SceneSketcher-v2: Fine-Grained Scene-Level Sketch-Based Image Retrieval Using Adaptive GCNs

Fang Liu, Xiaoming Deng, *Member, IEEE*, Changqing Zou, Yu-Kun Lai, *Member, IEEE*, Keqi Chen, Ran Zuo, Cuixia Ma, Yong-Jin Liu, *Senior Member, IEEE*, and Hongan Wang

***Abstract*— Sketch-based image retrieval (SBIR) is a long-standing research topic in computer vision. Existing methods mainly focus on category-level or instance-level image retrieval. This paper investigates the fine-grained scene-level SBIR problem where a free-hand sketch depicting a scene is used to retrieve desired images. This problem is useful yet challenging mainly because of two entangled facts: 1) achieving an effective representation of the input query data and scene-level images is difficult as it requires to model the information across multiple modalities such as object layout, relative size and visual appearances, and 2) there is a great domain gap between the query sketch input and target images. We present SceneSketcher-v2, a Graph Convolutional Network (GCN) based architecture to address these challenges. SceneSketcher-v2 employs a carefully designed graph convolution network to fuse the multi-modality information in the query sketch and target images and uses a triplet training process and end-to-end training manner to alleviate the domain gap. Extensive experiments demonstrate SceneSketcher-v2 outperforms state-of-the-art scene-level SBIR models with a significant margin.**

Manuscript received August 10, 2021; revised February 6, 2022 and March 22, 2022; accepted April 29, 2022. Date of publication May 20, 2022; date of current version May 26, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61872346 and Grant 61725204, in part by the Tsinghua University Initiative Scientific Research Program, in part by the Natural Science Foundation of Beijing under Grant 4212029, in part by the China Postdoctoral Science Foundation under Grant 2021M701889, and in part by the Newton Prize 2019 China Award under Grant NP2PB/100047. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Soma Biswas. *(Fang Liu and Xiaoming Deng contributed equally to this work.) (Corresponding authors: Cuixia Ma; Yong-Jin Liu.)*

Fang Liu is with the Institute of Software, Chinese Academy of Sciences, University of Chinese Academy of Sciences, Beijing 100190, China, and also with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: lfang@tsinghua.edu.cn).

Xiaoming Deng, Keqi Chen, Ran Zuo, Cuixia Ma, and Hongan Wang are with the State Key Laboratory of Computer Science and the Beijing Key Laboratory of Human-Computer Interaction, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100190, China (e-mail: xiaoming@iscas.ac.cn; chenkeqi19@mails.ucas.ac.cn; zuoran18@mails.ucas.ac.cn; cuixia@iscas.ac.cn; hongan@iscas.ac.cn).

Changqing Zou is with the State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310027, China, and also with the Research Center for Artificial Intelligence and Fine Arts, Zhejiang Laboratory, Hangzhou 310058, China (e-mail: aaronzou1125@gmail.com).

Yu-Kun Lai is with the School of Computer Science and Informatics, Cardiff University, Cardiff CF24 4AG, U.K. (e-mail: yukun.lai@cs.cardiff.ac.uk).

Yong-Jin Liu is with the MOE-Key Laboratory of Pervasive Computing, BNRist, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: liuyongjin@tsinghua.edu.cn).

Digital Object Identifier 10.1109/TIP.2022.3175403

***Index Terms*— Sketch-based image retrieval, graph convolutional network, scene sketch, fine-grained image retrieval.**

I. Introduction

SKETCHING is a natural and intuitive form of communication for a human being to express their concepts or ideas. Using a sketch as query data for image retrieval is an increasingly important research topic because of the popularity of touch-screen devices in recent years. Although the research towards Sketch Based Image Retrieval (SBIR) has spanned over two decades, most of the existing SBIR methods are mainly category-level or object-level as illustrated in Fig. 1. Those category-level SBIR methods mainly aim at searching for the images belonging to a specific category depicted by the query input sketch, while those object-level SBIR methods predominantly focus on retrieving the images having the target objects with a sketch which usually includes a single free-hand drawn object.

Scene-level images exist in a large portion of the image data in real world and more importantly more and more images would share similar content or capture similar scenes as the amount of images increases. Despite many conventional SBIR methods are object-level, i.e., conducting image retrieval using a sketch containing only a single object instance and simple background, very few studies have addressed scene-level SBIR problem [3], [4]. Existing scene-level SBIR works classify sketches into dozens of scene categories (e.g. bedroom, forest, ballroom, etc.), and their goal is to retrieve an image of the same scene category as the query scene sketch [4] (see the bottom left part of Fig. 1). These methods, together with those text or image based retrieval methods, are not able to effectively meet the user's specific requirements in some application scenarios, such as searching a target image having a few airplanes with specific poses and relative size as shown in the bottom right part of Fig. 1. Therefore, a SBIR method focusing on fine-grained scene-level image retrieval is required. Recently, Zou *et al.* [5] present a scene sketch dataset with semantic and instance segmentation annotations, named SketchyScene, and conduct a preliminary study of scene-level SBIR. Since the goal of the SBIR task in SketchyScene is to retrieve the specific image corresponding to the input sketch, it can be seen as fine-grained SBIR to some extent. However, the retrieval method used in SketchyScene is largely a pilot study, which

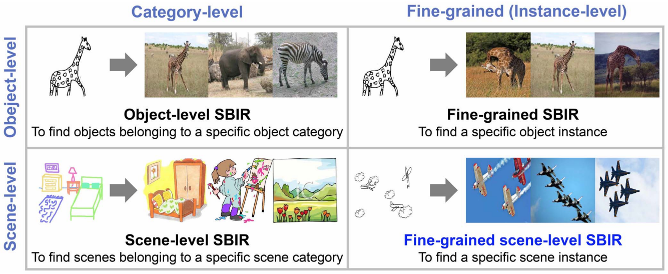
Category-level
Fine-grained (Instance-level)
Obeject-level
Scene-level
Object-level SBIR
To find objects belonging to a specific object category
Fine-grained SBIR
To find a specific object instance
Scene-level SBIR
To find scenes belonging to a specific scene category
Fine-grained scene-level SBIR
To find a specific scene instance

Fig. 1. Illustration of the whole spectrum of SBIR problems. The proposed method, focusing on retrieving the fine-grained scene-level images satisfying the user's specific requirements via a freehand sketch, is in stark contrast to those of object-level SBIR methods [1], [2] and those focusing on retrieving scene-level images of the same scene class [3].

is built upon an object-level SBIR model, and the images are all cartoon images.

This paper investigates the problem of fine-grained SBIR at the scene level (see the bottom right part of Fig. 1). Our method aims to retrieve target images that are consistent with the scene sketch query input in terms of fine-grained object instances and scene information, such as object instances' poses, the relative size and layout of the objects in a scene. This problem is challenging because modeling the information across multiple modalities such as object layout, relative size and visual appearances within a unified network is difficult and there is a great domain gap between the query sketch input and target images. To address these challenges, We propose SceneSketcher-v2, a graph convolutional network that is capable of fusing hierarchical information of scene-level sketches and images, including scene-level spatial layout information, category-level information of objects' semantic attributes and instance-level information of objects' visual appearance. Specifically, we first use a graph-based representation to explicitly model each scene sketch and image, and then leverage an adaptive graph convolutional module to model the spatial and semantic correlations between object categories. We finally train the adaptive graph convolutional network and the visual feature extraction network of sketches and images in an end-to-end manner through a triplet training process. Our network can be well generalized to different scene data because of its favoring flexible graph feature learning. Because the fine-grained scene-level "sketch-photo" pair database is scarce, to verify the superiority of our SceneSketcher-v2, we use modified public scene sketch-photo databases (SketchyCOCO [6] and SketchyScene [5]) to evaluate our SBIR method.

Our main contributions can be summarized as follows:

1) We propose a new GCN-based architecture for fine-grained scene-level SBIR, in which we encode effective hierarchical scene information for feature embedding, including global layout, category-level semantic attributes and instance-level visual features. Moreover, we train the GCN model as well as the visual feature extraction network in an end-to-end manner through a triplet training process;
2) We adopt an adaptive graph convolutional module to model the spatial and semantic correlations between object categories, which increases the flexibility of our model for graph feature learning;
3) Extensive experiments show that our SceneSketcher-v2 achieves retrieval performance that exceeds state-of-the-art SBIR models by a significant margin.

A preliminary version of our work, SceneSketcher, was published in [7]. Compared to the earlier version, the improvements of this work named as SceneSketcher-v2 are three fold:

1) Instead of choosing each object instance as a graph node in SceneSketcher, SceneSketcher-v2 sets each object category as a graph node, which leads to a fixed graph structure and also allows semantic context of different object categories to be used for more effective feature embedding;
2) SceneSketcher-v2 is more general for different scene representation than SceneSketcher. It employs two adjacency matrices to model the scene layout and the correlations between different categories and adopts an adaptive graph for each data sample that denotes its specific pattern rather than a fixed network topology in SceneSketcher. Moreover, the graph building is more powerful in SceneSketcher-v2 where the graph edge weight takes into account the semantic context between different object categories, not just the spatial distance between object instances in SceneSketcher;
3) SceneSketcher-v2 can be trained in an end-to-end manner which can boost the performance of our SBIR framework. As a comparison, the multi-stage modules of SceneSketcher can only be trained in a stage-by-stage manner due to the graph similarity computing process is non-differentiable.

II. Related Work

A. Sketch-Based Image Retrieval (SBIR)

Aiming at using a free-hand sketch to find a specific image from a gallery of natural photos, sketch-based image retrieval has been extensively studied since 1990s [8], and has attracted more attention recently due to the proliferation of touch devices. Most existing SBIR works focus on category-level image retrieval, where the goal is to search the images from the same category. They usually extract representative and shared hand-crafted image descriptors (e.g. SIFT, HOG, etc.) to conduct shape matching between sketches and edge maps of natural images [9]–[12]. Eitz *et al.* [13] utilize descriptors based on the bag-of-features approach for SBIR and present a benchmark for evaluating the performance of large-scale SBIR systems. Their later works include inheriting the GF-HOG and BoVW paradigm for SBIR and extending it by proposing a bag-of-regions (BoR) representation which decomposes images into region representations at multiple scales [10]. Several deep learning based SBIR methods have been introduced recently [14]–[19] and set new records in the major SBIR benchmarks. The first large-scale dataset of sketch-photo pairs is proposed by Sangkloy *et al.* [2], which is used to train cross-domain neural networks and set up an object-level

SBIR benchmark. Hu *et al.* [20] use a semi-supervised metric learning method for anchor graph hashing to conduct SBIR. Zhang *et al.* [21] propose to discover the object representative landmarks and learn the discriminative structural representations for sketch recognition and SBIR.

