Fine-Grained Video Retrieval with Scene Sketches

Ran Zuo, Xiaoming Deng, Keqi Chen, Zhengming Zhang, Yu-Kun Lai, Fang Liu,
Cuixia Ma, Hao Wang, Yong-Jin Liu, Hongan Wang

***Abstract*—Benefiting from the intuitiveness and naturalness of sketch interaction, sketch-based video retrieval (SBVR) has received considerable attention in the video retrieval research area. However, most existing SBVR research still lacks the capability of accurate video retrieval with fine-grained scene content. To address this problem, in this paper we investigate a new task, which focuses on retrieving the target video by utilizing a fine-grained storyboard sketch depicting the scene layout and major foreground instances' visual characteristics (e.g., appearance, size, pose, etc.) of video; we call such a task "fine-grained scene-level SBVR". The most challenging issue in this task is how to perform scene-level cross-modal alignment between sketch and video. Our solution consists of two parts. First, we construct a scene-level sketch-video dataset called SketchVideo, in which sketch-video pairs are provided and each pair contains a clip-level storyboard sketch and several keyframe sketches (corresponding to video frames). Second, we propose a novel deep learning architecture called Sketch Query Graph Convolutional Network (SQ-GCN). In SQ-GCN, we first adaptively sample the video frames to improve video encoding efficiency, and then construct appearance and category graphs to jointly model visual and semantic alignment between sketch and video. Experiments show that our fine-grained scene-level SBVR framework with SQ-GCN architecture outperforms the state-of-the-art fine-grained retrieval methods. The SketchVideo dataset and SQ-GCN code are available in the project webpage https://iscas-mmsketch.github.io/FG-SL-SBVR/.**

***Index Terms*—Fine-grained sketch-based video retrieval, sketch-video dataset, scene sketch, graph convolutional networks.**

I. Introduction

THE rapid growth of video resources has led to the demand for accurate video retrieval. When people recall events happened in videos, their episodic memory is evoked to describe the event including spatial and temporal contexts as well as other event details [1]. Compared to text queries (which are limited to the intrinsic abstractness of the text modality) and image and video queries (which suffer from the difficulties in data acquisition to timely express users' diverse retrieval intention), free-hand sketch is a kind of flexible visual recollections which can intuitively provide fine-grained information of objects or scenes by depicting their approximate appearance and layout. Another advantage of free-hand sketch is that it can be easily obtained with the aid of popular sketching interfaces on various touch devices [2]. Therefore, sketch-based video retrieval (SBVR) has become a highly desired tool.

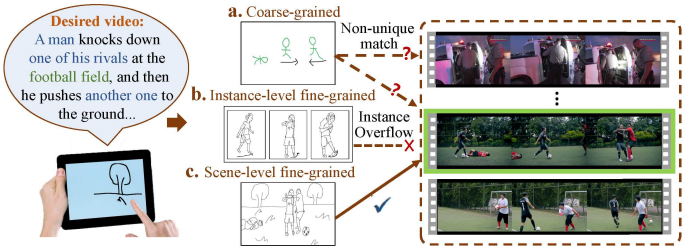


Fig. 1. The application scenario of fine-grained scene-level sketch-based video retrieval (SBVR), where different types of sketch queries are compared: (a) coarse-grained sketch without appearance details, infeasible for accurate retrieval; (b) fine-grained sketch(es) without background elements, limited to single instance; (c) fine-grained scene-level sketch, providing sufficient visual descriptions for accurate video match. The sketch formats of (a) and (b) refer to [3] and [8].

Most existing SBVR research mainly concentrates on coarse-grained retrieval [3], [4], [5], [6], [7], which does not make use of subtle differences in individual objects. The only instance-level SBVR work is confined to single-instance retrieval [8], which still overlooked the presence of multiple objects and scene context of background information in the real-world videos (see Figure 1b). Our work is motivated by the key observation that if users are capable of providing sketches with fine-grained scene content, much more accurate video retrieval can be achieved. Figure 1 shows an application scenario of fine-grained scene-level SBVR, where the user tries to search for a sports video clip specifically with intense physical touch. Rather than using abstract and complex text description, the user can sketch the desired scene content using a Tablet PC. We observe that sketching without multiple object instances or background details still leads to inaccurate retrieval. As a comparison, fine-grained sketches which emphasize the poses of players and the background such as several trees and grasses can significantly improve the accuracy of retrieval.

To the best of our knowledge, we are the first to study the fine-grained scene-level SBVR problem, which aims to retrieve the target video using a fine-grained scene sketch as input. Instead of providing a sequence of sketches similar to [8] (see Figure 1b), we use a single storyboard sketch (see Figure 1c) as query. Storyboard sketch was first proposed in [3], where both scene content and dynamic information are

This work was supported by the National Natural Science Foundation of China under Grant 62272447, the Natural Science Foundation of Beijing under Grant 4212029, Newton Prize 2019 China Award under Grant NP2PB/100047, and the Natural Science Foundation of Beijing under Grant L222008. Cuixia Ma, Xiaoming Deng, and Yong-Jin Liu are the corresponding authors.

Ran Zuo, Xiaoming Deng, Keqi Chen, Zhengming Zhang, Cuixia Ma and Hongan Wang are with Beijing Key Lab of Human-Computer Interaction, Institute of Software, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, e-mail: zuoran18@mails.ucas.ac.cn, xiaoming@iscas.ac.cn, chenkeqi19@mails.ucas.ac.cn, zhangzhengming16@mails.ucas.ac.cn, cuixia@iscas.ac.cn, hongan@iscas.ac.cn.

Yu-Kun Lai is with Cardiff University, e-mail: Yukun.Lai@cs.cardiff.ac.uk.

Hao Wang is with Alibaba, e-mail: cashenry@126.com.

Fang Liu, Yong-Jin Liu is with BNRist, Department of Computer Science and Technology, Tsinghua University, Beijing, China, e-mail: lfang@tsinghua.edu.cn, liuyongjin@tsinghua.edu.cn.

depicted using arrows and streak lines (see Figure 1a). In our work, the definition of storyboard sketch is slightly different: (1) the foreground instances and background elements are depicted with more fine-grained details (i.e. appearance, size, pose, etc.) to support fine-grained scene-level SBVR; (2) our sketch can contain instances that appear in different video frames (see Figure 2 for some examples); (3) the motion cues indicating instantaneous movement are removed due to the semantic ambiguity caused by the incapability of representing complex activities and the misalignment between motion cues and instances based on our user study (presented in Section III-A). The fine-grained scene-level SBVR problem is challenging mainly in three aspects: (1) the intrinsic domain gap between real-world videos and sketches that only contain sparse strokes; (2) the inaccurate correspondence of instances between videos and sketches due to freehand sketching; and (3) lack of fine-grained scene-level SBVR datasets.

To address the dataset issue, we construct a fine-grained scene-level SBVR dataset called SketchVideo, which contains 6,713 sketches and 1,126 video clips. The distinct characteristics of our dataset include: (1) sketch-video pairs are provided, and each pair contains one clip-level storyboard sketch (which can contain instances in different video frames) and several keyframe sketches (each of which contains instances in one video frame); (2) each video contains one consistent scene; (3) 43 foreground and 18 background object categories are labeled in the dataset; (4) each sketch in the dataset (including both storyboard sketches and keyframe sketches) depicts multiple foreground instances and iconic background elements.

Based on our SketchVideo dataset, we propose a novel Sketch Query Graph Convolutional Network (SQ-GCN) to model the spatial-temporal content matching of videos and storyboard sketches. We firstly design an adaptive video frame sampling strategy called FrameSampler, using the visual feature matching between keyframe sketches and video frames to train a sketch-image correlation model. During video retrieval, we use this model to select video frames that are highly relevant to the input storyboard sketch for video feature encoding. FrameSampler is useful for efficient video encoding because the selected frames can cover major object instances in the sketch, which is beneficial for content alignment between video and sketch. Then we construct *a spatial storyboard sketch graph* and *a spatial-temporal video graph*, and design two feature encoding branches, in which appearance and category features are aligned respectively through graph convolutions. After feature encoding in each branch, the sketch and video embeddings are fed to a triplet network training process. During the inference stage, we utilize appearance and category graph features with a late fusion strategy to compute the overall distance between sketch and video features for video retrieval.

As a summary, in this paper, we make the following contributions:

1) We construct a scene-level multi-instance sketch-video dataset *SketchVideo*. The sketches in SketchVideo depict not only fine-grained instances, but also multiple objects in diverse scenes. The dataset contains fine-grained clip-level (storyboard sketch) and frame-level (keyframe sketches) sketch-video pairs. In addition to video retrieval, our dataset can also support other related scene-level video tasks, such as video synthesis, video summarization, etc.
2) We propose a novel fine-grained scene-level video retrieval solution, using clip-level storyboard sketches as query input and designing a SQ-GCN structure to model the semantic and visual correlation between two modalities of videos and sketches.
3) We propose an adaptive video sampling strategy based on frame-level sketch-video pairs to reduce the computational cost and improve the efficiency of video encoding.
4) Extensive experimental results on the SketchVideo dataset demonstrate that our method outperforms the state-of-the-art fine-grained retrieval methods and the dataset is useful to promote sketch-based video research.

