Cross-Modal Relation and Sketch Prototype Learning for Zero-Shot Sketch-Based Image Retrieval

Yuanping Song
School of Software Engineering, University Of Science And Technology Of China
sa202254@mail.ustc.edu.cn

Yanwei Yu*
School of Software Engineering, Suzhou Institute for Advanced Research, University Of Science And Technology Of China
ywyu@ustc.edu.cn

Hao Tang
School of Software Engineering, University Of Science And Technology Of China
SA21225402@mail.ustc.edu.cn

Junyuan Guo
School of Software Engineering, University Of Science And Technology Of China
prophetgvo@mail.ustc.edu.cn

Yibo Wang
School of Software Engineering, University Of Science And Technology Of China
wrainbow@mail.ustc.edu.cn

ABSTRACT

Zero-Shot Sketch-Based Image Retrieval (ZS-SBIR) is an innovative cross-modal task that utilizes a sketch to retrieve corresponding images in the zero-shot learning scene. At present, most algorithms treat ZS-SBIR as a typical image classification problem, using image-level features with triplet or cross entropy loss to achieve retrieval, while ignore the correspondence between sketches and images on local features. Therefore, we propose a new Local Feature Contrastive Network (LFCN) for ZS-SBIR from the perspective of contrastive learning. More specifically, a local feature contrastive method is proposed to establish the cross-modal relationships between images and sketches with creatively applying transformers to extract similarity representation from sketch-image pairs to narrow the domain gap. Furthermore, a feature prototype memory bank is designed to learn sketch prototypes to address the in-class diversity problem in sketch domain. A large number of experiments show that our method notably superior to the state-of-the-art algorithms in both TU-Berlin and Sketchy datasets.

CCS CONCEPTS

• **Computing methodologies → Visual content-based indexing and retrieval**.

KEYWORDS

contrastive learning, local features, reranking transformer, sketch-based image retrieval

*Corresponding author.

CSAI 2022, December 9–11, 2022, Beijing, China

ACM ISBN 978-1-4503-9777-3/22/12…$15.00
https://doi.org/10.1145/3577530.3577550

ACM Reference Format:
Yuanping Song, Yanwei Yu, Hao Tang, Junyuan Guo, and Yibo Wang. 2022. Cross-Modal Relation and Sketch Prototype Learning for Zero-Shot Sketch-Based Image Retrieval . In *2022 6th International Conference on Computer Science and Artificial Intelligence (CSAI) (CSAI 2022), December 9–11, 2022, Beijing, China.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3577530.3577550

1 INTRODUCTION

Zero-Shot Sketch-Based Image Retrieval (ZS-SBIR) [24] is an interesting and widely used task derived from Sketch-Based Image Retrieval (SBIR) [1, 23], which aims to retrieve natural images from unknown categories using hand-drawn sketches as queries. However, solving ZS-SBIR is challenging, because it requires simultaneously addressing large domain gaps, intra class diversity in the sketch domain, and knowledge transfer to unknown categories [5].

Previous ZS-SBIR methods aim to learn a general representation for sketches and images through triplet loss or cross entropy loss [34]. However, cross entropy loss classifies sketch and image equally, ignoring the large domain gap between sketch and image in the same category. In addition, due to the small number of positive and negative samples in every batch, triplet loss cannot fully explore the relationship between samples. Furthermore, everyone has their own drawing style, resulting in a diverse appearance of sketches in the same category, which is neglected by many previous algorithms.

Aiming to solve problems mentioned above, we propose a Local Feature Contrastive Network (LFCN) for ZS-SBIR, mainly including a local feature contrastive method using transformers to process sketch-image paired features (see Fig. 1), a well-designed feature prototype memory bank (see Fig. 2), and a teacher network [18] preserving transferable knowledge with a pretrained teacher model. In detail, the feature maps of image I and sketch S are first extracted by a pretrained CNN. To reduce the domain gap, a local feature contrastive method is then conducted to obtain the similarity score of sketch-image pairs with a Reranking Transformer (RRT) [27] from these local descriptors which transform from feature maps. After pooling the feature map to obtain image-level embedding, the cross-modal contrastive method [32] is conducted to reduce the domain

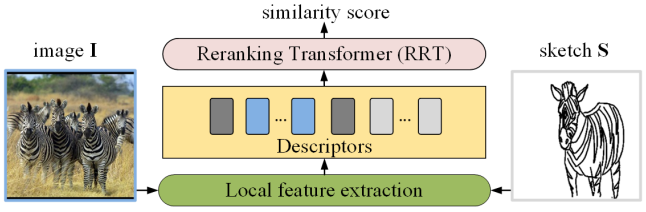

Figure 1: The local feature contrastive method used to mine the relationship of sketch-image pairs on local features to reduce the domain gap.