There is a growing number of studies addressing the zero-shot sketch-based image retrieval (ZS-SBIR) task, which can conduct the retrieval task on unseen object classes [22]. ZS-SBIR is treated as a domain adaptation problem [23] in most circumstances. Yelamarthi *et al.* [24] consider SBIR as the task of generating additional information that is absent in the sketch in order to retrieve similar images, and they propose a generative ZS-SBIR model by taking sketches as inputs and filling in the missing information stochastically for image search. Deng *et al.* [25] employ cross-reconstruction loss and propose a progressive cross-modal semantic network for ZS-SBIR. Dutta and Akata [26] aim to address any-shot (i.e., zero-shot and few-shot) SBIR. Although extensive efforts have been made to make SBIR more efficient and practical, these coarse-grained SBIR works only focus on whether the retrieved image has the same category as the input sketch but the instances' visual details and characteristics gain little attention.

B. Fine-Grained Sketch-Based Image Retrieval (FG-SBIR)

Compared with traditional query form of text or tags, a key advantage of sketch query is that sketch can depict outlines and main characteristics in a simple way. However, these advantages cannot be achieved in category-level SBIR since it only cares if the category of the retrieved image is correct and overlooks the detailed characteristics. Compared to object-level SBIR, FG-SBIR requires the retrieved images contain fine-grained details described in the input scene sketch. Yu *et al.* [1] investigate the problem of instance-level FG-SBIR via a deep triplet-ranking model and introduce a database of sketch-photo pairs with fine-grained annotations. They later expanded this database to 4 datasets, including 3,000+ photos and 8,000+ sketches, accompanied by 32,000+ human triplet annotations to train a better triplet retrieval network [27]. Rather than focusing on feature extraction for cross-domain matching, Song *et al.* [28] instead propose to learn semantic attributes and deep features in a complementary way. They further construct a spatially aware model which combines coarse and fine semantic information in [29]. Li *et al.* [30] aim at bridging the image-sketch gap via combine low level information with high level object parts and attributes, and they collect a dataset with 304 photos and 912 sketches, where each sketch and image is annotated with its semantic parts and associated part-level attributes. In order to retrieve the target image with the least number of strokes possible, Bhunia *et al.* [31] propose an on-the-fly FG-SBIR framework based on a reinforcement learning scheme. Though a growing number of FG-SBIR research works have been proposed in recent years, those works focus on retrieving a single object, which cannot be applied to many real-world applications. In this paper, we explore the problem of scene-level fine-grained SBIR instead, which utilizes local features such as object instances and their visual details, as well as global context, e.g. scene layout.

In addition to the fine-grained scene-level SBIR addressed in this paper, other related fine-grained computer vision tasks include fine-grained classification and retrieval using other modalities, such as image-text fine-grained retrieval [32], image-video fine-grained cross-modal retrieval [33], image-3D fine-grained tasks [34], etc. Targeting fine-grained visual classification, Du *et al.* [35] combine a progressive training framework to learn category-consistent features at specific granularities. Huang *et al.* [36] tackle fine-grained image categorization under the few-shot setting. Wei *et al.* [37] conduct a systematic survey of fine-grained image analysis studies with deep learning methods, and consolidate fine-grained image recognition and retrieval as two fundamental research areas in the FG image analysis field.

C. Scene-Level Sketch-Based Applications

Scene sketch have been studied and applied in scene image composition [38], scene image synthesis [6], scene image retrieval (not fine-grained) [3], and scene sketch semantic segmentation [5]. Compared to Sketch2Photo [38], which composites a photo-realistic scene image with a hand-drawn sketch and text as input and retrieves initial candidates of object instances for later scene image blending, our work aims to retrieve a specific image from an image gallery instead of compositing a synthesized image. Dey *et al.* [39] propose a cross-modal deep network to conduct multi-object image retrieval, which can use both sketches and text as inputs. Castrejon *et al.* [3] collect a cross-modal scene dataset and propose a cross-modal scene data representation learning framework for cross-modal retrieval tasks (including real images, clip art, sketches, spatial text and descriptions). Xie *et al.* [4] conduct a ZS-SBIR task on this cross-modal scene dataset by utilizing the overall visual features of scenes. Zou *et al.* [5] present a scene sketch dataset named SketchyScene with semantic and instance segmentation annotations, and conduct a pilot study of scene-level SBIR using an object-level SBIR method [1]. Although the goal of SBIR in SketchyScene is similar to the fine-grained scene-level SBIR in this paper, SketchyScene mainly proposes a baseline using an object-level SBIR method [1], and does not exploit the rich scene context for SBIR.

D. Image Retrieval With Graph Convolutional Networks

Graph convolutional networks (GCNs) [40] are effective neural network architectures to model and process graph data, and they have been used in many applications such as social recommendation [41], traffic prediction [42], action recognition [43], layout generation [44] and text matching [45] in the last few years. Jia *et al.* [46] develop CA-GCN for personalized image retrieval, which leverages user behavior data in a GCN model to learn user and image embeddings simultaneously. Chen *et al.* [47] propose a multi-label image classification model based on GCN and a re-weighted scheme to capture the label dependencies of co-occur objects in an image. In the realm of SBIR, Zhang *et al.* [48] utilize GCN

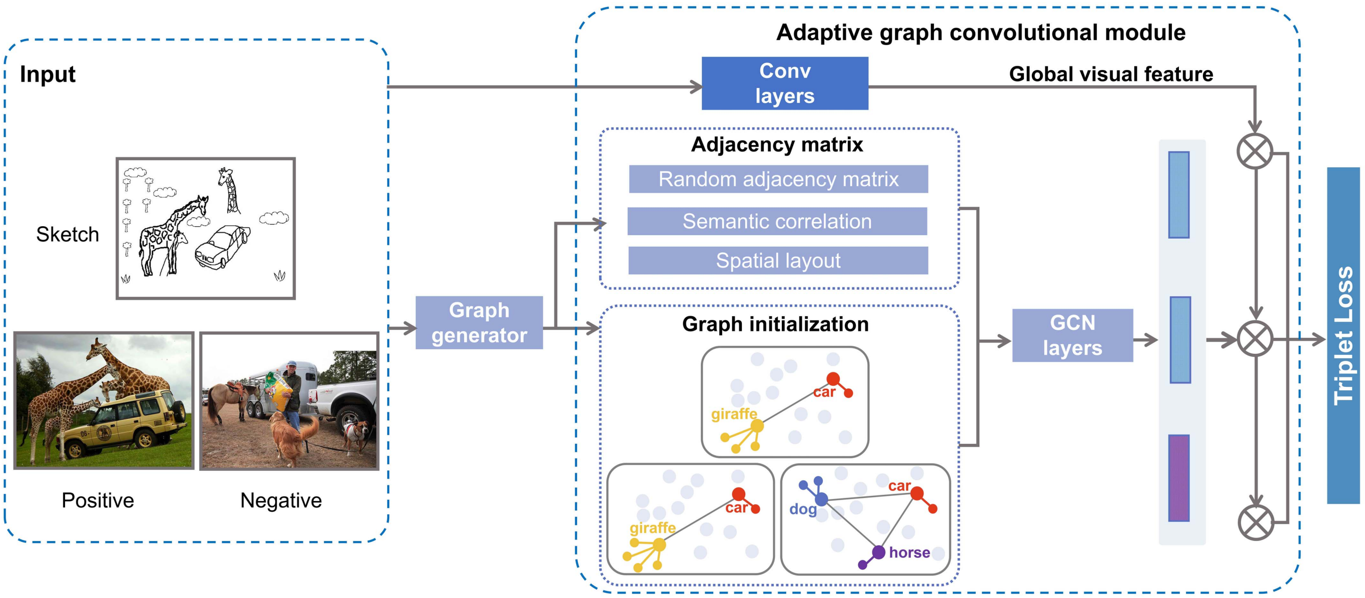

Fig. 2. Framework of the proposed SceneSketcher-v2. Our network mainly consists of a graph generator, an adaptive graph encoder module and a triplet loss for training. We first construct graphs for the input scene sketch and the images to be retrieved, and then utilize our adaptive GCNs for feature encoding of the graphs. Finally, we conduct retrieval using the extracted graph features via a triplet loss.

in zero-shot SBIR task, and propose SketchGCN model to use both visual and semantic information, which enhances the generalization ability of the retrieval model.

Compared to these approaches, our method leverages multi-modal features of object categories (i.e. global layout, visual features and semantic features) to construct graph nodes and learn adaptive graphs for encoders.

III. Methodology

A. Overview

The architecture of our proposed network is illustrated in Fig. 2. Our method mainly consists of a graph generation module, an adaptive graph convolution module and a triplet similarity module. The overall network extracts feature embeddings of scene sketches and images, and feeds them to a triplet ranking loss to enforce the distance in the feature space reflects how close scene sketches and images are in terms of global layout, appearance and semantic information. During the training process, each time the network takes a query scene sketch, a positive image and a negative image as input. In order to model the key scene context in fine-grained scene-level SBIR, we adopt adaptive graph convolutional networks (GCNs) as the graph encoders, which integrate the hierarchical information in scene sketches and images, including global scene visual features, global layout, semantic correlations between object categories, and projected features of visual and location features of each object category.

B. Graph Initialization

We employ a weighted, undirected scene graph to model the global layout, the semantic correlation and the visual appearance (size, pose and other fine-grained details) of the object instances in a scene sketch or a scene image explicitly. Our scene graph can be formulated as $G = (N, E)$, where $N = \{n_i\}$ is the node set and $E = \{e_{i,j}\}$ is the edge set, and $e_{i,j} = (n_i, n_j)$ is the edge connecting nodes n_i and n_j. The category set of the nodes in the graph is denoted as $C = \{c_i\}$, where c_i is the category label of node n_i.

