Computers & Graphics 116 (2023) 150–161

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on 3DOR 2023

SketchANIMAR: Sketch-based 3D animal fine-grained retrieval

Trung-Nghia Le [a,b], Tam V. Nguyen [c], Minh-Quan Le [a,b], Trong-Thuan Nguyen [a,b], Viet-Tham Huynh [a,b], Trong-Le Do [a,b], Khanh-Duy Le [a,b], Mai-Khiem Tran [a,b], Nhat Hoang-Xuan [a,b], Thang-Long Nguyen-Ho [a,b], Vinh-Tiep Nguyen [d,b], Nhat-Quynh Le-Pham [a,b], Huu-Phuc Pham [a,b], Trong-Vu Hoang [a,b], Quang-Binh Nguyen [a,b], Trong-Hieu Nguyen-Mau [a,b], Tuan-Luc Huynh [a,b], Thanh-Danh Le [a,b], Ngoc-Linh Nguyen-Ha [a,b], Tuong-Vy Truong-Thuy [a,b], Truong Hoai Phong [a,b], Tuong-Nghiem Diep [a,b], Khanh-Duy Ho [a,b], Xuan-Hieu Nguyen [a,b], Thien-Phuc Tran [a,b], Tuan-Anh Yang [a,b], Kim-Phat Tran [a,b], Nhu-Vinh Hoang [a,b], Minh-Quang Nguyen [a,b], Hoai-Danh Vo [a,b], Minh-Hoa Doan [a,b], Hai-Dang Nguyen [a,b], Akihiro Sugimoto [e], Minh-Triet Tran [a,b,*]

[a] *University of Science, VNU-HCM, Ho Chi Minh City, Viet Nam*
[b] *Vietnam National University, Ho Chi Minh City, Viet Nam*
[c] *University of Dayton, OH, USA*
[d] *University of Information Technology, VNU-HCM, Ho Chi Minh City, Viet Nam*
[e] *National Institute of Informatics, Tokyo, Japan*

ARTICLE INFO

Article history:
Received 11 April 2023
Received in revised form 15 July 2023
Accepted 15 July 2023
Available online 17 July 2023

Keywords:
3D object retrieval
Fine-grained retrieval
Animal models

ABSTRACT

The retrieval of 3D objects has gained significant importance in recent years due to its broad range of applications in computer vision, computer graphics, virtual reality, and augmented reality. However, the retrieval of 3D objects presents significant challenges due to the intricate nature of 3D models, which can vary in shape, size, and texture, and have numerous polygons and vertices. To this end, we introduce a novel SHREC challenge track that focuses on retrieving relevant 3D animal models from a dataset using sketch queries and expedites accessing 3D models through available sketches. Furthermore, a new dataset named ANIMAR was constructed in this study, comprising a collection of 711 unique 3D animal models and 140 corresponding sketch queries. Our contest requires participants to retrieve 3D models based on complex and detailed sketches. We receive satisfactory results from eight teams and 204 runs. Although further improvement is necessary, the proposed task has the potential to incentivize additional research in the domain of 3D object retrieval, potentially yielding benefits for a wide range of applications. We also provide insights into potential areas of future research, such as improving techniques for feature extraction and matching and creating more diverse datasets to evaluate retrieval performance.

1. Introduction

The rapid development of 3D technologies has produced a remarkable number of 3D objects. Consequently, 3D object retrieval has garnered considerable attention and is beneficial in real-life applications [1–3], including but not limited to video games, artistic pursuits, cinematography, and virtual reality.

Sketch-based 3D object retrieval aims to retrieve 3D models from a user's hand-drawn 2D sketch. Due to the innate intuitive appeal of freehand drawings, sketch-based 3D object retrieval has drawn a significant amount of attention and is being utilized in numerous critical applications such as 3D scene reconstruction [4–6], 3D geometry video retrieval [7–9], and 3D augmented/virtual reality entertainment [10,11]. However, sketch-based 3D object retrieval poses a formidable challenge in 3D object retrieval research, primarily due to the large discrepancy between the 2D and 3D modalities: non-realistic 2D sketches differ significantly from their 3D counterparts and respective views.

Several SHREC challenge tracks [12–17] have been organized to facilitate research on sketch-based 3D object retrieval. However, the existing datasets incorporated in these tracks primarily

[*] Corresponding author at: University of Science, VNU-HCM, Ho Chi Minh City, Viet Nam.
E-mail address: tmtriet@fit.hcmus.edu.vn (M.-T. Tran).

https://doi.org/10.1016/j.cag.2023.07.035
0097-8493/© 2023 Elsevier Ltd. All rights reserved.

comprise generic objects with simplistic shapes and poses. To augment sketch-based 3D object retrieval research, we organize a new SHREC challenge track dedicated to ***Sketch**-based 3D **ANIMA**l model fine-grained **R**etrieval (**SketchANIMAR**).*[1] This track aims to retrieve relevant 3D animal models from a dataset using sketch queries and expedites accessing 3D models through available sketches. Previous SHREC challenge tracks have focused on a limited number of general object categories, often lacking realism. Our challenge track for SHREC 2023 is significantly more challenging and can simulate real-life scenarios more effectively than its predecessors. After the challenge concluded, the dataset has been made publicly available for academic purposes.

First, conventional 3D object retrieval tasks consider only the object category, where the training and test samples are characterized by the same category settings. Consequently, features extracted from these methods are often optimized to fit the seen categories while lacking generalizability for unseen categories. Under such circumstances, the classification-based retrieval embedding learning methods become invalid in practice. Meanwhile, open-set 3D object retrieval can address this issue more effectively by dealing with unseen categories better. This technique involves training retrieval and representation models using seen-category 3D objects, with unseen-category 3D data subsequently used for retrieval. Nevertheless, our fine-grained retrieval task requires participants to conduct an accurate search to get 3D animal models whose shapes correspond to the query, necessitating consideration of unseen categories and poses (*cf*. Table 1). Compared to searching for 3D general objects of a given category, 3D animal model fine-grained retrieval poses a more significant challenge due to the substantial discrepancy in animal breeds and poses.

Second, participants in our track challenge must solve the considerable domain gap between sketches and 3D shapes when dealing with differently posed animals. Furthermore, human sketches on existing datasets tend to be semi-photorealistic and drawn by experts. In contrast, our dataset comprises more diverse sketches, including abstract sketches drawn by amateurs, semi-photorealistic sketches, and sketches in different styles (*cf*. Fig. 1). As such, this task proves significantly more challenging than conventional sketch-based object retrieval tasks. We anticipate that the sketch-based 3D animal fine-grained retrieval task can pave the way for a new research direction and exciting, practical applications.

The remainder of this paper is organized as follows. Section 2 provides an overview of related work. Section 3 presents the ANIMAR dataset and the evaluation measures used in this SHREC contest. Section 4 describes the participant statistics. In Section 5, the methods of the participating teams are presented. The evaluation results and an in-depth analysis of their performance are reported in Section 6. Finally, Section 7 concludes the paper and suggests directions for future work.

2. Related work

To recover 3D objects from a database, content-based 3D object retrieval examines the visual contents of the objects, such as color, texture, form, and geometric aspects. Many tracks concentrating on similar problems have been held in previous SHREC competitions (see Table 1) to promote research on content-based 3D object retrieval.

Several SHREC tracks concentrate on retrieving 3D items in a database that resemble the 3D objects used as a query. The attractiveness of sketch-based 3D object retrieval, in particular, stems from the organic and intuitive quality of freehand sketches, and it has garnered much attention in recent years.

Li et al. [12–14] promoted this intriguing study by organizing SHREC tracks of sketch-based 3D shape retrieval. At that time, deep learning was not popular; thus, submitted solutions were based on hand-crafted features such as Scale-Invariant Feature Transform (SIFT), Histogram of Oriented Gradient (HOG), fourier descriptors, bag-of-features, and sparse coding. After that, Juefei et al. [15,16] extended the task to 2D scene sketch-based 3D scene retrieval. Domain adaptation algorithms, such as two-stream CNN with triplet loss, adversarial training, and different data augmentation techniques, were used to resolve the disagreement between two domains (*i.e.*, sketch and 3D object). In addition, a competition for sketch-based 3D form retrieval in the wild was conducted by Qin et al. [17], further advancing the task. They used a variety of 3D forms, including models created by scanning genuine objects, as well as large-scale sketches created by amateur artists with a range of sketching abilities. Furthermore, technologies, such as point cloud and multi-view learning using various deep learning architectures, were created to emulate actual retrieval circumstances.

Sketch-based 3D object retrieval methods can be grouped into two categories: model-based and view-based approaches. Model-based methods commonly utilize 3D CNN to extract 3D shape features directly from the original 3D representations. In the view-based approach, 2D convolutional neural networks (CNN) are frequently used to analyze shape features from a set of 2D view projections.

