Retrieval-Specific View Learning for Sketch-to-Shape Retrieval

Shuaihang Yuan, Congcong Wen, *Member, IEEE*, Yu-Shen Liu, *Member, IEEE*, and Yi Fang, *Member, IEEE*

***Abstract*—Sketch-based 3D shape retrieval (SBSR) can be approached by learning domain-invariant descriptors or ranking metrics from sketches and 2D view images of 3D shapes rendered through numerous viewpoints. However, determining the most appropriate viewpoints that convey discriminative geometric features to benefit the task of SBSR became an essential yet not fully explored area. Existing works extract 3D features from multi-view images observed through pre-defined viewpoints to match 2D sketches. Those methods, however, fail to dynamically select viewpoints by considering the SBSR task. In this work, we introduce a fully differentiable viewpoint learning paradigm driven by the downstream SBSR task, which supports the task-aware and sketch-dependent dynamic viewpoint determination process. We naturally integrate this task-specific and sketch-dependent viewpoint learning process into a meta-learning framework to develop a novel Dynamic Viewer (DV) module for category-level SBSR. DV module comprises a Meta View Learner (MVL) block and a View Generator (VG) block. Specifically, as the first part of the DV module, the MVL block learns to initiate the necessary network parameters of the VG block. Then, the VG block that serves as the second part learns the best viewpoints to render 2D images. To learn the optimal viewpoints for category-level SBSR, we further introduce a view mining loss that aims to maximize the similarity of feature-level information among rendered 2D views and the query sketch. Further, we adopt a variational autoencoder (VAE) to retrieve 3D shapes by setting the newly rendered images and query sketch as inputs. Comprehensive experimental results on popular SBSR datasets demonstrate that our proposed category-level SBSR framework have achieved state-of-the-art performance for category-level SBSR task. Furthermore, our approach can be easily adapted to address instance-level SBSR task with promising results.**

***Index Terms*—Optimal 2D views, sketch-based 3D shape retrieval, task-specific viewpoint learning.**

I. Introduction

DUE to cross-modality divergences between the sketch and shape domain, sketch-based 3D shape retrieval (SBSR) has become an extremely challenging research topic [1], [2], [3], [4], [5], [6], [7]. To cope with modality divergences, existing methods intentionally narrow the domain gap by using projection-based methods. These techniques involve projecting 3D shapes onto multi-view images in the 2D domain, and subsequently assessing the similarity between 3D shapes and 2D sketches through the matching of category information of the sketches with the corresponding 2D views. The quality of retrieved 3D shapes and 2D sketches is subject to the expressiveness of the rendered views. It is, therefore, of great significance to determine the most appropriate 2D views from 3D shapes, providing promising performance in producing accurate results for multi-view-based downstream tasks such as SBSR.

To this end, researchers manually examine the dataset to locate appropriate camera positions which can reflect geometric contents of a 3D shape to the utmost [8]. Although the aforementioned methods have achieved promising results on certain datasets, it is not practical to directly adopt those approaches in real-world situations that contain noise, occlusions, and arbitrary orientations. Chen et al. [8], [9] approach this problem by exhaustively generating 2D views from the surface of a sphere where 3D shapes are placed at the sphere center. A significant additional computational cost, however, is required for shape descriptors extraction, since this method has to aggregate information from every 2D view, which also introduces a large amount of redundant information.

Distinct from existing methods that unintentionally use preselected viewpoints and ineffective multi-view selection mechanisms for the SBSR, in this paper, our approach tackles this not fully explored area of SBSR by designing a novel framework. We argue that the ability to dynamically render 3D shapes based on query sketches is a crucial factor in achieving optimal results in SBSR. Fig. 1 highlights the primary distinctions between our proposed approach and the existing frameworks. Additionally, the ability to handle different drawing styles of 2D sketches serves as another important factor. As previously mentioned, query sketches guide the prediction of retrieval-specific camera viewpoints, highlighting the importance of extracting structural information from 2D sketches. Training the model to learn statistical structural information using a sufficiently large amount of 2D sketch training data is one intuitive approach. However, in real-world scenarios, 2D sketches may be acquired with significant variations in drawing styles due to the diverse drawing skills of real users. Collecting a sketch dataset that covers all possible drawing styles

Manuscript received 14 October 2022; revised 16 March 2023, 6 May 2023, and 26 May 2023; accepted 1 June 2023. Date of publication 19 June 2023; date of current version 21 February 2025. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. Yang Liu. *(Shuaihang Yuan and Congcong Wen contributed equally to this work.) (Corresponding author: Yi Fang.)*

Shuaihang Yuan, Congcong Wen, and Yi Fang are with the NYUAD Center for Artificial Intelligence and Robotics, New York University, Abu Dhabi 10012-1126, UAE, and with the NYU Multimedia and Visual Computing Lab, New York University, New York, NY 10012 USA, and also with the New York University, Abu Dhabi 10012-1126, UAE (e-mail: sy2366@nyu.edu; wencc@nyu.edu; yfang@nyu.edu).

Yu-Shen Liu is with the School of Software, Tsinghua University, Beijing 100084, China (e-mail: liuyushen@tsinghua.edu.cn).

Digital Object Identifier 10.1109/TMM.2023.3287332

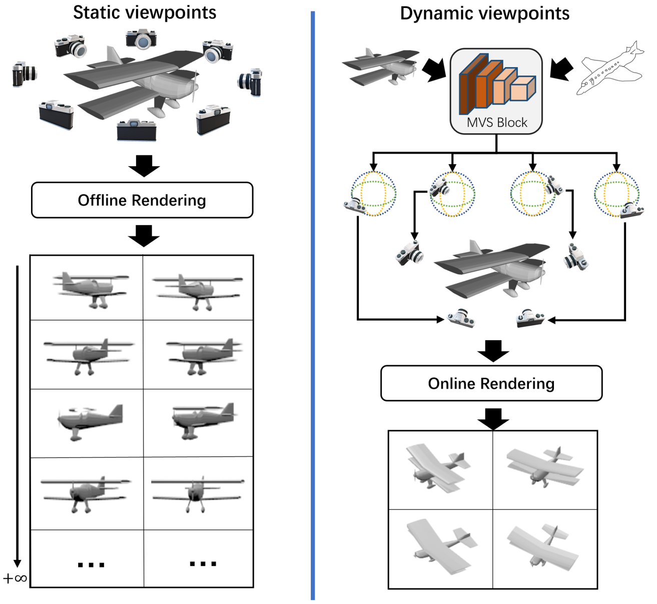

Fig. 1. Left: Existing view-based SBSR frameworks generate static views rendered from pre-selected viewpoints offline. Right: Our proposed method dynamically generates viewpoints according to sketches to render 3D objects.

is not practical. Therefore, it is essential to extract common patterns and structures across sketches, regardless of the different drawing styles.

In this work, we propose a new view learning paradigm specifically designed for category-level sketch-based 3D shape retrieval by considering the aforementioned two critical factors simultaneously. Our pipeline is presented in Fig. 2. As freehand sketches possess large style variations, the view learning method should be robust to various sketch styles. To this end, we propose a Dynamic Viewer (DV) module to render multi-view images for 3D objects through learned viewpoints such that the rendered 2D views are optimal for SBSR with the given query sketch. The proposed DV module contains two parts and adopts the model-based meta-learning method to train the DV module. Specifically, we introduce a Meta View Learner block, serving as a meta learner, to learn the network parameters for a part of the View Generator (VG) block. The VG block, working as the second part of the DV module, learns task-specific viewpoints from 3D point set according to the given 2D sketch for multi-view rendering. To ensure the DV module can generate optimal multi-view images according to a given sketch for the SBSR, we propose an unsupervised View Mining loss that maximizes the feature-level similarity of sketch and 2D views in latent embeddings. However, simply minimizing the distances of different embeddings in the latent space makes the training meaningless since such embeddings contain semantic information. As an illustration, the maximization of feature-level similarity in a common latent space between sketches and 3D shapes that belong to distinct categories would lead to the misclassification of their respective labels by the network. To circumvent this issue, inspired by [10], we adopt a variational autoencoder (VAE) to decompose an embedding of 2D view/sketch into a semantic embedding and an view-specific embedding. The semantic embedding represents the domain-invariant high-level semantic information of sketches and views, and view-specific embedding corresponds to the feature-level information of a sketch/view. The unsupervised view mining loss is computed over the projected view-specific embedding of multi-view images and the sketch, which tries to minimize the discrepancies of feature-level information in the common latent space. We evaluate our proposed novel approach using semantic embedding on SHREC 13, and SHREC 14 benchmark datasets for the task of category-level SBSR. The experimental results validate our argument by outperforming previous state-of-the-art approaches. Furthermore, with a slight modification, our method can be adapted to the task of instance-level SBSR. Our experimental results on the instance-level retrieval dataset, AmateurSketch [11], validate the effectiveness of our proposed method. In addition, an in-depth analysis of each proposed component and design validates their necessity. Our main contributions are summarized as follows:

- We propose a novel differentiable viewpoint learning paradigm that represents a pioneering endeavor in dynamically determining the optimal viewpoints for a given sketch, particularly for the sketch has not been previously encountered and is absent from the training data.
- We propose a new meta-learning framework designed to tackle the inherent challenges associated with non-unique mappings between two-dimensional sketches and three-dimensional shapes. These challenges arise due to the variability of observation viewpoints and the broad spectrum of styles present in human-drawn sketches.
- Extensive experimental results demonstrate the effectiveness of our approach for category-level sketch-based 3D shape retrieval tasks, as well as its generalizability to instance-level sketch-based 3D shape retrieval tasks.