1) Node Construction: In this paper, we model each object category c_i as a node n_i in the graph G. Fig. 3 illustrates an overview of the graph node initialization process. Given an object category c_i, we construct a corresponding node n_i by integrating the characteristics of all the instances $\{o_{ij}\}$ from the same object category c_i. There are two types of information in each node n_i, i.e., the visual features v_i and the spatial position p_i. Specifically, we resize the bounding boxes of the instances to a fixed size of 128 × 128 and adopt Inception-V3 [49] to extract a 2048-d visual feature v_{ij} for each instance. Then we concatenate the visual feature v_{ij} of instance o_{ij} with its spatial position p_{ij} (i.e., the coordinates of the upper left and bottom right corners of its bounding box). Finally, for each graph node n_i representing an object category, we get a 2052-d fused feature x_i by fusing the characteristics of instances $\{o_{ij}\}$ with the same category c_i via a perception layer (See Fig. 3). In the experiment, when the number of instances in a certain category is more than three, we choose the top three instances with the max sizes to construct the node for this category.

2) Edge Construction: The object nodes are connected with undirected weighted edges, and the edge weight between a pair of object nodes shows their spatial correlation. For each category node n_i, we define its position p_i by computing the coordinates of the center point of the bounding boxes of all instances in the category c_i. Each node position p_i is denoted as a 2-d vector and the coordinates are normalized to $(0, 1)$. Given two nodes n_i and n_j with positions p_i and p_j, we define the edge weight $A_{i,j} \in (0, 1)$ between them based

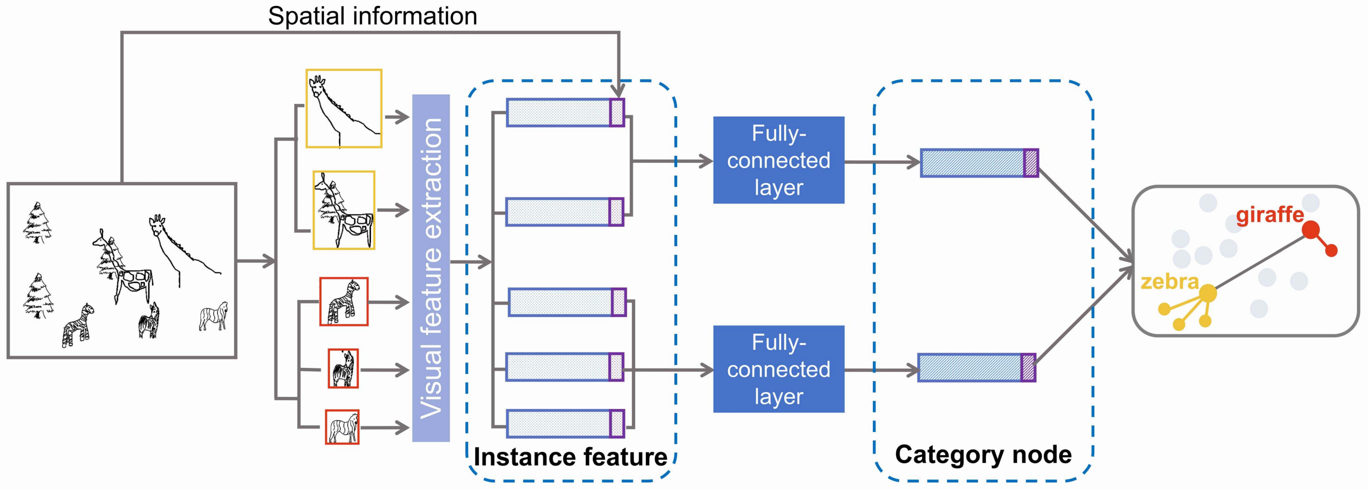

Fig. 3. Illustration of our graph node construction. We model each object category as a node in the graph. For each object instance from the same category, we concatenate the extracted visual feature and its spatial position. We then get a graph node with a 2052-d fused feature for each object category by integrating the features of all the object instances from the same category through a fully-connected layer.

on normalized Euclidean distance as follows:

$$A_{i,j} = 1 - D_{i,j} \quad (1)$$

where $D_{i,j} = ||p_j - p_i||_2$ is the Euclidean distance of the spatial position of node n_i and node n_j.

C. Graph Convolutional Networks

After generating scene graphs for sketches and images, we adopt GCNs to learn node-level representations for our scene graph where we update the node features by propagating information between nodes. The l-th layer of a GCN takes a feature matrix $\mathbf{H}^{l-1}$ and the corresponding adjacency matrix $\mathbf{A} = \{A_{ij}\}$ as inputs and learns a function $f(\cdot,\cdot)$ to extract features on a graph $G = (N, E)$. The l-th layer of the GCN can be formulated as

$$\mathbf{H}^{(0)} = \{x_i\}_{i=1}^{n} \quad (2)$$

$$\mathbf{H}^{(l)} = f(\mathbf{H}^{(l-1)}, \mathbf{A}), \quad l > 1 \quad (3)$$

Then we adopt the propagation rule introduced in [40], and the feature extraction function $f(\cdot,\cdot)$ can be written as

$$f(\mathbf{H}^{(l)}, \mathbf{A}) = \sigma(\hat{\mathbf{D}}^{-\frac{1}{2}}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}) \quad (4)$$

where $\sigma(\cdot)$ is the leaky ReLU activation function, $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, and $\hat{\mathbf{D}}$ is the diagonal node degree matrix of $\hat{\mathbf{A}}$, and $\mathbf{W}^{(l)}$ is a weight matrix to be learned.

We denote the outputs of the last layer of graph convolution networks for sketches and images to be two scene graph embeddings G_S and G_I, respectively.

D. Adaptive Graph Convolutional Module

Our previous SceneSketcher [7] use a fixed single graph in the graph convolutional network for both training and testing stages, which is not ideal to model the semantic relationships and the dependency of object categories. Inspired by the spatial attention module in 2SA-GCN [50] and the temporal attention module in STA-GCN [51] that were designed for action recognition, we adopt an adaptive graph convolution module with a powerful attention mechanism for our fine-grained scene-level SBIR task. In order to integrate different aspects of graph structures, we use the sum of three adjacency matrices as the adjacency matrix in Eq. (4), which represents three different graph structures, i.e., a fixed adjacency matrix $\mathbf{A}_1$ that denotes the category-level characteristics and spatial layout of the scene sketch, a semantic adjacency matrix $\mathbf{A}_2$ modeling the correlations and dependencies between different categories, and a learnable adjacency matrix $\mathbf{A}_3$ denoting the unique pattern of each sketch-photo dataset. The construction process of the adjacency matrices is as follows:

1) As for $\mathbf{A}_1$, we use the original spatial graph adjacency matrix in Eq. (1);
2) In order to effectively capture the correlations between object categories, we use a semantic graph $\mathbf{A}_2$ to model the semantic correlation of the category labels. Specifically, the category label c_i of each node is encoded as a 300-d vector $\tilde{c}_i$ by Word2Vec [52], and then we use the cosine distance between them to model the correlation of the two nodes;
3) The third matrix $\mathbf{A}_3$ is a trainable matrix. Compared to $\mathbf{A}_1$ and $\mathbf{A}_2$ which are both fixed after initialization, the adjacency matrix $\mathbf{A}_3$ can be learned during the training process. In this data-driven way, the model can learn a specific graph that can help to achieve better performance of fine-grained scene-level task on a particular dataset.

We show the overall architecture of our adaptive graph convolutional layer in Fig. 4. Given the input scene sketch, we first get a graph feature map f_{in} of $N \times (f_v + 4)$-d via our node construction module (see Sec. III B), where f_v denotes the size of the extracted feature of the visual feature extraction network. And we also extract a $1 \times N$ global visual feature of the whole scene sketch using Inception-V3. Then we use graph convolutional layers to embed f_{in} with the sum of the

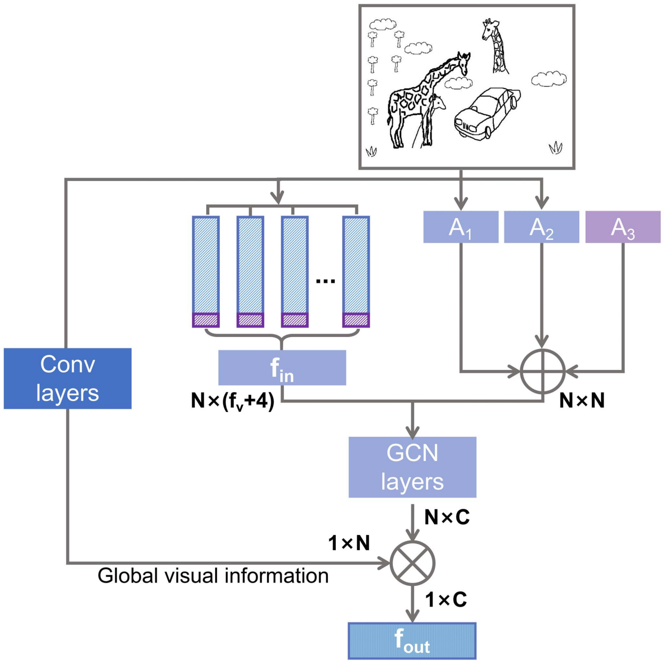

Fig. 4. Illustration of the adaptive graph convolution module. Given the input scene sketch, we can embed it into a $1 \times C$ feature f_{out} with our adaptive graph convolution module by integrating the global visual feature, the fused feature f_{in} of each category and the graph adjacency matrices $\{\mathbf{A}_i\}_{i=1}^{3}$ of categories. N is the number of nodes in the graph, C is the output dimension of the GCN layers, and f_v is the size of instances' visual feature extracted by the convolutional layers.

proposed three adjacent matrices into an $N \times C$ graph feature. Finally, the $N \times C$ graph feature and the $1 \times N$ global visual feature are multiplied into a scene-level feature vector of the size $1 \times C$.

Compared with the graph network in SceneSketcher [7] in which the different scene graphs output graph embeddings with variable sizes, our proposed adaptive graph convolutional module produces fixed-size feature vectors which can be directly fed into the later triplet ranking network. Our adaptive graph convolutional module enables an end-to-end training process for both the GCN layers and the visual feature extraction networks. As a comparison, the graph embeddings in SceneSketcher [7] need to be further compared with a non-differentiable graph matching strategy and the parameters of its visual feature network are fixed after initialization, thus the extracted features may not be optimal for feature embedding.

E. Loss Function

We use triplet loss to update our fine-grained scene-level SBIR framework. The input of our SceneSketcher-v2 is a triplet (S, I^+, I^-), where S is a scene sketch, I^+ is the corresponding image of S, and I^- is an image of a different scene. We describe the construction of our loss function as follows.