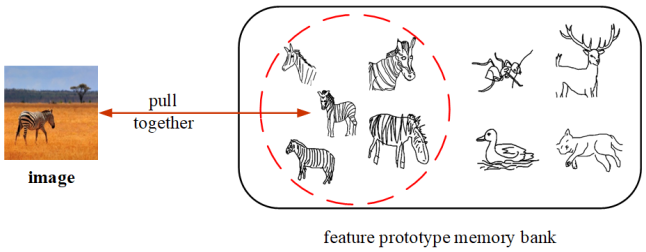

Figure 2: A feature prototype memory bank used to help learn the feature prototypes of sketches to solve the in-class diversity problem in the sketch domain.

gap on image-level retrieval embeddings. Meanwhile, a feature prototype memory bank is proposed to learn sketch prototypes to solve the in-class diversity problem. Besides, to obtain discriminating features, our model is trained using knowledge of teacher model, word embeddings and category label [4, 18, 19]. To confirm the superiority of our LFCN algorithm, we conduct extensive experiments and the results show that our approach markedly outperforms the state-of-the-art algorithms on two large-scale datasets.

The main contributions of this work are as follows:

1) A novel LFCN model for ZS-SBIR task is proposed, the local feature contrastive method of this model creatively employs a reranking transformer to mine the correspondence between the sketch and the image on local features, which effectively reducing the domain gap.
2) To solve the inherent in-class diversity problem in the sketch domain, a feature prototype memory bank is constructed to learn the feature prototypes of sketches, so as to reduce the appearance diversity.
3) Comprehensive ZS-SBIR experiments and ablation studies on two popular large-scale datasets (Sketchy and TU-Berlin) demonstrating the superiority of the proposed LFCN.

2 RELATED WORK

2.1 Metric-Based Learning

Metric-based learning aims to employ a multilevel Convolutional Neural Network (CNN) to extract deep embeddings from photos while maintaining intra-class compactness and large inter-class gap. Most existing algorithms use pair-based or classification-based loss as the objective function, The pair-based algorithms usually focus on mining the relations between photo pairs or triples by forcing positive and negative sample pairs to maintain a minimum margin in the embedding space. Such as Contrastive loss [15], N-Pair Loss [25], Triplet Loss [11], Angular Loss [30] and Multi-Similarity loss [31]. The classification-based algorithms usually represent the learnable category centers with a weight matrix, so as to obtain the global relationship between each sample and all category centers. For example, Softmax Loss [26], CosFace [29], SoftTriplet Loss [21] and ProxyNCA [22]. All these methods align and classify image-level embeddings, but ignore the relationship between sketches and images on local features. In this work, we propose a local feature contrastive algorithm to mine the connections between local features of sketch-image pairs.

2.2 Zero-Shot Sketch-Based Image Retrieval

ZS-SBIR is a sketch-based image retrieval (SBIR) task under zero-shot setting. Since training and testing sets do not have the same category, it is more interesting and practical in realistic scenarios. Shen [24] first combined zero-shot learning with SBIR, and proposed a novel hashing algorithm for ZS-SBIR. Yelamarthi [35] constructed an auto-encoder model to generate image embeddings for alignment through corresponding sketches. Dutta [8] built a cyclic consistent pair generation model and adversarial training is carried out to align sketch and image in semantic space. And Liu [18] regarded ZS-SBIR as a knowledge forgetting problem and adopted a CNN network pretrained on ImageNet [7] as a teacher network to retain the learned knowledge. However, the above approaches neglect the inherent in-class diversity problem within the sketch domain. Therefore, we construct a feature prototype memory bank to learn prototype representations of sketch categories in order to reduce in-class diversity in the sketch field.

2.3 Transformers for image retrieval

The transformer architecture was introduced by Deng [6] for machine translation. It solely relies on self-attention mechanism and fully-connected layers, and achieving an attractive trade-off between efficiency and performance. Recently, transformers have become the mainstream models for extracting image representations in purely visual tasks, such as object detection [2] and visual recognition [16]. These works use the transformers as backbone to obtain representations from a single image and apply them to the prediction and encoding. Inspired by [27], which employs transformers to extract deep representations from natural image pairs in the same domain and apply them to instance-level image retrieval. We innovatively introduces transformers to cross-modal image retrieval. Different from existing algorithms, our approach is to learn sketch-image pair correspondences on local features using transformers under the supervision of its category label.

3 METHODOLOGY

3.1 Problem Definition

In ZS-SBIR tasks, the dataset consists of two subsets, the training set used to train the model and the test set used to verify its performance. The training set is denoted as $\mathcal{D}^{known} = \{\mathcal{I}^{known}, \mathcal{S}^{known}\}$, where I^{known} and S^{known} indicate images and sketches from the

known category, respectively. Similarly, the testing set is denoted as $\mathcal{D}^{un} = \{\mathcal{I}^{un}, \mathcal{S}^{un}\}$. Statistically, $\mathcal{I}^{\text{known}} = \left\{(I_j, y_j) \mid y_j \in C^{\text{known}}\right\}_{j=1}^{N_1}$, $\mathcal{S}^{\text{known}} = \left\{(S_i, y_i) \mid y_i \in C^{\text{known}}\right\}_{i=1}^{N_2}$. where y_i is the category label, C^{known} is the training class set, N_1 and N_2 is the quantity of images and sketches in the training set respectively.

