Context-based Sketch Classification

Zhang, Jianhui; Chen, Yilan; Li, Lei; Fu, Hongbo; Tai, Chiew-Lan

Published in:
Expressive '18 - Proceedings of the Joint Symposium on Computational Aesthetics Sketch-Based Interfaces and Modeling Non-Photorealistic Animation and Rendering

Published: 01/08/2018

Document Version:
Post-print, also known as Accepted Author Manuscript, Peer-reviewed or Author Final version

License:
Unspecified

Publication record in CityU Scholars:
Go to record

Published version (DOI):
10.1145/3229147.3229154

Publication details:

Citing this paper
Please note that where the full-text provided on CityU Scholars is the Post-print version (also known as Accepted Author Manuscript, Peer-reviewed or Author Final version), it may differ from the Final Published version. When citing, ensure that you check and use the publisher's definitive version for pagination and other details.

General rights
Copyright for the publications made accessible via the CityU Scholars portal is retained by the author(s) and/or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights. Users may not further distribute the material or use it for any profit-making activity or commercial gain.

Publisher permission
Permission for previously published items are in accordance with publisher's copyright policies sourced from the SHERPA RoMEO database. Links to full text versions (either Published or Post-print) are only available if corresponding publishers allow open access.

Take down policy
Contact lbscholars@cityu.edu.hk if you believe that this document breaches copyright and provide us with details. We will remove access to the work immediately and investigate your claim.

Download date: 19/12/2019
Context-based Sketch Classification

Jianhui Zhang
Hong Kong UST

Yilan Chen
City University of Hong Kong

Lei Li
Hong Kong UST

Hongbo Fu
City University of Hong Kong

Chiew-Lan Tai
Hong Kong UST

Figure 1: Given a sketched scene consisting of semantically segmented objects as input, our framework produces object categories with higher contextual compatibility compared to the predictions of the state-of-the-art single-object classification method (indicated within the parentheses). Our framework can be applied to both sketch co-classification (a)(b) and incremental sketch classification (c). The processing order is indicated next to the categories in (c).

ABSTRACT
We present a novel context-based sketch classification framework using relations extracted from scene images. Most of existing methods perform sketch classification by considering individually sketched objects and often fail to identify their correct categories, due to the highly abstract nature of sketches. For a sketched scene containing multiple objects, we propose to classify a sketched object by considering its surrounding context in the scene, which provides vital cues for alleviating its recognition ambiguity. We learn such context knowledge from a database of scene images by summarizing the inter-object relations therein, such as co-occurrence, relative positions and sizes. We show that the context information can be used for both incremental sketch classification and sketch co-classification. Our method outperforms a state-of-the-art single-object classification method, evaluated on a new dataset of sketched scenes.

CCS CONCEPTS
• Human-centered computing → Human computer interaction (HCI); • Computing methodologies → Shape analysis;

KEYWORDS
Sketch classification, context, object relations, co-analysis

ACM Reference Format:

1 INTRODUCTION
Freehand sketching is arguably one of the most accessible and efficient means for communicating ideas. However, semantically recognizing roughly-sketched objects remains an algorithmic challenge. Unlike natural images with rich texture details, sketches are a unique 2D art form that tends to be drawn at different levels of abstraction and in different styles, commonly with noticeable distortions. A robust sketch recognition approach capable of accommodating variations can facilitate better human-human communications and human-computer interactions in a high-level language and empower many practical downstream applications, such as graphical design, sketch-based retrieval or modeling.

Existing studies have mainly focused on identifying categories of individually sketched objects by casting the problem as an image classification task and applying available techniques from the natural image domain. Classifiers, such as Support Vector Machines (SVMs), are trained on certain image features that are extracted from a database of collected sketches [7, 33]. Recently, researchers have also employed deep Convolutional Neural Networks (CNNs) to learn more discriminant features for single sketched object classification [32, 47]. Although their accuracies (e.g., close to 80% on their...
tested datasets) surpass the performance of traditional approaches based on hand-crafted features (69%) [33], they still often produce wrong predictions due to the inherent interpretation ambiguity of individual sketches. Such a problem deteriorates considerably when the sketches are highly abstract and created in a short period, like millions of doodles in the QuickDraw dataset [12].

We observe that users often sketch multiple objects together to compose a scene that visualizes a complex concept [5, 37, 44]. Recognition uncertainty can be alleviated by considering relations among the objects in the scene as additional cues. For example, in Fig. 1-(b), it is very challenging for existing approaches to recognize each object individually (e.g., keyboard or mouse), since other categories in the training dataset (e.g., pillow or vase) may also contain sketches of similar shapes. However, if all the sketched objects are jointly considered, the compatibility of their predicted categories, in terms of co-occurrence and spatial relations, can strongly indicate that the drawing is more likely to be a workspace and that the ambiguous objects are more likely a keyboard and a mouse.