II. Related Work

A. *Sketch-Based Shape Retrieval*

Sketch-based Shape Retrieval is a critical topic in computer vision, which typically involves two main objectives. The first objective is category-level SBSR, which aims to retrieve 3D objects from a database that belong to the same category as the query 2D sketch. With the thriving of deep learning, numerous sophisticated methods were proposed to address SBSR by extracting modality-invariant descriptors from sketches, and 3D object data [1], [2], [3], [4], [5], [6], [7], [12]. Wang et al. [12] learn descriptors of sketches and 3D shapes through two Siamese CNNs by minimizing intra-modality similarity and inter-modality discrepancy. Dai et al. [5] and Xie et al. [13] introduce Siamese Metric Networks to tackle the task of SBSR. In addition to the Siamese networks, evidence shows that Generative Adversarial Networks can learn a domain invariant descriptor to address this problem [14]. Those approaches use off-the-shelf renderers to generate multiple 2D views of 3D objects on behalf of the 3D descriptor extraction process, which requires human labor to manually examine the dataset to locate appropriate camera positions such that the resulting 2D views can provide sufficient details of 3D objects. However, it is not practical to manually

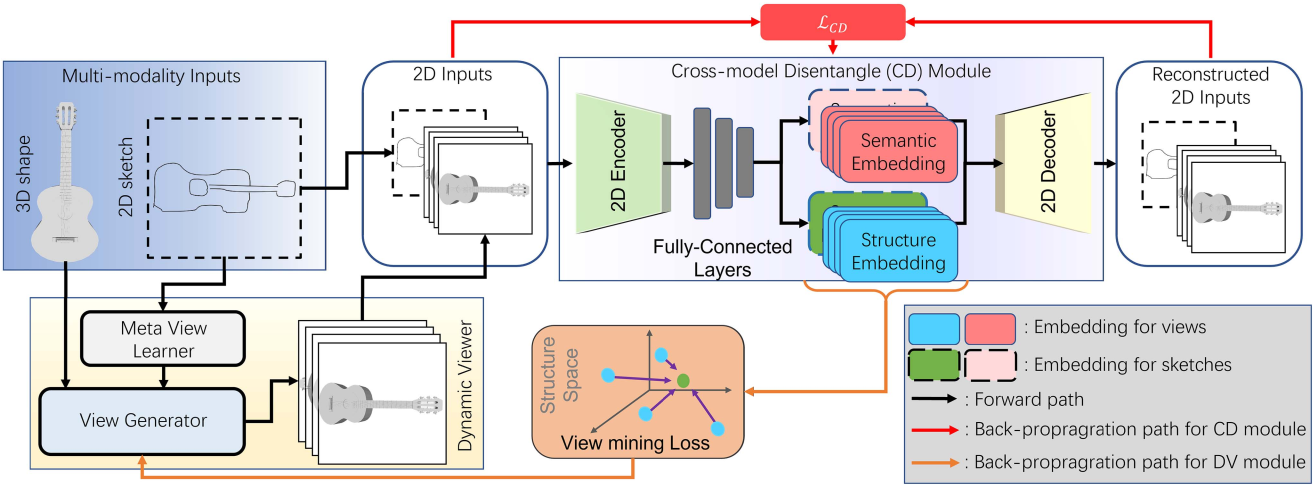

Fig. 2. Illustration of our proposed framework. Our framework first adopts a Dynamic Viewer module, represented in the yellow box, to render views according to a given sketch. The detailed information on the DV module can be found in Section III-B. Based on rendered views, as well as the input sketches, a cross-model disentangle (CD) module is adopted to embed views and sketches into disentangled latent embeddings.

place cameras for every 3D shape acquired from the real world. To address this issue, DSSH [9] introduces a stochastic sampling method that randomly selects rendering views from the sphere around a 3D shape. As the number of sampling iterations increases, the acquired views can cover the whole shape. Although this method avoids manually finding camera positions for view rendering, it requires sufficient iterations to achieve a promising performance. Besides category-level SBSR, instance-level (fine-grained) SBSR is increasingly gaining researchers' attention due to its widespread applications in various real-world scenarios. In comparison to category-level SBSR, instance-level SBSR is more challenging as the model must discern the most similar 3D objects in terms of both structure and geometry rather than relying solely on semantic similarities. Qi et al. [11] propose the first work for instance-level SBSR. This paper proposes a cross-modal-deep embedding model that includes a novel cross-modal view attention module that automatically computes the optimal combination of 2D projections of a 3D shape given a query sketch. While this method has shown promising results, it relies on a static set of viewpoints to project 3D shapes. To set us apart from the aforementioned methods that achieve SBSR by either finding camera positions to render views manually or adopting an exhaustive sampling strategy, we introduce a novel DV module that can learn to derive expressive views according to a given sketch.

B. *Viewpoint Selection in SBSR*

Due to the ill-defined optimal viewpoints, various methods have been proposed to improve their selection process. Eitz et al. [2] use projected area and occluded contours of a 3D shape to define whether a viewpoint is optimal. Further, Yasseen et al. [15] introduce a method to quantify viewpoints by measuring the occlusion area and symmetry parts of projected 3D shapes. In addition, other researchers [16], [17] defined the optimal viewpoint as a position where people usually draw shapes from it. To this end, those works use the similarity between the rendered image and the target images as the objective function to learn optimal viewpoints. Xu et al. [18] exhaustively render 2D views from the surface of a sphere where 3D shapes are placed at the sphere center. By ranking feature level similarity between rendered image and the target offline, Xu et al. select first top-k viewpoints as optimal viewpoints. Then, those pre-selected viewpoints are used for all sketches for the task of SBSR. Distinct from the approaches mentioned above, we propose dynamically determining the minimum number of viewpoints that are optimal for SBSR based on the query sketch online. We further introduce a view mining loss to train the network by maximizing the feature-level similarity between the sketch and multi-view images. Furthermore, our method is fully differentiable and can be learned in an end-to-end fashion.

C. *Meta-Learning*

Meta-learning is employed to address the situation where there are limited training data available for each object category. As a solution for few-shot learning, meta-learning can be categorized into three branches:

1) Metric-Based Methods: tackle issues that arise from directly training the deep neural network model with a small training dataset, thus causing the model to overfit. To avoid this, metric-learning-based methods adopt a non-parametric learner as a classifier to directly compare the feature of input data and predict the label. A parametric learner is further used to opt for an appropriate feature with large inter-class variation and small intra-class variation. Different frameworks, such as Siamese Networks [19], Matching Networks [20], Prototypical Networks [21], and Relation Networks [22], realize the metric-based methods.

2) Gradient-Based Methods: construct a meta-learner that learns knowledge from a few data benefiting a classifier to achieve a quick convergence. For example, in MAML [23], a meta-learner learns the initial parameters of a classifier from training data so that after a small number of back-propagation steps, the classifier achieves good performance on few-shot tasks. In addition to learning initial network parameters, this method can also tune the network hyper-paremeters [24].

3) Model-Based Methods: Refs. [25], [26] rely on two different parametric learners where the first one works as a meta-learner and the second one works as a classifier. The classifier has only a feed-forward pass to predict the corresponding data labels, following this process, the meta-learner infers its network parameters. These methods yield efficient learning depending on back-propagating the gradient of the meta-learner to the classifier. In this study, we utilize meta-learning techniques to enhance the capability of our model to identify common patterns and structures of 2D sketches, enabling it to generalize well to unseen sketch styles.

