM	91.3 (-1.2)
Ours w/o CMAM	91.4 (-1.1)
Our SCA-PVNet	**92.5**

with ArcFace Loss slightly outperformed its counterpart trained with cross-entropy loss. Compared to PVRNet trained with cross-entropy loss, SCA-PVNet with cross-entropy loss produced a 1.6% performance gain in the retrieval mAP. From the experimental observations, we conclude that the loss function does not play a dominant role in performance gain.

Table 7: mAP for the ablation models for the loss functions (in %)

Method	mAP
SCA-PVNet with Cross-Entropy Loss	92.1
SCA-PVNet with ArcFace Loss	**92.5**

We also conducted a sensitivity analysis of the scaling factor s and margin m for ArcFace loss in Eq. (15) using the ModelNet40 dataset. For s, we fixed the margin m at 0.5 and selected s from a set of values, that is, $\{16, 32, 64, 128, 256\}$. Fig. 7(a) shows mAP with respect to s. It is observed that mAP is robust to different values of s where comparable performance is achieved. We set s to 64 because this produced the best performance, that is, 92.5%. For m, we fixed the scale s at 64 and selected m from $\{0.1, 0.3, 0.5, 0.7, 0.9\}$. From Fig. 7(b), we note that the mAP improves as s increases from 0.1 to 0.5 and then significantly decreases from 92.5% to 85.4% as s decreases from 0.5 to 0.9. Therefore, in this paper, m is set to 0.5.

4.4.3. ***On Model Complexity.***

We also analyzed the relationship between performance gain and model complexity. We compared the different ablation models of our SCA-PVNet with the baseline method, PVRNet [14] because PVRNet shares the same feature extraction networks (DGCNN and MVCNN) as our method. Table 8 presents the retrieval performance of the ModelNet40 and model complexity (in terms of the # of parameters) of the four ablation models. By progressively incorporating the proposed feature aggregation branches, the performance of our method was consistently improved, with only a slight increase in model complexity.

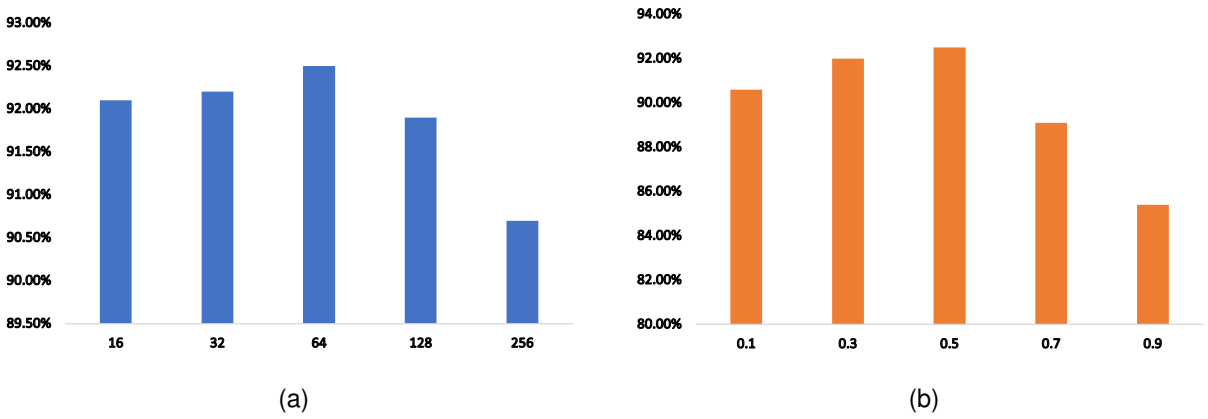

Figure 7: Sensitive analysis in mAP on ModelNet40 with respect to (a) s and (b) m.