This motivates us to propose a new sketch classification framework that is endowed with relations of object categories to better address the above ambiguity issue. To this end, we need to solve two main challenges. Firstly the extraction of relation priors on categories of sketched objects requires voluminous sketch data. Unfortunately, there is a lack of large-scale datasets of sketched scenes at present. Existing sketch datasets, such as [7, 12, 32], only comprise drawings of single objects. We resort to existing image datasets that contain rich annotations of objects in real-world scenes, and we show that it is a viable solution to transfer and apply the learned relation priors across different domains (i.e., image to sketch). The other challenge is to identify and quantify relations that are effective for ambiguity resolution, and then to unify them.

Taking a scene sketch composed of semantically segmented objects as input, we leverage existing CNNs, which are trained for single-object classification, to produce candidate categories for each sketched object in the scene. With the relation priors extracted, we define a compatibility function to evaluate the plausibility of specifically assigned categories of a pair of sketched objects. Then a prioritizing process is designed to select the most probable candidates that respect the extracted relation priors well so that a semantically valid scene is formed. Specifically, we present two context-based sketch classification algorithms that target different sketching scenarios: an incremental algorithm (Fig. 1-(c)) for recognizing a new object given confirmed categories of existing objects as context, and a co-classification algorithm (Fig. 1-(a)(b)) for jointly identifying the categories for all the objects in the scene at once. To test the performance of the proposed algorithms, we collect an evaluation dataset that contains 332 freehand scene sketches, covering various object categories and scene types. Overall our incremental classification algorithm is 6.7% higher and our co-classification algorithm is 6.3% higher than the CNN-only method [32] in terms of classification accuracy. We will release this dataset to the community for future relevant research.

To sum up, our contributions in this work are three-fold: 1) extracting and transferring relation priors from the image domain to the sketch domain to alleviate recognition ambiguity in sketches; 2) a context-based sketch classification framework with two specific algorithms for different sketching scenarios, each achieving higher accuracy than the state-of-the-art CNN for single-object classification; 3) a new dataset of scene sketches for benchmarking the performance of relevant recognition algorithms.

2 RELATED WORK

The seminal work Sketchpad [41] introduced sketching as a means of human-computer interaction in the earliest days of the computer graphics field. Since then, much research effort has been devoted to low-level understanding of sketches, such as identifying geometric primitives (e.g., lines or circles) [26, 36] or discovering repetitive patterns for gesture recognition [6]. Information like stroke ordering has been used in sketch recognition [34]. Prior knowledge can also be utilized to facilitate the recognition of domain-specific drawings [1], such as mathematical expressions [17], chemical drawings [25], electronic circuit diagrams [15, 35] or architectural drawings [21].

Recently, high-level semantic understanding of sketches, in terms of objectness, has received an increasing amount of research attention. However, due to different levels of abstraction and styles in freehand drawings, recognizing a single sketched object semantically is challenging, especially for highly abstract ones. Given a dataset of sketched objects as training data, a common paradigm among existing studies is to first extract certain hand-crafted features (most of them originally proposed for natural images) from the sketch images and then to train a classifier (e.g., SVMs) on the feature representations to predict the categories of unseen sketched objects. The first work for large-scale recognition analysis of human sketched objects, by Eitz et al. [7], used a bag-of-features representation [39] of sketch images and trained multi-class SVMs on a dataset of 250 object categories, collected via crowd-sourcing, achieving a recognition accuracy of 56%. In contrast, humans can recognize on average 73.1% of sketches in this dataset. Several follow-up studies differ mostly in the use of feature representations originated from natural images, for instance, by replacing bag-of-features with Fisher Vectors [33] (with the recognition accuracy improved to 68.9%) or by combining several types of local features [20] (65.8%).

Visual recognition tasks [31], such as object detection and classification in natural images, have advanced significantly thanks to the rapid development of deep learning. Similar approaches have been applied to single sketched object recognition. Instead of relying on hand-crafted features of sketch images, Yu et al. [47] adopted a CNN architecture, matured in visual recognition tasks, to learn more discriminant hierarchical features for classifying sketches more robustly. Their carefully-designed sketch classification network outperforms humans, achieving an accuracy of 77.95% on the TU-Berlin dataset [7]. To address the lack of texture details in sketches, Zhang et al. [48] introduced another CNN-based method that uses natural images as an intermediate for learning shared features of sketches more effectively. The recent work done by Sangkloy et al. [32] reported the state-of-the-art accuracy (80.85%) using GoogLeNet [42] on the above dataset.