II. Related Work

A. Sketch-based Video Datasets

Most existing sketch datasets are designed for sketch understanding and sketch-based image applications [9], [10], [11], [12]. TU-Berlin [9] and Sketchy [10] are two large-scale sketch datasets with category-level sketch annotations. Yu et al. [11] constructed the first instance-level sketch-image dataset where each image has one corresponding sketch. While previous datasets only contain sketches of single object without background information, Zou et al. [12] proposed the first scene-level sketch dataset SketchyScene, consisting of scene sketch and image pairs with both instance-level and scene-level annotations.

For sketch-based video retrieval, Collomosse et al. [3] built several sketch-video datasets and used a combination of sketches and motion cues (arrows and streak-lines) to retrieve videos. Sketches used in [4], [5], [6], [7] are similar to [3] with different video types and scales. These sketches are roughly-drawn stick figures without fine-grained details, which can only be used for coarse-grained video retrieval.

Recently, Xu et al. [8] established a fine-grained sketch-video pair dataset named FG-SBVR, which contains 528 skating video clips and 1,448 sketches depicting the appearance and motion of the skaters. However, each skating video only contains one skater and a motion vector without scene context. FG-SBVR lacks instance amount, category diversity, and background elements, so there still lacks suitable datasets for scene-level SBVR research.

B. Cross-modal Video Retrieval

Video retrieval is a challenging problem due to the complex spatial and temporal information in video content. A few works use a set of example videos as query inputs [13], [14], [15], which is impractical for in-time query acquisition due to the variability of users' retrieval intention. Therefore, cross-modal video retrieval has drawn researchers' attention, with query forms including image, text and sketch. Image-based video retrieval [16], [17], [18], [19] still suffers from the same difficulty as example videos. Although text [20], [21], [22], [23] is capable of expressing rich semantic information,

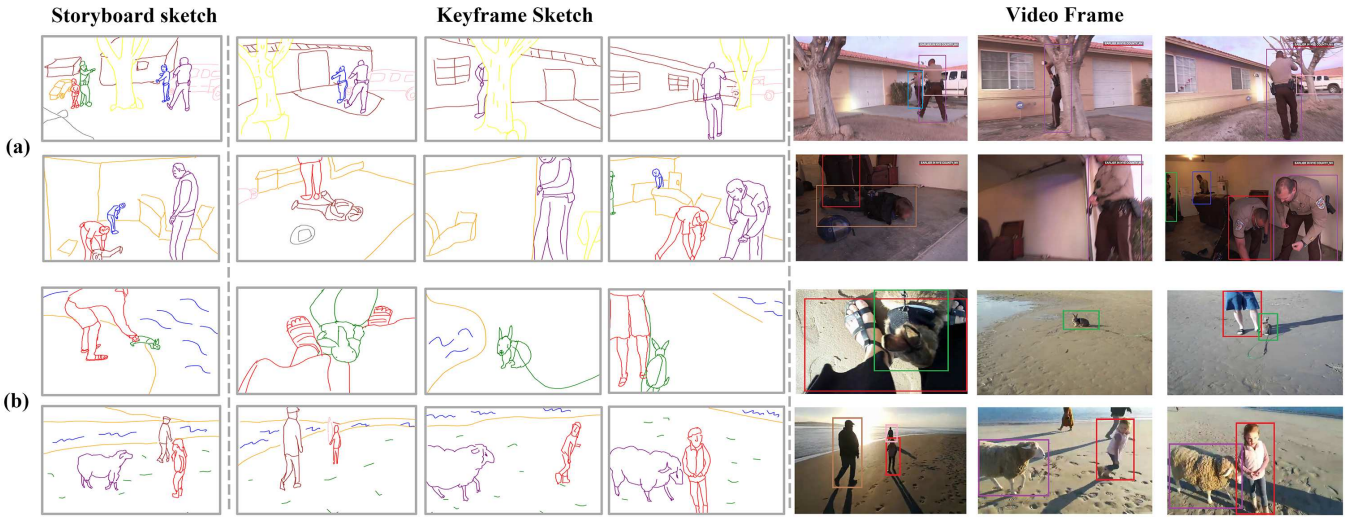


Fig. 2. Examples of our SketchVideo dataset where the sketches and video frames are temporally colored to show the dynamics of the individuals over time. Besides the fine-grained diversity in objects' size, viewpoint and appearance, the dataset also has fine-grained scene variations: (a) background variation, where the two videos contain the same policemen and fugitive outside and inside a house respectively; (b) foreground object variation, where there are different individuals on the beach in two videos.

it is too abstract to convey fine-grained details (i.e. layouts, appearance, size, etc.) in videos.

A few sketch-based video retrieval (SBVR) studies were conducted in the past decades. VideoQ [24] firstly took motion depiction in query sketches into consideration and designed strict rules for precise motion cue depiction. Collomosse et al. [3] accepted greater flexibility of drawing sketches, and proposed to use storyboard sketch as query, which is composed of roughly depicted objects with motion cues. Hu et al. [4] transformed keypoint trajectories corresponding to the camera motion in videos by space-time keypoint clustering, and matched the video and sketch tokens based on motion and color using the Viterbi shortest path algorithm. Hu et al. [5] later extended their work by leveraging the motion, color and semantic distribution of videos for retrieval. Furthermore, Hu et al. [6] used the Markov Random Field for video segmentation, and they built video graphs with space-time sub-volumes to match the sketch by appearance, motion and semantic category of the objects. James et al. [7] proposed an indexing method to fuse the object shape, color, semantic category and motion information into a spatial-temporal descriptor. All these methods belong to *coarse-grained* SBVR, which ignores the detailed depiction capability of sketches and lacks effective mechanisms to retrieve a particular video with sketch.

Our work is related to [8], a prior art on the fine-grained *instance-level* SBVR task. In [8], a sketch sequence query is used to retrieve the target instance video. However, since the method only uses sketches with a single foreground instance without scene context, their method can not be generalized well to many scenarios. It remains an open problem to perform fine-grained scene-level SBVR with multiple foreground instances so far.

C. Graph Convolutional Networks

Graph Convolutional Networks (GCNs) [25] are beneficial for feature extraction of graph-structured data by message passing through edge connection, and it has been widely used in sketch-based image retrieval (SBIR) tasks [26], [27]. Specifically, Liu et al. [27] used GCNs for fine-grained scene-level SBIR, where they constructed scene graphs with foreground instances as nodes and normalized Euclidean distances between nodes as edges. To extract the effective spatial and temporal features in videos, Yan et al. [28] proposed Spatial-Temporal Graph Convolutional Networks (ST-GCN) for action recognition via constructing spatial-temporal graph of a skeleton sequence.

Inspired by these works, we propose a SQ-GCN model for the fine-grained scene-level SBVR task, where we conduct feature embedding of storyboard sketches and videos by constructing spatial graphs for storyboard sketches and spatial-temporal graphs for videos.

III. The SketchVideo Dataset

We construct a new sketch-video dataset *SketchVideo* which is proposed for fine-grained scene-level sketch-video research (illustrated in Figure 2). SketchVideo contains 1,126 video clips collected from YouTube, and 6,713 corresponding sketches forming clip-level and frame-level representations, drawn by 30 amateur painters with diverse personal painting skills and styles. All strokes in the sketches are annotated with instance-level labels. The sketches were stored in the Scalable Vector Graphic (SVG) format following [8], [9], [10].

A. User Study for Storyboard Sketch Depiction

To investigate whether motion cues [3], [8] are suitable for storyboard scene sketch depiction, we carried out a user study and recruited 24 participants (12 males and 12 females) who rarely draw pictures and have no professional drawing skills. Given randomly-selected videos in the SketchVideo dataset, the participants were asked to draw scene sketches with and without motion cues. Then they were invited to rate

two storyboard sketch formats in three aspects, including "*It is easy to depict*", "*It is capable of describing fine-grained scene content*" and "*It has great generality* " (It can be applied on common video retrieval scenarios) [29]. In order to find out motion cues' influences upon scene understanding, the participants were split into six groups, where each group member would describe their understanding of others' depicted sketches without and with motion cues respectively. In the end, they rated whether they could well understand the meanings of motion cues depicted in the sketches.