Table 8: Model Complexity

Model	mAP	# of Params (M)
PVRNet	90.5	70.00
Ours w/o View-Level Feature Aggregation	91.2	70.53
Ours w/o Object-Level Feature Aggregation	91.4	74.19
Our SCA-PVNet	**92.5**	75.76

*4.4.4. **On Robustness Towards Missing Testing Data***

In this section, the robustness of the proposed method for missing testing data is analyzed. In real-world applications, the testing data are only partially available. In our case, there may have been missing views or missing points. One merit of multimodal approaches is their robustness towards missing data in one of the modalities because the other modality can compensate for the information loss for the object to be retrieved.

Motivated by PVRNet [14], we investigated the influence of scenarios with missing views and points. Our SCA-PVNet was trained using 12 views and 1024 points. For the missing-view scenario, during testing, we retained the number of points for each point cloud at 1024 and varied the number of views in the range of [2, 12] with an interval of 2. Fig 8 shows the mAP for different numbers of missing views for PVRNet and the proposed method. It was demonstrated that reducing the number of testing views leads to a degradation in retrieval performance. By reducing the number of testing views from 12 to 2, compared with the severe performance drop of 9.3% for PVRNet, the performance drop of our method was only 4.7%. For the edge case of only two views available during testing, our method still obtained 87.8% in mAP compared with 81.2% for PVRNet. In summary, our method exhibits better robustness towards missing views than PVRNet.

In scenarios where points were missing, we maintained the number of views at 12 during testing; however, the number of points was adjusted within a range of 1281024, increasing in steps of 128. The retrieval mAPs under different settings are shown in Fig. 9. Similarly, it was observed that missing points caused performance degradation.

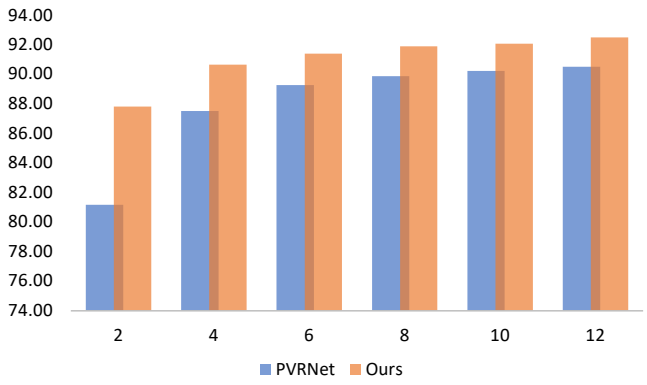

Figure 8: Retrieval MAPs for both methods under different settings in the missing views scenario.

Compared to PVRNet, whose retrieval mAP decreased from 90.5% to 70.3% when the number of points was reduced from 1024 to 128, the mAP of our method decreased from 92.5% to 84.1%. Therefore, we conclude that the proposed method is more robust to missing points.

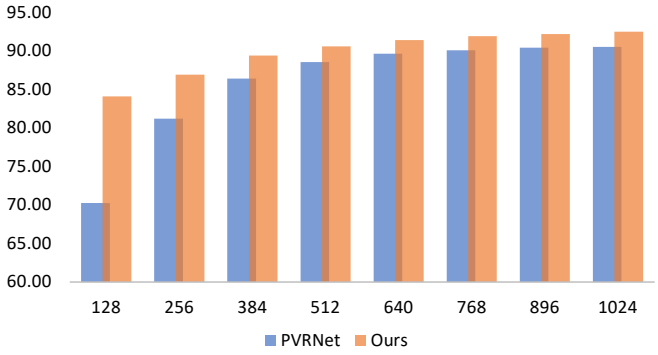

Figure 9: Retrieval MAPs for both methods under different settings in the missing points scenario.

In this experiment, we focused on evaluating our models robustness towards missing data in either modality, following the prior work on PVRNet [14]. It would be interesting to evaluate the robustness of SCA-PVNet under other degradation, noise effects, and variabilities during the data generation process, as analyzed in [66]. This topic will be explored in future studies.

4.4.5. ***Qualitative Results***

In Fig.10, we show the top 10 retrieval results for four testing samples (with category "table," "chair," "car," and "monitor") from the ShapeNetCore55 testing set. The query sample contains both the point cloud and view image modalities, and we use one view image to represent each retrieved result. From the figure, it can be observed that our method can achieve good performance even for those with large within-class variations, such as chairs with different styles, as shown in the second row of Fig.10.