We reiterate that the aforementioned methods for high-level semantic recognition of sketches, including the ones with hand-crafted features and the ones with CNNs, are designed for individually sketched objects. In practice, multiple objects are often drawn
together to depict a more complex scene, for example, for the composition of new images [5, 13], diagrams [45] or 3D scenes [37, 44]. In such scenarios, the relations among the sketched objects can provide strong cues for alleviating recognition uncertainty, which is the main focus of our work.

Leveraging object relations has been explored extensively in scene-object classification tasks in natural images, but rarely in scene sketches. Galleguillos and Belongie [11] summarized three types of relations mainly used in image classification: co-occurrence, positions and sizes. Generally, such relation features are utilized after an independent categorization step to filter out incompatible results [9, 30], or they are concatenated with other image features for joint classification [19, 23, 24]. The latter however, requires a larger number of annotated images for training. As there is a lack of labeled scene-sketch data, we choose to transfer relations from the image domain to the sketch domain, and combine them with single sketched object recognition methods. Thus our work is more similar to the first approach.

Fisher et al. [10] presented a context-based 3D model retrieval method, which uses relations between objects to predict potential candidates for subsequent retrieved objects. Xu et al. [44] proposed a sketch co-analysis approach, which is intended for finding semantically more meaningful 3D model combinations from individually retrieved results for sketch-based indoor scene modeling. Our work extends their co-analysis idea of utilizing context information to a more general sketch classification problem and aims to accommodate various object categories and scene types beyond indoor scenes. Similar to [44], we assume that an input scene sketch is composed of semantically segmented objects drawn one by one. Semantic sketch segmentation [2, 14, 40] is a challenging problem on its own. In this work, we mainly focus on the context-based recognition of sketched scene objects, expecting segmentations from user inputs.

Sketch-based image retrieval (SBIR), another line of studies, is related to sketch recognition, but generally aims to find visually the most relevant image content with respect to an input sketch [4, 5, 8, 46], instead of attempting on semantic understanding of the sketch. Sketch recognition, as demonstrated by [18], can help to reduce the retrieval space significantly if the category of the input sketch is robustly identified. Recent works utilize Siamese CNN to minimize the distance between relevant sketches and images [3, 28]. Several studies on SBIR, such as [27, 29], also investigated supplementing existing SBIR systems with a refinement post-process to group and re-rank the retrieval results. Despite sharing a similar refinement idea with our context-based framework for sketch classification, their designs heavily utilize the content and features of the retrieved real images while our work strives to use the learned relation priors to prioritize the candidate categories of input sketches that contain extremely limited feature information.

3 OVERVIEW

Our task is to take a scene sketch as input, which contains semantically segmented objects drawn one by one, and recognize the objects therein of unknown categories by considering both each sketched object itself and its context. We first use a pre-trained CNN for individual object recognition, which produces a set of candidate categories for each object. We then employ relation knowledge learned from a large-scale scene-image dataset, together with the CNN predictions, and select the most probable candidates by assessing the object category compatibility in the scene, yielding a more robust and accurate sketch classification framework (see Fig. 2).

Similar to [32], we adopt the GoogLeNet [42] as the base of our CNN architecture due to its demonstrated state-of-the-art performance on sketches. Other single-object classification methods may be used alternatively. This network is pre-trained on ImageNet [31] and then fine-tuned on a sketch dataset with the last layer modified to produce estimations over our used categories (Section 6.1).

While there is no large dataset of scene-sketches available for learning the relation knowledge of object categories, we use existing scene-image datasets that are of large volume and contain rich annotations [16, 43]. We assume that, when drawing multiple objects, users still loosely follow the relations found in real scenes. We construct a graph to represent the extracted relations such as co-occurrence and spatial relations of object categories (Section 4).

Next we define a relation compatibility function that measures how well a pair of objects and their assigned categories respect the learned relation priors. We use this function as a building block and design context-based incremental classification and co-classification algorithms for recognizing the sketched objects in a scene (Section 5), catering two different sketching paradigms. We show that the proposed algorithms, leveraging both CNNs and the relation graph, are effective at alleviating recognition ambiguity existing in freehand sketches.