As illustrated in Figure 3a, storyboard sketches with motion cues can convey slightly more scene content details, but they are more difficult to depict and can lead to a decrease of the sketch's generality. Moreover, the average rating in Figure 3b is below 3-Neutral which indicates that it is difficult to understand the meanings of the motion cues that others depicted. After analyzing participants' comprehension records of storyboard sketches, the percentage of incorrect alignment of motion cues and foreground objects is 48.57%. As illustrated in Figure 4, users' understanding of the main activities in the scene sketch with or without motion cues is similar. However, adding motion cues may cause semantic ambiguity due to (1) the incapability of describing complex activities and (2) the misalignment between multiple foreground objects and motion cues. Several participants commented that "*it is unlikely to use motion cues to describe activities with complex trajectories*", "*it's difficult to assign motion cues to multiple foreground objects especially when the scene is very complex*", "*maybe there needs to be a strict standard of motion cue depiction for comprehension, but that will harm the flexibility of sketch creation*".

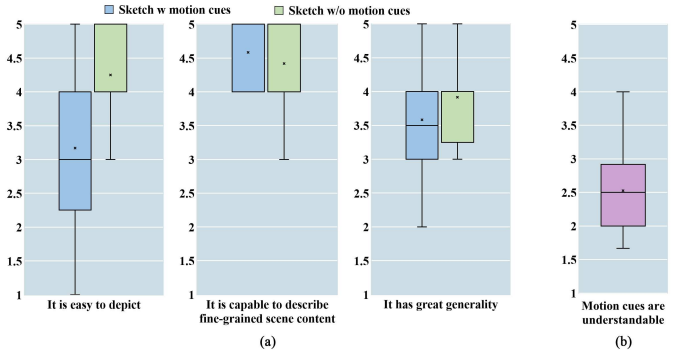


Fig. 3. Boxplots of user study results towards the usage of motion cues in sketch depiction. (a) User evaluation of storyboard sketches with and without motion cues; (b) User comprehension about motion cues depicted in the sketch. We use a 5-point Likert scale for rating (1-Strongly Disagree to 5-Strongly Agree).

Based on the user feedback, we can draw the following conclusions. Although motion cues are useful to depict objects' directional and instantaneous movement, there are non-negligible problems in practical usage including (1) the increase of depicting difficulty, (2) the incapability to describe complex activities, and (3) the misalignment of motion cues and multiple foreground objects. In contrast, the basic understanding towards storyboard sketches without motion cues has provided strong clues (scene layout, object appearance and pose, main activity, etc.) for scene-level video retrieval. Therefore, the storyboard sketches in our dataset are depicted without motion cues.

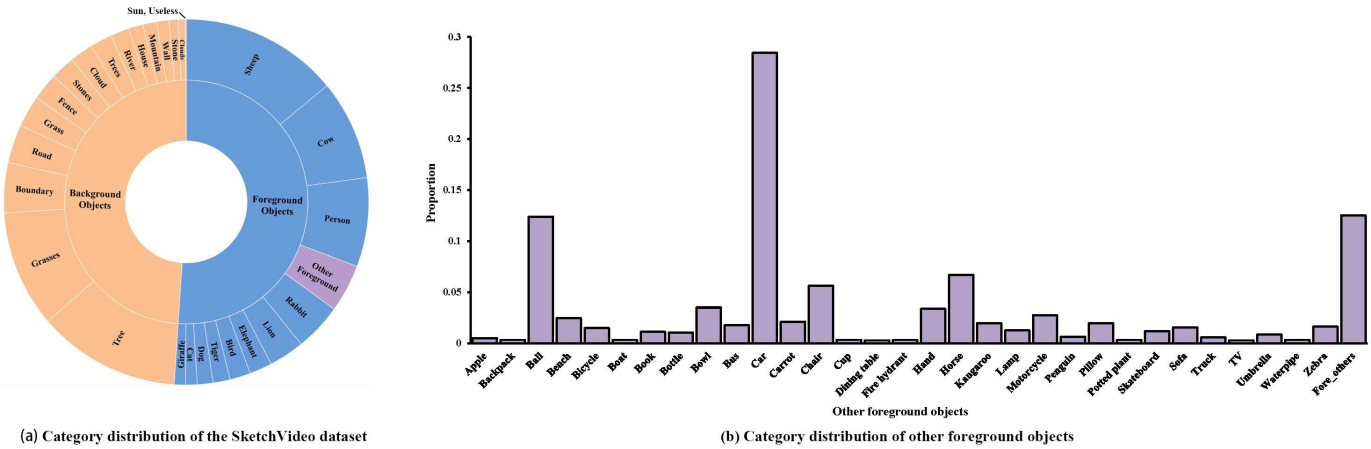
Fig. 4. The comparison of the understandings between storyboard sketches without and with motion cues. Users first describe the sketch without motion cues, and then share their new understandings when motion cues are added. The activities are stressed in red. They can easily understand the main activities with or without motion cues, but adding motion cues leads to ambiguous understanding, such as failing to represent complex activities, and misalignment between motion cues and multiple foreground objects.

B. Video Collection and Processing

To construct our scene-level dataset with diverse foreground and background content, we select several common scenes as the background including home, wild, park, beach, sky, road, etc., and then 11 common animal categories including person, rabbit, sheep, cow, etc. were selected from [10], [33] as the foreground objects. Then we used the combinations of the foreground and background keywords to search videos on YouTube, and downloaded the relevant video clips as the basis of our dataset. We used the SceneDetect tool [34] to segment them into clips, as to maintain the scene consistency of each video. Then we manually screened out substandard video clips when the SceneDetect tool fails due to dramatic shot changes. Finally, we obtained 1,126 video clips with an average duration of 13.3 seconds.

C. Sketch Collection

We developed a user interface for sketching and verification. The painters were required to watch the video clip first and drew one storyboard sketch by recalling the memorable content of the clip, including major foreground objects and background elements of the scene. Then they drew keyframe sketches corresponding to specific video frames indicating the detailed content variation process. The specific sketching principles are: (1) each object resembles its video counterpart;



(a) Category distribution of the SketchVideo dataset

(b) Category distribution of other foreground objects

Fig. 5. (a) Category distribution in the SketchVideo dataset. (b) Category distribution of other foreground objects.

TABLE I
COMPARISON WITH EXISTING SKETCH DATASETS. SKETCH AMOUNT AND MEAN STROKES OF SEVERAL DATASETS ARE CITED FROM [8].

Dataset	Objects			Sketch Amount	Paired Element		Instance-level Match?	Mean Strokes
	Fine-grained?	Multi-instance?	Categories		Type	Amount		
TU-Berlin[9]	-	-	250	20000	-	-	-	17.55
Sketchy[10]	✓	-	125	75471	Images	12500	-	16.06
QMUL-Shoe[11]	✓	-	1	419	Images	419	✓	-
QMUL-Chair[11]	✓	-	1	297	Images	297	✓	-
QMUL-Handbag[30]	✓	-	1	568	Images	568	✓	-
SketchyScene[12]	✓	✓	44	29056	Images	4730	✓	-
SketchyCOCO[31]	✓	✓	17	14081	Images	14081	✓	-
FS-COCO[32]	✓	✓	150	10000	Images	10000	✓	74.3
TSF[3]	-	✓	3	100	Videos	298	-	-
FG-SBVR[8]	✓	-	1	1448	Videos	528	✓	102.4
Our Dataset (SketchVideo)	✓	✓	**61**	**6713**	**Videos**	**1126**	✓	**1922.58**

(2) typical background elements are depicted to represent the current scene; (3) storyboard sketches depict major objects of certain activities at different timings; (4) keyframe sketches depict slight changes of each object. Auditors can monitor the sketching process of painters and provide timely feedback on the verification interface.

D. Ground Truth Annotation

In order to get the category pool for annotation, we selected the main categories from [33] and typical background categories of scenes (home, wild, park, beach, sky, road, etc.), resulting in 61 categories for annotation. We design a user interface for data annotation and annotation check. Each foreground object is required to annotate. For background categories with high redundancy (clouds, grasses, stones and trees), we merge adjacent multiple objects of the same category into a single object.

We provide three types of annotations for sketches and videos: (1) instance-level sketch annotation of category, strokes, orientation, integrity (describing the degree of occlusion), similarity between the sketch and the video and quality (i.e., how easy it can be recognized); (2) bounding boxes of each foreground object in keyframe sketches and video frames; (3) temporal alignment between sketches and video clips in original untrimmed videos.