III. Method

A. Framework Overview

As presented in Fig. 2, our SBSR framework is comprised of two modules: 1) the Dynamic Viewer module and 2) the Cross-Model Disentangle (CD) module. Considering a training dataset with K categories, $\{C^k\}_{\{k=1,2,\ldots,K\}}$, we assume that each category is composed of N_s^k sketches and N_o^k 3D objects. We represent the input training data by pairing sketches and 3D objects of the same category. Specifically, we divide the whole training data into mini-batches. For each mini-batch, we randomly select N_c categories, and for each category, we sample N_m sketches and 3D objects. The training mini-batch is defined as $D = \{(\{s_i^1\}_{i=1}^{N_m}, \{o_i^1\}_{i=1}^{N_m}), \ldots, (\{s_i^{N_c}\}_{i=1}^{N_m}, \{o_i^{N_c}\}_{i=1}^{N_m})\}$, where the training sketch mini-batch is $S = \{\{s_i^1\}_{i=1}^{N_m}, \ldots, \{s_i^{N_c}\}_{i=1}^{N_m}\}$, the training 3D object mini-batch is $O = \{\{o_i^1\}_{i=1}^{N_m}, \ldots, \{o_i^{N_c}\}_{i=1}^{N_m}\}$, and the corresponding label set is $Y = \{C_1, \ldots, C_k\}$. Existing SBSR methods [3], [4], [5], [6], [12] use offline renderers to render 3D shapes from pre-selected viewpoints and apply view-based 3D representation learning methods to extract shape descriptors. This offline rendering process prevents researchers from using learning-based methods to learn to generate optimal 2D views for the task of SBSR since the gradient cannot be back-propagated from the rendered multi-view images to the learned viewpoints. In this work, we use a differentiable renderer as a part of our framework such that the optimal viewpoints can be learned via back-propagation. Specifically, we feed each sketch-3D object pair to the Dynamic Viewer module to render a set of 2D views. Then, rendered 2D views and sketches are sent to a cross-model disentangle module to disentangle the latent embeddings into semantic and view-specific parts. The proposed view mining loss is calculated over view-specific embedding to mine the camera positions such that the similarity of feature-level information between the rendered views and the query sketch can be maximized in a shared latent space. We evaluate the object retrieval performance upon the semantic embedding part.

B. Dynamic Viewer

We integrate the rendering process into the SBSR framework by introducing a Dynamic Viewer (DV) module. The DV module learns to predict camera viewpoints to render the optimal sketch-dependent 2D views guided by query sketches, which requires our model to extract structural information from 2D sketches. As mentioned earlier, 2D sketches often involved with significant variations in drawing styles resulting from diverse user drawing skills. To this end, extracting common patterns and structures across sketches becomes crucial regardless of the varying drawing styles. To this end, we leverage the power of model-based meta-learning to learn drawing style features for sketch-dependent viewpoints prediction. And we use a differentiable renderer to render 3D objects through the predicted viewpoints to generate multi-view images.

1) Task Sampling: To simulate sketch style variations at training, we randomly select p sketch-3D object pairs from each category of the training mini-batch D as the support set $D_{train} = \{(\{s_i^1\}_{i=1}^{p}, \{o_i^1\}_{i=1}^{p}), \ldots, (\{s_i^{N_c}\}_{i=1}^{p}, \{o_i^{N_c}\}_{i=1}^{p})\}$, and the remaining $N_m - p$ pairs of each category from the query set D_{val}. To incorporate the model-based meta-learning method, we define our base (or inner) learning algorithm as a deep learning model that solves the problem of sketch descriptor extraction. We further employ another deep learning network to modulate the base learning algorithm by learning its network parameters as the meta (or outer) algorithm. We use sketches from D_{train} to update parameters of the inner algorithm and D_{val} from the same category to calculate the loss term defined in Section III-D. In this case, the optimizations of inner and outer algorithms are tightly coupled and make the training process entirely feed-forward. We propose two new blocks to achieve this feed-forward meta-learning module.

2) View Generator: The first block, working as an inner learning algorithm, aims to render 2D views for a given 3D object, which are optimal for SBSR. To integrate this rendering process into the feed-forward training process, we leverage a differentiable renderer $r_\epsilon(\cdot) : o \rightarrow I$ to render a set of 2D views I by inputting a 3D shape and cameras parameterized by ϵ, i.e., azimuth and elevation angle of cameras. The gradient with respect to camera parameters can be computed and propagated backward through the differentiable renderer. It is, therefore, possible to design a network to learn optimal viewpoints since the whole training pipeline is fully differentiable.

We introduce the Dynamic Viewer Module, composed of a 3D descriptor learning backbone $h_{o_{\omega_o}}(\cdot) : o \rightarrow z_o$ with network parameter ω_o, a descriptor projection network $m_\varphi(\cdot) : z_o \rightarrow z_o^*$ and a viewpoints predictor $v_\phi(\cdot) : [z_o^*, z_o] \rightarrow \epsilon$ are proposed in this block, where $[\cdot, \cdot]$ is the concatenation operation, and φ and ϕ represent the network parameters of two network, respectively. We use Multilayer Perceptrons (MLPs) as $m_\varphi(\cdot)$, which have N_h hidden layers to project the 3D descriptor according to the 2D sketch descriptor. The computation of each layer is defined as $y = \alpha x + \gamma$, where y and x stand for the output and the input of each layer, respectively. α and γ represent the weight matrix and the bias vector. The projected descriptor concatenated with the 3D descriptor is then sent to $v_\phi(\cdot)$ to regress the camera parameters ϵ.

3) Meta View Learner: In order to allow the View Generator to acquire the ability to produce the most similar views as the given sketches, we have to leverage structure information, such as orientations, of 2D sketches to guide the prediction of viewpoints. One intuitive way is to fuse the information of sketches with 3D shapes using a concatenation operation or attention mechanism. Although those methods have achieved promising results, there is a sever performance drop as models encounter sketch drawings in styles that have never been seen before. Inspired by the model-based meta-learning, we propose a Meta View Learner (MVL) block as the meta-learner to address this issue. Specifically, the proposed MVL aggregate information from two modalities and enable the View Generator to handle different drawing styles of 2D sketches by predicting the network parameters of $m_\varphi(\cdot)$ from the input sketches. More concretely, the MVL block uses a 2D descriptor learning backbone $h_{s_{\omega_s}}(\cdot) : s \rightarrow z_s$ parameterized by ω_s to obtain the sketch descriptor. To predict the parameters φ of $m_\varphi(\cdot)$ with N_h layers, we adopt the same number of MLPs $\{f_{\psi_l}(\cdot) : z_s \rightarrow \varphi_l\}_{l=1}^{N_h}$ to regress α and γ for the lth layer of $m_\varphi(\cdot)$. By adopting the MVL block, the training of $m_\varphi(\cdot)$ becomes the training of $f_{\psi_l}(\cdot)$. By leveraging the meta-learning mechanism DV learns more generalized and common patterns and structures across different drawing styles which reduces the amount of training data needed for training.

The final DV module consists of $h_{o_{\omega_o}}(\cdot)$, $h_{s_{\omega_s}}(\cdot)$, $m_\varphi(\cdot)$, $v_\phi(\cdot)$, and $\{f_{\psi_l}(\cdot)\}_{l=1}^{N_h}$. The network parameters of $m_\varphi(\cdot)$ are the output of $\{f_{\psi_l}(\cdot)\}_{l=1}^{N_h}$. Thus, during the training, φ will not be updated by back-propagation. The final meta-learning based DV module is represented by the following functions:

$$\varphi = \{f_{\psi_l}(h_{s_{\omega_s}}(s_{train}))\}_{l=1}^{N_h}, \tag{1}$$

$$\epsilon = v_\phi([m_\varphi(o_{val}), h_{o_{\omega_o}}(o_{val})]), \tag{2}$$

$$I = r_\epsilon(o_{val}), \tag{3}$$

where s_{train} is a sketch from D_{train} and o_{val} is a 3D shape from D_{val}. In this work, we use the DGCNN [27] as the descriptor learning backbone $h_{o_{\omega_o}}(\cdot)$. We adopt Inception-V2 as $h_{s_{\omega_s}}(\cdot)$ Detailed configuration and implementation of $m_\varphi(\cdot)$, $v_\phi(\cdot)$, and $\{f_{\psi_l}(\cdot)\}_{l=1}^{N_h}$ can be found in Section IV. The final learning process can be formulated as following:

$$\underset{\omega_s,\omega_o,\theta,\psi}{\arg\min} \underset{\substack{T\sim p(T)\\(D_{train},D_{val})\in T}}{\mathbb{E}} \sum_{D_{val}} [\mathcal{L}(D_{train}, D_{val})], \tag{4}$$

where the $\mathcal{L}$ is the final loss term, which is described in Section III-C. From (4), we meta-train the DV module by optimizing over sampled tasks from a task distribution. $\{f_{\psi_l}(\cdot)\}_{l=1}^{N_h}$ maps a sketch of D_{train} into the weights of $m_\varphi(\cdot)$ to predict sketch style descriptors. Therefore, $m_\varphi(\cdot)$ should acquire the ability to adapt to new sketches sampled from the test set. To train the DV module to render the optimal 2D views for SBSR, we further introduce a viewpoint mining strategy that uses a cross-modality disentangle module and a view mining loss.

C. Viewpoint Mining

Since the ground truth viewpoints that are optimal for SBSR are not practical to obtain for different sketches, we propose a self-supervised surrogate method to mine viewpoints by maximizing the similarity of their feature-level information, i.e., latent space embedding. Since the latent space embedding contains the semantic information, directly constructing a loss function to maximize the similarity of latent embeddings makes training of SBSR meaningless. For example, we assume that the input sketch and 3D shape are taken from different categories. Maximizing the similarity of their embeddings by simply minimizing their distance in the latent space will faulty make the network classify them into the same category. Motivated by the [10], we have implemented a cross-modal disentanglement module that decomposes the embeddings into semantic parts and view-specific parts, aiming to tackle this problem. We calculate the view mining loss over view-specific parts to maximize the similarity of their semantic feature-level information.