4 RELATION EXTRACTION

Now we describe the procedure of extracting relation knowledge from a large-scale scene-image dataset. We use the Visual Genome dataset [16], which contains rich annotations such as attributes and relationships of object instances. We aim at constructing a directed relation graph \( G = (V, E) \), where the nodes \( V \) denote the set of object categories of interest, and the edges \( E \) denote the set of pairwise relations among the categories. For an ordered category pair \((u, v)\) (\(u, v \in V\)), if \(u\) has certain valid relations with \(v\) (Fig. 3), a link edge \(e_{uv}\) from \(u\) to \(v\) is inserted into \(E\), and in turn, so as is \(e_{vu}\) for \((v, u)\). For simplicity, in the following sections, we use \((u, v)\) to illustrate the extracted relations; the inverse relations can...
4.2 Spatial Association

To measure the spatial association of an ordered category pair \((u, v)\), we again make use of the object relation pairs in the Visual Genome dataset and introduce a set of discrete attributes \(R_{uv}\) to describe the annotations (e.g., person wearing hat) with respect to the bounding boxes of objects. The attribute set, \(R_{uv}\), includes relative position, relative size and overlap ratio of the objects of categories \(u\) and \(v\). Specifically, \(R_{uv}\) includes:

- the objects of \(u\) are above (below) the objects of \(v\) w.r.t their bounding box centers;
- the objects of \(u\) are near to (far from) the objects of \(v\) w.r.t their bounding box centers;
- the objects of \(u\) are wider (narrower) than the objects of \(v\) w.r.t their bounding box widths;
- the objects of \(u\) are taller (shorter) than the objects of \(v\) w.r.t their bounding box heights;
- the objects of \(u\) have full (partial, no) overlap regions with the objects of \(v\).

The details of computation and thresholds are provided in the supplementary material. We iterate all the images with object relation pairs of \(u\) and \(v\), and compute the frequencies for all the listed attributes. Only the attributes with high-frequency (\(\geq 70\%\) in our implementation) are kept in \(R_{uv}\). For example, as shown in Fig. 3, since a mouse may appear above or below a keyboard with approximately equal frequency when being captured from arbitrary viewpoints, the category pair (keyboard, mouse) discards the above/below attribute that conveys unreliable vertical position information. We then associate the filtered attribute set \(R_{uv}\) with edge \(e_{uv}\). Note that for edge \(e_{uv}\), the attribute values in \(R_{uv}\) can be easily derived from \(R_{uv}\). For example, in Fig. 3, the category pair (hat, person) has \(\{\text{partial overlap, above, narrower, shorter, near}\}\). Unlike the above and below attributes, which often stem from physical supporting constraints, the left and right attributes may be easily affected by the camera viewpoints and are thus less reliable. Thus, we excluded the left and right attributes from spatial association.

Discussions. For a category pair \((u, v)\), we initially tried a different approach to quantifying the spatial association by estimating continuous probability density functions on relative positions and sizes of object instances. Specifically, we tried to use kernel density estimation to capture the distribution of object centers of \(u\) with respect to the object centers of \(v\) in the dataset. However, we found that for many category pairs, it is hard to estimate reliable probability models, possibly because the supporting samples of relative positions are often quite noisy. A similar problem was also encountered during the estimation of probability models for relative sizes. Our adopted approach based on discrete attributes, as described above, works well as shown in Section 6.2.

5 CONTEXT-BASED CLASSIFICATION

In this section, we first define a pairwise score function to assess the relation compatibility of two sketched objects being assigned with specific candidate categories (Section 5.1). The pairwise score function is built upon the extracted relation priors from Section 4. Then we present two context-based sketch classification algorithms for different scenarios: incremental classification (Section 5.2) and
co-classification (Section 5.3) of sketches, showing the usefulness of employing relation knowledge to alleviate recognition ambiguity.

## 5.1 Relation Compatibility Function

Given two sketched objects \( o_1 \) and \( o_2 \) and their assigned object categories \( u \) and \( v \) as input, the relation compatibility function considers the co-occurrence and spatial association of the categories to produce a score, indicating the plausibility of \( (o_1, o_2) \) being labeled as \( (u, v) \) with respect to the extracted relation priors.

We first define a spatial association score \( p \) for measuring how well the bounding boxes of \( (o_1, o_2) \) satisfy the spatial attributes in \( R_{uv} \) as follows:

\[
p(o_1, o_2, u, v) = \begin{cases} 
q_{\Delta o_1 o_2 R_{uv}} \cdot b |R_{uv}|^{-\Delta o_1 o_2 R_{uv}} & \text{if } |R_{uv}| > 0 \\
1 & \text{otherwise} 
\end{cases}
\]

where \( \Delta o_1 o_2 R_{uv} \) counts the number of attributes in \( R_{uv} \) that are satisfied by the bounding boxes of \( o_1 \) and \( o_2 \), \( a \) is a bonus factor and \( b \) is a penalty factor. We set \( a = 1.5 \) and \( b = 0.5 \). Intuitively, objects \( o_1 \) and \( o_2 \) are more compatible to each other if their bounding boxes satisfy more relation priors.