As evidenced by pioneering SBIR networks [1], [27], the triplet ranking loss is able to express fine-grained appearance details and relationships of sketches better than Siamese loss [2]. The goal of the triplet loss is to enforce the embedding features of two examples with the same label to be close to each other and the embedding features of two examples with different labels far away. The triplet loss L_{tri} of a given triple (S, I^+, I^-) can be computed by

$$L_{tri} = \max(d(S, I^+) - d(S, I^-) + m, 0) \tag{5}$$

where $d(\cdot, \cdot)$ is the distance function in the embedding space, and m is a margin between the anchor-positive distance and the anchor-negative distance, which is set to 0.4 in our experiments.

With three scene graph embeddings G_S, G_{I^+} and G_{I^-} of the triple (S, I^+, I^-), we define $d(S, I^+)$ and $d(S, I^-)$ of Eq. (5) by computing the Euclidean distance between them.

IV. Experiments

A. Datasets

Although several sketch datasets [1]–[3], [5], [53], [54] have been constructed for SBIR or other sketch-oriented applications (see Fig. 5), none of them fit our problem. They either just contain a single object instance in one photo, or no fine-grained annotations of objects are available. We show several examples of the existing sketch databases in Fig. 5.

1) Object-Level Sketch Datasets: TU-Berlin [53] and QuickDraw [54] only contain sketches without photos, thus they cannot be used in SBIR task. The Sketchy Database [2] was originally used for object-level SBIR, where there is a single object instance in each sketch or image. TU-Berlin Extended contains photos of the same classes of the TU-Berlin dataset, which is a main benchmark dataset for coarse-grained sketch-based image retrieval; QuickDraw-Extended proposed in [55] contains photos of the same classes of the QuickDraw sketches; Sketchy-Extended was implemented by Shen *et al.* [22] and Yelamarthi *et al.* [24], and these three extended datasets are commonly used in zero-shot SBIR. QMUL Shoe&Chair dataset [1] is the first dataset introduced for fine-grained SBIR task, containing a few hundred sketch-photo pairs. Although QMUL Shoe&Chair dataset [1] facilitates the fine-grained sketch-related applications, all the sketches and images in this dataset have single instances and clean backgrounds, thus it cannot be used in our scene-level SBIR task. Moreover, there are only two object classes (shoes and chairs) in QMUL Shoe&Chair dataset, which is insufficient for large-scale SBIR.

2) Scene-Level Sketch Datasets: SketchyScene [5], CMPlaces [3] and SketchyCOCO [6] are the three available scene-level sketch datasets. SketchyScene was originally used for the scene sketch segmentation task and is not suitable to be directly used to train and evaluate our fine-grained scene-level SBIR network. Though SketchyScene contains large amount of sketch-image pairs, the images of SketchyScene are all cartoon clips, while the focus of our work is to retrieve natural photos. Moreover, SketchyScene does not contain the bounding box or object instance segmentation annotation in images, thus it cannot offer the visual feature and spatial information of object instances for our SceneSketcher-v2 framework. CMPlaces was originally used for category-level

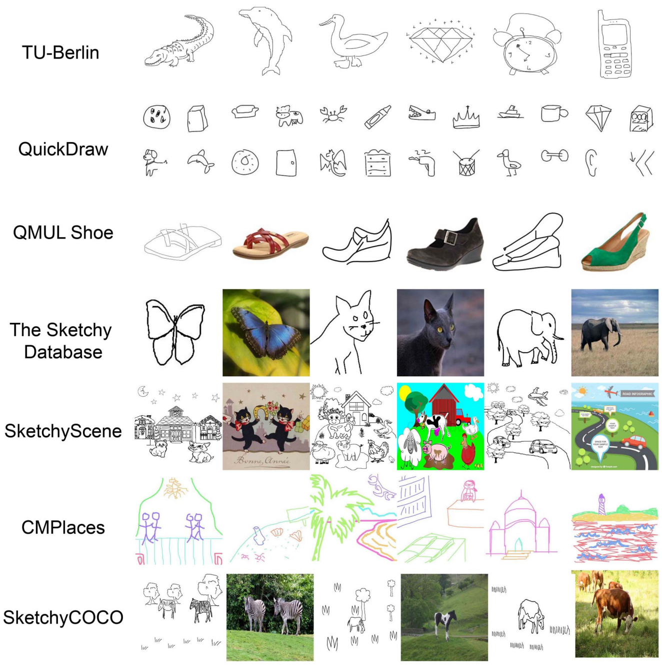

Fig. 5. Examples of the existing sketch databases.

cross-modal retrieval. It only contains scene-level category labels, thus cannot be used for our fine-grained retrieval task either. On the one hand, it does not contain paired image and sketch data. On the other hand, it does not contain object instance segmentation annotations as in the SketchyScene database. SketchyCOCO is a fine-grained scene-level sketch dataset containing sketch-image pairs, and it is designed for sketch-based image synthesis. Most of the images in SketchyCOCO only contain single foreground object, and the correspondences between objects in sketches and images are usually inaccurate, therefore, SketchyCOCO is not an ideal dataset for fine-grained scene level SBIR.

In our experiment, we modified existing sketch databases SketchyCOCO [6] and SketchyScene [5] for evaluations.

3) SketchyCOCO-SL: We collect a scene sketch-image dataset (named SketchyCOCO-SL, where we use “SL” to emphasize “scene-level”) by modifying SketchyCOCO [6], and utilize the scene sketch dataset for our fine-grained scene-level SBIR task. SketchyCOCO is constructed for sketch-based image generation task, containing over 14,000 scene-level sketch-photo pairwise examples, but most of them only contain one foreground instance. We use sketch-photo pairs containing more than one object instance from SketchyCOCO, that is 1,225 scene sketch-photo pairs in total, covering 14 object categories (bicycle, car, motorcycle, airplane, traffic light, fire hydrant, cat, dog, horse, sheep, cow, elephant, zebra, giraffe). We split SketchyCOCO-SL into training and testing sets, each containing 1,015 and 210 scene sketch-image pairs. The first two columns of the last row of Fig. 5 shows two examples of SketchyCOCO-SL dataset. We display two samples with multiple object instances, and our fine-grained scene-level SBIR models are needed to differentiate a specific scene.

4) SketchyCOCO-SL Extended: We extend SketchyCOCO-SL with natural images from COCO-stuff [56] to form a larger image gallery, named SketchyCOCO-SL Extended, to further investigate the performance of our method. We select 5,000 natural images, the objects of which are within the 14 categories in SketchyCOCO-SL. These natural images do not have corresponding sketches in the SketchyCOCO-SL dataset and are not used in our training process. Then we combine them with the images of the testing dataset in the SketchyCOCO-SL dataset. In total, there are 210 sketches and 5,210 images in the testing set.

5) SketchyScene: SketchyScene [5] is a scene-level sketch dataset designed for segmentation tasks, and a pilot study of scene-level SBIR also has been conducted on it. In this work, we use the same 2,472 and 252 pairs of sketch-photo data as SketchyScene [5] for evaluation.

B. Implementation Details

We adopt the Yolo-V4 object detector [57] to obtain the instances' bounding boxes in sketches and images. In order to extract object instances in scene sketches, we use the training set of SketchyCOCO-SL to train a Yolo-V4 object detector, and we use the trained Yolo-V4 model to obtain the instances' bounding boxes in scene sketches during testing. Similarly, for the SketchyScene dataset, we select 150 images from the training set of SketchyScene, manually label the bounding boxes of the object instances, and then train a Yolo-V4 network to detect the object instances in the images for retrieval.

As mentioned in the *Node Construction* part in Sec. III-B, if the number of instances in a certain category is more than three, we choose the top three instances with the max sizes to construct the node for this category. We have conducted experiments to analyze the SBIR performance with respect to the maximum number of instances to be retained for each category in our model. We set the maximum number of instances in a certain category as three, four, and five, respectively, and we have observed that the network obtains the best retrieval performance when setting the maximum number of instances in a certain category as three. And the performance is almost reaching a steady state over all the three models. This may be because over 94% of the scene sketches in the SketchyCOCO-SL database contain no more than 3 instances of the same category. Furthermore, the max number of instances in a certain category can be regarded as a hyper-parameter used in the graph node construction process, and it can be modified according to the sketches and images of different databases.

C. Evaluation Metrics

We adopt a standard and the most commonly used evaluation metric for retrieval as [1], recall at rank K (Recall@K), which is computed as the percentage of test queries where the target image is ranked within the top K retrieved images.

D. Comparison With Baselines

We show several fine-grained SBIR examples with our method on the SketchyCOCO-SL Extended dataset in Fig. 6

TABLE I

COMPARISON OF SCENE-LEVEL SBIR PERFORMANCE WITH EXISTING SBIR METHODS ON THE SKETCHYCOCO-SL DATABASE (210 TESTING IMAGES), SKETCHYCOCO-SL EXTENDED DATABASE (5,210 TESTING IMAGES), AND SKETCHYSCENE DATABASE (252 TESTING IMAGES). THE MISSING VALUES INDICATE THE RECALLS OF THE SBIR METHODS ARE CLOSELY TO ZERO

Dataset	Method	Recall@1	Recall@5	Recall@10
SketchyCOCO-SL	HOG+BoW+RankSVM [11]	0.48	1.43	4.76
	Dense HOG+RankSVM [58]	0.48	3.81	5.71
	Sketch-a-Net+RankSVM [59]	0.48	3.33	4.76
	Sketch me that shoe [1]	6.19	17.15	32.86
	DSSA [29]	0.48	3.81	7.62
	SketchyScene [5]	1.43	4.76	8.57
	SceneSketcher [7]	31.91	66.67	86.19
	SceneSketcher-v2	**68.10**	**87.62**	**95.24**
Dataset	Method	Recall@10	Recall@50	Recall@100
SketchyCOCO-SL Extended	HOG+BoW+RankSVM [11]	0.48	0.48	0.48
	Dense HOG+RankSVM [58]	-	0.95	1.91
	Sketch-a-Net+RankSVM [59]	-	0.95	2.86
	Sketch me that shoe [1]	1.90	6.19	8.57
	DSSA [29]	-	0.95	1.90
	SketchyScene [5]	0.48	0.95	2.86
	SceneSketcher [7]	38.10	68.10	82.86
	SceneSketcher-v2	Recall@1	Recall@5	Recall@10
		54.76	**68.10**	**71.90**
Dataset	Method	Recall@1	Recall@5	Recall@10
SketchyScene	HOG+BoW+RankSVM [11]	-	3.20	5.60
	Dense HOG+RankSVM [58]	0.80	2.80	4.80
	Sketch-a-Net+RankSVM [59]	0.80	2.00	3.20
	Sketch me that shoe [1]	13.20	29.20	38.00
	DSSA [29]	13.60	28.00	39.60
	SketchyScene [5]	13.60	24.40	32.40
	SceneSketcher [7]	10.00	23.20	41.20
	SceneSketcher-v2	**23.60**	**45.60**	**58.00**

and on SketchyScene dataset in Fig. 7. For each query sketch, there are typically a handful of visually very similar photos; since in this paper our goal is to conduct fine-grained scene-level SBIR, the lower-rank accuracy is a better indication on how well the model is capable of distinguishing fine-grained subtle differences between candidate photos. Some sketches do not match the photos exactly in the SketchyCOCO and SketchyScene dataset, thus there are cases that no images in the database can fully match the input sketch.