Regarding the model-based approach, Furuya et al. [23] propose Deep Local Feature Aggregation Network (DLAN), which extracts rotation-invariant 3D local features and aggregates them in a single deep architecture. More concretely, the DLAN uses a set of 3D geometric features invariant to local rotation to characterize local 3D regions of a 3D model. The DLAN then compiles the set of features into a (global) rotation-invariant and compact feature for each 3D model. Furthermore, an Octree-based Convolutional Neural Network (O-CNN) [24] is also proposed for 3D shape analysis. O-CNN executes 3D CNN operations on the octants filled by the 3D shape surface using the average normal vectors of a 3D model sampled in the smallest leaf octants as input.

Concerning the view-based approach, Wang et al. [25] propose two Siamese Convolutional Neural Networks for the views and the sketches. Moreover, the loss function is designed for within-domain and cross-domain similarities. Similarly, two deep CNNs are proposed by Xie et al. [26] for deep feature extraction of sketches and 2D projections of 3D shapes. Next, the authors compute the Wasserstein barycenters of deep features of multiple projections of 3D shapes to form a barycentric representation. Last but not least, Multi-view Convolutional Neural Network (MVCNN) [27] creates a single, compact shape descriptor from data from multiple views of a 3D shape, which improves recognition performance.

In recent competitions of 3D Shape Retrieval Contests (SHREC) [17,28,29], teams achieving high performances followed the view-based approach.

3. Dataset and evaluation

3.1. Dataset

In this competition, we constructed a new dataset, namely ANIMAR, which encompasses a corpus of 711 distinct 3D animal models along with 140 sketch queries.

We collected an assemblage of 186 mesh models depicting over 50 diverse categories of animals. These models were diligently sourced from an array of publicly available online resources and video games, including the well-known Planet Zoo

[1] https://aichallenge.hcmus.edu.vn/sketchanimar.

Table 1
SHREC challenge tracks for 3D object retrieval (See [12–22]).

SHREC challenge	Year	Query type	Training category	Testing category
Hameed et al. [18]	2018	Image	Seen	Seen
Hameed et al. [19]	2019	Image	Seen	Seen
Li et al. [20]	2019	Image	Seen	Seen
Li et al. [21]	2020	Image	Seen	Seen
Feng et al. [22]	2022	Image	Seen	Unseen
Li et al. [12]	2012	Sketch	Seen	Seen
Li et al. [13]	2013	Sketch	Seen	Seen
Li et al. [14]	2014	Sketch	Seen	Seen
Juefei et al. [15]	2018	Sketch	Seen	Seen
Juefei et al. [16]	2019	Sketch	Seen	Seen
Qin et al. [17]	2022	Sketch	Seen	Seen
SketchANIMAR	2023	Sketch	Unseen	Unseen

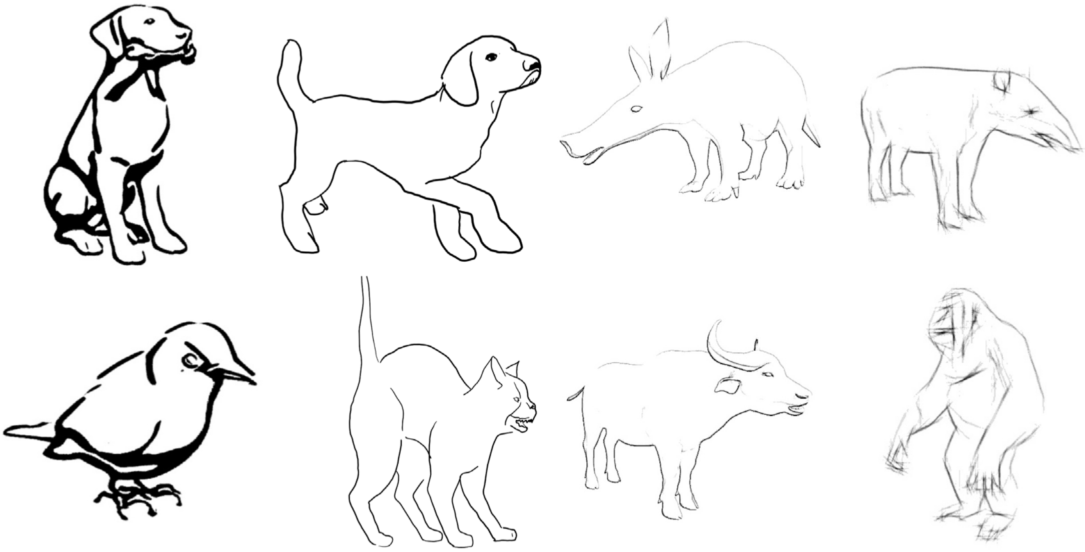

Fig. 1. Sketch ground-truth in ANIMAR dataset, including various sketch types and animal poses.

video game[2] [30]. The primary goal of our competition track was to simulate real-life scenarios in which users endeavor to identify and explore a diverse range of animal species. To achieve this, **we purposely concealed categorical information during both the training and retrieval stages**. Furthermore, we refined our model database by generating a series of watertight mesh models by reducing the number of faces by 25%, 50%, and 75%, yielding a total of 525 models. Following the work of Douze et al. [31], our 3D animal model database is employed for both the training and retrieval phases.

From 186 original mesh models, we randomly selected 60 models for sketch image creation. For each model, we rotated the model and generated 2–3 sketches from distinct viewpoints, thereby producing a total of 140 sketch images to describe the 3D animal models. Notably, we intentionally chose not to create sketches for all animal models in order to prevent participants from utilizing them to train retrieval solutions. Of the 140 sketches, 74 were aligned with their corresponding models in the database, yielding a set of 297 query-model pairs that were utilized for training purposes. The remaining 66 sketches were designated as queries, resulting in 265 query-model pairs employed during the retrieval phase. Unlike existing datasets, which primarily featured semi-photorealistic sketches drawn by experts, **our dataset comprises more diverse sketches, including abstract sketches drawn by amateurs, semi-photorealistic sketches, and sketches in different styles.** This diversity is exemplified in Fig. 1, where the varied nature of the sketches can be observed.

3.2. Evaluation metrics

We provide a comprehensive evaluation of the performance of different methods in this track. The following metrics are utilized:

- **Nearest Neighbor (NN)** evaluates top-1 retrieval accuracy.
- **Precision-at-10 (P@10)** is the ratio of relevant items in the top-10 returned results.
- **Normalized Discounted Cumulative Gain (NDCG)** is a measure of ranking quality defined as $\sum_{i=1}^{p} \frac{rel_i}{log_2(i+1)}$, where p is the length of the returned rank list, and rel_i denotes the relevance of the ith item.
- **Mean Average Precision (mAP)** is the area under the precision–recall curve. It measures the precision of methods at different levels and then takes the average. mAP is calculated as $\frac{1}{r}\sum_{i=1}^{r} P(i)(R(i)-R(i-1))$, where r is the number of retrieved relevant items, $P(i)$ and $R(i)$ are the precision and recall at the position of the ith relevant item, respectively.
- **First Tier (FT)** denotes the recall of the top m retrieval results, where m is the number of relevant images in the whole database. It measures the accuracy of retrieving the most relevant images among all the possible matches. The FT score is calculated as: FT = (number of relevant images retrieved in the top m)/m.

[2] https://www.planetzoogame.com.

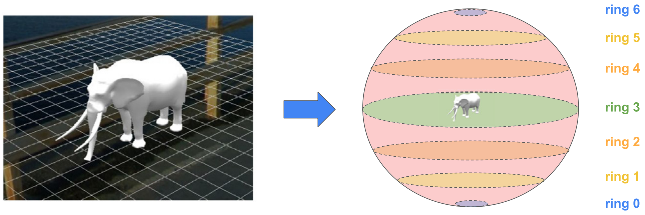

Fig. 2. 3D object represented as view sequences of 7 rings with 12 views on each ring. The chosen latitudes were 0 (the equator), ±90 (the poles), and ±30, ±60.

- **Second Tier (ST)** denotes the recall of the top $2m$ retrieval results, where m is the number of relevant images in the whole database. It measures the ability to retrieve relevant images within a broader set of results. The ST score is calculated as: ST = (number of relevant images retrieved in the top $2m$)/m.
- **Fallout Rate (FR)** shows the ratio of non-relevant retrieved items in relation to the total number of non-relevant items available. It measures the system's ability to avoid retrieving non-relevant items. The FR score is calculated using the formula: FR = (number of non-relevant items retrieved)/ (number of total non-relevant items in the database)

4. Participants

Eight groups participated in the SketchANIMAR challenge track. Each group was provided with three weeks to complete the challenge. Throughout the contest, a total of 204 runs were submitted. All participating groups were required to register and submit their results along with a detailed description of their methods. It is important to note that the organizers did not participate in the challenge. We remark that three teams opted not to disclose the methods they used in the competition against the SHREC spirit, which was born to compare the performance of algorithms on common data. Thus, they are not reported in this paper. The participant details are provided below (team members will be added upon acceptance):

- TikTorch team submitted by Nhat-Quynh Le-Pham, Huu-Phuc Pham, Trong-Vu Hoang, Quang-Binh Nguyen, and Hai-Dang Nguyen (see Section 5.2).
- THP team submitted by Truong Hoai Phong (see Section 5.3).
- Etinifni team submitted by Tuong-Nghiem Diep, Khanh-Duy Ho, Xuan-Hieu Nguyen, Thien-Phuc Tran, Tuan-Anh Yang, Kim-Phat Tran, Nhu-Vinh Hoang, and Minh-Quang Nguyen (see Section 5.4).
- V1olet team submitted by Trong-Hieu Nguyen-Mau, Tuan-Luc Huynh, Thanh-Danh Le, Ngoc-Linh Nguyen-Ha, and Tuong-Vy Truong-Thuy (see Section 5.5).
- DH team submitted by Hoai-Danh Vo and Minh-Hoa Doan (see Section 5.6).