1) Cross-Model Disentanglement: We adopt a variational autoencoder (VAE) to disentangle sketches and 2D views embeddings into a semantic part and a view-specific part such that the semantic part is invariant for sketches and views belonging to the same category, and the view-specific part is unique for each sketch or view. Prior to presenting our disentanglement module, we first briefly review how the VAE works and how to apply VAE to the cross-modality disentangle module.

VAE optimizes evidence lower bound (ELBO) for log-likelihood of given data distribution using Kullback-Leibler (KL) divergence, which is defined as follows:

$$\log p(x) \geq \underset{z\sim q(z|x)}{E}[\log p(x|z)] - D_{KL}(q(z|x)||p(z)). \tag{5}$$

We set $p(z)$ to be a normal distribution $\mathcal{N}(z|0, 1)$. The conditional probability distributions $q(z|x)$ and $p(x|z)$ are represented by the encoder and the decoder of VAE. μ and σ^2 of a normal distribution $\mathcal{N}(\mu, \sigma^2)$ will be returned by the encoder for the latent embedding z, where $z \sim \mathcal{N}(\mu, \sigma^2)$. We then extend the basic VAE to the cross-modality setting followed by the [28]. Specifically, we have data from two modalities, A and B, and (5) is reformulated:

$$\log p(A) \geq \underset{z\sim q(z|B)}{E}[\log p(A|z)] - D_{KL}(q(z|B)||p(z)). \tag{6}$$

We further follow [10] to disentangle sketches and 2D view embeddings. Concretely, the encoder produces three outputs, including a semantic embedding z_{sem}, a mean μ, and a variance σ. With μ and σ, we derive the view-specific embedding z_{vs} following the idea of [28], where $z_{vs} = \mu + \sigma \bigodot \mathcal{N}(0, 1)$. We perform an element-wise summation over z_{sem} and z_{vs} forming the input to the decoder, z, which is used to reconstruct views or sketches $\hat{x}$. By learning the mapping function that maps the input 2D data into semantic embedding and view-specific embedding, we can further construct our view mining loss.

2) View Mining Loss: In our work, view mining locates a set of viewpoints to render the 3D object such that rendered views are optimal for the SBSR based on a query sketch. We introduce a self-supervised view mining loss to facilitate the SBSR process by maximizing the feature-level information similarly between each sketch and 2D view. Specifically, Given the view-specific embedding of a sketch z_{vs}^s and a set of 2D views $\{z_{vs}^I\}_{I=1}^{N_v}$, where N_v is the number of multi-view images, two different MLPs project z_{vs}^s and $\{z_{vs}^I\}_{I=1}^{N_v}$ to a common latent space and form

$\hat{z}^s_{vs}$ and $\{\hat{z}^I_{vs}\}_{I=1}^{N_v}$. Our proposed view mining loss is formulated as follows:

$$\mathcal{L}_{\textbf{view}} = \sum_{I=1}^{N_v} \|\hat{z}^I_{vs} - \hat{z}^s_{vs}\|^2 \tag{7}$$

D. Objective Function

We train the cross-modal disentangle module with four objective functions following the same setting in [10] to ensure its disentangling ability. The first objective function measures the reconstruction quality of the disentangle module:

$$\mathcal{L}_{rec} = \|decoder(z_{sem} + z_{vs}) - x\|_2 \tag{8}$$

where x, z_{sem}, and z_{vs} are input data, corresponding semantic embedding, and corresponding view-specific embedding coming from either the sketch domain or the 2D view domain. $decoder$ denotes the decoder of the CD module. The second function measures the KL divergence:

$$\mathcal{L}_{KL} = D_{KL}[q(z|x)||p(z)], \tag{9}$$

where $p(z) = \mathcal{N}(z; 0, I)$. Further, a triple loss on the semantic embeddings and final fused embedding are adopted to train the VAE model:

$$\begin{aligned}\mathcal{L}_{tri} = &\max\left\{0, m_{z_{sem}} + \|z^s_{sem} - z^I_{sem}\|^2 - \|z^s_{sem} - z^{I_n}_{sem}\|^2\right\}\\ &+ \max\left\{0, m_z + \|z^s - z^I\|^2 - \|z^s - z^{I_n}\|^2\right\},\end{aligned} \tag{10}$$

where $m_{z_{sem}}$ and $m_{z_{sem}}$ are margin hyper-parameters. I_n is a view from other categories that are different from the category where I belongs. Empirically, we add the second term of (10) to help the reconstruction process. The final objective of training the cross-domain disentangle module is:

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda_1\mathcal{L}_{KL} + \lambda_2\mathcal{L}_{tri} + \lambda_3\mathcal{L}_{view}, \tag{11}$$

where λ_1, λ_2, and λ_3 are hyper-parameters.

IV. Experiments

In this section, to demonstrate that our model can learn to select viewpoints according to a given sketch dynamically, we perform experiments on multiple datasets to evaluate our proposed method. In Section IV-A, we first present a detailed introduction of datasets used in our experiments. In Section IV-B, we show the detailed implementation and present the evaluation metrics in Section IV-C. In Section IV-D, we compare our model with existing category-level SBSR approaches. In Section IV-F1 we extend our method to the instance-level SBSR. We then investigate the computational cost of our proposed module in Section IV-F2. In Section IV-E, we provide the in-depth analysis of our proposed model to validate the effectiveness as well as the necessity of designed component.

A. Dataset

In our work, we evaluate the proposed framework on two SBSR datasets.

SHREC 13 [29] is composed of 90 object categories with a total of 7,200 sketches and 1,258 shapes. The 3D objects of the SHREC 13 dataset are directly adopted from the PSB dataset, and there are around 14 3D objects for each category. SHREC 13 dataset contains 80 freehand sketches for each category, which are relevant to 3D models in the Princeton Shape Benchmark dataset [30]. Moreover, 50 sketches of each category are reserved for training, and the rest is for testing.

SHREC 14 [31] is an extension of the SHREC 13 dataset, which is composed of 171 object categories with a total of 13,680 sketches and 8,987 3D objects. The 2D sketch query set contains 80 sketches for each category; in particular, 8,550 sketches are the training data, and the rest are the test data. Moreover, 3D object sets classify 8,987 objects into 171 categories, with each class having around 53 models.

B. Implementation and Training Details

1) Dynamic Viewer: We use Inception-V2 as $h_{s_{\omega_s}}(\cdot)$ to map input sketches from D_{val} to 1024 dimensional sketch descriptors. $m_\varphi(\cdot)$ takes this sketch descriptor as input to predict the sketch descriptor. $m_\varphi(\cdot)$ includes three hidden layers l_1, l_2, and l_3, where $l_1 : \mathbb{R}^{1024} \rightarrow \mathbb{R}^{1024}$, $l_2 : \mathbb{R}^{1024} \rightarrow \mathbb{R}^{512}$, and $l_3 : \mathbb{R}^{512} \rightarrow \mathbb{R}^{512}$. The computation of each layer is defined by $y = \alpha x + \gamma$. The weight matrix α and the bias vector γ of each layer are learned by a $f_{\psi_l}(\cdot)$. $\{f^l_{\psi_l}(\cdot)\}_{l=1}^3$ take the sketch descriptors extracted from D_{train} as input and regress α and γ. Each $f_{\psi_l}(\cdot)$ is an MLP that consists of two successive hidden layers with sizes of 512 and 256. We utilize a DGCNN network [27] as our 3D backbone, $h_{o_{\omega_o}}(\cdot)$, to extract 3D descriptors represented in 512-dimensional vectors. The stacked 3D and 2D descriptors extracted by $h_{o_{\omega_o}}(\cdot)$ and $m_\varphi(\cdot)$ are sent to an MLP $v_\phi(\cdot)$ to regress N_v camera positions. Each camera position is represented by an azimuth angle and an elevation angle, and N_v varies according to different experimental settings. Pytorch3D differentiable renderer [32] takes predicted camera positions to online render multi-view images.

2) Viewpoint Mining: We follow [10] to implement the cross-modal disentangle module. We adopt the same $h_{s_{\omega_s}}(\cdot)$ as an encoder and use MLP to project a 2D descriptor into three 256-dimensional latent vectors representing μ, σ, and z_{sem}, respectively. For the decoder of VAE, we construct a sequence of fractionally-strided convolutions with Normalization layers. Each layer follows a ReLu activation function except the final output layer, which is activated by a tanh function. To calculate the view mining loss, we further use MLP with one hidden layer to project z_{vs} of sketch and 2D views into a common space to obtain $\hat{z}_{vs}$. We adopt a GAP global average pooling layer to fuse information from multi-view images. Our framework is implemented in Pytorch trained with an Adam optimizer using a batch size of 8 with a learning rate of 0.0001. The λ_1 λ_2 and λ_3 from (11) are set to be 0.001, 1, and 0.01, respectively.