E. Dataset Analysis

The SketchVideo dataset contains abundant sketch-video pairs with multiple categories. Each video corresponds to one storyboard sketch and 4.96 keyframe sketches on average. Each sketch is composed of 2.55 foreground objects and 2.68 background objects on average.

a) Category Analysis: In our dataset, there are 35,132 objects in total which consist of 48.82% foreground objects and 51.18% background objects, respectively, and the detailed category distribution is shown in Figure 5a. The foreground objects contain a few commonly used foreground such as person and ten animal categories, which are used to construct queries, and other foreground such as bowl, car, motorcycle, etc. (shown in Figure 5b).

b) Diversity: In order to make our dataset suitable for real-world applications, our dataset contains objects in diverse categories, appearances and scales, and varied scenes with diversity. The scene variation can be summarized in two aspects (see Figure 2): (1) the background variation and the foreground object variation, which demonstrates the capability of our dataset for fine-grained scene-level sketch studies; (2) the sketches are drawn by painters with different levels of painting skills (see Figure 6).

c) Dataset Augmentation: Inspired by the data augmentation strategy in [12], we also allow the users to create new scene sketches by arbitrarily combining different objects. Furthermore, we provide stroke-level manipulation, which can

Fig. 6. Examples of different painting levels in SketchVideo dataset. The good-painting level sketches contain vivid depiction of foreground objects (e.g. detailed depiction of appearance, size and pose, etc.) and background elements. The normal-painting level ones lack some details of foreground and background. And the poor-painting level sketches are roughly drawn and only depict the outline of objects.

further enable fine-tuning of each sketch object's appearance.

d) Quality Evaluation of Our Dataset: The integrity, similarity and quality of objects in medium and high levels (out of low, medium and high for annotation) occupy 91.43%, 90.75%, 90.69% of cases respectively. It demonstrates that the overall quality of our dataset is sufficient to support further fine-grained sketch-based video research.

e) Comparison with Existing Sketch Datasets: We made several statistical analysis on SketchVideo and the existing sketch datasets (see Table I). Although TU-Berlin [9] and Sketchy [10] have a large amount of sketches and object categories, they cannot enable fine-grained instance-level retrieval due to the lack of instance-level matches. Most of the remaining datasets in Table I support the fine-grained cross-modal retrieval task, among which SketchyScene [12], SketchyCOCO [31] and FS-COCO [32] are capable of fine-grained scene-level retrieval with multiple instances, yet they are all limited to the image domain. Compared with the video retrieval datasets TSF [3] and FG-SBVR [8], our dataset covers more object categories and contains more sketches, and the sketches in our dataset depict not only fine-grained single instances but also multiple objects in diverse scenes, which is more suitable for real-world sketch-related video research.

IV. Method

A. Overview

In this work, we propose a SQ-GCN model for fine-grained scene-level SBVR task by matching the spatial-temporal content between storyboard sketch and video (see Figure 7). The SQ-GCN model includes three components, i.e., storyboard sketch encoding, video encoding and feature matching. To efficiently sample video frames aligned with storyboard sketch content for video encoding, we train a sketch-image correlation model with frame-level sketch-video pairs to select the most relevant video frames given a storyboard sketch query. In order to encode features of the storyboard sketch and video, we design two encoding branches to perform the appearance and

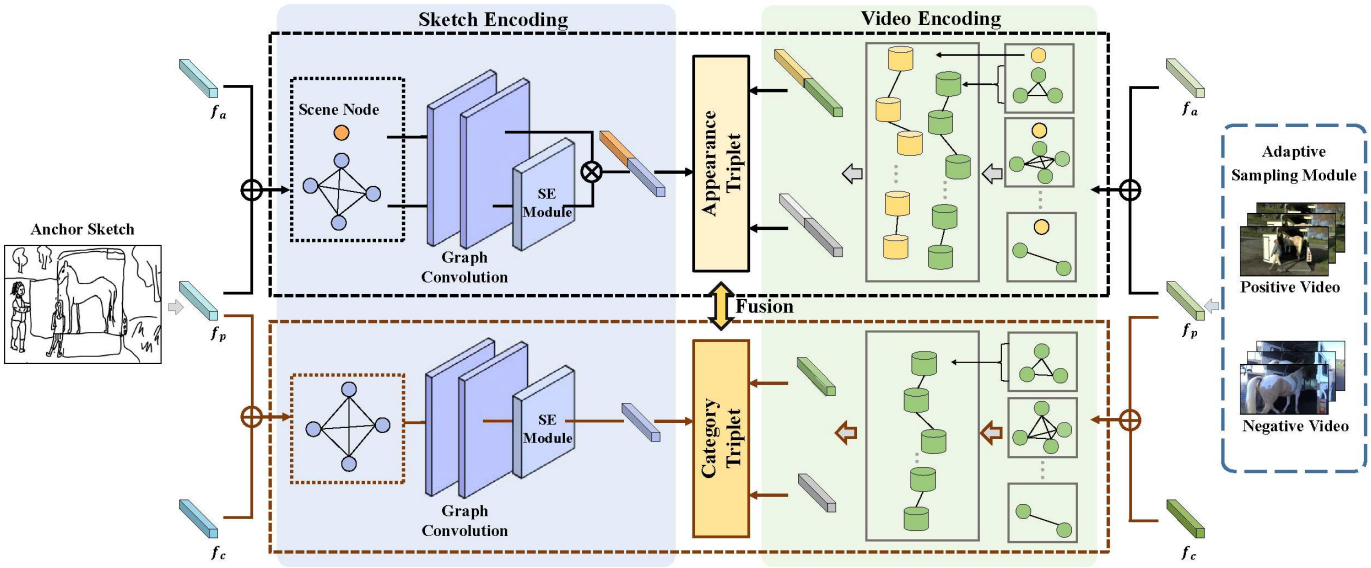


Fig. 7. The pipeline of our SQ-GCN for fine-grained scene-level SBVR with storyboard sketches. Our network includes three components, i.e. storyboard sketch encoding, video encoding and feature matching. We firstly use an adaptive video frame sampling strategy to select relevant video samples. Then we construct appearance graphs (appearance and position features) and category graphs (category and position features without scene node) for both sketch and video. Through GCN for sketch encoding and ST-GCN for video encoding, we embed sketch and video into a common feature space, and use a triplet network for training.

category alignment. In each branch, we construct a spatial graph of sketch input and a spatial-temporal graph for the sampled video frames, where each node represents a single object instance. The node features are initialized with appearance or category features in each branch, with the corresponding position features further added to indicate the scene layout information. Then the sketch and video embeddings are fed to a triplet network for feature matching.

During the training stage, we train the appearance branch and category branch, respectively. During the inference stage, we use a late fusion strategy where the sketch-video distances of both branches are computed together to comprehensively obtain the retrieval results.

B. Storyboard Sketch Encoding

a) Graph Construction: Given a storyboard sketch S, we construct a category graph $G_{s,c}$ and an appearance graph $G_{s,a}$. Both graphs contain n instance nodes $g_s^{(i)}(i = 1, 2, ..., n)$ representing instance-level information. Based on the nodes' positional relationship, we define the edge weight $A_{i,j} \in (0, 1)$ using normalized Distance-IoU [35] computed with their bounding boxes. Besides, the appearance graph $G_{s,a}$ has an additional scene node $g_s^{(0)}$ representing the global appearance. The scene node and instance nodes are updated through graph convolutions simultaneously. During feature updating, the scene node is separated from the instance nodes and does not conduct message passing with them.

b) Node Representation: The features of each node in $G_{s,c}$ and $G_{s,a}$ are initialized with appearance features and category features respectively, with positional embeddings added. We use pre-trained GoogLeNet Inception-V3 [36] to obtain 2048-d appearance features f_a, and apply pre-trained Bert model [37] to encode the category label into 768-d features f_c. Furthermore, to encode layout information, we apply the method proposed in [38], [39] using sine and cosine functions of different frequencies to obtain absolute position features f_p, and then add it to f_a and f_c respectively.

c) Graph Encoding: After constructing graphs and initializing the features, we adopt a two-layer GCN [25] for feature embedding, the node features F_s^{l+1} of graph G_s (the feature updating process of $G_{s,c}$, $G_{s,a}$ is similar) updated in the l-th layer can be represented as follows:

$$F_s^{l+1} = ReLU(AF_s^l W^l) \tag{1}$$

where A is the normalized adjacency matrix, and W^l is the trainable weights of the l-th layer.

After message passing through GCNs, we consider all the instance nodes' features $F_s^{(i)}(i = 1, 2, ..., n)$ as different channels, and then employ the Squeeze-and-Excitation (SE) module [40] to obtain the encoded local features:

$$F_s^I = \frac{1}{n}\sum_{i=1}^{n} \sigma(W_{se}^{(1)} ReLU(W_{se}^{(0)} \bar{F}_s^{(i)}))F_s^{(i)} \tag{2}$$

where σ is the Sigmoid activation function, and W_{se} is the weight matrix of the SE module.