5. Methods

5.1. Overview of submitted solutions

All submissions to our track are built upon the foundation of view-based learning. This approach captures the essence of each 3D object by presenting it as a sequence of ring images, as illustrated in Fig. 2. These images are acquired by strategically maneuvering a camera around the object along a predefined path, with each ring consisting of a series of images. In particular, when the camera's trajectory aligns parallel to the ground plane relative to the object, the multi-view method demonstrates remarkable effectiveness, generating valuable images that greatly assist in extracting features for representing three-dimensional objects.

The view-based learning shows more advantages than directly learning the point clouds. This matches well with the settings of our challenge. The 3D objects in our dataset have high-density point clouds with a large number of points. This detail can make it difficult for feature extraction models on point cloud such as PointNet [32] and PointMLP [33] when these models usually randomly sample a specific number of points (1024, for example) in the point cloud of 3D objects. This approach proves particularly useful in scenarios where 3D models are not readily available for querying, but sketches of the objects are, which is often the case in real-world applications.

To facilitate the retrieval task, TikTorch, THP, and Etinifni teams considered the problem as contrastive learning (as shown in Sections 5.2, 5.3, 5.4). Meanwhile, V1olet team formulated the task as a classical classification problem (as depicted in Section 5.5). On the other hand, DH team directly extracted and compared non-deep learning features between sketch queries and generated sketches from 3D objects (see Section 5.6).

5.2. TikTorch team

5.2.1. Proposed contrastive learning solution

To retrieve 3D objects from sketch queries, they propose a contrastive learning framework where embedding vectors of 3D objects and 2D sketches are learned. The embedding vectors of similar objects and sketches should be closer to each other and vice versa.

The overall architecture of their method is presented in Fig. 3, containing two separate feature extractors for 3D objects and sketch images. The extracted feature vectors are then embedded in the common vector space by two Multi-layer Perceptron (MLP) networks. The contrastive loss used for simultaneous learning of the parameters for models is a customized version of Normalized Temperature-scaled Cross Entropy Loss (NT-Xent) [34].

Sketch feature extractor. To extract the features of sketch images, they fine-tune EfficientNetV2-Small [35] pretrained on ImageNet dataset [36]. The models in the EfficientNetV2 family reduce the parameter size significantly while maintaining competitive accuracy on many datasets, which is desirable for simple images, especially sketch images.

3D object feature extractor. Each 3D object is represented as a set of 3 rings, and each ring contains 12 images. The 3D object feature extractor has two main phases: extracting the features of each ring (ring extractor) and combining the features of 3 rings to obtain the features of the object.

In the ring extractor, they also fine-tune EfficientNetV2-Small [35], similar to the sketch feature extractor module, to extract the features of 12 images of each ring. These 12 feature vectors then go through an encoder block called T-Encoder [37] to learn the relationship between images in the same ring to decide which image is essential in the current ring and which is not. After that, they combine these vectors by simply calculating their average to get a single feature vector for each ring.

When they obtain the feature vectors of 3 rings, these vectors are passed into 2 T-encoder blocks to know which ring is useful for the model to learn the features of the current object. Then, the vectors are averaged to get the feature vector of the 3D object.

Embedding into common space. To compute the similarity between objects and sketches, their feature vectors must be embedded into a shared space. Since the feature vectors of 3D objects and sketches may have different dimensions, two MLP networks with two layers are utilized. The output layer of each network has the same number of units, ensuring that the feature

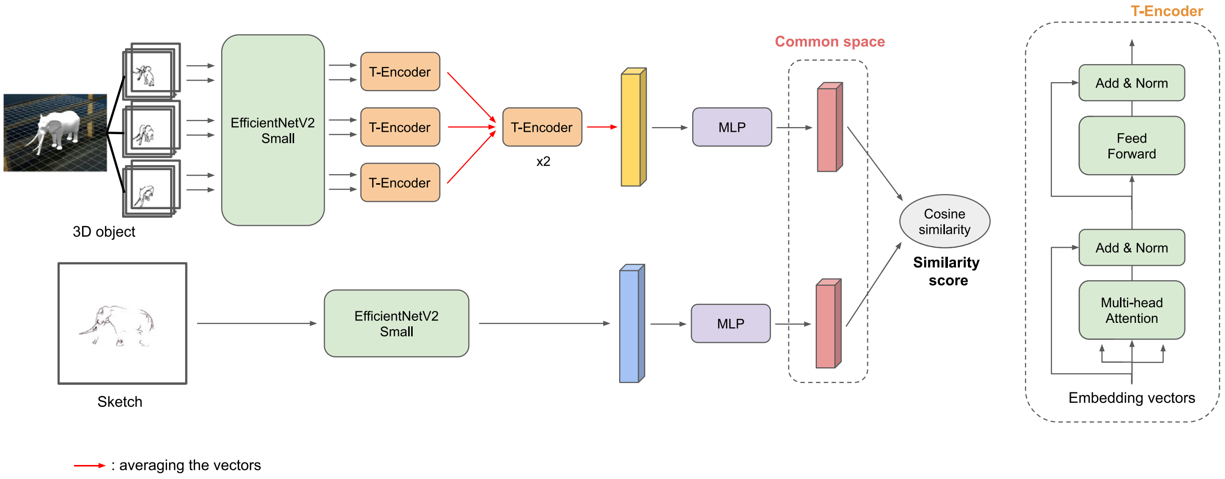

Fig. 3. Proposed framework of TikTorch team.

vectors are transformed into the same vector space. In addition, a Dropout layer [38] is added to each network to prevent overfitting. Once the feature vectors are embedded in the common space, the similarity between two embedding vectors, $\mathbf{u}$ and $\mathbf{v}$, can be computed using the cosine similarity metric.

$$\mathrm{sim}\left(\mathbf{u},\mathbf{v}\right)=\frac{\mathbf{u}\cdot\mathbf{v}}{\|\mathbf{u}\|\,\|\mathbf{v}\|} \tag{1}$$

Loss function. The contrastive loss function used is a customized version of Normalized Temperature-scaled Cross Entropy Loss (NT-Xent) [34]. Given a mini batch of $2N$ samples $\{\mathbf{x}_i\}_{i=1}^{2N}$ containing N objects and N sketches. They denote $\mathbf{z}_i$ as the embedding vector of the sample $\mathbf{x}_i$ in the common space. Let P_i be the set of indices of samples that are similar to $\mathbf{x}_i$ in the current mini-batch exclusive of i, *i.e.*, $(\mathbf{x}_i,\mathbf{x}_j)$ is a positive pair for $j\in P_i$. Here, $\mathbf{x}_i$ can belong to many positive pairs, such as two 3D objects that are similar to the same sketch. The loss function for a positive pair $(\mathbf{x}_i,\mathbf{x}_j)$ is defined as:

$$l_{i,j}=-\log\frac{\exp\left(\mathrm{sim}\left(\mathbf{z}_i,\mathbf{z}_j\right)/\tau\right)}{\sum_{k=1}^{2N}\mathbb{I}_{[k\neq i,k\notin P_i]}\exp\left(\mathrm{sim}\left(\mathbf{z}_i,\mathbf{z}_k\right)/\tau\right)}, \tag{2}$$

where $\mathbb{I}_{[k\neq i,k\notin P_i]}\in\{0,1\}$ is an indicator function evaluating to 1 if and only if $k\neq i$ and $k\notin P_i$, τ is a temperature parameter.

Training phase. During the training process, the optimizer used for training was AdamW [39], along with the StepLR algorithm to reduce the learning. They also applied the k-fold cross-validation technique with $k=5$.

Retrieval phase. They ensemble the results of models trained on k-fold by max-voting. The similarity between a 3D object and a sketch image is the largest value of the similarity score computed by the five models.

5.2.2. Data augmentation

Generation of multi-view images for 3D objects. Before generating batch images from 3D objects, it is essential to ensure axial synchronization of the objects so that the resulting multi-view images with their corresponding camera angles are consistent. To achieve this, they carefully examine the available dataset and identify several objects rotated at a 90-degree angle along the Ox axis. Then, they apply a consistent rotation to align these objects with the majority of the dataset as in Fig. 4.

Among seven rings, as in Fig. 2, they find that the most informative views are captured from rings 2, 3, and 4, which provide a 360-degree perspective around the object. Hence, they focus on processing the images from these rings to extract the relevant features and information.