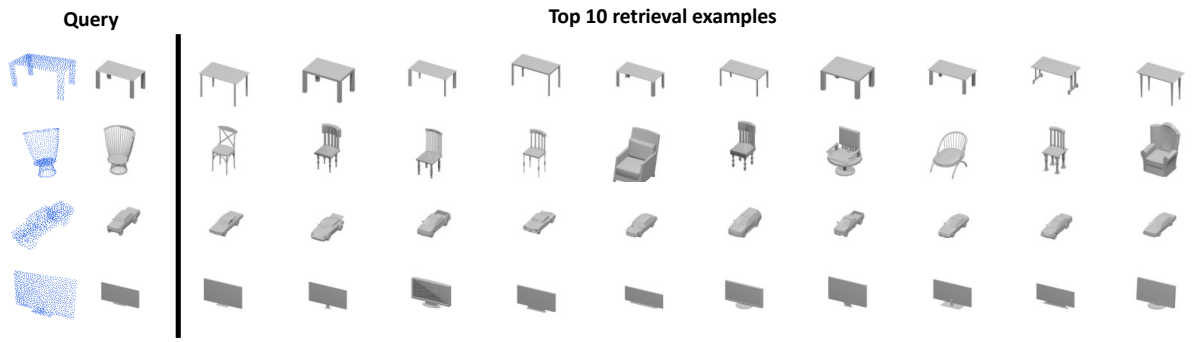

Figure 10: Qualitative results using the testing samples picked from the ShapeNetCore55 dataset.

5. Conclusion

This study proposes a self-and-cross attention-based aggregation of point clouds and multiview images (SCA-PVNet) for 3D object retrieval. To facilitate effective feature fusion, two types of aggregation modules, IMAM and CMAM, have been proposed by leveraging self-attention and cross-attention mechanisms, respectively. Extensive experiments and analyses were conducted on three datasets, ranging from small to large scales, to validate the superiority of the proposed SCA-PVNet over state-of-the-art methods. In addition to producing state-of-the-art retrieval performance, SCA-PVNet also shows more robustness in scenarios where there are missing views or missing points during inference. In future studies, we will further investigate extending the current self-and-cross attention framework to more representative modalities of 3D objects, such as voxels or panorama-view images. Additionally, it could be promising to extend the current method for large-scale real-world 3D object recognition. Another interesting future study could be the introduction of more advanced AI backbone networks to extract features from different modalities and further improve the retrieval performance of SCA-PVNet.

Acknowledgement

This research was supported by A*STAR under RIE2020 INDUSTRY ALIGNMENT FUND-INDUSTRY COLLABORATION PROJECTS (IAF-ICP) Grant No. I2001E0073.

References

[1] D. Maturana, S. Scherer, VoxNet: A 3D convolutional neural network for real-time object recognition, in: Proceedings of the IEEE International Conference on Intelligent Robots and Systems, 2015, pp. 922–928.

[2] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, J. M. Solomon, Dynamic graph CNN for learning on point clouds, ACM Transactions on Graphics 38 (2018) 1–12.

[3] H. Su, S. Maji, E. Kalogerakis, E. Learned-Miller, Multi-view convolutional neural networks for 3D shape recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 945–953.

[4] C. R. Qi, H. Su, K. Mo, L. J. Guibas, PointNet: Deep learning on point sets for 3D classification and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660.

[5] C. R. Qi, L. Yi, H. Su, L. J. Guibas, PointNet++: Deep hierarchical feature learning on point sets in a metric space, Advances in Neural Information Processing Systems (2017) 5105–5114.

[6] A. Cheraghian, L. Petersson, 3DCapsule: Extending the capsule architecture to classify 3D point clouds, in: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 2019, pp. 1194–1202.

[7] X. He, Y. Zhou, Z. Zhou, S. Bai, X. Bai, Triplet-center loss for multi-view 3D object retrieval, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1945–1954.