Next, we define the relation compatibility function \( \psi \) as

\[
\psi(o_1, o_2, u, v) = \eta_{uv} \cdot p(o_1, o_2, u, v),
\]

where the first term is the co-occurrence score and the second term is the spatial association score. In the following, we demonstrate the usage of \( \psi \) in two context-based sketch classification scenarios.

## 5.2 Incremental Classification

In an interactive scenario, the user may choose to confirm the category after sketching each object in the scene. Thus, to classify a newly sketched object, a definite scene context with confirmed categories of existing objects is given to the algorithm.

Suppose there are \( n \) context objects \( \{o_i\}_{i=1}^n \) with the corresponding known categories \( \{v_i\}_{i=1}^n \) (\( v_i \in V \)). Let \( o_{n+1} \) denote the newly sketched object that needs to be recognized. We first feed \( o_{n+1} \) into our pre-trained CNN to obtain the ranked top-\( k \) candidate categories \( \{\hat{v}_i\}_{i=1}^k \). Next, we define a re-ranking score function to select the most compatible category for \( o_{n+1} \) with respect to the given context as follows:

\[
x(o_{n+1}, \hat{v}_i | \{o_i\}, \{v_i\}) = p(\hat{v}_i | o_{n+1}) \cdot \max(m \sum_{i=1}^n \psi(o_{n+1}, o_i, \hat{v}_i, v_i), \lambda),
\]

where \( p(\hat{v}_i | o_{n+1}) \) is the probability of \( o_{n+1} \) being of category \( \hat{v}_i \), as predicted by the CNN, and \( m \) is the number of context objects whose categories \( \{v_i\} \) have valid co-occurrence relations with \( \hat{v}_i \) as defined in Eq. 1. For a candidate category that has zero or little compatibility with context objects, in order to maintain the influence of CNN, we assign a fixed value of re-ranking score \( \lambda = 0.001 \) in our implementation to it. By multiplying a factor \( m \), we amplify the influence of aggregated contexts. In other words, in Eq. 4, we consider the probability of an object to be assigned with a certain category by its appearance as well as the relation compatibility of the whole scene. We iterate all the top-\( k \) candidate categories (see Section 6.2), compute the re-ranking scores, and classify the sketched object \( o_{n+1} \) as the category with the highest score.

We designed a simple user interface for sketching with our incremental classification algorithm. In our UI, we allow the user to confirm the category of each individually sketched object in a scene and fix wrong predictions if any. Please refer to the supplementary video for demos of our UI and algorithm.

## 5.3 Co-classification

Co-classification of sketched objects shares the same spirit with co-analysis. Instead of solving problems individually, co-analysis correlates the problems and aims at achieving a certain degree of consistency across the results [38, 44]. Different from incremental classification, with co-classification, all the objects in the input scene sketch do not have any pre-confirmed labelings and are analyzed jointly for their final category predictions. Such an algorithm allows the user to draw the objects continuously without interruptions to confirm the categories one by one, and is particularly useful for recognizing objects in sketched scenes created offline.

Given a set of \( n \) sketched objects \( \{o_i\}_{i=1}^n \), the top-\( k \) candidate categories \( \hat{v}_1^k \) for each object \( o_i \) are first obtained via the CNN. Next we define a score function \( \delta \) to evaluate the plausibility of assigning each \( o_i \) with one of its candidates. Suppose one of the assignment combinations is \( \{\hat{v}_1^k\}_{i=1}^n \), where \( \hat{v}_1 \in \hat{v}_1^k \). Let

\[
\delta(\{o_i\}, \{\hat{v}_1^k\}) = \max_{1 \leq i \leq n} p(\hat{v}_i | o_i) \cdot \max_{1 \leq j \leq k} \psi(o_i, o_j, \hat{v}_i, \hat{v}_j, \lambda),
\]