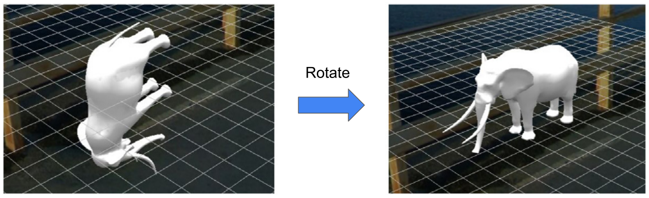

Fig. 4. Example of rotating a 3D object whose axis is not aligned with the majority of objects.

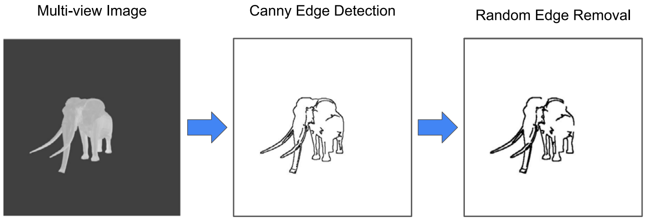

Fig. 5. Multi-view image processing steps.

To enhance the sketch-like appearance of the multi-view images, they utilize the Canny edge detector [40] to extract edge information. They also add some noises and variations to the edge information by randomly removing edges in the image using a traversal algorithm while preserving the underlying structure and content of the image (see Fig. 5). Fig. 6 illustrates the outcome of generating multi-view images for a 3D object.

Generation of training sketch query images. Firstly, they cluster similar 3D objects together by some algorithms to check the similarity between the distribution of points in the point clouds and also the manual checking as post-processing. After that, they identify the best-quality object in each cluster, which usually is the most fine-grained object (*i.e.*, the highest number of points in the point cloud). Then, when they generate a sketch-line image for each object in this cluster and use it for contrastive learning, they pick the image of the best-quality object as the representative sketch (depicted in Fig. 7).

To expand training samples for contrastive learning, they develop a method to generate three queries per object. Each query is randomly chosen from rings 2, 3, and 4 (see Fig. 2) with probabilities of 0.2, 0.6, and 0.2, respectively, as they observe that the majority of informative queries are in ring 3. Once a ring is selected, they randomly choose an image within that ring

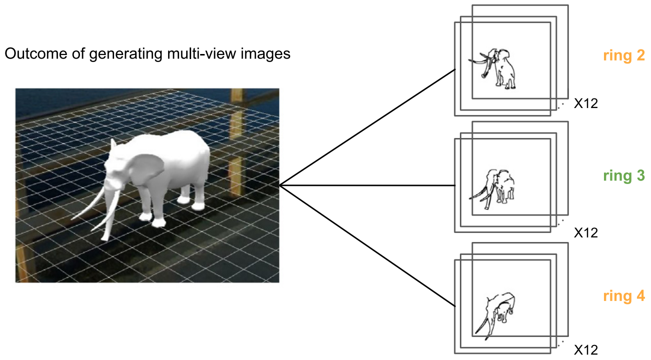

Fig. 6. Outcome of generating multi-view images for 3D objects. Each 3D object is represented by a set of 3 rings, and each ring is a collection of 12 images.

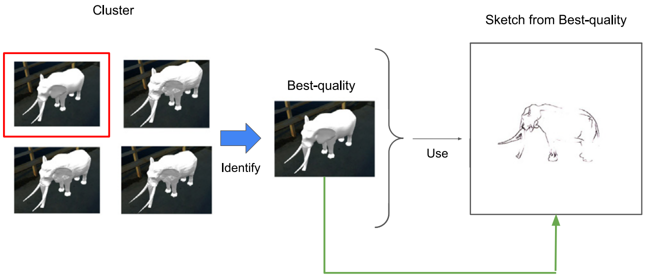

Fig. 7. Optimizing object sketches through clustering: the sketch of the best-quality object for all objects within a group of similar 3D objects is used.

from the cluster this object belongs to and apply random Canny edge [40] or Artline [41] techniques, along with image horizontal flip and rotation transformations. By implementing this process, they can significantly increase the number of our training samples from about 100 to 2500 while maintaining a high level of quality.

5.3. THP team

5.3.1. Architecture of proposed network

Fig. 8 illustrates an overview of the proposed network. To evaluate the similarity of two given sketch images, they compare the distance of global features and local features and then combine the results.

To extract global features, they use the pre-trained CLIP model [42]. The input of the CLIP model is the dilated Canny edge extracted from multi-view images. After that, CLIP feature vectors of view images and sketch queries are matched using cosine similarity. The final score is calculated as the maximum of scores of 4 views, in which each view score is calculated by the sum of the six highest similarities.

To increase local information awareness, they use the HOG descriptor [43] on both sketch and multi-view images. The HOG vectors are then matched using the L2 distance. They are also ranked like CLIP feature vectors.

Finally, they combine CLIP and HOG similarity scores as follows: $Score = \alpha * CLIP\,score + (1-\alpha) * HOG\,score$, where $\alpha = 0.7$.

5.3.2. 2D shape projection

They use four camera setups to take multiple views of 3D objects:

- For the first camera setup, assuming the 3D object is initially aligned along the z-axis, the camera is aligned on the Oxy plane and looks at the center of the object. The camera is moved around the subject to create 12 views from a distance of 30 degrees each time.
- For the second camera setup, the camera is raised to 30 degrees above the Oxy plane and moved around to create the next 12 views.
- For the third camera setup, the camera is placed on the Oyz plane and looks at the object's center. The camera is moved like the first setup to create the next 12 views.
- Camera for the last setup is raised from the third setup to 30 degrees to create the next 12 views.

There are a total of 48 views for each object. Note that it is possible to create images from other directions, but according to their tests, from these 48 views, they can observe the characteristics of objects.

5.3.3. Sketch pre-processing

To reduce the domain gap, they apply Canny edge algorithm [40] for each 2D projection image to create an image similar to the sketch image because the sketch is also a special type of edge. Both sketch and Canny edge images are then cropped and resized to 224 × 224 with the padding of 5 pixels. After that, edges are further clarified using the dilation morphology algorithm. The sketch pre-processing pipeline is shown in Fig. 9.

5.4. Etinifni team

5.4.1. Overview of proposed solution

Due to the nature of the retrieval tasks, they build a deep learning framework using the CLIP model [42] as the backbone (See Fig. 10). The framework performs the following steps:

1. Pre-process the dataset to re-direct the 3D objects into one single vertical orientation.
2. Extract multi-views (*i.e.*, 12 random views around the objects in uniform angles) from the 3D objects.
3. Encode the 2D sketches and the view images of 3D objects using the AutoEncoder built upon the ResNet50.
4. Reduce the size of the feature vectors from 512 to 128 through the projection head.
5. Compare feature vectors of sketches and the 3D objects using the cosine similarity function. After that, they can identify the matching pairs of sketches and 3D objects.

5.4.2. Data pre-processing

Resize objects: They resize the 3D objects to fit the 3D object inside an imaginary box of 2 × 2 × 2 by re-scaling the dimensions (x, y, z) of the 3D objects into the new dimension (x', y', z') not greater than 2 using the following formula:

$$(x', y', z') = \left(\frac{2x}{\max(x, y, z)}, \frac{2y}{\max(x, y, z)}, \frac{2z}{\max(x, y, z)}\right) \quad (3)$$

Re-orientate objects: They re-direct the orientation of the 3D objects manually so that the 3D objects are standing (*i.e.*, the objects are in an erect position).

5.4.3. Multi-view generation

They first rotate the camera as in Fig. 2. They apply 3D image rendering at each position to provide a more accurate and detailed view of objects. This enables the model to extract the necessary features for retrieval with greater precision, even where subtle variations in shape and texture can be crucial in determining an object's identity. Models then extract the most detailed and accurate information possible, leading to more robust and reliable results.

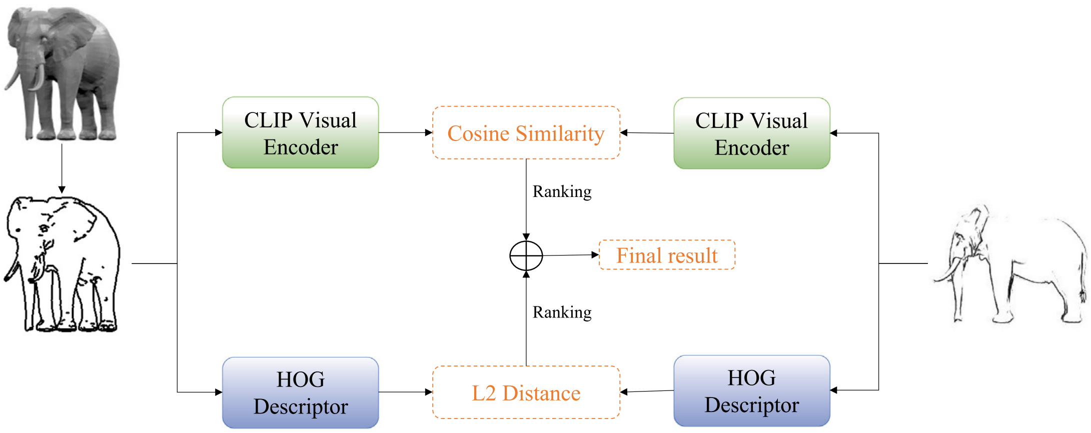

Fig. 8. Proposed framework of THP team.