Finally, for the category graph, we take instance nodes' features $F_{s,c}^I$ as category features $F_{s,c}$. For the appearance graph, we get its appearance features $F_{s,a}$ by concatenating $F_{s,a}^I$ with the scene node's features $F_{s,a}^{(0)}$.

$$F_{s,a} = (F_{s,a}^{(0)}; F_{s,a}^I). \tag{3}$$

C. Video Encoding

a) Adaptive Video Frame Sampling: The key issue of video encoding is that multiple video frames may convey

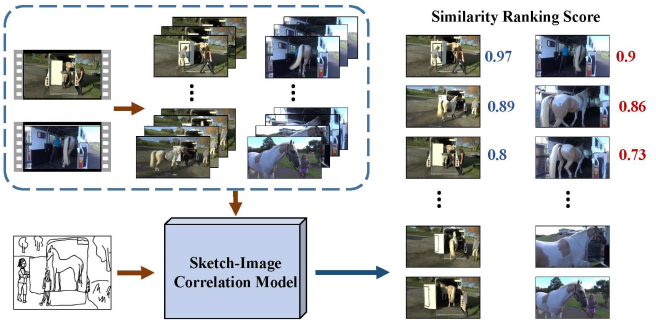


Fig. 8. Illustration of adaptive video sampling module. The module can adaptively sample the most visually relevant frames given a storyboard sketch as query.

the same information due to videos' temporal redundancy. To reduce huge computational cost in the commonly used frame-by-frame processing and align the content of sketch and video efficiently, we propose an adaptive video frame sampling method named *FrameSampler* to sample the most visually relevant frames to a storyboard sketch query (see Figure 8). Benefiting from the paired data of sketch and image in the keyframe of SketchVideo (see Figure 2), we train a sketch-image correlation model using the same triplet loss in Section IV-D for cross-modal semantic similarity analysis. During video retrieval, we first sparsely sample several video frames as the candidates, and then apply the correlation model to further select the most relevant k frames for video encoding.

Benefiting from the lightweight design of our video sampling network, the extra run-time cost caused by FrameSampler is negligible (about **1%** of the total retrieval run-time) (see Section V-B).

b) Graph Construction: Given k sampled frames, we construct spatial-temporal graphs to extract the spatial and temporal features. For each frame, we construct spatial graphs similar to that of the storyboard sketch, with appearance features f_a extracted by a ResNet-152 [41] network. The number of spatial graphs is determined when we achieve optimal experimental results on the training dataset. For the appearance graph, after fusing the spatial instance features through SE module into a single temporal instance node, we build two temporal graphs G_T^S and G_T^I, which consist of each frame's scene node $v_t^s(t = 1, .., T)$ and temporal instance node $v_t^I(t = 1, .., T)$, respectively. The nodes are connected in temporal order, representing global and local temporal features. For the category graph, we only construct one temporal graph with instance nodes since there are no global features.

c) Video Graph Encoding: We use our Storyboard Sketch Encoder to encode spatial video graphs, and apply the two-layer GCN and SE module to temporal video graphs successively. Then we obtain the video embeddings $F_{v,c}$ and $F_{v,a}$ representing overall information in both category and appearance levels.

D. Feature Matching

a) Loss Function: After obtaining storyboard sketch features F_s and video features F_v, we adopt a fine-grained cross-modal triplet loss to learn the semantic correlation of sketch and video, where an input triplet contains the features of the query sketch F_{s_i}, the features of the corresponding video F_{v_i}, and the features of the hard negative video $F_{v_i^h}$. The loss function is denoted as:

$$L = \frac{1}{b}\sum_{i=1}^{b} L_{s_i}^{v_i, v_i^h} \tag{4}$$

$$L_{s_i}^{v_i, v_i^h} = \max(0, d(F_{s_i}, F_{v_i}) - d(F_{s_i}, F_{v_i^h}) + \triangle) \tag{5}$$

$$v_i^h = \arg\min_{v_j} d(F_{s_i}, F_{v_j}), i \neq j \tag{6}$$

where $d(\cdot, \cdot)$ is the Euclidean distance function, $\triangle$ is the margin between positive and negative pairs, and b is the batch size.

b) Inference: During testing, we sort the candidate video pools based on the Euclidean distance D between sketch features F_s and video features F_v. Utilizing both category and appearance information, we fuse the two distances D_c and D_a with appropriate weights:

$$D = \alpha D_c + (1 - \alpha) D_a \tag{7}$$

where α is determined during experiments empirically.

E. Implementation Details

For graph construction where each node represents a foreground instance, we train two YOLOv4 models with annotated bounding boxes of the training set (the weights of the two models are not shared), and then use the models to detect the objects in sketch and video for testing, respectively.

For the adaptive video frame sampling, we use the keyframe sketches in the training set to train the sketch-image correlation model, whose weights are fixed during video retrieval.

We use the PyTorch framework to implement our method with a single RTX 2080Ti GPU. The GCNs' parameters are initialized with Kaiming Initialization. During training, we use the Adam optimizer with initial learning rate 0.0001 and batch size 100. The number of adaptively sampled video frames k is set to 3. The margin $\triangle$ in Eq. 5 is set to 100, and α in Eq. 7 is set to 0.65. The sketch and video embeddings are all 512-d vectors in different methods. The number of instance nodes n is a hyper-parameter which is fixed during graph construction and can be modified to fine-tune SQ-GCN. If the number of instances in the storyboard sketch and video frames is less than n, the features of the rest nodes will be initialized with zeros. Otherwise, we will calculate the size of each object instance and select the top n as instance nodes. Specifically, we set the number of graph nodes to 21, which includes 1 scene node (set to zero in the category model) and 20 instance nodes (n=20) by traversing the maximum number of instances contained in each storyboard sketch. We have conducted the experiment with different graph nodes to evaluate the model's best performance with 21 graph nodes (See Effect of Graph Nodes of Section V-B).

Fig. 9. Top-3 results of our scene-level SBVR. Green rectangles represent the corresponding ground truth videos.

V. Experiments

A. Datasets and Evaluation Metrics

a) Datasets: Since existing SBVR-related datasets in [3], [8] are not publicly available, we split our SketchVideo dataset into training set, validation set and test set, with 752, 187 and 187 video clips, respectively.

b) Evaluation Metrics: We evaluate the retrieval results by two evaluation metrics. First, we use retrieval accuracy [11] acc.@K, namely the percentage of sketch queries whose true-match videos are ranked among the top K retrieved results. In our case, we set K to 1, 5 and 10 respectively. Second, we also apply mean Average Precision (mAP) scores to evaluate the general ranking results.

TABLE II
Ablation study on the configuration of graph construction and feature fusion strategy. "T-GCN": temporal graph convolutional networks (in video graph encoding of Section IV-C), "App.": appearance features, "Cat.": category features, "Pos.": position features.

Method		mAP	Acc.@1	Acc.@5	Acc.@10
Baseline (w/o adaptive sampling)		30.57	19.35	38.22	50.78
Baseline (w/o SE module)		34.60	22.01	48.29	64.31
Baseline		35.22	22.68	49.19	63.90
Graph Feature	App. (w/o scene)	18.80	8.01	27.56	41.38
	App. (w/o SE Module)	37.15	23.29	54.28	66.88
	App. (w/o T-GCN)	37.64	23.72	52.56	64.10
	App.	39.92	25.64	55.98	70.94
	App. & Pos.	48.75	33.17	67.63	80.13
	Cat. (w/o SE Module)	37.94	18.16	63.89	81.84
	Cat. (w/o T-GCN)	37.96	18.27	65.06	81.73
	Cat.	40.17	21.15	63.78	82.53
	Cat. & Pos.	51.78	38.25	68.81	80.34
Fusion type (Full feature)	Early	58.86	45.19	75.32	88.78
	Collaborative	62.68	47.12	81.09	90.70
	Late (SQ-GCN)	**66.74**	**52.78**	**83.98**	**93.80**

B. Ablation Study

a) Baseline: We set up a baseline model, where two fully-connected layers are directly applied to the appearance features of sketch and adaptively sampled video frames, and then the frames' features are fused with the SE module. Finally, the cross-modal triplet loss is utilized to train the model. Row 1 in Table II shows the results of the baseline model, which represents coarse-grained image-level matching. Note that the adaptive sampling strategy can remarkably improve the retrieval performance.

b) Graph Construction Analysis: To evaluate the effectiveness of graph related configurations, we conduct several experiments on different combinations of graph features and specific graph modules.

The graph features can be initialized with appearance features, category features and position features. When utilizing appearance features, the results in Table II show that the graph model without the scene node performs significantly worse than the model with the scene node and our baseline model. Therefore, the global scene content plays an important role in appearance feature matching. Furthermore, the SE module also improves the performance of both appearance and category models, so it is important for instances' node feature encoding. In order to investigate whether the ST-GCN of our model can capture temporal information effectively, we set the nodes in temporal graphs to be completely separate from each other (indicated as "w/o T-GCN" in Table II). The results show that the temporal connections are useful for video encoding.