C. Evaluation Metrics and Protocols

We evaluate our proposed SBSR framework on both category-level and instance-level SBSR. We use the following evaluation metrics to evaluate the performance of category-level SBSR: 1) nearest neighbor (**NN**), 2) first tier (**FT**), 3) second tier (**ST**), 4) E-measure (**E**), 5) discounted cumulated gain (**DCG**), and 6)

TABLE I

SKETCH-BASED 3D SHAPE RETRIEVAL RESULTS ON SHREC 13 AND 14 DATASETS. WE USE THE TERM "OURS (N_v)" TO REPRESENT OUR METHOD FOR N_v RENDERING VIEWS

	SHREC 13						SHREC 14					
Methods	**NN**	**FT**	**ST**	**E**	**DCG**	**mAP**	**NN**	**FT**	**ST**	**E**	**DCG**	**mAP**
CDMR [33]	27.9	20.3	29.6	16.6	45.8	25.0	10.9	5.7	8.9	4.1	32.8	5.4
SBR-VC [29]	16.4	9.7	14.9	8.5	34.8	11.4	9.5	5.0	8.1	3.7	31.9	5.0
CAT-DTW [15]	23.5	13.5	19.8	10.9	39.2	14.1	13.7	6.8	10.2	5.0	33.8	6.0
Siamese [12]	40.5	40.3	54.8	28.7	60.7	46.9	23.9	21.2	31.6	14.0	49.6	22.8
DCML [5]	65.0	63.4	71.9	34.8	76.6	67.4	27.2	27.5	34.5	17.1	49.8	28.6
DCHML [6]	73.0	71.5	77.3	36.8	81.6	74.4	40.3	32.9	39.4	20.1	54.4	33.6
LWBR [34]	71.2	72.5	78.5	36.9	81.4	75.2	40.3	37.8	45.5	23.6	58.1	40.1
DCA [14]	78.3	79.6	82.9	37.6	85.6	81.3	77.0	78.9	82.3	39.8	85.9	80.3
Semantic [7]	82.3	82.8	86.0	40.3	88.4	84.3	80.4	74.9	81.3	39.5	87.0	78.0
DSSH [9]	83.1	84.4	88.6	41.1	89.3	85.8	79.6	81.3	85.1	41.2	88.1	82.6
SSM [35]	83.6	83.3	88.3	41.1	89.6	85.3	-	-	-	-	-	-
Guidance [36]	83.6	85.5	**90.2**	42.5	**90.5**	**87.1**	79.9	82.1	86.2	41.9	88.6	83.5
SLC [37]	82.7	84.1	89.5	41.8	89.5	86.4	78.1	80.4	84.7	41.5	87.2	82.0
JFLN [38]	**84.0**	**85.8**	89.9	42.3	89.7	86.6	79.2	82.3	84.7	**42.4**	87.3	83.3
Ours (1)	75.2	77.3	79.4	38.6	84.9	79.3	68.7	64.5	74.2	72.9	77.8	73.7
Ours (6)	80.3	82.4	85.2	38.4	84.5	83.4	82.2	83.8	83.6	39.0	84.3	82.5
Ours (12)	83.3	85.1	89.9	**42.5**	90.4	87.0	**80.6**	**84.2**	**88.7**	41.4	**88.8**	**85.1**

mean average precision (**mAP**). In the standard evaluation protocols, we calculate shape and sketch descriptors for all objects from the 3D database and query sketches for matching. This evaluation method can be directly adapted to the existing SBSR framework because the calculation of 3D descriptors solely depends on the trained network, which is principally different from our approach. Our method calculates 3D descriptors not only depending on the trained network but also depending on the given query sketches since rendered views vary for different sketches, which leads to dynamic 3D descriptors for each 3D object. In our setting, we calculate **NN**, **FT**, **ST**, **E**, **DCG**, and **mAP** over each query sketch and its corresponding 3D descriptors. We report the average score of each evaluation metrics. In addition, we follow the evaluation adopted in Qi et al. [11] to evaluate the instance-level SBSR. To assess the efficacy of instance-level SBSR, we follow Qi et al. [11] and employ a widely recognized evaluation metric known as the Top-K retrieval accuracy (acc@K). This metric computes the proportion of query sketches for which the corresponding 3D shapes appeared in the top-K retrieved results.

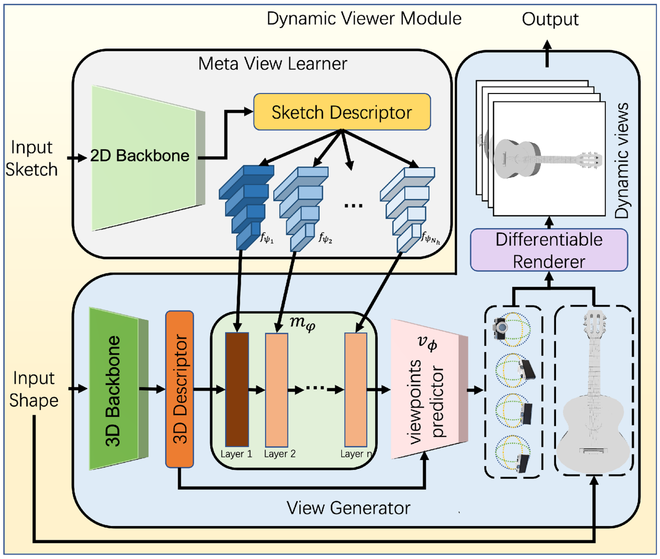

Fig. 3. Detailed illustration of the proposed DV Module. The DV module consists of two blocks. Meta View Learner block serves as a meta-learner to learn the network parameters of $v_\phi(\cdot)$. View Generator Module renders multi-view images from 3D shapes based on the given sketch.

D. Results

1) Experiment Setup: We compare our proposed framework with the state-of-the-art methods for sketch-based 3D shape retrieval [5], [6], [7], [12]. To validate the effect of using different numbers of camera positions, we provide the results of our framework with varied N_v. To evaluate the SBSR performance when $N_v > 1$, We use the global average pooling to obtain z_{sem} for multi-view images. Moreover, In all the experiments, we set the number of prediction heads N_h to be 3.

2) Experiment Result: Table I shows the SBSR result on SHREC 13 dataset. The performance of our proposed method achieves new state-of-the-art results compared with existing methods. As presented in Table I, our model achieves the performance of 87.0% mAP on the SHREC 13 dataset with $N_v = 12$ which is on-par with the most recent method. Even with an extremely small number of views (e.g., one view per 3D object), our framework still achieves reasonable results. Note that DSSH [9] adopts a view sampling strategy to sample one view from four non-overlapping spaces [9]. As the sampling time increase, the sampled views can cover the whole space. During the inference, DSSH repeats the view sampling ten times, which yields a significant number of views for the actual testing phase. Compared with DSSH, our framework can achieve comparable results, whilst only a few views are used because the designed DV module can dynamically select viewpoints that acquire the most similar visual content to the sketch; thus, the encoder network can produce similar 2D descriptors for matching. As we increase the N_v, better SBSR performance is achieved. The qualitative SBSR results on SHREC 13 can be found in Fig. 5. The

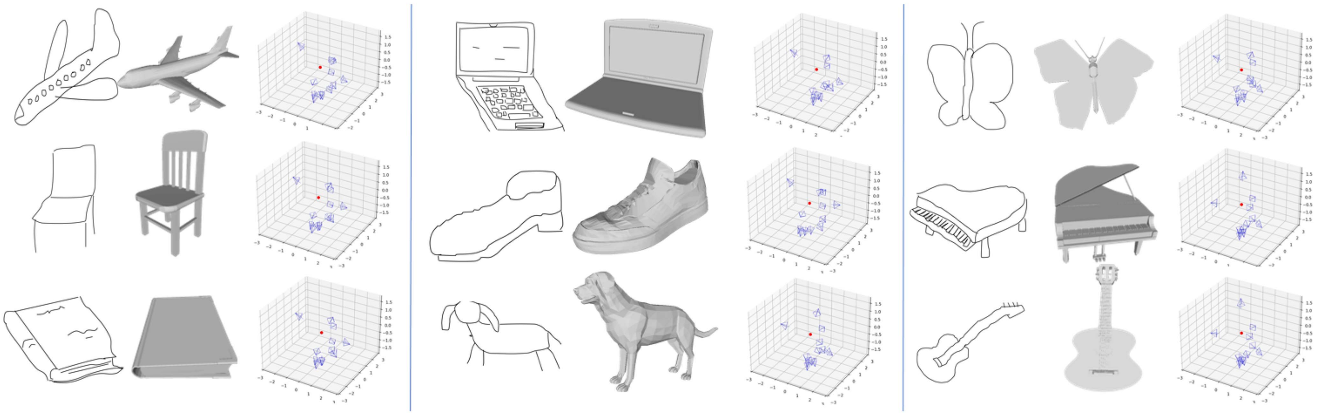

Fig. 4. Visualization of predicted camera positions.