In the following, we also compare our model with several state-of-the-art (SOTA) using hand-crafted features and deep learning based features. (1) Baselines using hand-crafted features include *HOG-BoW+RankSVM* [11] and *Dense HOG+RankSVM* [58]. We first compare our method with *HOG-BoW+RankSVM*. HOG-BoW descriptor is a widely-used visual feature for SBIR [11], [60]. We first extract HOG features from each image, and feed them to the BoW (Bag-of-Words) framework for feature encoding. Then we feed the features to train a RankSVM model to rank the results as [61]. During the comparison, we use the same triplets for training as those in the experiment of our method. We also compare our method with Dense HOG features-based method. We follow [1] to extract Dense HOG features, in which dense HOG features are extracted by concatenating HOG features over a dense grid [58]. (2) Baselines using deep learning based features include *Sketch-a-Net+RankSVM* [59], *Sketch me that shoe* [1], *DSSA* [29], and *SketchyScene* [5]. In Sketch-a-Net+RankSVM [59], we extract deep features using the Sketch-a-Net model and feed them to RankSVM to train a SBIR model. In order to compare Sketch me that shoe [1], we adopt a deep triplet ranking model for instance-level fine-grained SBIR, where free-hand sketches are used as queries for instance-level retrieval of images. Due to the lack of fine-grained scene-level SBIR models, we can only compare with the existing object-level SBIR methods mentioned above. To the best of our knowledge, SceneSketcher-v1 published in ECCV 2020 [7] is the only fine-grained scene-level SBIR model. Besides, Zou *et al.* [5] present a closely related work, where they construct a scene sketch dataset and conduct a preliminary study of scene-level SBIR based on a similar triplet ranking network proposed in [1]. We also compare our method with the SBIR method used in SketchyScene.

Table I shows the comparison of the retrieval recalls with our model and the compared methods. The results indicate that: (1) Our model achieves significantly higher recall than the other baselines on all three datasets, which demonstrates that our method is effective. (2) Baselines in Table I use hand-crafted image descriptors ([11], [58]) or deep features ([1], [5], [29], [59]) to conduct shape matching between sketches and edge maps of natural images. They are all designed for single object retrieval, thus produced poor results on multi-objects dataset. Deep learning based models are in general stronger compared with traditional hand-crafted features designed for SBIR. (3) Deep learning based models

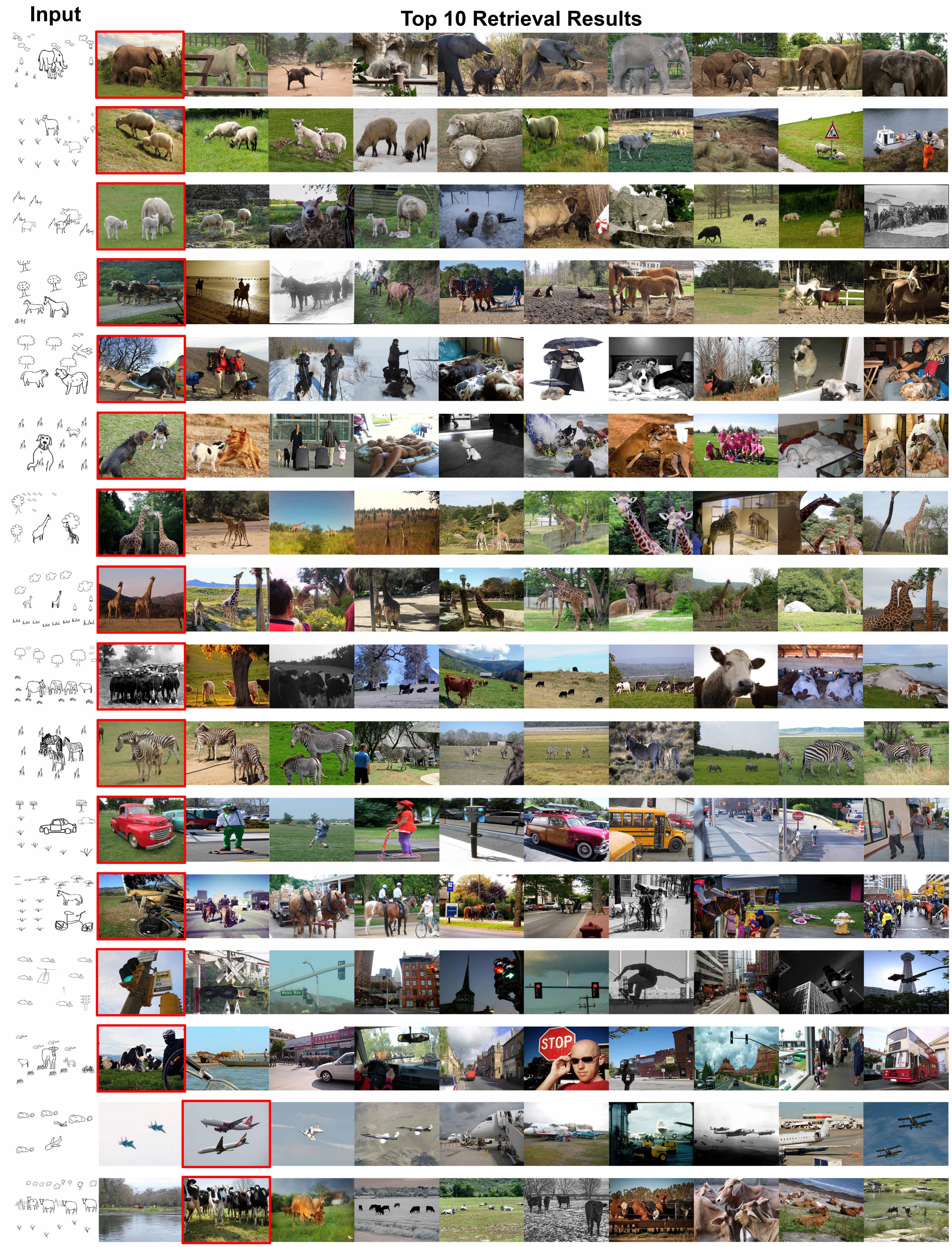

Fig. 6. Top-10 fine-grained scene-level SBIR results with our method. The true matches are highlighted with red borders.

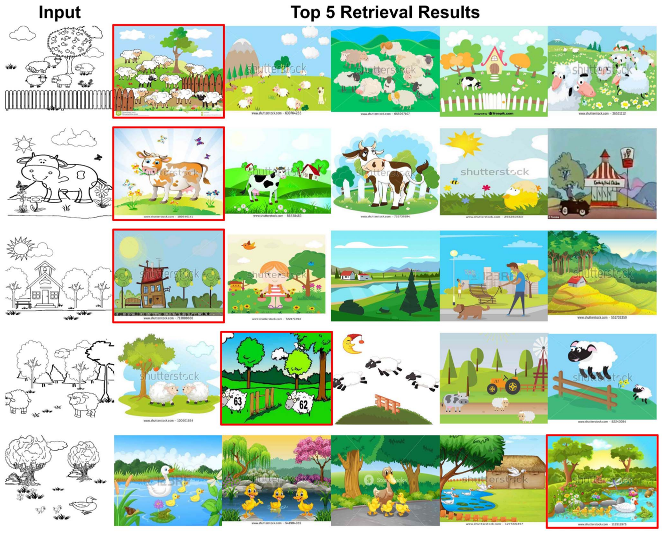

Fig. 7. Top-5 fine-grained scene-level SBIR results with our method on SketchyScene dataset. The true matches are highlighted with red borders.

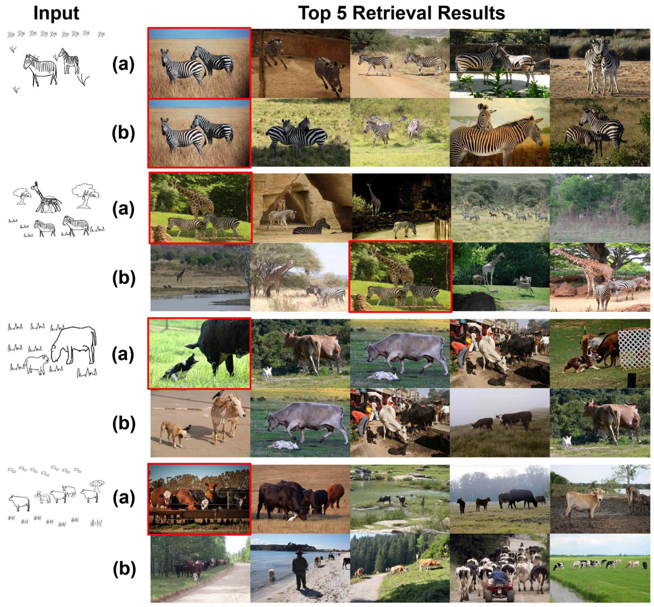

Fig. 8. Comparison of scene-level SBIR results with our method and SceneSketcher [7]. We show four panels. In each panel, (a) shows the results of SceneSketcher-v2, (b) shows the results of SceneSketcher. The ground truth matches are highlighted with red borders.

designed for SBIR with a single object (e.g. Sketch me that shoe [1], DSSA [29], and SketchyScene [5]) get poor performance on SketchyCOCO-SL dataset and SketchyCOCO-SL Extended dataset. However, compared to traditional methods, the recalls of these deep learning based models improve significantly on SketchyScene dataset, which may be because these three methods conduct SBIR between input sketches and the edge maps extracted from images, and the photos in SketchyScene are all cartoon images which makes edge map extraction more easier. Sketch me that shoe [1] is a more related SOTA SBIR model, which is also the first work on fine-grained SBIR task. (4) We also compare our method with SceneSketcher [7]. The Recall@1 with our method is about 37%, 16% and 13% higher than those with SceneSketcher on SketchyCOCO-SL, SketchyCOCO-SL Extended and SketchyScene datasets, respectively. We show the visual comparison of scene-level SBIR results with our method and SceneSketcher in Fig. 8.