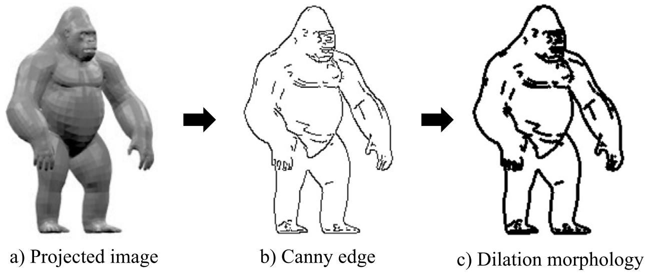

Fig. 9. Pipeline of sketch pre-processing.

5.4.4. Data augmentation

Due to the limited available training data, data augmentation techniques are applied to increase the number of training data:

Outline modifications:

- *Thickness:* is a significant attribute of 2D objects as it plays a crucial role in defining their physical properties and functionality. By adjusting the thickness, they generate new sketches with varying line lengths, thereby introducing diversity in the visual depiction of the objects.
- *Geometry:* explores the properties and relationships of shapes, sizes, positions, and dimensions of objects in space. By modifying the stroke of these edges, they can generate variations in the appearance of the zigzag patterns. This allows for a deeper investigation into the geometric characteristics of the objects and the effects of stroke adjustments on their overall geometry.

Image processing:

- *Random deletion:* To ensure the robustness of the sketch, they partition the 2D images into multiple blocks. Each block is equal in size and contains a subset of pixels from the original image. They randomly select a portion of these blocks and remove them from the image to simulate missing strokes on the data set.
- *Image compression:* Because the provided query images are of low quality, a compressor is utilized to reduce the image quality to generate the sketches.

View extraction:

- The camera is set up at a suitable distance, height, and orientation to ensure comprehensive coverage of the object's surface. The camera views are then randomly selected to capture diverse angles for robust 3D reconstruction.

5.5. V1olet team

5.5.1. Proposed classification approach

They formulate the retrieval task as a classical classification problem. Notably, several potential CNNs, including EfficientNet [44], EfficientNetV2 [35], and ConvNeXt [45] are employed to recognize whether the sketch query and 3D objects are the same class. These networks are renowned for their remarkable feature extraction capabilities and are considered state-of-the-art in image recognition. The used models also are lightweight and suitable for real-life applications while achieving considerable performance.

Ensemble Solution. Fig. 11 illustrates the proposed ensemble approach by averaging the predictions of each model. This approach effortlessly helps mitigate individual models' potential shortcomings, resulting in improved performance and robust generalization to varying data distributions.

To further improve the accuracy of models, they also utilize Test Time Augmentation (TTA), which involves applying a range of transformations such as rotations, flips, and translations to test images and averaging the results to obtain the final prediction. Specifically, they utilize horizontal flipping to provide additional perspectives of the original images. This technique not only enhances generalization but also enables the model to recognize objects that may be oriented differently from those in the training set.

Training Phase. To evaluate the performance of used models, they created a validation set by randomly leaving out 10% of samples from each class in the training data. Pre-trained models on ImageNet were fine-tuned using the remaining training sets. They also employed the cross-entropy loss with label smoothing of 0.1 to prevent overfitting and improve the generalizability of models. Sketchified multi-view images were jointly trained with the original sketch queries to enforce networks to recognize them as the same class. During both training and inference, an image size of 384 × 384 was utilized. All networks were trained for 20 epochs using the Adam optimizer [46] with a learning rate of 0.0001. Finally, they selected the models with the best validation accuracy for ensembling.

Retrieval Phase. Given a sketch query, the ensemble averaging of CNN models produces a set of softmax probabilities. These probabilities are then used to identify whether the sketch query and sketch-like images generated from the 3D object belong to the same class. The softmax probabilities also serve as a ranking metric, allowing for sorting the retrieved 3D objects by relevance.

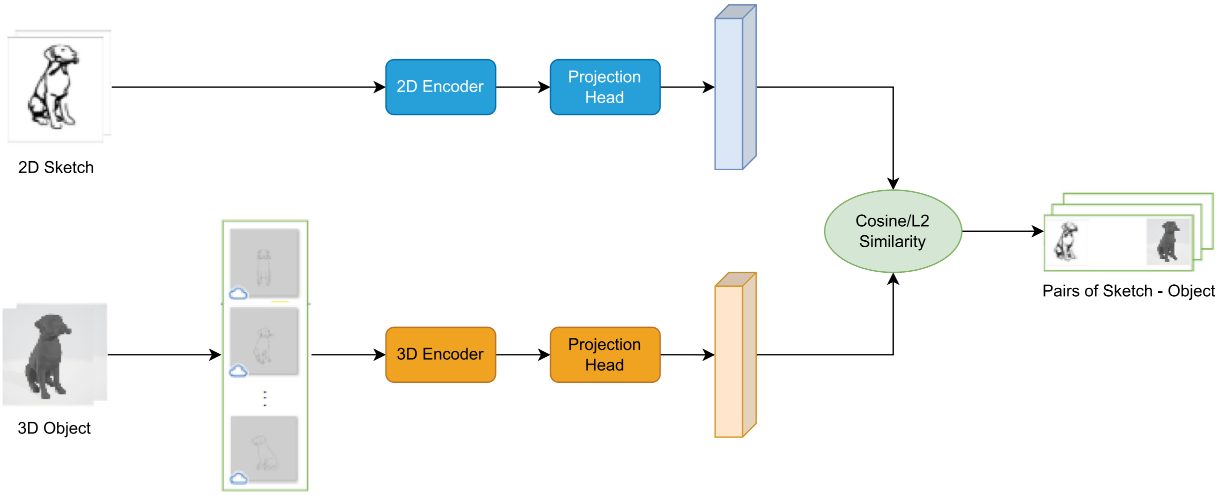

Fig. 10. Proposed framework of Etinifni team.

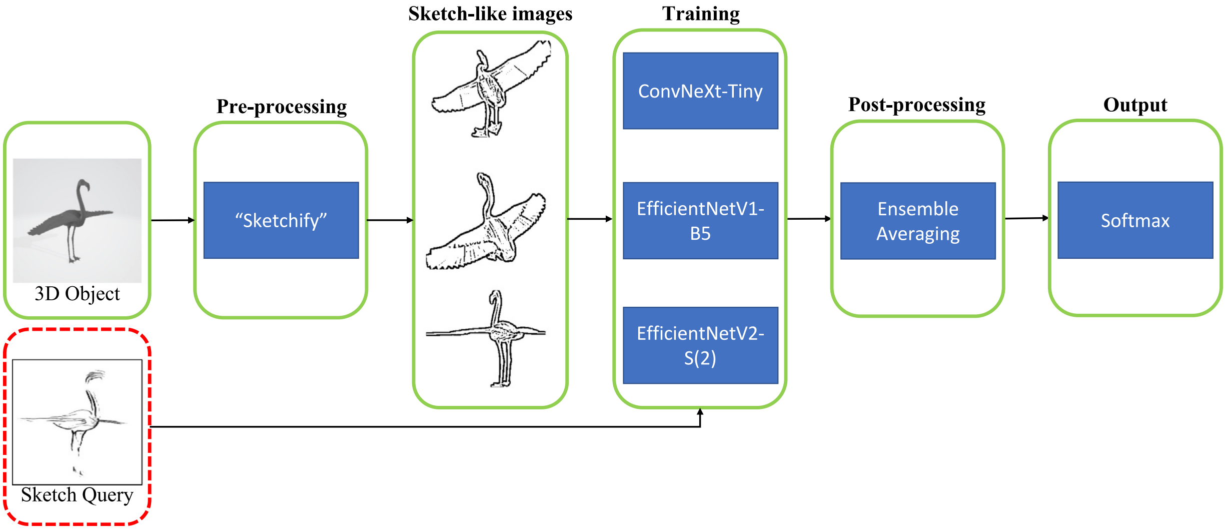

Fig. 11. Proposed framework of V1olet team.

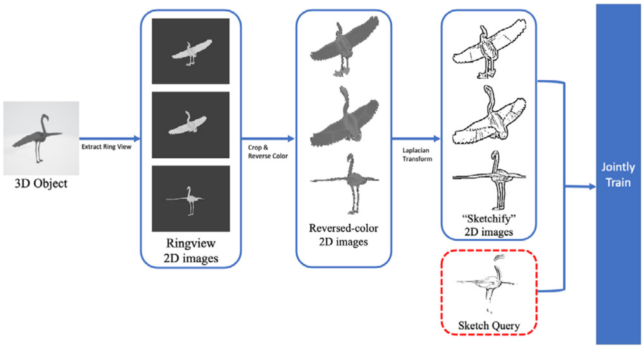

Fig. 12. The proposed data pre-processing pipeline.