[8] X. He, S. Bai, J. Chu, X. Bai, An improved multi-view convolutional neural network for 3D object retrieval, IEEE Transactions on Image Processing 29 (2020) 7917–7930.

[9] X. He, T. Huang, S. Bai, X. Bai, View N-gram network for 3D object retrieval, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 7515–7524.

[10] J. Jiang, D. Bao, Z. Chen, X. Zhao, Y. Gao, MLVCNN: Multi-loop-view convolutional neural network for 3D shape retrieval, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 8513–8520.

[11] H. Xu, C. Huang, D. Wang, Enhancing semantic image retrieval with limited labeled examples via deep learning, Knowledge-Based Systems 163 (2019) 252–266.

[12] X. Shi, X. Qian, Exploring spatial and channel contribution for object based image retrieval, Knowledge-Based Systems 186 (2019) 104955.

[13] H. You, Y. Feng, R. Ji, Y. Gao, PVNet: A joint convolutional network of point cloud and multi-view for 3D shape recognition, in: Proceedings of the ACM International Conference on Multimedia, 2018, pp. 1310–1318.

[14] H. You, Y. Feng, X. Zhao, C. Zou, R. Ji, Y. Gao, PVRNet: Point-view relation neural network for 3D shape recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 9119–9126.

[15] W. Nie, Q. Liang, Y. Wang, X. Wei, Y. Su, MMFN: multimodal information fusion networks for 3D model classification and retrieval, ACM Transactions on Multimedia Computing, Communications, and Applications 16 (4) (2020) 1–22.

[16] W. Nie, Q. Liang, A.-A. Liu, Z. Mao, Y. Li, MMJN: Multi-modal joint networks for 3D shape recognition, in: Proceedings of the ACM International Conference on Multimedia, 2019, pp. 908–916.

[17] B. Peng, Z. Yu, J. Lei, J. Song, Attention-guided fusion network of point cloud and multiple views for 3D shape recognition, in: Proceedings of the IEEE International Conference on Visual Communications and Image Processing, 2020, pp. 185–188.

[18] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, J. Xiao, 3D ShapeNets: A deep representation for volumetric shapes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1912–1920.

[19] A. Brock, T. Lim, J. M. Ritchie, N. Weston, Generative and discriminative voxel modeling with convolutional neural networks, arXiv preprint arXiv:1608.04236.

[20] H.-Y. Zhou, A.-A. Liu, W.-Z. Nie, J. Nie, Multi-view saliency guided deep neural network for 3D object retrieval and classification, IEEE Transactions on Multimedia 22 (6) (2019) 1496–1506.

[21] W. Nie, Y. Zhao, D. Song, Y. Gao, DAN: Deep-attention network for 3D shape recognition, IEEE Transactions on Image Processing 30 (2021) 4371–4383.

[22] D. Lin, Y. Li, Y. Cheng, S. Prasad, T. L. Nwe, S. Dong, A. Guo, Multi-view 3D object retrieval leveraging the aggregation of view and instance attentive features, Knowledge-Based Systems 247 (2022) 108754.

[23] C. Xu, Z. Li, Q. Qiu, B. Leng, J. Jiang, Enhancing 2D representation via adjacent views for 3D shape retrieval, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 3732–3740.

[24] Z. Han, H. Lu, Z. Liu, C.-M. Vong, Y.-S. Liu, M. Zwicker, J. Han, C. P. Chen, 3D2SeqViews: Aggregating sequential views for 3D global feature learning by CNN with hierarchical attention aggregation, IEEE Transactions on Image Processing 28 (8) (2019) 3986–3999.

[25] Z. Li, C. Xu, B. Leng, Angular triplet-center loss for multi-view 3D shape retrieval, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 8682–8689.

[26] C. Ma, Y. Guo, J. Yang, W. An, Learning multi-view representation with LSTM for 3D shape recognition and retrieval, IEEE Transactions on Multimedia 21 (5) (2018) 1169–1182.