E. Ablation Study of SceneSketcher-v2

Our fine-grained scene-level SBIR method adopts an adaptive GCN to fuse hierarchical information of scene sketches and images, including category nodes fusing instance-level characteristics with the same category label, graph embedding representing category-level overall information and correlation, and image attention representing global layout feature of scene. In order to demonstrate the contribution of each component, we compare our full model with the following three models:

1) *Graph only (average fusion).* To investigate the effect of the fused feature of the instances of the same object category in the node construction (See Section III-B), we use the average feature of the instances of the same object category to construct the category node for comparison. We remove the fully-connected layer in Fig. 3 and get the category nodes by directly averaging the features of the instances of the same category. Moreover, the global scene visual feature is not included. The rest parts of this model are the same as our full model.
2) *Graph + global (average fusion).* This model is similar to *Graph only (average fusion)*, but the global scene visual feature is used as our full model.
3) *Graph only (learned fusion).* To investigate the effect of the global scene feature in the adaptive graph convolution module (See Fig. 4), we remove the global scene visual feature from our full model, and use the same graph encoder as our full model. The node construction procedure is the same as we described in Section III-B. We fuse the features of instances with the same object category in a learned way via a fully-connected layer to construct the category node in our scene graph.

Table II shows the performances of our full model and the three models above on the fine-grained scene-level SBIR. We can observe that: **(1) Our global scene visual feature and the adaptive GCN module contribute greatly to the final performance of our SceneSketcher-v2.** The only difference between *Graph + global (average fusion)* (*Full model*) and *Graph only (average fusion)* (*Graph only (learned fusion)*) is that the former also employs additional global scene visual feature. *Graph + global (average fusion)* (*Full model*) outperforms *Graph only (average fusion)* (*Graph only (learned fusion)*) on all the three datasets. Besides, *Graph only (learned fusion)* models can also obtain relatively good performance, showing the effectiveness in applying GCN to the fine-grained scene-level SBIR task. **(2) The way to fuse instance features and construct category nodes has a great impact on the retrieval performance of the model** (See *Graph only (average fusion)* vs. *Graph only (learned fusion)*, and *Graph + global (average fusion)* vs. *Full model*). Instead

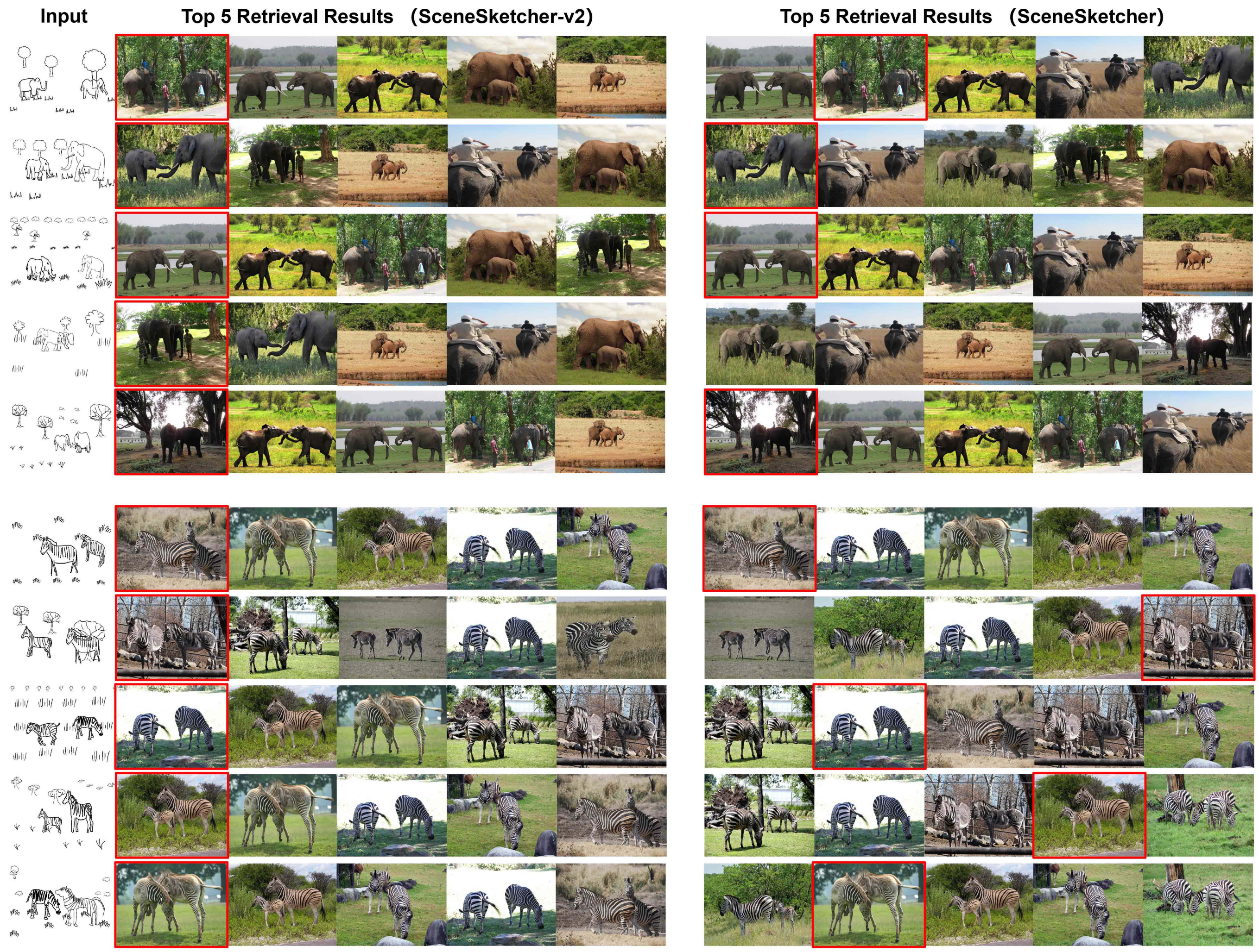
Input
Top 5 Retrieval Results (SceneSketcher-v2)
Top 5 Retrieval Results (SceneSketcher)

Fig. 9. Comparison of Top-5 fine-grained scene-level SBIR results with SceneSketcher-v2 and SceneSketcher. We use two sets of similar image collections of elephants (row 1 to row 5) and zebras (row 6 to row 10), respectively. The true matches are highlighted with red borders.

of fusing instance features using manually defined feature fusion rules (e.g., averaging instance features with the same category label to get the category node feature), we construct our category nodes by fusing the visual features and positions of the instances via a learned way (the fully-connected layer in Fig. 3), thus get better features of each object category and facilitate SBIR.

F. Comparisons Between Adjacency Matrices

We also compare the effect of different adjacency matrices in the adaptive graph layer (See Table III). From the SBIR results, we can see that: (1) The three components of the adjacency matrices all contribute to the excellent retrieval performance of our final model. (2) Although the adjacency matrix $\mathbf{A}_1$ which is a fixed graph structure denotes the spatial layout of the scene sketch performs poor when working solely, it can enhance the SBIR performance when $\mathbf{A}_1$ works together with $\mathbf{A}_2$ and $\mathbf{A}_3$. Conceptually, the spatial layout between instances alone is not enough to determine the similarity between scenes, but it can be used as an inherent constraint for SBIR. (3) The model achieves good results when working alone with $\mathbf{A}_2$ or $\mathbf{A}_3$. Compared to the performance with $\mathbf{A}_3$, the retrieval model performs better with $\mathbf{A}_2$. Since $\mathbf{A}_2$ provides category semantic information, we can conclude that semantic information is a key clue in the scene-level SBIR task.

TABLE II

COMPARISONS OF DIFFERENT COMPONENTS

SketchyCOCO-SL	Recall@1	Recall@5	Recall@10
Graph only (average fusion)	48.09	66.67	74.76
Graph + global (average fusion)	62.38	77.62	85.71
Graph only (learned fusion)	60.95	78.57	84.76
Full model	**68.10**	**87.62**	**95.24**
SketchyCOCO-SL Extend	Recall@1	Recall@5	Recall@10
Graph only (average fusion)	43.33	56.19	58.57
Graph + global (average fusion)	48.57	60.00	66.19
Graph only (learned fusion)	55.71	60.95	66.19
Full model	54.76	**68.10**	**71.90**
SketchyScene	Recall@1	Recall@5	Recall@10
Graph only (average fusion)	18.80	37.60	44.80
Graph + global (average fusion)	19.20	40.00	48.80
Graph only (learned fusion)	20.00	40.40	49.60
Full model	**23.60**	**45.60**	**58.00**

TABLE III

COMPARISONS USING DIFFERENT ADJACENCY MATRIX IN THE ADAPTIVE GRAPH LAYER. THESE THREE ADJACENCY MATRICES, $\mathbf{A}_1$, $\mathbf{A}_2$, AND $\mathbf{A}_3$, ARE DESIGNED TO MODEL SPATIAL, SEMANTIC, AND ADAPTIVE INFORMATION, RESPECTIVELY