In order to address whether the appearance and category models are complementary, we compare the performance of appearance and category models (see App. and Cat. in Table II). The mAP performance of the category model is slightly better, demonstrating that the category correlation context is a strong clue in multi-object video retrieval. As a comparison, the appearance model performs better on ACC.@1 index, which proves that the fine-grained appearance features that sketch provides are necessary for accurate retrieval. Therefore, we use both the appearance and category models for feature

extraction.

As shown in Table II (App. vs. App. & Pos., Cat. vs. Cat. & Pos.), position features indicating layout information are effective to boost the performance of both the appearance and the category models.

c) Fusion Methods: To find out the most effective way to utilize both category and appearance information, several experiments with different fusion methods are conducted. Table II shows the comparison results. The early fusion strategy is to simply add the two features. The collaborative method fuses appearance and category features after they go through the graph convolution, which performs better than the early fusion method. The late fusion method computes the weighted distance sum as in Eq. 7, where $\alpha = 0.65$. We observe that the late fusion method outperforms the other two methods by a large margin.

d) Video Frame Sampling: To prove the effectiveness of our adaptive sampling method, several experiments are conducted in Table III. As k increases to 4, the corresponding model already achieved excellent results for SketchVideo dataset. Note that we tried all the possible combinations of uniformly sampled video frames as fixed sampling points and displayed the best performance in Table III. The results demonstrate the superiority of our adaptive sampling method compared to the fixed sampling method.

TABLE III
THE ABLATION STUDY ON SBVR WITH DIFFERENT SAMPLING STRATEGIES AND DIFFERENT NUMBERS OF SAMPLED VIDEO FRAMES k.

Sampling Method		mAP	Acc.@1	Acc.@5	Acc.@10
Fixed Sampling	k=1	57.90	45.08	73.29	85.04
	k=2	60.50	46.15	76.28	86.54
	k=3	63.03	49.36	79.49	88.46
	k=4	63.08	48.40	80.77	89.74
Adaptive Sampling	k=1	60.87	47.76	77.88	86.86
	k=2	64.84	51.92	81.73	90.06
	k=3	**66.74**	**52.78**	83.98	**93.80**
	k=4	66.67	51.71	**84.83**	92.31

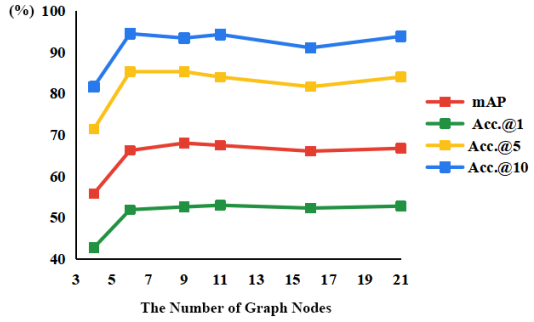


Fig. 10. The ablation study on SBVR with different numbers of graph nodes during graph construction.

e) Effect of Graph Nodes: To evaluate the effectiveness of graph nodes which includes 1 scene node (set to zero in the category model) and n instance nodes, we conduct several experiments with different numbers of graph nodes. As demonstrated in Figure 10, the model's performance is almost the same after the number of graph nodes reaches 9.

C. Robustness of SQ-GCN

To evaluate the robustness of SQ-GCN for fine-grained scene-level SBVR, we adopt two ways to randomly remove strokes from storyboard sketches, including (1) randomly removing 10%-50% strokes at any position or (2) randomly removing 10%-50% continuous strokes of each instance in storyboard sketches to generate the modified test-set and test our model, Scene Sketcher [27] and FG-SBVR [8] on it. Figure 11 shows the performance of the state-of-the-art methods. Our method consistently achieves the best performance, which demonstrates the capabilities of our method for handling incomplete scene sketches. Figure 12 shows several retrieval examples with incomplete storyboard sketches.

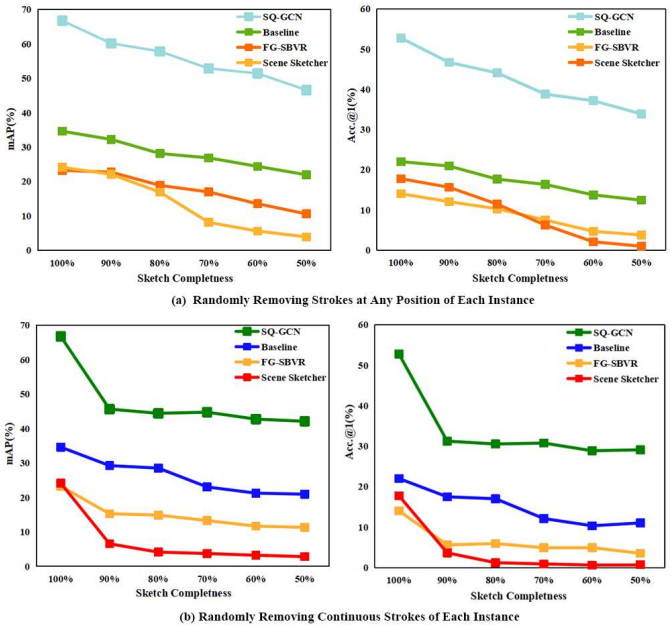


Fig. 11. Performance results of different methods (including SQ-GCN, Baseline, Scene Sketcher [27] and FG-SBVR [8]) with different levels of sketch completeness to demonstrate their model robustness, where certain proportions of (a) strokes at any position or (b) continuous strokes of each instance are randomly removed.

D. Performance of SQ-GCN with Fine-tuned Inception-V3

To extract more effective sketch appearance features, we adopt the sketch classification task and use the Sketchy dataset [10] which contains a large amount of object-level sketches with diverse categories to fine-tune the Inception-V3 [36] pre-trained on ImageNet. As shown in Table IV, we compare the performance of appearance model with pre-trained Inception-V3 and fine-tuned Inception-V3 and the results demonstrate that fine-tuned Inception-V3 can extract more effective sketch appearance features and helps improve the performance of SQ-GCN.

E. Comparison with State-of-the-Art Methods

We compare our method with two related state-of-the-art methods [8][27]. Xu et al. [8] proposed the only deep learning-based method so far. Scene Sketcher [27] is the state-of-the-art method for fine-grained scene-level sketch-based image

Fig. 12. Examples of Top-5 retrieval results given incomplete storyboard sketches of different levels (with 10%, 30% and 50% (a) strokes at any position or (b) continuous strokes of each instance randomly removed respectively).

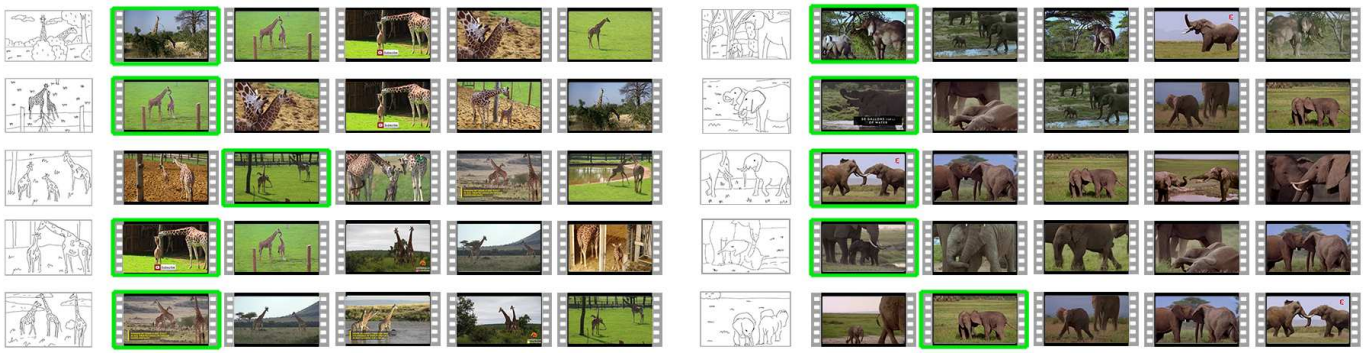


Fig. 13. Examples of Top-5 retrieval results given giraffe/elephant storyboard sketches with similar scene content.

TABLE IV
COMPARISON OF THE APPEARANCE MODEL WITH PRE-TRAINED INCEPTION-V3 AND FINE-TUNED INCEPTION-V3.

Sketch Feature Extration Method	mAP	Acc.@1	Acc.@5	Acc.@10
Pretrained Inception-V3	48.75	33.17	67.63	80.13
Fine-tuned Inception-V3	53.07	38.25	69.23	80.34

TABLE V
COMPARISON BETWEEN OUR MODEL AND THE STATE-OF-THE-ART METHODS INCLUDING FG-SBVR [8] AND SCENE SKETCHER [27].