5.5.2. Data pre-processing

Fig. 12 demonstrates an overview of the proposed data pre-processing pipeline. In general, it can be divided into three steps: Ringview extraction, color reversal, and sketchify.

Ringview extraction. Extracting multiple views of an object can be highly advantageous for various applications, including 3D object retrieval. They extract multiple views from 7 rings with 12 views on each ring, like Fig. 2. By providing different perspectives of the 3D object, these multiple views can extract more robust and detailed features to train models with greater accuracy for 3D object retrieval. Thus, ringview processing is a valuable technique that can improve the accuracy of models for 3D object retrieval.

Color reversal. It is also crucial to consider the background of the images when matching 3D objects with the querying sketches. Typically, these sketches usually have a white background, while the multiple 2D images obtained from the ringview extraction step have a gray background. To solve this problem, they merely flip the color of the ringview image, making the backdrop translucent to match the background of the target sketch queries. Therefore, the resulting images better resemble the sketch queries and support the further "sketchify" procedure.

Sketchify. Laplacian Transform is utilized to produce images that are more similar to sketches. Particularly, the Laplacian Transform, as a linear operator, is applied to the gray-scale image to generate a second-order derivative image that enhances the edges and transitions of the image [47]. The operator produces a sketch-like version of an image by thresholding the Laplacian image to obtain a binary edge map, which is used to synthesize a sketch-like representation of the image. This transformation enables more efficient comparison of the views with the sketch

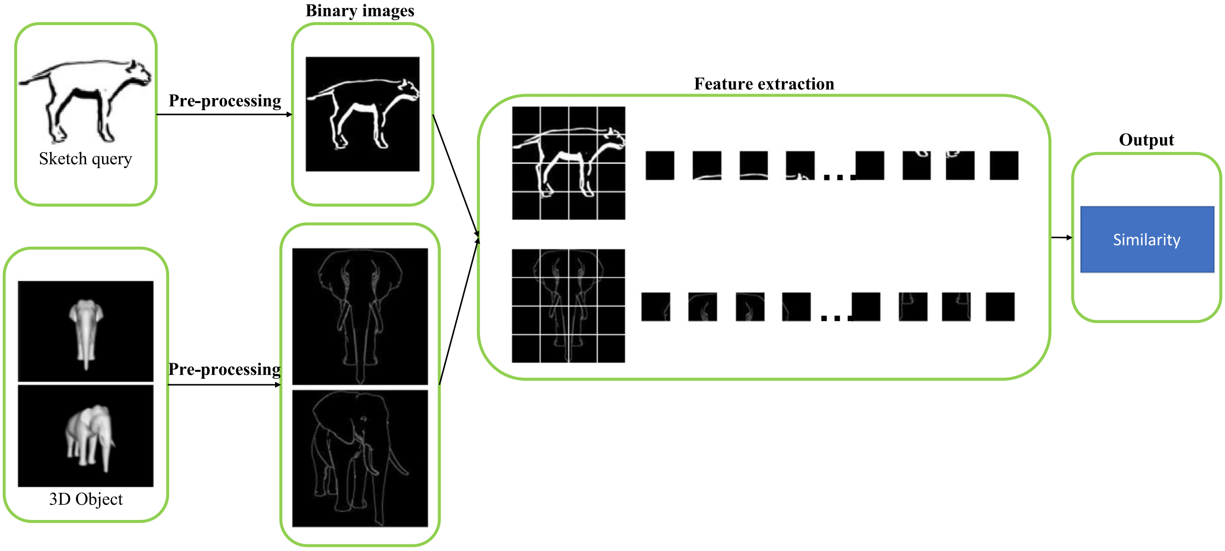

Fig. 13. Proposed framework of DH team.

query as the critical matching features become more pronounced, facilitating accurate matching.

5.6. DH team

5.6.1. Proposed method

They propose a simple method to measure the distance between 2 images with fast execution speed due to the small number of computations. Based on the observation that 3D objects with the same shape will have the same distribution of pixels in 2D projection space, they propose dividing the image into small parts and then comparing each area to measure the similarity between the two images. Fig. 13 shows the proposed framework, including four main modules: 2D sketch processing, 3D model processing, image feature extraction, and feature matching.

2D sketch processing. They crop images intending to keep only the part containing the animals and remove the background. Since sketches in the dataset only contain the animal and no other extraneous details in the background, they define the animal's bounding box as follows: First, sketches are converted to binary images and then inverted. Then they find the top and bottom image lines with the value 255. These two lines correspond to the top and bottom edges of the bounding box. The same method is applied to the left and right edges. After that, they crop the part containing the animal into a square with the size of a maximum bounding-box height and width and then resize the image to 224 × 224.

3D model processing. They first rotate the 3D objects using the Open3D library and capture the 3D model from 21 perspectives. After that, view images are cropped similarly to 2D sketches.

Image feature extraction. The images are divided into 4 × 4 squares, and then the ratio of total pixel values in each square to total pixel values of the whole image is considered at the score of the square. At the end of this step, a 16-dimensional feature vector represents an image, including the sketch and view images.

Feature matching. Given a sketch query image Q and a 3D model R, features are extracted to obtain f^Q representing the sketch image and $f_i^R, i = 1..21$ representing the 21 view images corresponding to the 3D model. The similarity score between the sketch query Q and the 3D model R is defined as follows:

$$D(Q,R) = min_i \|f^Q - f_i^R\|_2. \quad (4)$$

6. Results and discussions

In our SketchANIMAR track, the submitted runs are evaluated on two subsets: the public and private tests. The private test comprised 66 sketch queries, resulting in 265 query-model pairs. To ensure fairness and prevent potential cheating, approximately half of the private test (30 sketch queries) was randomly selected and designated as the public test subset. The leaderboard for the private test was revealed after the challenge concluded.

Tables 2 and 3 display the leaderboard outcomes for the public and the private test, reporting only the best-performing runs submitted by each team. However, to ensure a fair comparison, our analysis focuses solely on the results from the private test, which evaluated all the submitted sketch queries.

As seen in Table 3, the TikTorch and V1olet team's presented methods repeatedly stood out as the top effective strategies. On 6 out of 7 performance metrics (P@10, NDCG, mAP, FT, ST, and FR), TikTorch outperformed rival teams by a wide margin. In terms of NN metric, the V1olet team secured the top 1 position, indicating that this team focuses on the best search instead of neighboring search. Meanwhile, THP took up the third position on public and private leaderboards. Most teams achieving high performances apply the view-base approach, which analyzes shape features from 2D view projections. For the public test, similar findings are also shown in Table 2, excepting that the top-1 team (TikTorch) only takes the fourth position on FT and ST metrics. The first two teams, TikTorck and V1olet, achieve the top two ranks. In contrast, the performance of the other teams (*i.e.*, THP, Etinifni, and DH) is constant across test sets, both private and public. Summary, the findings show the challenges of our ANIMAR dataset when participants did not achieve excellent performance (the best NN result is only around 0.5, and P@10 results are smaller than 0.3). It also suggests that there is room to improve the performance of this research direction.

Fig. 14 illustrates the precision–recall curves of submissions on the private test of teams. It is clear that among the teams, the TikTorch team has the best average performance with the biggest area under the curve, while V1olet has the maximum precision at low recall (recall ≤ 0.2). The main difference between TikTorch's and V1olet's methods is that TikTorch follows a contrastive learning approach while V1olet leverages a classification-based one with a softmax layer. The softmax score just implies whether an object and a sketched query belong to the same category

Table 2
Leaderboard results of SketchANIMAR competition. Best run results on the public test.

Team	NN	P@10	NDCG	mAP	FT	ST	FR
TikTorch	**0.533 (1)**	**0.280 (1)**	**0.708 (1)**	**0.570 (1)**	0.192 (4)	0.333 (4)	**0.0102 (1)**
V1olet	0.467 (2)	0.213 (2)	0.613 (2)	0.411 (2)	0.317 (2)	0.492 (2)	0.0111 (2)
THP	0.433 (3)	0.207 (3)	0.601 (3)	0.399 (3)	0.300 (3)	0.450 (3)	0.0112 (3)
Etinifni	0.200 (4)	0.147 (4)	0.489 (4)	0.303 (4)	**0.475 (1)**	**0.650 (1)**	0.0121 (4)
DH	0.100 (5)	0.080 (5)	0.361 (5)	0.140 (5)	0.133 (5)	0.192 (5)	0.0130 (5)

Table 3
Leaderboard results of SketchANIMAR competition. Best run results on the private test.