SketchyCOCO-SL	Recall@1	Recall@5	Recall@10
$\mathbf{A}_1$	13.33	34.76	51.90
$\mathbf{A}_2$	57.14	67.14	74.29
$\mathbf{A}_3$	53.80	61.43	70.00
$\mathbf{A}_1 + \mathbf{A}_2$	67.62	82.86	88.57
$\mathbf{A}_1 + \mathbf{A}_3$	67.14	82.86	91.90
$\mathbf{A}_2 + \mathbf{A}_3$	60.48	76.67	86.67
Full model	**68.10**	**87.62**	**95.24**
SketchyCOCO-SL Extend	Recall@1	Recall@5	Recall@10
$\mathbf{A}_1$	7.62	13.33	16.67
$\mathbf{A}_2$	53.33	56.19	57.14
$\mathbf{A}_3$	45.71	58.57	61.42
$\mathbf{A}_1 + \mathbf{A}_2$	44.76	67.62	70.00
$\mathbf{A}_1 + \mathbf{A}_3$	48.09	63.33	68.10
$\mathbf{A}_2 + \mathbf{A}_3$	45.71	62.86	66.19
Full model	**54.76**	**68.10**	**71.90**
SketchyScene	Recall@1	Recall@5	Recall@10
$\mathbf{A}_1$	19.60	35.60	45.20
$\mathbf{A}_2$	20.00	37.60	44.40
$\mathbf{A}_3$	20.80	36.00	44.00
$\mathbf{A}_1 + \mathbf{A}_2$	20.80	39.60	47.60
$\mathbf{A}_1 + \mathbf{A}_3$	21.20	41.80	51.20
$\mathbf{A}_2 + \mathbf{A}_3$	22.00	42.40	55.20
Full model	**23.60**	**45.60**	**58.00**

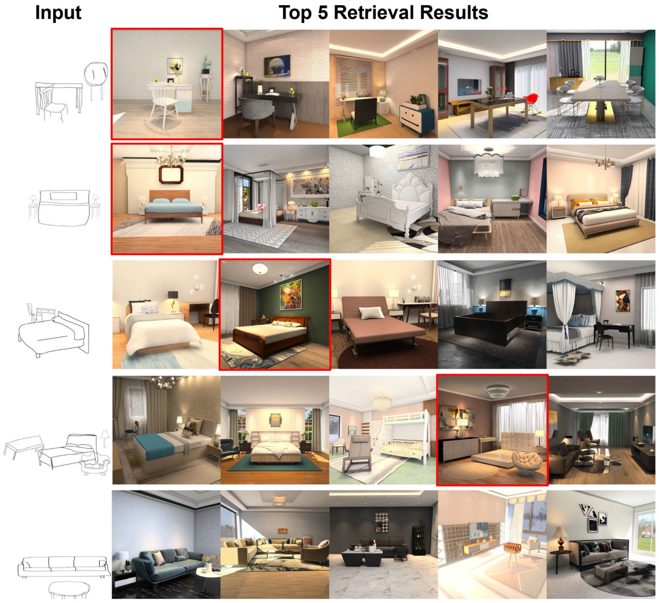

Fig. 10. Fine-grained scene-level SBIR application: furnished room retrieval with scene sketches. We show top-5 fine-grained scene-level SBIR results with our SceneSketcher-v2. The true matches are highlighted with red borders.

G. Fine-Grained Retrieval

To analyze the performance of our fine-grained scene-level SBIR, we pick some images that are extremely similar in overall layout of sketches, category of objects, and their position and shape. We pick up 10 extremely similar images of elephants (or zebras) from our SketchyCOCO-SL Extended dataset. Corresponding scene sketches are used to conduct the SBIR task. We aim at exploring the sorting results of the 10 images with different sketches as inputs. We also use our previous SceneSketcher [7] to conduct image retrieval on these elephant and zebra images. Results are shown in Fig. 9. Although both SceneSketcher and SceneSketcher-v2 can retrieve the desired fine-grained images effectively, our SceneSketcher-v2 is able to capture more details of object positions and relationships.

H. Application

As an example, we demonstrate that our method can enable the application of sketch-based interior scene retrieval. In the following, we conduct a pilot study of this application.

To the best of our knowledge, there is no large-scale indoor scene sketch-image paired data publicly available. In order to train and test our model, we selected indoor scene images of furnished rooms from a large-scale indoor scene dataset, 3D-FRONT (3D Furnished Rooms with layOuts and semaNTics) [62] and made their corresponding scene sketches. Since the appearance of furniture in each indoor scene image varies greatly and no pattern can be followed to automatically generate a large number of sketches, we construct the dataset manually. In this pilot study, we first selected 110 indoor scene images of furnished rooms from 3D-FRONT. Then we generate the sketches of the selected scene images via composition of single instance sketches as [5]: (1) we firstly select sketches of object categories contained in those scene images from TU-Berlin [53]; The scene objects in 3D-FRONT are clustered into 7 major categories (i.e. cabinet, bed, chair, table, sofa, stool and lighting), and the object categories of sketches selected from TU-Berlin include 5 classes (i.e. bed, chair, table, couch and tablelamp). Since the categories "sofa" and "lighting" do not exist in TU-Berlin, we select the sketches of similar object categories from "couch" and "tablelamp" instead. (2) we choose the appropriate object sketch for each object instance, and construct scene sketches by placing them in proper places using dragging, rotation, and scaling operations under the guidance of the reference image.

We get 110 pairs of "sketch-image" indoor scene data. We use 95 pairs of "sketch-image" data to train our retrieval model, and the rest 15 pairs of sketches to find the scene images from the 110 images. We obtain 13.33% on recall@1, 46.67% on recall@5, and 60.00% on recall@10 on the 110 testing images. In addition, we constructed another image gallery for evaluation by selecting additional 500 furnished room scene images from the 3D-FRONT dataset. We also use the 15 sketches to retrieve scene images from the indoor image gallery with 515 images (combining 15 relevant images with the additional 500 images). The recall@1, recall@5, and recall@10 retrieval results are 13.33%, 26.67% and 33.33%, respectively. Fig. 10 shows several retrieval examples. Our SceneSketcher-v2 is able to find out rooms which are decorated with similar furniture in similar positions. Fine-grained scene-level SBIR techniques provide potential solutions to indoor scene retrieval and style selection.

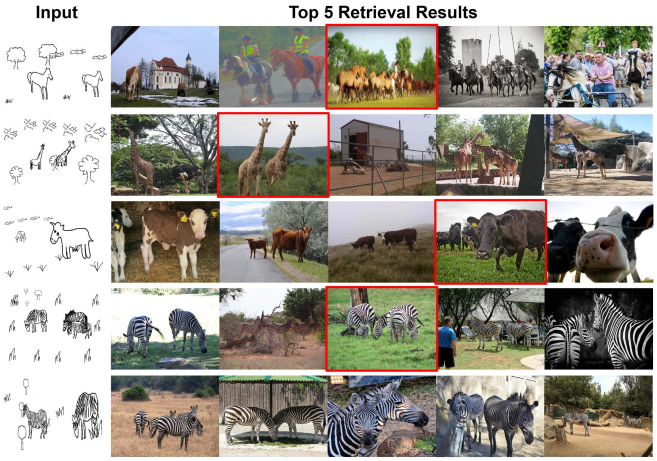

Fig. 11. Failure cases on SketchyCOCO-SL Extended. We show Top-5 fine-grained scene-level SBIR results with our method. The true matches are highlighted with red borders.

I. Failure Cases and Limitations

Although our SceneSketcher-v2 has achieved promising results, it still has several limitations. Fig. 11 shows several failure retrieval examples on our SketchyCOCO-SL Extended dataset. Some failure cases come from the inaccurate or wrong data annotations (see the first row of Fig. 11). Because our model does not enforce object instance orientation constraints, our retrieval model may find false fine-grained images with different object orientations (see top-1 results in the second and third rows of Fig. 11, which have the correct categories and similar appearances, but wrong orientations). In addition, when the input scene sketch contains complex occlusions, the retrieval performance may drop a little (see the last two rows of Fig. 11).

V. Conclusion and Future Works

In this work, we propose a new network called SceneSketcher-v2 for fine-grained scene-level sketch-based image retrieval. SceneSketcher-v2 incorporates an adaptive graph-based framework, together with a global image attention, to model the layout and fine-grained details of sketch scenes at the same time in an explicit way. Its end-to-end training manner enables the updating of visual feature learning network together with the graph convolutional network via a triplet loss, which greatly boosts the final SBIR performance. We show our method is superior to SceneSketcher as well as other existing sketch-based image retrieval methods on several popular datasets.

Although promising results have been obtained in this work, our SBIR framework can be further improved in three aspects: (1) The instances in the input scene sketches are treated equally, however, in real SBIR applications, users may want to give different retrieval priorities to the instances, e.g., draw sketch objects in a variable levels of sketch abstraction and detail to express their different attention to the instances' similarities between input sketch and retrieved image. Our method could be incorporated with users' extra interactive information to achieve image re-ranking or incremental image retrieval. (2) Our method only uses the position, size, and geometrical visual information in the scene sketch for image retrieval. In some scenarios, extra user input from other modalities, such as the color information, may allow the user to search the target image more efficiently. In the future, we may also consider providing a flexible and hybrid query interface which integrates sketch as well as other modality input for fine-grained scene-level sketch-based image retrieval. (3) Our SceneSketcher-v2 cannot be directly used for zero-shot retrieval, and we plan to extend our model to obtain scene graphs in a dynamic way, and conduct scene-level fine-grained SBIR on unseen instances in the future work.

References

[1] Q. Yu, F. Liu, Y.-Z. Song, T. Xiang, T. M. Hospedales, and C. C. Loy, "Sketch me that shoe," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 799–807.

[2] P. Sangkloy, N. Burnell, C. Ham, and J. Hays, "The sketchy database: Learning to retrieve badly drawn bunnies," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 1–12, Jul. 2016.

[3] L. Castrejon, Y. Aytar, C. Vondrick, H. Pirsiavash, and A. Torralba, "Learning aligned cross-modal representations from weakly aligned data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2940–2949.

[4] Y. Xie, P. Xu, and Z. Ma, "Deep zero-shot learning for scene sketch," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 3661–3665.

[5] C. Zou *et al.*, "SketchyScene: Richly-annotated scene sketches," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 421–436.

[6] C. Gao, Q. Liu, Q. Xu, L. Wang, J. Liu, and C. Zou, "Sketchy-COCO: Image generation from freehand scene sketches," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5174–5183.

[7] F. Liu *et al.*, "SceneSketcher: Fine-grained image retrieval with scene sketches," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 718–734.

[8] Y. Cao, C. Wang, L. Zhang, and L. Zhang, "Edgel index for large-scale sketch-based image search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 761–768.

[9] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "An evaluation of descriptors for large-scale image retrieval from sketched feature lines," *Comput. Graph.*, vol. 34, no. 5, pp. 482–498, 2010.

[10] R. Hu, T. Wang, and J. Collomosse, "A bag-of-regions approach to sketch-based image retrieval," in *Proc. 18th IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2011, pp. 3661–3664.