Method	mAP	Acc.@1	Acc.@5	Acc.@10
FG-SBVR[8]	23.25	14.02	32.23	42.99
Scene Sketcher[27]	24.14	17.76	31.78	37.78
Baseline	35.22	22.68	49.19	63.90
SQ-GCN	66.74	52.78	83.98	93.80
SQ-GCN w/ Fine-tuned Inception-V3	**69.20**	**55.98**	**84.61**	**95.08**

retrieval. Both works support retrieval using one storyboard sketch as query input. Therefore, we reproduce the FG-SBVR method which did not release original codes, and use Scene Sketcher to compare with our own baseline and final model. For FG-SBVR, we randomly sample one video frame as [8] did to extract the appearance feature. For Scene Sketcher, we provide appearance, category, and position features and average the features of sampled video frames. The results in Table V show that SQ-GCN and SQ-GCN with fine-tuned Inception-V3 greatly outperform the previous fine-grained methods. Therefore, our method is effective in fine-grained scene-level SBVR task. Figure 9 shows several retrieval examples with our method on the SketchVideo dataset.

F. Hard Case Analysis

To evaluate our method's fine-grained scene-level retrieval capability, we select 20 giraffe/elephant videos as hard cases for detailed analysis. Their scene contents are similar while the size, amount, movement, direction, etc. of foreground objects and background elements have subtle differences. As shown in Figure 13, our method can capture detailed scene content information in storyboard sketches and retrieve the target video accurately.

VI. APPLICATION

Sketch is a natural input modal that facilitates great creative freedom, thus fine-grained scene-level SBVR has a wide range of applications, such as finding videos in a cellphone album, and collecting video materials for video creation.

In this work, we construct a prototype system that supports interactive fine-grained scene-level SBVR on a PC or a tablet. The user interface (Figure 14) provides two ways of sketch

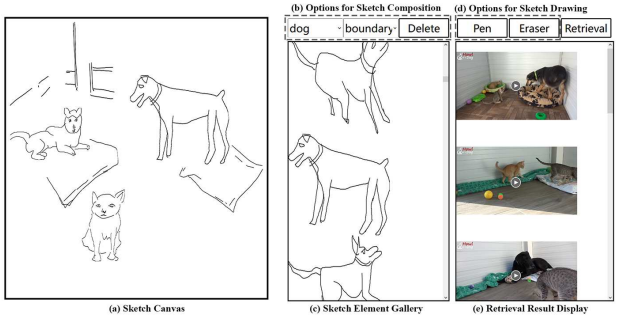


Fig. 14. The user interface of the fine-grained video retrieval system with scene sketch, which supports two types of storyboard sketch depiction (including sketch composition and sketch drawing) and retrieval results display.

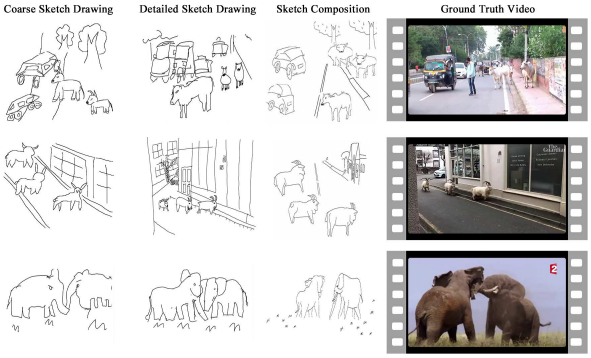


Fig. 15. Examples of user-created sketches by sketch drawing (including coarse drawing and detailed drawing) and sketch composition in the user study.

creation, including sketch composition and sketch drawing. For sketch composition, users change the category options (Figure 14b) and select sketch materials existing in our dataset from the sketch element gallery (Figure 14c). Then they can compose refined scene sketches by resizing, dragging and rotating sketch materials on the sketch canvas (Figure 14a). For sketch drawing, users use pen and eraser (Figure 14d) to depict the desired scene content. The retrieval results are displayed (Figure 14e), and users can watch the retrieved videos by clicking them.

We conducted a user study to evaluate the performance of our design. 8 participants (4 males and 4 females) who rarely draw pictures and have no professional drawing skills were asked to retrieve a randomly-selected video based on sketch drawing and sketch composition methods, respectively. They were asked to draw the scene sketch twice, including coarse drawing without any restrictions and detailed drawing with careful descriptions of scene content. We evaluate our system from three aspects, i.e. effectiveness (the system meets user's retrieval requirement), users' satisfaction with the system (users are relaxed and pleasant when using this system), and efficiency of the system (the system spends little time to obtain correct retrieval results), and we use a 5-point Likert scale for rating (1-Strongly Disagree to 5-Strongly Agree). Our system gets average ratings of 4.25, 4.63 and 4.13 for the three evaluation aspects, so our fine-grained video retrieval system with scene sketches has great potential for general users to retrieve desired videos in a natural and effective way. Some participants commented that "*This retrieval method is quite flexible because I can retrieve the exact scene content whatever I imagined.*", "*Sketch is a general tool for kids, e.g. retrieve their desired cartoon videos through sketching.*"

TABLE VI
THE TIMINGS OF SKETCH CREATION BY DRAWING AND COMPOSITION, AND THE RANKING OF GROUND TRUTH (GT) VIDEOS IN THE USER STUDY.

Participant ID	Time Consumption			GT Video Ranking		
	Coarse Drawing	Detailed Drawing	Composition	Coarse Drawing	Detailed Drawing	Composition
1	59″	2′30″	1′14″	4	2	2
2	1′57″	4′55″	2′5″	4	1	1
3	1′14″	2′27″	1′16″	9	3	2
4	1′38″	3′36″	2′2″	3	1	1
5	1′45″	2′1″	1′42″	6	4	3
6	1′3″	3′32″	1′26″	15	2	4
7	2′12″	4′11″	2′	2	1	2
8	1′48″	3′35″	2′2″	2	1	1

Moreover, we recorded the timings of sketch creation and the rankings of the ground truth videos. Several examples of user-created sketches are displayed in Figure 15. As shown in Table VI and Figure 15, coarse sketch drawings describe instances with sparse lines and vague contours while sketch composition can generate high-quality scene sketches close to the detailed sketch drawing. Besides, sketch composition costs slightly more time than coarse sketch drawing but can retrieve the target video much more accurately. Detailed drawing has a good retrieval performance but takes nearly double the time than sketch composition. Participants commented that "*Sketch composition is a great design. I always feel overwhelmed when I'm drawing because I don't have such skills. Sketch composition not only helps me retrieve videos more accurately, but also makes me feel more comfortable during retrieval.*", "*The searching of the right materials takes time, but I don't have to worry about my poor painting skills anymore, which is fantastic.*". In summary, our system gets positive feedback from participants, which shows that our system design is effective and practical and our system has great potential for the applications of fine-grained scene-level SBVR.

VII. CONCLUSION

In this paper, we present the first fine-grained scene-level sketch-video-paired dataset named SketchVideo, with clip-level storyboard sketch and frame-level keyframe sketches depicted for each video. Benefiting from our proposed dataset SketchVideo, we investigate the new scene-level SBVR task with the storyboard sketch query, and propose a novel SQ-GCN model to perform feature matching between sketches and videos. In order to improve video encoding efficiency, we design FrameSampler, an adaptive video sampling strategy based on frame-level sketch-video pairs. Extensive experimental results demonstrate that our method achieves the-state-of-the-art performance for fine-grained scene-level SBVR. These results also evaluate the efficiency of our SketchVideo dataset, which has great application potential in sketch-based video research. For example, the keyframe sketches can be used for video synthesis, and the storyboard sketches can be used for scene-level video localization and summarization. Specifically, sketch-based video summarization aims to automatically generate storyboard sketches from video clips, which provides

an interactive representation to annotate and visualize the major scene content of video clips [42] and supports flexibly editing or adding object sketches in a sketch-based interface. Furthermore, we will try CLIP [43] to simplify our SQ-GCN model inspired by [44], [32] and adapt it to scene sketches' feature encoding in the future.

REFERENCES

[1] E. Tulving, "Elements of episodic memory," 1983.

[2] J. A. Landay and B. A. Myers, "Sketching interfaces: Toward more human interface design," *Computer*, vol. 34, no. 3, pp. 56–64, 2001.

[3] J. P. Collomosse, G. McNeill, and Y. Qian, "Storyboard sketches for content based video retrieval," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2009, pp. 245–252.

[4] R. Hu and J. Collomosse, "Motion-sketch based video retrieval using a trellis levenshtein distance," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2010, pp. 121–124.

[5] R. Hu, S. James, and J. Collomosse, "Annotated free-hand sketches for video retrieval using object semantics and motion," in *International Conference on Multimedia Modeling (MMM)*, 2012, pp. 473–484.

[6] R. Hu, S. James, T. Wang, and J. Collomosse, "Markov random fields for sketch based video retrieval," in *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval (ICMR)*, 2013, pp. 279–286.

[7] S. James and J. Collomosse, "Interactive video asset retrieval using sketched queries," in *Proceedings of the 11th European Conference on Visual Media Production*, 2014, pp. 1–8.