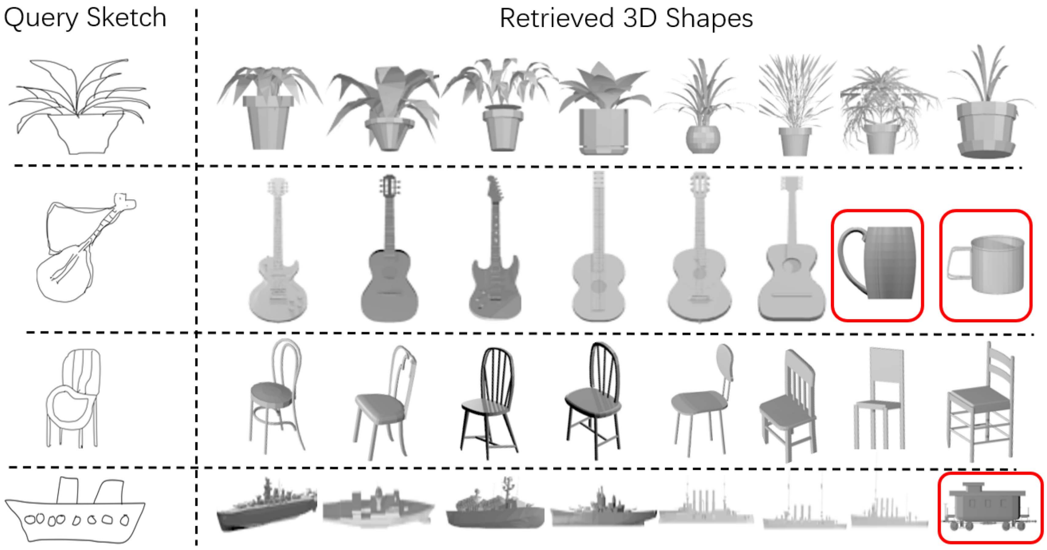

Fig. 5. 3D shapes retrieved by given query sketches on SHREC 13 dataset.

quantitative results on SHREC 14 dataset are summarized in Table I, where the model achieves the best retrieval map of 85.1%. Our proposed method outperforms the most recent method in category-level SBSR on the SHREC 14 dataset. This improvement can be attributed to the meta-learning strategies that we have developed, which are adept at handling large variations in drawing styles. The SHREC 14 dataset is nearly twice the size of the SHREC 13 dataset and consequently exhibits a greater range of drawing style variations. The improvement of SBSR performance on the SHREC 14 dataset validate the efficacy of our meta-learning design. Moreover, we also show the camera position predicted by our method in Fig. 4. We demonstrate that the learned camera position can capture the necessary information to categorize an object with the utmost precision.

E. Ablation Studies

1) Effect of the Dynamic Viewer Module: We assess the importance of the Dynamic View module by removing the DV module from our pipeline. We use an off-the-shelf renderer to render different numbers of views for each 3D object following the offline rendering process of existing SBSR methods [5], [6], [14]. Then, disentangle module takes each view and sketch as input and produces latent vectors for retrieval. We use the experimental setup for $N_v > 1$ cases presented in Section IV-D. Quantitative results can be found in Table II. By adopting the DV module, promising SBSR performance can be achieved even with a few views. As the number of views increases, the performance of the framework without a DV module increases whilst the performance of our proposed framework (with a DV module) remains relatively unchanged. However, the performance

TABLE II
ABLATION STUDIES ON SHREC 13 DATASET. WE EXAM THE EFFECTIVENESS OF OUR PROPOSED DV MODULE. #V REPRESENTS NUMBER OF VIEWS

Method	**#V**	**NN**	**FT**	**ST**	**E**	**DCG**	**mAP**
Ours (w/o DV)	6	76.3	78.9	80.5	34.1	80.6	79.2
Ours	6	80.3	82.4	85.2	38.4	84.5	83.4
Ours (w/o DV)	12	79.0	81.2	83.6	37.1	84.7	82.8
Ours	12	**83.3**	**85.1**	**89.9**	**42.5**	**90.4**	**87.0**
Ours (w/o DV)	18	81.4	83.2	85.9	38.9	87.9	84.1
Ours	18	83.3	85.1	89.8	42.8	90.5	**87.0**

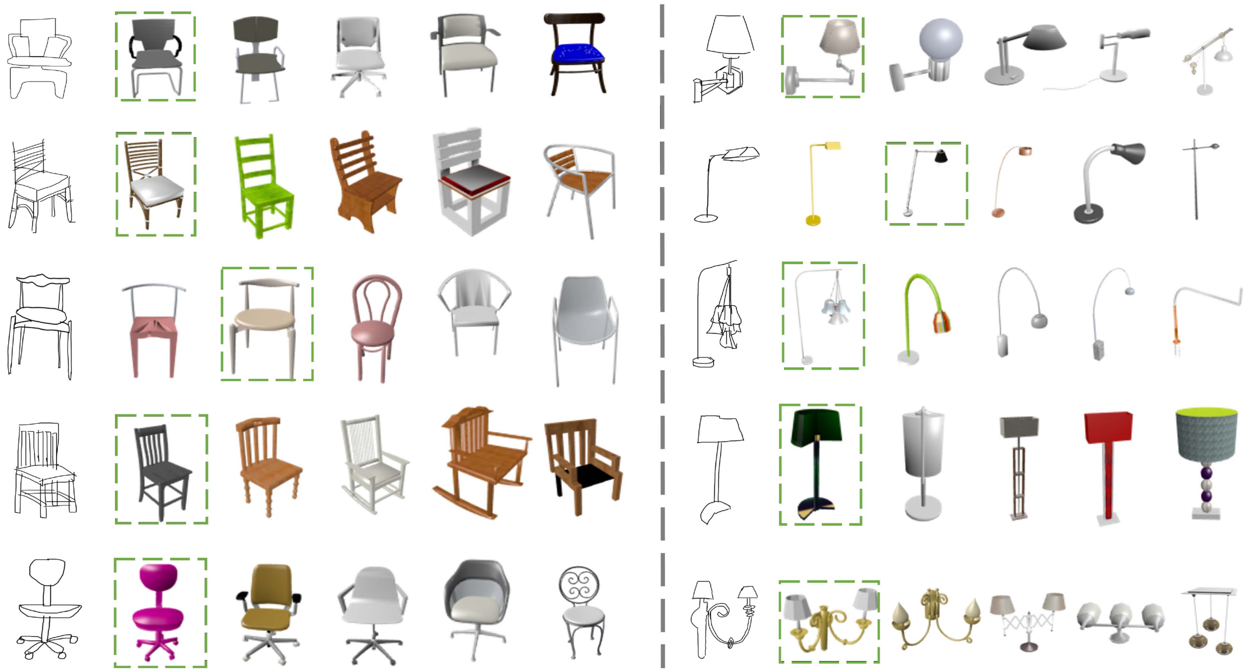

Fig. 6. Qualitative results for instance-level sketch-based 3D shape retrieval of our proposed method on the AmateurSketch Dataset. Follow the work [11]. We show the top 5 ranked 3D shapes in the database in each row. Green-dashed line rectangles represent the ground truth for each sketch.

achieved by using a DV module is still better than the performance without utilizing a DV module.

2) Effect of the Meta View Learner: To address the challenge of a single 3D shape being represented by an infinite number of 2D sketches, which can vary in viewpoints and drawing styles, traditional approaches of training a single view generator have limitations such as being unable to generate viewpoints for unseen sketches and not capturing variations resulting from sketch differences. As a solution, our proposed Meta View Learner (MVL) models the problem within a meta-learning framework and predicts a novel view generator for a given 2D sketch, achieving exceptional performance with only a few 2D sketch-to-3D shape pairs. In order to demonstrate the effectiveness of MVL, we perform a modification on the training set of SHREC 14 by randomly removing 5%, 10%, 15%, 20%, 25%, 30%, 35%, 40%, 45%, 50% of training sketches from each category. The results of our experiments, as illustrated in Fig. 7, demonstrate that the proposed method can achieve reasonable performance even when only 50% of the training data is provided. This ablation study confirms the ability of our approach to handle different and unseen sketches by outperforming the most recent method on a larger dataset, such as SHREC 14. This ablation study further supports the superiority of our proposed approach over the most recent method on a larger dataset such as SHREC 14, as the testing samples (i.e., unseen sketches with potentially different drawing styles) are more diverse compared to those in SHREC 13. The findings from this study provide strong evidence for the ability of the proposed module to handle diverse drawing styles of 2D sketches, which is a significant advantage in real-world applications where the drawing styles may vary widely.