[11] R. Hu and J. Collomosse, "A performance evaluation of gradient field HOG descriptor for sketch based image retrieval," *Comput. Vis. Image Understand.*, vol. 117, no. 7, pp. 790–806, Jul. 2013.

[12] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 831–837.

[13] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "Sketch-based image retrieval: Benchmark and bag-of-features descriptors," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 11, pp. 1624–1636, Nov. 2011.

[14] L. Liu, F. Shen, Y. Shen, X. Liu, and L. Shao, "Deep sketch hashing: Fast free-hand sketch-based image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2862–2871.

[15] G. Tolias and O. Chum, "Asymmetric feature maps with application to sketch based retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2377–2385.

[16] P. Xu *et al.*, "SketchMate: Deep hashing for million-scale human sketch retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 8090–8098.

[17] T. Bui, L. Ribeiro, M. Ponti, and J. Collomosse, "Sketching out the details: Sketch-based image retrieval using convolutional neural networks with multi-stage regression," *Comput. Graph.*, vol. 71, pp. 77–87, Apr. 2018.

[18] D. Xu, X. Alameda-Pineda, J. Song, E. Ricci, and N. Sebe, "Cross-paced representation learning with partial curricula for sketch-based image retrieval," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4410–4421, Sep. 2018.

[19] X. Qian, X. Tan, Y. Zhang, R. Hong, and M. Wang, "Enhancing sketch-based image retrieval by re-ranking and relevance feedback," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 195–208, Jan. 2016.

[20] H. Hu, K. Wang, C. Lv, J. Wu, and Z. Yang, "Semi-supervised metric learning-based anchor graph hashing for large-scale image retrieval," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 739–754, Feb. 2019.

[21] H. Zhang, P. She, Y. Liu, J. Gan, X. Cao, and H. Foroosh, "Learning structural representations via dynamic object landmarks discovery for sketch recognition and retrieval," *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4486–4499, Sep. 2019.

[22] Y. Shen, L. Liu, F. Shen, and L. Shao, "Zero-shot sketch-image hashing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3598–3607.

[23] J. Zhang *et al.*, "Generative domain-migration hashing for sketch-to-image retrieval," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 297–314.

[24] S. K. Yelamarthi, S. K. Reddy, A. Mishra, and A. Mittal, "A zero-shot framework for sketch based image retrieval," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 316–333.

[25] C. Deng, X. Xu, H. Wang, M. Yang, and D. Tao, "Progressive cross-modal semantic network for zero-shot sketch-based image retrieval," *IEEE Trans. Image Process.*, vol. 29, pp. 8892–8902, 2020.

[26] A. Dutta and Z. Akata, "Semantically tied paired cycle consistency for any-shot sketch-based image retrieval," *Int. J. Comput. Vis.*, vol. 128, no. 10, pp. 2684–2703, 2020.

[27] Q. Yu, J. Song, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Fine-grained instance-level sketch-based image retrieval," *Int. J. Comput. Vis.*, vol. 129, pp. 1–17, Sep. 2020.

[28] J. Song, Y.-Z. Song, T. Xiang, T. M. Hospedales, and X. Ruan, "Deep multi-task attribute-driven ranking for fine-grained sketch-based image retrieval," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, vol. 1, 2016, p. 3.

[29] J. Song, Q. Yu, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Deep spatial-semantic attention for fine-grained sketch-based image retrieval," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5551–5560.

[30] K. Li, K. Pang, Y.-Z. Song, T. M. Hospedales, T. Xiang, and H. Zhang, "Synergistic instance-level subspace alignment for fine-grained sketch-based image retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5908–5921, Dec. 2017.

[31] A. K. Bhunia, Y. Yang, T. M. Hospedales, T. Xiang, and Y.-Z. Song, "Sketch less for more: On-the-fly fine-grained sketch-based image retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9779–9788.

[32] J. Li, L. Liu, L. Niu, and L. Zhang, "Memorize, associate and match: Embedding enhancement via fine-grained alignment for image-text retrieval," *IEEE Trans. Image Process.*, vol. 30, pp. 9193–9207, 2021.

[33] P. Xu *et al.*, "Fine-grained instance-level sketch-based video retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 5, pp. 1995–2007, May 2021.

[34] X. Liu, Z. Han, Y.-S. Liu, and M. Zwicker, "Fine-grained 3D shape classification with hierarchical part-view attention," *IEEE Trans. Image Process.*, vol. 30, pp. 1744–1758, 2021.

[35] R. Du, J. Xie, Z. Ma, D. Chang, Y.-Z. Song, and J. Guo, "Progressive learning of category-consistent multi-granularity features for fine-grained visual classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Nov. 13, 2021, doi: 10.1109/TPAMI.2021.3126668.

[36] H. Huang, J. Zhang, L. Yu, J. Zhang, Q. Wu, and C. Xu, "TOAN: Target-oriented alignment network for fine-grained image categorization with few labeled samples," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 2, pp. 853–866, Feb. 2022.

[37] X.-S. Wei *et al.*, "Fine-grained image analysis with deep learning: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Nov. 13, 2021, doi: 10.1109/TPAMI.2021.3126648.

[38] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, "Sketch2Photo: Internet image montage," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 1–10, Dec. 2006.

[39] S. Dey, A. Dutta, S. K. Ghosh, E. Valveny, J. Lladós, and U. Pal, "Learning cross-modal deep embeddings for multi-object image retrieval using text and sketch," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 916–921.

[40] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[41] L. Wu, P. Sun, R. Hong, Y. Fu, X. Wang, and M. Wang, "SocialGCN: An efficient graph convolutional network based model for social recommendation," 2018, *arXiv:1811.02815*.

[42] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2019, vol. 33, no. 1, pp. 922–929.

[43] M. Guo, E. Chou, D.-A. Huang, S. Song, S. Yeung, and L. Fei-Fei, "Neural graph matching networks for fewshot 3d action recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 653–669.

[44] R. Hu, Z. Huang, Y. Tang, O. Van Kaick, H. Zhang, and H. Huang, "Graph2Plan: Learning floorplan generation from layout graphs," *ACM Trans. Graph.*, vol. 39, no. 4, pp. 1–118, Aug. 2020.

[45] T. Zhang, B. Liu, D. Niu, K. Lai, and Y. Xu, "Multiresolution graph attention networks for relevance matching," in *Proc. ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2018, pp. 933–942.

[46] X. Jia, H. Zhao, Z. Lin, A. Kale, and V. Kumar, "Personalized image retrieval with sparse graph representation learning," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 2735–2743.

[47] Z.-M. Chen, X.-S. Wei, P. Wang, and Y. Guo, "Multi-label image recognition with graph convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5177–5186.

[48] Z. Zhang, Y. Zhang, R. Feng, T. Zhang, and W. Fan, "Zero-shot sketch-based image retrieval via graph convolution network," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 12943–12950.

[49] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[50] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12026–12035.

[51] W. Zhang, Z. Lin, J. Cheng, C. Ma, X. Deng, and H. Wang, "STA-GCN: Two-stream graph convolutional network with spatial–temporal attention for hand gesture recognition," *Vis. Comput.*, vol. 36, nos. 10–12, pp. 2433–2444, Oct. 2020.

[52] *Word2vec Software*. Accessed: 2013. [Online]. Available: https://code.google.com/archive/p/word2vec/

[53] M. Eitz, J. Hays, and M. Alexa, "How do humans sketch objects?" *ACM Trans. Graph.*, vol. 31, no. 4, pp. 1–10, Aug. 2012.

[54] D. Ha and D. Eck, "A neural representation of sketch drawings," 2017, *arXiv:1704.03477*.

[55] S. Dey, P. Riba, A. Dutta, J. L. Llados, and Y.-Z. Song, "Doodle to search: Practical zero-shot sketch-based image retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2179–2188.

[56] H. Caesar, J. Uijlings, and V. Ferrari, "COCO-stuff: Thing and stuff classes in context," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1209–1218.

[57] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[58] B. J. Prosser *et al.*, "Person re-identification by support vector ranking," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2010, vol. 2, no. 5, pp. 1–11.

[59] Q. Yu, Y. Yang, F. Liu, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Sketch-a-net: A deep neural network that beats humans," in *Proc. Int. J. Comput. Vis.*, vol. 122, no. 3, pp. 411–425, May 2017.

[60] R. Hu, M. Barnard, and J. Collomosse, "Gradient field descriptor for sketch based retrieval and localization," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2010, pp. 1025–1028.

[61] A. Yu and K. Grauman, "Fine-grained visual comparisons with local learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 192–199.

[62] H. Fu *et al.*, "3D-FUTURE: 3D furniture shape with TextURE," *Int. J. Comput. Vis.*, vol. 129, no. 12, pp. 3313–3337, Dec. 2021.

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

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

Changqing Zou received the B.E. degree from the Harbin Institute of Technology, Harbin, China, the M.E. degree from the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing, China, and the Ph.D. degree from the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His research interests include computer graphics, computer vision, and machine learning.

Yu-Kun Lai (Member, IEEE) received the bachelor's and Ph.D. degrees in computer science from Tsinghua University in 2003 and 2008, respectively. He is currently a Professor of visual computing with the School of Computer Science and Informatics, Cardiff University. His research interests include computer graphics, geometry processing, image processing, and computer vision. He is on the editorial boards of *Computer Graphics Forum* and *The Visual Computer*.

Keqi Chen received the B.Eng. degree from Southeast University, China, in 2019. He is currently pursuing the master's degree with the School of Computer Science and Technology, University of Chinese Academy of Sciences. His research interests include computer vision and human–computer interaction.

Ran Zuo received the B.S. degree from Beijing Normal University, China, in 2018. She is currently pursuing the Ph.D. degree with the State Key Laboratory of Computer Science and the Beijing Key Laboratory of Human-Computer Interaction, Institute of Software, Chinese Academy of Sciences, China. Her research interests include sketch-based video retrieval and computer vision.

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 was a Research Associate with the Department of Computer Science, Naval Postgraduate School, Monterey, CA, USA, from 2005 to 2006. She is currently a Professor with the Institute of Software, Chinese Academy of Sciences. Her research interests include sketch interaction, multimodal interaction and cognitive computation.

Yong-Jin Liu (Senior Member, IEEE) 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 currently a Professor with the 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 currently a Professor with the Institute of Software, Chinese Academy of Sciences. He is the Director of the Intelligence Engineering Laboratory. His research interests include human–computer interaction, real-time intelligence, and real-time active database.