[27] A.-A. Liu, H. Zhou, W. Nie, Z. Liu, W. Liu, H. Xie, Z. Mao, X. Li, D. Song, Hierarchical multi-view context modelling for 3D object classification and retrieval, Information Sciences 547 (2021) 984–995.

[28] Z. Han, M. Shang, Z. Liu, C.-M. Vong, Y.-S. Liu, M. Zwicker, J. Han, C. P. Chen, SeqViews2SeqLabels: Learning 3D global features via aggregating sequential views by RNN with attention, IEEE Transactions on Image Processing 28 (2) (2018) 658–672.

[29] A. Kanezaki, Y. Matsushita, Y. Nishida, Rotationnet for joint object categorization and unsupervised pose estimation from multi-view images, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (1) (2019) 269–283.

[30] X. Wei, R. Yu, J. Sun, View-GCN: View-based graph convolutional network for 3D shape analysis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 1850–1859.

[31] J.-C. Su, M. Gadelha, R. Wang, S. Maji, A deeper look at 3D shape classifiers, in: Proceedings of the European Conference on Computer Vision Workshops, 2018, pp. 0–0.

[32] W.-Z. Nie, W.-W. Jia, W.-H. Li, A.-A. Liu, S.-C. Zhao, 3D pose estimation based on reinforce learning for 2D image-based 3D model retrieval, IEEE Transactions on Multimedia 23 (2020) 1021–1034.

[33] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: Proceedings of the International Conference on Machine Learning, PMLR, 2021, pp. 8748–8763.

[34] D. Hong, B. Zhang, H. Li, Y. Li, J. Yao, C. Li, M. Werner, J. Chanussot, A. Zipf, X. X. Zhu, Cross-city matters: A multimodal remote sensing benchmark dataset for cross-city semantic segmentation using high-resolution domain adaptation networks, Remote Sensing of Environment 299 (2023) 113856.

[35] D. Hong, B. Zhang, X. Li, Y. Li, C. Li, J. Yao, N. Yokoya, H. Li, X. Jia, A. Plaza, et al., SpectralGPT: Spectral foundation model, arXiv preprint arXiv:2311.07113.

[36] M. Afham, I. Dissanayake, D. Dissanayake, A. Dharmasiri, K. Thilakarathna, R. Rodrigo, Crosspoint: Self-supervised cross-modal contrastive learning for 3D point cloud understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2022, pp. 9902–9912.

[37] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, in: International Conference on Learning Representations, 2021.

[38] C.-F. R. Chen, Q. Fan, R. Panda, CrossViT: Cross-attention multi-scale vision transformer for image classification, in: Proceedings of the IEEE International Conference on Computer Vision, 2021, pp. 357–366.

[39] J. Deng, J. Guo, N. Xue, S. Zafeiriou, Arcface: Additive angular margin loss for deep face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 4690–4699.

[40] S. Kim, H.-g. Chi, X. Hu, Q. Huang, K. Ramani, A large-scale annotated mechanical components benchmark for classification and retrieval tasks with deep neural networks, in: Proceedings of the European Conference on Computer Vision, 2020.

[41] G. Qian, Y. Li, H. Peng, J. Mai, H. Hammoud, M. Elhoseiny, B. Ghanem, PointNeXt: Revisiting PointNet++ with improved training and scaling strategies, in: Advances in Neural Information Processing Systems (NeurIPS), 2022.

[42] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems 25 (2012) 1097–1105.

[43] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[44] D. Wang, H. Yao, F. Tombari, S. Zhao, B. Wang, H. Liu, Learning descriptors with cube loss for view-based 3-D object retrieval, IEEE Transactions on Multimedia 21 (8) (2019) 2071–2082.

[45] M. Kazhdan, T. Funkhouser, S. Rusinkiewicz, Rotation invariant spherical harmonic representation of 3D shape descriptors, in: Symposium

on Geometry Processing, Vol. 6, 2003, pp. 156–164.

[46] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, M. Ouhyoung, On visual similarity based 3D model retrieval, in: Computer graphics forum, Vol. 22, Wiley Online Library, 2003, pp. 223–232.