where \( p(\hat{v}_i | o_i) \) is the probability, predicted by the CNN, of category \( \hat{v}_i \) assigned to object \( o_i \) and \( \lambda \) is a small constant which is same as that in Eq. 4 to replace the context score when the candidate category has no relation with its context. Similar to the incremental classification method, the formula above aims to maximize the category compatibility of the assigned combination for each object, leaving other objects as context. However, a brute-force search strategy to maximize the score function \( \delta \) is in \( O(n^2 \cdot k^n) \), which is computationally prohibitive even for a moderate number of sketched objects. Instead, we adopt a beam search algorithm similar to [44]. More specifically, we first consider all possible category pairs for every pair of sketched objects and compute the corresponding scores with \( \delta \) (i.e., by setting \( n = 2 \)). Each category pair is called a size-2 category set. The time complexity for enumerating all size-2 sets is \( O(n^2 k^2) \) and we leave only \( r \) sets with the highest scores for further processing. Next, we iteratively generate the optimal size-\( p \) sub-set via adding one new object to each size \( (p-1) \) set or adding two new objects to each size \( (p-2) \) set. At each iteration, the algorithm will produce at most \( O(n^2 k^2) \) size \( p \) sets for size \( (p-1) \) sets, and \( O(r^2) \) sets for size \( (p-2) \) sets. The above steps are repeated until the size of sub-sets reaches \( n \). Finally, we leave the full-size set with the highest score of \( \delta \) as our co-classification result.

## 6 EVALUATION

To validate our proposed relation-supported sketch classification, we first collected a scene-sketch evaluation dataset along with the ground truth labelings (Section 6.1). Then we performed a
series of experiments on the collected dataset, comparing both the incremental classification and co-classification algorithms with the state-of-the-art CNN for single sketched object recognition, to show the effectiveness of incorporating relations for ambiguity resolution (Section 6.2).

### 6.1 Data Collection

We used the QuickDraw dataset [12] which contains 345 object categories and 70K individually sketched objects per category to train the CNN, and used the Visual Genome dataset [16] which contains 108K scene-images, 80K object categories, 3.8 million object instances to extract relation knowledge. Furthermore, to get a set of object categories with less redundancy and noise, we first obtained the intersection of categories of the QuickDraw and Visual Genome datasets, and then removed the categories that have no co-occurrence. We used this filtered set of categories (in total 70 categories) to construct our sketch scene dataset. The category selection step allows us to unify the pre-trained CNN and the extracted relation priors that are used in our proposed framework. We used a simple sketching interface similar to the one for incremental classification (Section 5.2) for data collection.

We recruited 15 participants with basic drawing skills and asked them to quickly draw simple scene sketches that are semantically meaningful to humans. There were no strict time limits, though we found that most participants finished one scene sketch within three minutes. The participants needed to draw in two different ways: 1) freely choosing categories from the full set to compose a scene; 2) choosing categories only from a given subset to compose a specific type of scene. For the second case, we roughly divided the used object categories into six subsets, representing different scene types, which include dining room, home, outdoor activity, outdoor nature, street view and workplace. The purpose of dividing into different scene types is to test whether our context-based method can perform better under specific scene types where the relations among objects tend to be stronger compared to freely drawn scenes. More specifically, each participant was asked to draw seven groups of sketches in total (i.e., one group of the freely composed scene type and six groups of pre-defined scene types) and three to four sketches per group, and in each scene sketch, at least three objects and ground truth labelings were required.

After the data collection, similar to [7, 32], we manually reviewed all the sketches for verification. One of the users seemed to have misconceptions of our requirements and created sketches that mostly do not represent real-world scenes (Fig. 4-(e)(f)), which violate our assumption that users loosely follow the relations found in real scenes during sketching. We thus removed the collected data of this user and kept the sketches of all the remaining 14 users. Fig. 4 shows some examples of our collected data. As a result, we retained 332 out of 354 collected sketches, which cover all 70 object categories and contain 1,568 individual objects. We then randomly selected 1/3 sketches from each group of sketched scenes drawn by each user as the validation set and left the rest as the test set, resulting in 110 validation samples and 222 test samples. The validation set was used for parameter fine-tuning. Please refer to the supplementary material for the summarized statistics of our dataset.

![Diagram](image-url)

**Figure 4:** Examples of our collected scene sketches along with ground truth labelings. Sketches (a)-(d) are kept in the dataset while sketches (e)-(f), representing invalid scenes, are discarded from the dataset after a data verification process.

### 6.2 Experiments

We evaluate the performance of our proposed context-based sketch classification framework on the test set of the above collected scene-sketch dataset, and as a baseline, we also compare it with the state-of-the-art CNN for single sketched object classification [32].