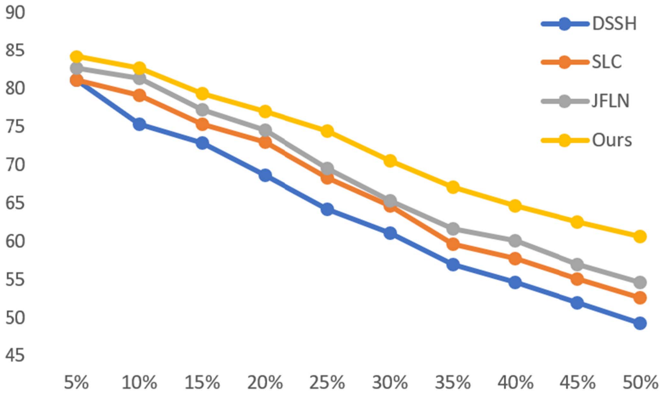

Fig. 7. Ablation studies on the SBSR performance when using different number of training sketches.

In summary, the proposed meta-learning approach offers a promising solution to the complex problem of generating optimal viewpoints for a variety of 2D sketches representing 3D shapes. By utilizing a meta-learning framework, our approach can adapt to diverse drawing styles and viewpoints without requiring an exhaustive training dataset.

3) Effect of the Number of Predict Heads: In this part, we conduct an ablation study to investigate the impact of the number of prediction heads which is controlled by the hyperparameters N_h. We conduct the task of SBSR on the SHREC 13 dataset. Specifically, we follow the similar setting presented in Section IV-D that uses 12 views for this experiment. Table III shows the quantitative results when we vary N_h from 1 to 5 and the size of the DV module. The findings presented in Table III

TABLE III
ABLATION STUDIES ON THE DESIGN CHOICE OF $m_{\varphi}(\cdot)$. WE REPORT THE RETRIEVAL RESULT ON THE SHREC 13 DATASET

Configuration	**NN**	**FT**	**ST**	**E**	**DCG**	**mAP**
$N_h - 1$	80.1	82.4	88.8	40.2	85.5	85.3
$N_h - 2$	83.2	84.8	89.0	42.0	88.9	86.7
$N_h - 3$	83.3	85.1	89.9	42.5	90.4	87.0
$N_h - 4$	83.3	85.2	90.0	42.4	90.3	87.0
$N_h - 5$	**83.3**	**85.2**	**90.0**	**42.6**	**90.6**	**87.1**

TABLE IV
ABLATION STUDIES ON SHREC 13 DATASET. WE EXAMINE THE EFFECTIVENESS OF OUR PROPOSED VIEW MINING LOSS. OUR PROPOSED VIEW MINING LOSS ACHIEVES THE BEST RESULT

Method	**NN**	**FT**	**ST**	**E**	**DCG**	**mAP**
Ours (w/o $\mathcal{L}_{\mathbf{view}}$)	81.1	82.6	87.4	40.7	88.2	85.4
Ours (with $\mathcal{L}_{pix}$)	82.0	83.5	90.4	41.1	88.8	86.2
Ours	**83.3**	**85.1**	**89.9**	**42.5**	**90.4**	**87.0**

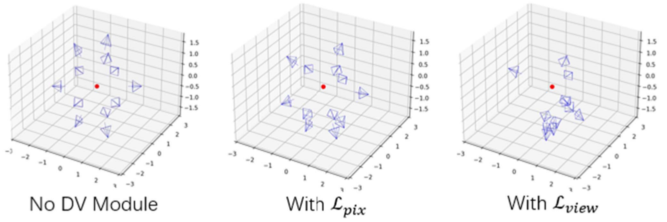

Fig. 8. Different camera positions predicted by using different loss.

demonstrate that the correlation between the number of layers and the resulting performance gain is not significant. While there is an increase in performance with an increase in the number of layers, the rate of improvement is not substantial. On the other hand, it is noteworthy that there is a linear relationship between the number of layers and the size of the model. This implies that as the number of layers in a model increases, so does its complexity and size. This may have practical implications, as larger models require more computing resources and memory. Therefore, it is crucial to strike a balance between model performance and resource requirements. Therefore, in our model, we chose N_h to be 3, which provided us with a good result and not much size cost.

4) Effect of the Viewpoint Mining: Then, we perform an ablation study to examine the effectiveness of the proposed view mining loss. To achieve this, we compare our model with 1) the same model trained without using view mining loss and 2) the same model trained using pixel-wise loss $\mathcal{L}_{pix}$. Specifically, we add an edge detector to the rendered image to obtain I_{edg}. The pixel-wise loss is defined as: $\mathcal{L}_{pix} = \|s - I_{edg}\|_2^2$. Table IV show the comparison between the aforementioned two models. Clearly, the model trained with view mining loss outperforms the other two models, which confirms the effectiveness of the proposed view mining loss. The camera positions predicted by adopting different loss terms can be found in Fig. 8.

TABLE V
INSTANCE-LEVEL SBSR RESULT ON THE AMATEURSKETCH DATASET

Method	Chair Dataset		Lamp Dataset	
	acc.@1	acc.@5	acc.@1	acc.@5
SBSVSR	44.94	77.94	48.05	79.87
FG-T-M	47.60	81.26	49.35	83.48
FG-T-A-M	47.10	79.10	48.95	81.68
FG-T-V	54.56	83.25	54.05	85.58
FG-T-P	0.50	4.48	0.90	3.60
FG-T-S	11.47	40.13	12.61	38.44
Qi et al. [11]	56.72	**87.06**	**57.66**	**87.39**
Ours	**57.28**	86.54	56.41	86.20

F. Discussion

1) Instance-Level Sketch-Based 3D Shape Retrieval: In this part of the experiment, we extend our method to the instance-level SBSR. Different from using global average pooling to get the z_{sem} for the retrieval, in this setting, we apply max pooling over view-specific embeddings of multi-view images. To conduct instance-level retrieval, it is necessary to establish a methodology for determining the similarity of sketches and 3D shapes. The similarity is computed within a feature space by calculating the Euclidean distance between view-specific embeddings of the sketch and 3D shapes.

Experiment dataset: AmateurSketch [11] addresses the challenge of instance-level sketch-based 3D shape retrieval, which has been hindered by the lack of datasets with one-to-one sketch-3D correspondences. The paper presents a new dataset comprising 4,680 sketch-3D pairings from two object categories generated through crowd-sourcing, where 80% are used for training, and the rest of those data will be used for testing.

Experiment setup: We compare our method with instance-level SBSR methods [11]. In addition, we follow the experiment setting presented in Qi et al. [11] to compare our method with several baseline methods, including Fine-Grained Triplet Based MVCNN [39], Fine-Grained Triplet With Spatial Attention [40], Fine-Grained Triplet Based on VNN [41], and Fine-Grained Triplet Based on Non-Projection Based 3D Deep Embeddings [42], [43]. In this experiment, we use configurations as mentioned in Section IV-B for the hyper-parameters and use $N_h = 12, N_h = 3$ to train our model.

Experiment result: Table V presents the quantitative results obtained from the AmateurSketch dataset, where the Top-1 and Top-5 retrieval accuracy were used to evaluate the performance of our method. Despite not being specifically designed for instance-level Sketch-Based 3D Shape Retrieval (SBSR), our method outperforms most of the listed methods. In the chair category, our method achieved a Top-1 retrieval accuracy of 57.28% and a Top-5 retrieval accuracy of 86.54%. Similarly, for the lamp category, our method achieved a Top-1 retrieval accuracy of 56.41% and a Top-5 retrieval accuracy of 86.20%. It is worth noting that although our model is not specifically designed for instance-level SBSR, our model still achieves the highest retrieval performance in terms of Top-1 retrieval accuracy on Chair Dataset. In addition to the quantitative results, we also present qualitative results in Fig. 6.

TABLE VI
ANALYSIS OF COMPUTATIONAL COMPLEXITY

Method	Query Time (sec.)	mAP
Siamese	$1.50 * 10^{-3}$	22.8
DCML	$3.95 * 10^{-1}$	28.6
DCHML	$3.95 * 10^{-1}$	33.6
LWBR	$3.95 * 10^{-1}$	40.1
DCA	$1.96 * 10^{-1}$	80.3
DSSH	$2.61 * 10^{-4}$	82.6
Ours	$1.94 * 10^{-1}$	85.1

2) Computational Complexity: In this section, we provide an analysis of the computational complexity of our method on the SHREC 14 dataset. For fair comparison with other models, we employed the same offline processing as in previous work [9], where descriptors were computed for each sketch and 3D shape in advance. During the retrieval phase, we selected the 3D shape whose descriptor is most similar to that of a given sketch as the retrieval result. The results of our analysis are presented in Table VI. The results show that the average query time per sketch in our method is similar to that of most previous work, while our accuracy surpasses that of existing methods. In other words, our approach achieves state-of-the-art retrieval performance with acceptable efficiency.

V. CONCLUSION

In this paper, we propose a novel category-level sketch-based 3D shape retrieval framework. First, we introduce a Dynamic Viewer (DV) module to explore expressive viewpoints according to a query sketch dynamically. Then, a cross-model disentangle module is subsequently adopted to decompose the sketch descriptor and the view descriptor. Finally, we optimize the DV module by minimizing the introduced view mining loss based on the resulting structural embedding. The proposed SBSR framework achieves promising results on SHREC 13 and SHREC 14 datasets and outperforms the state-of-the-art methods in some metrics. Furthermore, our method can easily be extended to the instance-level SBSR. In addition, the in-depth network analysis proves the effectiveness of each proposed module.