[47] T. Furuya, R. Ohbuchi, Deep aggregation of local 3D geometric features for 3D model retrieval., in: British Machine Vision Conference, Vol. 7, 2016, p. 8.

[48] B. Shi, S. Bai, Z. Zhou, X. Bai, Deeppano: Deep panoramic representation for 3D shape recognition, IEEE Signal Processing Letters 22 (12) (2015) 2339–2343.

[49] S. Bai, X. Bai, Z. Zhou, Z. Zhang, L. Jan Latecki, GIFT: A real-time and scalable 3D shape search engine, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5023–5032.

[50] S. Bai, Z. Zhou, J. Wang, X. Bai, L. Jan Latecki, Q. Tian, Ensemble diffusion for retrieval, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 774–783.

[51] Y. Feng, Z. Zhang, X. Zhao, R. Ji, Y. Gao, GVCNN: Group-view convolutional neural networks for 3D shape recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 264–272.

[52] B. Leng, C. Zhang, X. Zhou, C. Xu, K. Xu, Learning discriminative 3D shape representations by view discerning networks, IEEE Transactions on Visualization and Computer Graphics 25 (10) (2018) 2896–2909.

[53] L. Xu, H. Sun, Y. Liu, Learning with batch-wise optimal transport loss for 3D shape recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3333–3342.

[54] Z. Han, X. Wang, C.-M. Vong, Y.-S. Liu, M. Zwicker, C. Chen, 3DViewGraph: Learning global features for 3D shapes from a graph of unordered views with attention, arXiv preprint arXiv:1905.07503.

[55] Z. Shi, Z. Quan, J. Shi, Z. Guo, M. Zhang, Z. Xiao, 3D model retrieval algorithm based on attention and multi-view fusion, in: Proceedings of the International Conference on Computer Science and Software Engineering, 2022, pp. 474–480.

[56] R. Zhang, Z. Guo, W. Zhang, K. Li, X. Miao, B. Cui, Y. Qiao, P. Gao, H. Li, PointCLIP: Point cloud understanding by CLIP, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2022, pp. 8552–8562.

[57] S. Bai, X. Bai, Z. Zhou, Z. Zhang, Q. Tian, L. J. Latecki, GIFT: Towards scalable 3D shape retrieval, IEEE Transactions on Multimedia 19 (6) (2017) 1257–1271.

[58] X. Bai, S. Bai, X. Wang, Beyond diffusion process: Neighbor set similarity for fast re-ranking, Information Sciences 325 (2015) 342–354.

[59] P. Daras, A. Axenopoulos, A compact multi-view descriptor for 3D object retrieval, in: Proceedings of the International Workshop on Content-Based Multimedia Indexing, IEEE, 2009, pp. 115–119.

[60] Z. Lian, A. Godil, X. Sun, J. Xiao, CM-BOF: visual similarity-based 3D shape retrieval using clock matching and bag-of-features, Machine Vision and Applications 24 (8) (2013) 1685–1704.

[61] B. Li, H. Johan, 3D model retrieval using hybrid features and class information, Multimedia Tools and Applications 62 (3) (2013) 821–846.

[62] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, B. Chen, PointCNN: Convolution on X-transformed points, Advances in Neural Information Processing Systems 31 (2018) 820–830.

[63] Y. Xu, T. Fan, M. Xu, L. Zeng, Y. Qiao, SpiderCNN: Deep learning on point sets with parameterized convolutional filters, in: Proceedings of the European Conference on Computer Vision, 2018, pp. 87–102.

[64] A. Kanezaki, Y. Matsushita, Y. Nishida, Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5010–5019.

[65] T. Furuya, R. Ohbuchi, Diffusion-on-manifold aggregation of local features for shape-based 3D model retrieval, in: Proceedings of the ACM on International Conference on Multimedia Retrieval, 2015, pp. 171–178.

[66] D. Hong, N. Yokoya, J. Chanussot, X. X. Zhu, An augmented linear mixing model to address spectral variability for hyperspectral unmixing, IEEE Transactions on Image Processing 28 (4) (2018) 1923–1938.