Team	NN	P@10	NDCG	mAP	FT	ST	FR
TikTorch	0.470 (2)	**0.255 (1)**	**0.669 (1)**	**0.522 (1)**	**0.424 (1)**	**0.583 (1)**	**0.0105 (1)**
V1olet	**0.500 (1)**	0.232 (2)	0.640 (2)	0.453 (2)	0.379 (2)	0.534 (2)	0.0109 (2)
THP	0.409 (3)	0.226 (3)	0.608 (3)	0.421 (3)	0.333 (3)	0.504 (3)	0.0110 (3)
Etinifni	0.136 (4)	0.158 (4)	0.473 (4)	0.274 (4)	0.174 (4)	0.345 (4)	0.0119 (4)
DH	0.136 (4)	0.088 (5)	0.372 (5)	0.158 (5)	0.133 (5)	0.205 (5)	0.0129 (5)

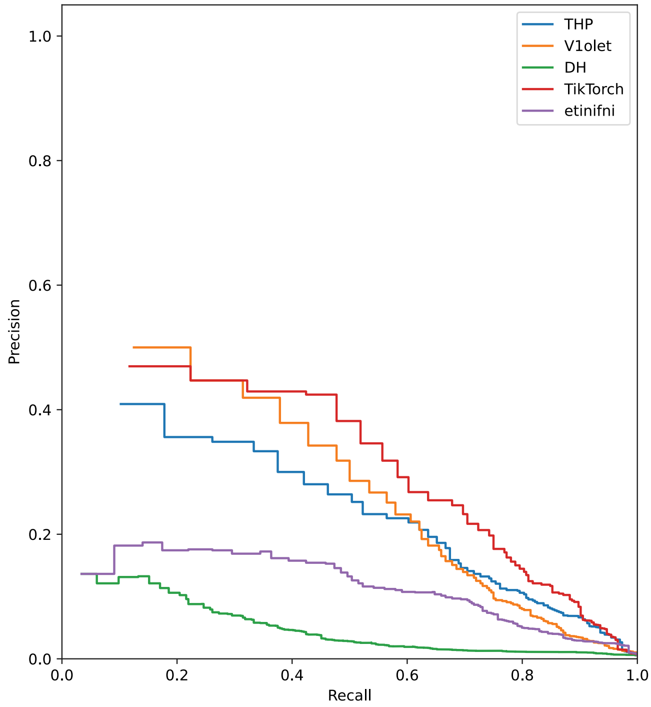

Fig. 14. The visualization of precision–recall curves of submissions on the private test of teams. It can be seen that the TikTorch team achieves the best average performance with the highest area under the curve, while V1olet obtains the highest precision at low recall (recall ≤ 0.2) among the eight teams. These insights are compatible with results in Table 3, TikTorch and V1olet secure the top 1 position regarding mAP and NN, respectively.

and cannot indicate much similarity between them. In general, V1olet's method works with objects in the same category as sketched queries and is less robust to negative samples than the contrastive learning approach. These conclusions are consistent with Table 3's findings, which show that TikTorch and V1olet rank first in terms of mAP and NN. Furthermore, the precision–recall curves of three teams, TikTorch, V1olet, and THP, share a similar shape, which shows the effectiveness at low recall threshold (recall ≤ 0.4) but drops significantly at high one (recall ≥ 0.8).

In conclusion, view-based learning methods have proven effective in achieving high performance. The difficulty of feature extraction models [32,48] can be attributed to the high point cloud density of the 3D objects in the ANIMAR dataset. It is important to remember that these models often randomly sample a certain number of pointclouds (*e.g.*, 1024). Contrarily, by utilizing the semantic data and representation of 3D objects, the use of view pictures obtained by moving the trajectory camera, as demonstrated in Fig. 2, enhances feature learning. This further demonstrates the effectiveness of the view-based learning strategy for retrieving 3D objects.

7. Conclusion

This paper introduces a novel track for sketch-based retrieval of fine-grained 3D animal models along with a newly constructed ANIMAR dataset. Our SHREC 2023 challenge track is designed to simulate real-life scenarios and requires participants to retrieve 3D animal models based on complex and detailed sketches. The challenge received submissions from eight teams; however, the evaluated results in this paper include methods from five of the eight teams. These submissions resulted in a total of 204 runs with different approaches. The evaluated results of this track were satisfactory but also revealed the difficulties of the task at hand.

In future research, we aim to expand the dataset by collecting a more diverse range of 3D animal models that encompass a wider variety of species, environmental contexts, and postures. This can enhance the generalization capability of potential solutions and improve performance on unseen 3D animal models. Additionally, we intend to generate synthetic data and texture to augment 3D animal models with different postures, backgrounds, and patterns to train more effective and robust representation models. We believe that by exploring these research avenues, we can advance the state-of-the-art in 3D object retrieval.

CRediT authorship contribution statement

Trung-Nghia Le: Conceptualization, Writing – review & editing, Project administration, Supervision. **Tam V. Nguyen:** Conceptualization, Writing – review & editing. **Minh-Quan Le:** Software, Writing – review & editing. **Trong-Thuan Nguyen:** Software, Writing – review & editing. **Viet-Tham Huynh:** Data curation. **Trong-Le Do:** Software, Investigation. **Khanh-Duy Le:** Visualization. **Mai-Khiem Tran:** Data curation. **Nhat Hoang-Xuan:** Software. **Thang-Long Nguyen-Ho:** Software. **Vinh-Tiep Nguyen:** Conceptualization. **Nhat-Quynh Le-Pham:** Methodology, Writing – original draft. **Huu-Phuc Pham:** Methodology, Writing – original draft. **Trong-Vu Hoang:** Methodology, Writing – original draft. **Quang-Binh Nguyen:** Methodology, Writing – original draft. **Trong-Hieu Nguyen-Mau:** Methodology, Writing – original draft. **Tuan-Luc Huynh:** Methodology, Writing – original draft. **Thanh-Danh Le:** Methodology, Writing – original draft. **Ngoc-Linh Nguyen-Ha:** Methodology, Writing – original draft. **Tuong-Vy Truong-Thuy:** Methodology, Writing

– original draft. **Truong Hoai Phong:** Methodology, Writing – original draft. **Tuong-Nghiem Diep:** Methodology, Writing – original draft. **Khanh-Duy Ho:** Methodology, Writing – original draft. **Xuan-Hieu Nguyen:** Methodology, Writing – original draft. **Thien-Phuc Tran:** Methodology, Writing – original draft. **Tuan-Anh Yang:** Methodology, Writing – original draft. **Kim-Phat Tran:** Methodology, Writing – original draft. **Nhu-Vinh Hoang:** Methodology, Writing – original draft. **Minh-Quang Nguyen:** Methodology, Writing – original draft. **Hoai-Danh Vo:** Methodology, Writing – original draft. **Minh-Hoa Doan:** Methodology, Writing – original draft. **Hai-Dang Nguyen:** Methodology, Writing – original draft. **Akihiro Sugimoto:** Conceptualization. **Minh-Triet Tran:** Conceptualization, Supervision, Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The dataset has been made publicly available for academic purposes on our website.

Acknowledgments

This work was funded by the Vingroup Innovation Foundation, Vietnam (VINIF.2019.DA19) and National Science Foundation, United States Grant (NSF#2025234).

References

[1] Stotko P, Krumpen S, Hullin MB, Weinmann M, Klein R. Slamcast: Large-scale, real-time 3D reconstruction and streaming for immersive multi-client live telepresence. IEEE Trans Vis Comput Graph 2019;25(5):2102–12.

[2] Liu X, Kofman J. Real-time 3D surface-shape measurement using background-modulated modified Fourier transform profilometry with geometry-constraint. Opt Lasers Eng 2019;115:217–24.

[3] Wang J, Mueller F, Bernard F, Sorli S, Sotnychenko O, Qian N, Otaduy MA, Casas D, Theobalt C. RGB2hands: real-time tracking of 3D hand interactions from monocular RGB video. ACM Trans Graph (ToG) 2020;39(6):1–16.

[4] Guo H, Peng S, Lin H, Wang Q, Zhang G, Bao H, Zhou X. Neural 3D scene reconstruction with the manhattan-world assumption. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 5511–20.

[5] Yookwan W, Chinnasarn K, So-In C, Horkaew P. Multimodal fusion of deeply inferred point clouds for 3D scene reconstruction using cross-entropy ICP. IEEE Access 2022;10:77123–36.

[6] Li J, Gao W, Wu Y, Liu Y, Shen Y. High-quality indoor scene 3D reconstruction with RGB-D cameras: A brief review. Comput Vis Media 2022;8(3):369–93.

[7] Gümeli C, Dai A, Nießner M. ROCA: robust CAD model retrieval and alignment from a single image. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 4022–31.

[8] Manda B, Kendre PP, Dey S, Muthuganapathy R. SketchCleanNet—A deep learning approach to the enhancement and correction of query sketches for a 3D CAD model retrieval system. Comput Graph 2022;107:73–83.

[9] Salihu D, Steinbach E. SGPCR: Spherical Gaussian point cloud representation and its application to object registration and retrieval. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. 2023, p. 572–81.

[10] Koca BA, Çubukçu B, Yüzgeç U. Augmented reality application for preschool children with unity 3D platform. In: 2019 3rd international symposium on multidisciplinary studies and innovative technologies. ISMSIT, IEEE; 2019, p. 1–4.

[11] Guo C, Jiang T, Chen X, Song J, Hilliges O. Vid2Avatar: 3D avatar reconstruction from videos in the wild via self-supervised scene decomposition. 2023, arXiv preprint arXiv:2302.11566.