REFERENCES

[1] J. M. Saavedra, B. Bustos, M. Scherer, and T. Schreck, "STELA: Sketch-based 3D model retrieval using a structure-based local approach," in *Proc. 1st ACM Int. Conf. Multimedia Retrieval*, 2011, pp. 1–8.

[2] M. Eitz, R. Richter, T. Boubekeur, K. Hildebrand, and M. Alexa, "Sketch-based shape retrieval," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 1–10, 2012.

[3] B. Li et al., "A comparison of methods for sketch-based 3D shape retrieval," *Comput. Vis. Image Understanding*, vol. 119, pp. 57–80, 2014.

[4] F. Zhu, J. Xie, and Y. Fang, "Learning cross-domain neural networks for sketch-based 3D shape retrieval," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, vol. 30, pp. 3683–3689.

[5] G. Dai, J. Xie, F. Zhu, and Y. Fang, "Deep correlated metric learning for sketch-based 3D shape retrieval," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4002–4008.

[6] G. Dai, J. Xie, and Y. Fang, "Deep correlated holistic metric learning for sketch-based 3D shape retrieval," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3374–3386, Jul. 2018.

[7] A. Qi, Y.-Z. Song, and T. Xiang, "Semantic embedding for sketch-based 3D shape retrieval," in *Proc. Brit. Mach. Vis. Conf.*, 2018, vol. 3, pp. 11–12.

[8] X. Wei, R. Yu, and J. Sun, "View-GCN: View-based graph convolutional network for 3D shape analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1847–1856.

[9] J. Chen et al., "Deep sketch-shape hashing with segmented 3D stochastic viewing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 791–800.

[10] A. Sain, A. K. Bhunia, Y. Yang, T. Xiang, and Y.-Z. Song, "StyleMeUp: Towards style-agnostic sketch-based image retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8500–8509.

[11] A. Qi et al., "Toward fine-grained sketch-based 3D shape retrieval," *IEEE Trans. Image Process.*, vol. 30, pp. 8595–8606, 2021.

[12] F. Wang, L. Kang, and Y. Li, "Sketch-based 3D shape retrieval using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1875–1883.

[13] J. Xie, G. Dai, F. Zhu, E. K. Wong, and Y. Fang, "DeepShape: Deep-learned shape descriptor for 3D shape retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 7, pp. 1335–1345, Jul. 2017.

[14] J. Chen and Y. Fang, "Deep cross-modality adaptation via semantics preserving adversarial learning for sketch-based 3D shape retrieval," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 605–620.

[15] Z. Yasseen, A. Verroust-Blondet, and A. Nasri, "View selection for sketch-based 3D model retrieval using visual part shape description," *Vis. Comput.*, vol. 33, no. 5, pp. 565–583, 2017.

[16] D. Lu, H. Ma, and H. Fu, "Efficient sketch-based 3D shape retrieval via view selection," in *Proc. Adv. Multimedia Inf. Process.: 14th Pacific-Rim Conf. Multimedia*, 2013, pp. 396–407.

[17] L. Zhao, S. Liang, J. Jia, and Y. Wei, "Learning best views of 3D shapes from sketch contour," *Vis. Comput.*, vol. 31, no. 6, pp. 765–774, 2015.

[18] Y. Xu, J. Hu, K. Wattanachote, K. Zeng, and Y. Gong, "Sketch-based shape retrieval via best view selection and a cross-domain similarity measure," *IEEE Trans. Multimedia*, vol. 22, pp. 2950–2962, 2020.

[19] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learn. Workshop*, 2015, vol. 2.

[20] O. Vinyals et al., "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3630–3638.

[21] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4077–4087.

[22] F. Sung et al., "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1199–1208.

[23] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1126–1135.

[24] Z. Li, F. Zhou, F. Chen, and H. Li, "Meta-SGD: Learning to learn quickly for few-shot learning," 2017, *arXiv:1707.09835*.

[25] L. Bertinetto, J. F. Henriques, J. Valmadre, P. Torr, and A. Vedaldi, "Learning feed-forward one-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 523–531.

[26] A. Graves, G. Wayne, and I. Danihelka, "Neural turing machines," 2014, *arXiv:1410.5401*.

[27] Y. Wang et al., "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, 2019.

[28] A. Spurr, J. Song, S. Park, and O. Hilliges, "Cross-modal deep variational hand pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 89–98.

[29] B. Li et al., "SHREC'13 track: Large scale sketch-based 3D shape retrieval," in *Proc. 6th Eurographics Workshop 3D Object Retrieval*, 2013, pp. 89–96.

[30] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The Princeton Shape Benchmark," in *Proc. Shape Model. Appl.*, 2004, pp. 167–178.

[31] B. Li et al., "SHREC'14 track: Extended large scale sketch-based 3D shape retrieval," in *Proc. Eurographics Workshop 3D Object Retrieval*, 2014, vol. 2014, pp. 121–130.

[32] N. Ravi et al., "Accelerating 3D deep learning with Pytorch3D," 2020, *arXiv:2007.08501*.

[33] T. Furuya and R. Ohbuchi, "Ranking on cross-domain manifold for sketch-based 3D model retrieval," in *Proc. Int. Conf. Cyberworlds*, 2013, pp. 274–281.

[34] J. Xie, G. Dai, F. Zhu, and Y. Fang, "Learning barycentric representations of 3D shapes for sketch-based 3D shape retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3615–3623.

[35] Y. Xia, S. Wang, L. You, and J. Zhang, "Semantic similarity metric learning for sketch-based 3D shape retrieval," in *Proc. 21st Int. Conf., Comput. Sci.*, 2021, pp. 59–69.

[36] Q. Liu and S. Zhao, “Guidance cleaning network for sketch-based 3D shape retrieval,” *J. Phys.: Conf. Ser.*, vol. 1961, no. 1, 2021, Art. no. 012072.

[37] H. Yang et al., “Sequential learning for sketch-based 3D model retrieval,” *Multimedia Syst.*, vol. 28, pp. 761–778, 2022.

[38] Y. Zhao, Q. Liang, R. Ma, W. Nie, and Y. Su, “JFLN: Joint feature learning network for 2D sketch based 3D shape retrieval,” *J. Vis. Commun. Image Representation*, vol. 89, 2022, Art. no. 103668.

[39] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, “Multi-view convolutional neural networks for 3D shape recognition,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 945–953.

[40] J. Song, Q. Yu, Y.-Z. Song, T. Xiang, and T. M. Hospedales, “Deep spatial-semantic attention for fine-grained sketch-based image retrieval,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5552–5561.

[41] X. He, T. Huang, S. Bai, and X. Bai, “View N-gram network for 3D object retrieval,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7514–7523.

[42] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, “PointNet: Deep learning on point sets for 3D classification and segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 77–85.

[43] C. Esteves, C. Allen-Blanchette, A. Makadia, and K. Daniilidis, “Learning SO(3) equivariant representations with spherical CNNs,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 52–68.

Shuaihang Yuan received the B.S. degree in computer science from Stony Brook University, Stony Brook, NY, USA, in 2017, and the M.S. degree in computer science from the Tandon School of Engineering, New York University, New York, NY, USA, in 2019. Since 2019, he has been working toward the Ph.D. degree with the Department of Computer Science, Tandon School of Engineering, New York University, supervised by Prof. Yi Fang. His research interests include computer vision and machine learning, specifically in 3D computer vision.

Congcong Wen (Member, IEEE) received the B.S. degree in geographic information system from the China University of Petroleum, Beijing, China, and the Ph.D. degree from the Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, China. He is currently a Postdoctoral Associate with the Department of Electrical and Computer Engineering, New York University Tandon School of Engineering, New York, NY, USA, and New York University Abu Dhabi, Abu Dhabi, UAE. His research interests include remote sensing image recognition and three-dimensional computer vision.

Yu-Shen Liu (Member, IEEE) received the B.S. degree in mathematics from Jilin University, Changchun, China, in 2000, and the Ph.D. degree from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 2006. From 2006 to 2009, he was a Postdoctoral Researcher with Purdue University, West Lafayette, IN, USA. He is currently an Associate Professor with the School of Software, Tsinghua University. His research interests include shape analysis, pattern recognition, machine learning, and semantic search

Yi Fang (Member, IEEE) received the B.S. and M.S. degrees in biomedical engineering from Xi’an Jiaotong University, Xi’an, China, in 2003 and 2006, respectively, and the Ph.D. degree in mechanical engineering from Purdue University, West Lafayette, IN, USA, in 2011. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, New York University Tandon School of Engineering, New York, NY, USA, and New York University Abu Dhabi, Abu Dhabi, UAE. His research interests include three-dimensional computer vision and pattern recognition, large-scale visual computing, deep visual computing, deep cross-domain and cross modality multimedia analysis, and computational structural biology.