[8] P. Xu, K. Liu, T. Xiang, T. M. Hospedales, Z. Ma, J. Guo, and Y.-Z. Song, "Fine-grained instance-level sketch-based video retrieval," *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, vol. 31, no. 5, pp. 1995–2007, 2020.

[9] M. Eitz, J. Hays, and M. Alexa, "How do humans sketch objects?" *ACM Transactions on graphics (TOG)*, vol. 31, no. 4, pp. 1–10, 2012.

[10] P. Sangkloy, N. Burnell, C. Ham, and J. Hays, "The sketchy database: learning to retrieve badly drawn bunnies," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–12, 2016.

[11] Q. Yu, F. Liu, Y.-Z. Song, T. Xiang, T. M. Hospedales, and C.-C. Loy, "Sketch me that shoe," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 799–807.

[12] C. Zou, Q. Yu, R. Du, H. Mo, Y.-Z. Song, T. Xiang, C. Gao, B. Chen, and H. Zhang, "Sketchyscene: Richly-annotated scene sketches," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 421–436.

[13] Z. Chen, J. Lu, J. Feng, and J. Zhou, "Nonlinear structural hashing for scalable video search," *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, vol. 28, no. 6, pp. 1421–1433, 2017.

[14] J. Song, H. Zhang, X. Li, L. Gao, M. Wang, and R. Hong, "Self-supervised video hashing with hierarchical binary auto-encoder," *IEEE Transactions on Image Processing (TIP)*, vol. 27, no. 7, pp. 3210–3221, 2018.

[15] S. Li, Z. Chen, J. Lu, X. Li, and J. Zhou, "Neighborhood preserving hashing for scalable video retrieval," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 8212–8221.

[16] Y. Li, R. Wang, Z. Huang, S. Shan, and X. Chen, "Face video retrieval with image query via hashing across euclidean space and riemannian manifold," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2015, pp. 4758–4767.

[17] A. Araujo and B. Girod, "Large-scale video retrieval using image queries," *IEEE transactions on circuits and systems for video technology (TCSVT)*, vol. 28, no. 6, pp. 1406–1420, 2017.

[18] R. Xu, L. Niu, J. Zhang, and L. Zhang, "A proposal-based approach for activity image-to-video retrieval." in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020, pp. 12 524–12 531.

[19] L. Liu, J. Li, L. Niu, R. Xu, and L. Zhang, "Activity image-to-video retrieval by disentangling appearance and motion," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2021, pp. 1–9.

[20] B. Zhang, H. Hu, and F. Sha, "Cross-modal and hierarchical modeling of video and text," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 374–390.

[21] M. Wray, D. Larlus, G. Csurka, and D. Damen, "Fine-grained action retrieval through multiple parts-of-speech embeddings," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 450–459.

[22] Y. Song and M. Soleymani, "Polysemous visual-semantic embedding for cross-modal retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1979–1988.

[23] S. Chen, Y. Zhao, Q. Jin, and Q. Wu, "Fine-grained video-text retrieval with hierarchical graph reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 638–10 647.

[24] S.-F. Chang, W. Chen, H. J. Meng, H. Sundaram, and D. Zhong, "Videoq: an automated content based video search system using visual cues," in *Proceedings of the fifth ACM international conference on Multimedia (MM)*, 1997, pp. 313–324.

[25] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2017.

[26] Z. Zhang, Y. Zhang, R. Feng, T. Zhang, and W. Fan, "Zero-shot sketch-based image retrieval via graph convolution network," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, no. 07, 2020, pp. 12 943–12 950.

[27] F. Liu, C. Zou, X. Deng, R. Zuo, Y.-K. Lai, C. Ma, Y.-J. Liu, and H. Wang, "Scenesketcher: Fine-grained image retrieval with scene sketches," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020, pp. 718–734.

[28] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the AAAI conference on artificial intelligence (AAAI)*, 2018.

[29] K. Hornbæk and A. Oulasvirta, "What is interaction?" in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI)*, 2017, pp. 5040–5052.

[30] 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 *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5551–5560.

[31] C. Gao, Q. Liu, Q. Xu, L. Wang, J. Liu, and C. Zou, "Sketchycoco: Image generation from freehand scene sketches," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5174–5183.

[32] P. N. Chowdhury, A. Sain, A. K. Bhunia, T. Xiang, Y. Gryaditskaya, and Y.-Z. Song, "Fs-coco: towards understanding of freehand sketches of common objects in context," in *European Conference on Computer Vision*. Springer, 2022, pp. 253–270.

[33] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proceedings of the European conference on computer vision (ECCV)*, 2014, pp. 740–755.

[34] https://pyscenedetect.readthedocs.io/en/latest/.

[35] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-iou loss: Faster and better learning for bounding box regression," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, no. 07, 2020, pp. 12 993–13 000.

[36] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2016, pp. 2818–2826.

[37] H. Xiao, "bert-as-service," https://github.com/hanxiao/bert-as-service, 2018.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems (NIPS)*, 2017, pp. 5998–6008.

[39] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proceedings of the European conference on computer vision (ECCV)*, 2020, pp. 213–229.

[40] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2018, pp. 7132–7141.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2016, pp. 770–778.

[42] C.-X. Ma, Y.-J. Liu, H.-A. Wang, D.-X. Teng, and G.-Z. Dai, "Sketch-based annotation and visualization in video authoring," *IEEE Transactions on Multimedia (TMM)*, vol. 14, no. 4, pp. 1153–1165, 2012.

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

[44] P. Sangkloy, W. Jitkrittum, D. Yang, and J. Hays, "A sketch is worth a thousand words: Image retrieval with text and sketch," in *European Conference on Computer Vision*. Springer, 2022, pp. 251–267.

Ran Zuo received the BS degree from Beijing Normal University, China, in 2018. She is currently pursuing her Ph.D. degree with Beijing Key Lab of Human-Computer Interaction, Institute of Software, Chinese Academy of Sciences, China. Her research interests include sketch interaction and computer vision.

Xiaoming Deng received the bachelor's and master's degrees from Wuhan University, and the PhD degree from the Institute of Automation, Chinese Academy of Sciences (CAS). He is currently a professor with the Institute of Software, CAS. He has been a research fellow with the National University of Singapore, and a postdoctoral fellow with the Institute of Computing Technology, CAS, respectively. His main research topics are in computer vision, and specifically related to 3D reconstruction, human motion tracking and synthesis, and natural user interfaces.

Keqi Chen received the Bachelor's degree from Southeast University, China in 2019, and the Master's degree from the University of Chinese Academy of Sciences (UCAS), China in 2022. His research interests include computer vision and human-computer interaction.

Zhengming Zhang received his B.S. degree from the China University of Petroleum, Beijing in 2016. He is currently pursuing his Ph.D. degree with the University of Chinese Academy of Sciences, Beijing, China. His current research interests include human-computer interaction and computer vision.

Yu-Kun Lai received his Bachelor's and Ph.D. degrees in computer science from Tsinghua University in 2003 and 2008, respectively. He is currently a Professor in the School of Computer Science & Informatics, Cardiff University. His research interests include computer graphics, geometry processing, image processing, and computer vision. He is on the editorial board of *The Visual Computer*.

Fang Liu received her Ph.D. degree from the University of the Chinese Academy of Sciences (UCAS), Beijing, China, in 2021. She is currently a postdoc at Tsinghua University. Her research interests include computer vision, sketch interaction, and affective computing.

Cuixia Ma received the B.S. and M.S. degrees from Shandong University, China, in 1997 and 2000, respectively, and the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2003. She is now a Professor with the Institute of Software, Chinese Academy of Sciences. Her research interests include Human-computer interaction, Sketch interface, and Multi-modal fusion.

Hao Wang received his Ph.D. degree from the University of Tokyo, co-supervised by professors from the University of California, Berkeley. He is currently leading the AI product development team at Alibaba Cloud, Alibaba Group. Before that, he was a Chief Data Scientist at Qihoo 360 Inc and a professor at the Chinese Academy of Sciences. His current research interests include Large Language Models and conversational AI.

Yong-Jin Liu received the B.Eng. degree from Tianjin University, Tianjin, China, in 1998, and the M.Phil. and Ph.D. degrees from the Hong Kong University of Science and Technology, Hong Kong, China, in 2000 and 2004, respectively. He is now a Professor with BNRist, Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include computational geometry, computer vision, cognitive computation, and pattern analysis. For more information, visit http://cg.cs.tsinghua.edu.cn/people/~Yongjin/Yongjin.htm

Hongan Wang received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 1999. He is a Professor with the Institute of Software, Chinese Academy of Sciences. He is currently the Director of Intelligence Engineering Laboratory. His research interests include human-computer interaction, real-time intelligence, and real-time active database.