[12] Li B, Schreck T, Godil A, Alexa M, Boubekeur T, Bustos B, Chen J, Eitz M, Furuya T, Hildebrand K, et al. SHREC'12 track: Sketch-based 3D shape retrieval. In: 3DOR@ eurographics. 2012, p. 109–18.

[13] Li B, Lu Y, Godil A, Schreck T, Aono M, Johan H, Saavedra JM, Tashiro S. SHREC'13 track: Large scale sketch-based 3D shape retrieval. 2013.

[14] Li B, Lu Y, Li C, Godil A, Schreck T, Aono M, Burtscher M, Fu H, Furuya T, Johan H, et al. SHREC'14 track: Extended large scale sketch-based 3D shape retrieval. In: Eurographics workshop on 3D object retrieval, Vol. 2014. 2014, p. 121–30.

[15] Yuan J, Li B, Lu Y, Bai S, Bai X, Bui N-M, Do MN, Do T-L, Duong A-D, He X, Le T-K, Li W, Liu A, Liu X, Nguyen K-T, Nguyen V-T, Nie W, Ninh V-T, Su Y, Ton-That V, Tran M-T, Xiang S, Zhou H, Zhou Y, Zhou Z. SHREC'18 track: 2D scene sketch-based 3D scene retrieval. In: Eurographics workshop on 3D object retrieval, Vol. 18. 2018, p. 70.

[16] Yuan J, Abdul-Rashid H, Li B, Lu Y, Schreck T, Bui N-M, Do T-L, Nguyen K-T, Nguyen T-A, Nguyen V-T, Tran M-T, Wang T. Shrec'19 track: Extended 2D scene sketch-based 3D scene retrieval. In: Eurographics workshop on 3D object retrieval, Vol. 18. 2019, p. 70.

[17] Qin J, Yuan S, Chen J, Ben Amor B, Fang Y, Hoang-Xuan N, Chu C-B, Nguyen-Ngoc K-N, Cao T-T, Ngo N-K, Huynh T-L, Nguyen H-D, Tran M-T, Luo H, Wang J, Zhang Z, Xin Z, Wang Y, Wang F, Tang Y, Chen H, Wang Y, Zhou Q, Zhang J, Wang H. SHREC'22 track: Sketch-based 3D shape retrieval in the wild. Comput Graph 2022.

[18] Abdul-Rashid H, Yuan J, Li B, Lu Y, Bai S, Bai X, Bui N-M, Do MN, Do T-L, Duong A-D, He X, Le T-K, Li W, Liu A, Liu X, Nguyen K-T, Nguyen V-T, Nie W, Ninh V-T, Su Y, Ton-That V, Tran M-T, Xiang S, Zhou H, Zhou Y, Zhou Z. 2D image-based 3D scene retrieval. In: Telea A, Theoharis T, Veltkamp R, editors. Eurographics workshop on 3D object retrieval. 2018.

[19] Abdul-Rashid H, Yuan J, Li B, Lu Y, Schreck T, Bui N-M, Do T-L, Holenderski M, Jarnikov D, Le KT, Menkovski V, Nguyen K-T, Nguyen T-A, Nguyen V-T, Ninh TV, Rey P, Tran M-T, Wang T. SHREC'19 track: Extended 2D scene image-based 3D scene retrieval. In: Eurographics workshop on 3D object retrieval, Vol. 700. 2019, p. 70.

[20] Li W, Liu A, Nie W, Song D, Li Y, Wang W, Xiang S, Zhou H, Bui N-M, Cen Y, Chen Z, Chung-Nguyen H-H, Diep G-H, Do T-L, Doubrovski EL, Duong A-D, Geraedts JM, Guo H, Hoang T-H, Li Y, Liu X, Liu Z, Luu D-T, Ma Y, Nguyen V-T, Nie J, Ren T, Tran M-K, Tran-Nguyen S-T, Tran M-T, Vu-Le T-A, Wang CC, Wang S, Wu G, Yang C, Yuan M, Zhai H, Zhang A, Zhang F, Zhao S. SHREC 2019-monocular image based 3D model retrieval. In: Eurographics workshop 3D object retrieval. 2019, p. 1–8.

[21] Li W, Song D, Liu A, Nie W, Zhang T, Zhao X, Ma M, Li Y, Zhou H, Zhang B, et al. SHREC 2020 track: extended monocular image based 3d model retrieval. In: Eurographics workshop 3D object retrieval. 2020.

[22] Feng Y, Gao Y, Zhao X, Guo Y, Bagewadi N, Bui N-T, Dao H, Gangisetty S, Guan R, Han X, et al. SHREC'22 track: Open-set 3D object retrieval. Comput Graph 2022;107:231–40.

[23] Furuya T, Ohbuchi R. Deep aggregation of local 3D geometric features for 3D model retrieval. In: Proceedings of the British machine vision conference. BMVC, 2016.

[24] Wang P-S, Liu Y, Guo Y-X, Sun C-Y, Tong X. O-CNN: Octree-based convolutional neural networks for 3D shape analysis. ACM Trans Graph 2017.

[25] Wang W, Shen J, Porikli F. Saliency-aware geodesic video object segmentation. In: CVPR. 2015, p. 3395–402.

[26] Xie S, Girshick R, Dollár P, Tu Z, He K. Aggregated residual transformations for deep neural networks. In: CVPR. 2017, p. 1492–500.

[27] Su H, Maji S, Kalogerakis E, Learned-Miller EG. Multi-view convolutional neural networks for 3D shape recognition. In: ICCV. 2015.

[28] Savva M, Yu F, Su H, Kanezaki A, Furuya T, Ohbuchi R, Zhou Z, Yu R, Bai S, Bai X, et al. Shrec'17 track large-scale 3D shape retrieval from ShapeNet core55. In: Proceedings of the workshop on 3D object retrieval. 2017.

[29] Moscoso Thompson E, Biasotti S, Giachetti A, Tortorici C, Werghi N, Obeid AS, Berretti S, Nguyen-Dinh H-P, Le M-Q, Nguyen H-D, Tran M-T, Gigli L, Velasco-Forero S, Marcotegui B, Sipiran I, Bustos B, Romanelis I, Fotis V, Arvanitis G, Moustakas K, Otu E, Zwiggelaar R, Hunter D, Liu Y, Arteaga Y, Luxman R. SHREC 2020: Retrieval of digital surfaces with similar geometric reliefs. Comput Graph 2020.

[30] Wu* Y, Chen* Z, Liu S, Ren Z, Wang S. CASA: Category-agnostic skeletal animal reconstruction. In: Neural information processing systems. 2022.

[31] Douze M, Tolias G, Pizzi E, Papakipos Z, Chanussot L, Radenovic F, Jenicek T, Maximov M, Leal-Taixé L, Elezi I, et al. The 2021 image similarity dataset and challenge. 2021, arXiv preprint arXiv:2106.09672.

[32] Qi CR, Su H, Mo K, Guibas LJ. Pointnet: Deep learning on point sets for 3D classification and segmentation. In: Conference on computer vision and pattern recognition. 2017, p. 652–60.

[33] Ma X, Qin C, You H, Ran H, Fu Y. Rethinking network design and local geometry in point cloud: A simple residual MLP framework. 2022, arXiv preprint arXiv:2202.07123.

[34] Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. In: International conference on machine learning. 2020, p. 1597–607.

[35] Tan M, Le Q. Efficientnetv2: Smaller models and faster training. In: International conference on machine learning. 2021, p. 10096–106.

[36] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC, Fei-Fei L. ImageNet large scale visual recognition challenge. Int J Comput Vis 2015;115(3):211–52.

[37] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. Adv Neural Inf Process Syst 2017;30.

[38] Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. 2012, arXiv preprint arXiv:1207.0580.

[39] Loshchilov I, Hutter F. Decoupled weight decay regularization. 2017, arXiv preprint arXiv:1711.05101.

[40] Canny J. A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 1986;(6):679–98.

[41] Madhavan V. Artline. 2021, https://github.com/vijishmadhavan/ArtLine. [Online; accessed 15-March-2023].

[42] Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, Sastry G, Askell A, Mishkin P, Clark J, et al. Learning transferable visual models from natural language supervision. In: International conference on machine learning. 2021, p. 8748–63.

[43] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: CVPR. 2005, p. 886–93.

[44] Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. 2019, p. 6105–14.

[45] Liu Z, Mao H, Wu C-Y, Feichtenhofer C, Darrell T, Xie S. A convnet for the 2020s. In: Conference on computer vision and pattern recognition. 2022, p. 11976–86.

[46] Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014, arXiv preprint arXiv:1412.6980.

[47] Nealen A, Sorkine O, Alexa M, Cohen-Or D. A sketch-based interface for detail-preserving mesh editing. In: ACM SIGGRAPH. 2005, p. 1142–7.

[48] Muzahid A, Wan W, Sohel F, Wu L, Hou L. CurveNet: Curvature-based multitask learning deep networks for 3D object recognition. IEEE/CAA J Autom Sin 2020;8(6):1177–87.