Table 1 shows the top-1 accuracies of the CNN, our incremental classification and co-classification algorithms in different scenes. The accuracy is calculated as the number of objects with correctly predicted categories over the total number of objects of a specific scene type in the test set. To evaluate the CNN-only method, we

<table>
<thead>
<tr>
<th>Scene Type</th>
<th>CNN</th>
<th>Increment</th>
<th>User Input</th>
<th>Random</th>
<th>Co-class.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dining room</td>
<td>80.1</td>
<td>88.5</td>
<td>82.4</td>
<td>88.5</td>
<td></td>
</tr>
<tr>
<td>Home</td>
<td>69.1</td>
<td>78.1</td>
<td>76.3</td>
<td>75.6</td>
<td></td>
</tr>
<tr>
<td>Outdoor activity</td>
<td>65.7</td>
<td>69.7</td>
<td>64.1</td>
<td>70.4</td>
<td></td>
</tr>
<tr>
<td>Outdoor nature</td>
<td>69.4</td>
<td>79.1</td>
<td>74.0</td>
<td>78.4</td>
<td></td>
</tr>
<tr>
<td>Street view</td>
<td>79.1</td>
<td>83.2</td>
<td>81.0</td>
<td>81.5</td>
<td></td>
</tr>
<tr>
<td>Workplace</td>
<td>71.2</td>
<td>79.7</td>
<td>73.9</td>
<td>83.8</td>
<td></td>
</tr>
<tr>
<td>Free</td>
<td>81.3</td>
<td>86.7</td>
<td>84.0</td>
<td>85.3</td>
<td></td>
</tr>
<tr>
<td>All</td>
<td>74.5</td>
<td>81.2</td>
<td>78.1</td>
<td>80.8</td>
<td></td>
</tr>
</tbody>
</table>

*Table 1: Top-1 accuracy (%) of each method for each scene. "User input" means the use of the input order of sketched objects and "Random" means the random drawing order of objects.*
feed each individually sketched object into the CNN and test the top-1 prediction against the ground truth. To evaluate our incremental classification algorithm, we simulate the interactive sketching scenario, that is, we progressively classify each object in a scene in the drawing order given by the users during data collection. (See User Input in Table 1). For the first object in an input scene, only the CNN is used for prediction and no context computation is involved. Then for each subsequent object, we assign the ground truth to the already tested objects that serve as context, and perform predictions as described in Section 5.2. Specifically, we use all the object categories ranked by the CNN as the candidates, since Eq. 4 can be evaluated efficiently. To evaluate our co-classification algorithm, the categories of all the objects in an input scene are jointly predicted, as described in Section 5.3, without considering the drawing order. For each object, we use top-10 categories predicted by the CNN as candidates. Fig. 8 shows the classification results of several sketched scenes with the above three methods.

Overall our incremental classification algorithm is 6.7% higher and our co-classification algorithm is 6.3% higher than the CNN-only baseline. The one-tail, paired t-tests confirm that both the incremental classification and the co-classification algorithms significantly outperform the CNN-only method (p-value < 0.05). Our incremental classification and co-classification algorithms have corrected 26.2% and 25.8% of the wrong predictions by the CNN, respectively, and made mistakes only for 1.1% and 1.5% of the correct predictions by the CNN.

Table 1 shows the experiment results. It is expected that the accuracy declines with random order. We deduce that this is because every two objects drawn sequentially tend to have stronger relations. The randomized drawing order may cut this underlying connection in the incremental classification scenario.

To validate the effectiveness of our relation formulation, we disable the spatial association term in Eq. 3 by setting it to 1 and then calculate the corresponding accuracies of our modified classification methods. As shown in Table 2, together with Table 1, the overall accuracy of all the sketches declines by 1.1% and 2.0% respectively for the incremental classification and co-classification algorithms without the spatial association term. The t-tests shows that both the incremental classification and co-classification algorithms outperform the ones without the spatial term with p-value < 0.05.

7 DISCUSSION AND FUTURE WORK

In this work, we have developed a context-based framework for sketch classification. We propose to co-analyze the objects within a scene to improve recognition accuracy.

<table>
<thead>
<tr>
<th>Scene Type</th>
<th>Increment.</th>
<th>Co-class.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dining room</td>
<td>87.0</td>
<td>88.5</td>
</tr>
<tr>
<td>Home</td>
<td>77.2</td>
<td>75.6</td>
</tr>
<tr>
<td>Outdoor activity</td>
<td>68.4</td>
<td>67.1</td>
</tr>
<tr>
<td>Outdoor nature</td>
<td>77.7</td>
<td>75.0</td>
</tr>
<tr>
<td>Street view</td>
<td>83.2</td>
<td>79.2</td>
</tr>
<tr>
<td>Workplace</td>
<td>77.1</td>
<td>79.6</td>
</tr>
<tr>
<td>Free</td>
<td>86.2</td>
<td>84.4</td>
</tr>
</tbody>
</table>

Table 2: Top-1 accuracy (%) of our context-based classification methods without the spatial association term.
scene sketch and optimize their compatibility with respect to relation priors extracted from an existing image dataset. Our approach can be applied to two scenarios, namely incremental classification and co-classification. We demonstrate the superior performance of our approach over the state-of-the-art individual sketch classification method through quantitative evaluation on a newly collected dataset of scene sketches.

Our context-based sketch classification framework will be useful for various applications. For example, incremental classification can be seamlessly integrated with interactive drawing systems to better support human-computer interactions, as shown in the supplementary video. In addition to labeling, our method will also facilitate transforming a sketch into other interesting forms, such as clip-art illustrations. More specifically, given a user-sketched scene, multiple objects can be more reliably recognized and then replaced with homogeneous clip-art pieces to synthesize artistic images. Our approach could also be extended to enable a suggestive interface, where users can specify a location in the scene with simple gestures and then compatible objects will be automatically suggested. This could be achieved by simply ignoring the CNN prediction term in Eq. 4 and ranking all the available categories in the relation graph by the computed scores. Such suggestive interfaces would be of help to design exploration.

We have tried an end-to-end learning approach. The main challenge is that we do not have sufficient sketch scene data to train a deep neural network. Therefore, similar to [22], we used the bounding boxes and labels in natural images as templates and filled in sketched objects to generate scene sketches for training. For each scene sketch image, we randomly chose an object as target and the others as context. We then conducted a joint training method to classify the target object with the context information as additional input. The result was promising but not competitive to our work. This is partially because the layouts of natural images are different from those of sketch images, and it may be challenging for the neural network to learn transferable relationships directly. In the future, we will consider to collect a database of sketched scenes consisting of various objects and learn the context information directly from such a database. Our another future work is to unify the object-level sketched scene segmentation and sketch recognition into a deep learning framework.

ACKNOWLEDGMENTS

This work was partially supported by grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU11204014, CityU11204014, HKUST16201315), and ACIM-SCM.

REFERENCES

Figure 8: More classification results. Left column: co-classification results; Right column: incremental classification results.

The labeled text indicates the prediction of our algorithm and the prediction of the CNN within the parentheses. For incremental classification results, the sequence of drawing is indicated before each label. The underlined labels are consistent with the given ground truth. If none of the predictions are correct, we show the given ground truth at the last.

Tomasz Malisiewicz and Alyosha Efros. 2009. Beyond categories: The visual

Yonggang Qi, Yi-Zhe Song, Honggang Zhang, and Jun Liu. 2016. Sketch-based

Kemal Tugrul Yesilbek and T Metin Sezgin. 2017. Sketch recognition with few

Ivan E. Sutherland. 1964. Sketchpad: A Man-machine Graphical Communication

Zhenbang Sun, Changhu Wang, Liqing Zhang, and Lei Zhang. 2012. Free hand-


Oana Sidi, Oliver van Kaick, Yanir Kleiman, Hao Zhang, and Daniel Cohen-

Tevfik Metin Sezgin and Randall Davis. 2001. Sketch Based

Tevfik Metin Sezgin and Randall Davis. 2005. HMM-based efficient sketch recogni-

Rosália G. Schneider and Tinne Tuytelaars. 2014. Sketch Classification and


Brandon Paulson and Tracy Hammond. 2008. PaleoSketch: Accurate Primitive

Tiziano Portenier, Qiyang Hu, Paolo Favarro, and Matthias Zwicker. 2017. SmartS-

X. Qian, X. Tan, Y. Zhang, R. Hong, and M. Wang. 2016. Enhancing Sketch-based

Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. 2016. The


Expressive ’18, August 17–19, 2018, Victoria, BC, Canada

https://doi.org/10.1101/j.cvii.2015.02.003


Zhaoliang Lun, Changqing Zou, Habin Huang, Evangelos Kalogerakis, Ping Tan, Marie-Paule Cani, and Hao Zhang. 2017. Learning to group discrete graphical

Tomaz Malisiewicz and Alyosha Efros. 2009. Beyond categories: The visual


Tevfik Metin Sezgin and Randall Davis. 2005. HMM-based efficient sketch recogni-


Oana Sidi, Oliver van Kuijk, Yunar Kleiman, Hao Zhang, and Daniel Cohen-


Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M. Hospedales, and Chen-


Zhenbang Sun, Changhu Wang, Liqing Zhang, and Lei Zhang. 2012. Free hand-