During training, we randomly sample a batch of training images and sketches with their labels to guide the model to learn the semantic representations. During testing, a sketch query $S' \in \mathcal{S}^{\text{un}}$ with its label $y \in C^{un}$ is given, and the goal is to search for the corresponding images $I' \in \mathcal{I}^{\text{un}}$ with the same category label, where C^{un} represents the set of unknown categories. We consider it a successful retrieval if the retrieved image has the same class label y' as the given sketch, that is $y' = y$. The zero-shot setting means that no testing classes were present during the training phase, that is $C^{\text{known}} \cap C^{\text{un}} = \varnothing$.

3.2 Local Feature Contrastive Network (LFCN)

The framework of our LFCN model is shown in Fig. 3. Specifically, we use a CSE-ResNet-50 as the backbone (CSE) to extract feature embeddings for sketches and images [18]. To narrow the domain gap between sketch and image, we use a local feature contrastive method to align these two domains with RRT [27]. In order to solve the inherent in-class diversity problem in the sketch field, we design a feature prototype memory bank. Furthermore, in order to transfer knowledge to unknown categories, a teacher network [18] is adopted to preserve historical knowledge. This section discusses the details of these modules.

Local Feature Contrastive Method

Previous methods usually extract image-level features and align them in semantic space to establish connections between photos, this approach neglects the cross-modal relations between sketches and images on local features. Therefore, we use a local feature contrastive method to establish this connection. Specifically, for each original image I and sketch S from $\mathcal{D}^{known}$, two samples are first obtained through augmentation method [3] respectively, denoted as $\{\hat{I}, \tilde{I}, \hat{S}, \tilde{S}\}$. Then apply CSE to extract their local feature maps:

$$\begin{aligned}[m_{im}, \hat{m}_{im}, \tilde{m}_{im}] &= [\mathrm{CSE}(I), \mathrm{CSE}(\hat{I}), \mathrm{CSE}(\tilde{I})], \\ [m_{\mathrm{sk}}, \hat{m}_{\mathrm{sk}}, \tilde{m}_{\mathrm{sk}}] &= [\mathrm{CSE}(S), \mathrm{CSE}(\hat{S}), \mathrm{CSE}(\tilde{S})].\end{aligned} \tag{1}$$

After obtaining local feature maps m, the global average pooling layer is used to obtain the image-level representations f, and a projection head with single fully-connected layer, called P_c, is used for mapping f to vector $v = \mathrm{P_c}(f)$. So we can get the vectors of original photos and their augmentations, which can be denoted as $\{v_{im}, \hat{v}_{im}, \tilde{v}_{im}, v_{sk}, \hat{v}_{sk}, \tilde{v}_{sk}\} \subseteq V$. Since $\hat{v}_{im}$ and $\tilde{v}_{im}$ are generated from $\hat{I}$ and $\tilde{I}$ respectively, their category labels are same as v_{im}. Similarly, the category label of $\hat{v}_{sk}$ and $\tilde{v}_{sk}$ are same as v_{sk}.

Algorithm. 1 illustrates the whole process of local feature contrastive method producing the similarity score q with reranking transformer (RRT). Specifically, after feature maps m are obtained, a series of local descriptors are extracted from the feature maps m with a CNN projection head P_l. Then, by calculating the cosine similarity between vectors in V, a positive sample with the same category label and the farthest distance and a negative sample with different category label and the closest distance are obtained. The local descriptors of original sketches, images and their augmentations are then paired with positive and negative samples, respectively, to form sample pairs, called Z. Then, each pair is converted into a feature sequence, the sequence together with ⟨CLS⟩ and ⟨SEP⟩ are fed into RRT producing the correlation representation $Z_C^{<CLS>}$, where ⟨CLS⟩ is a special token to summarize the feature sequence and ⟨SEP⟩ is an additional delimiter token. Finally, a binary classifier is conducted to produce the similarity score q.

Algorithm 1 Pseudocode of the local feature contrastive method to obtain similarity score q.

Require: two special tokens ⟨CLS⟩ and ⟨SEP⟩, vectors set $V = \{v_i : i \in (1, \ldots, N)\}$, both local feature maps $m = \{m_i : i \in (1, \ldots, N)\}$ and category labels $y = \{y_i : i \in (1, \ldots, N)\}$ of original sketches, images and their augmentations

Ensure: the similarity score of sample pairs q

$(Z, L, Seq) \leftarrow ([\,], [\,], [\,])$
$Desc \leftarrow P_l(m)$ ▷ obtain local descriptors
$Sim \leftarrow cosine\_similarity(V \cdot V^{\mathrm{T}})$ ▷ correlation matrix
for all $i \in 1, ..., N$ **do**
  $pos_i \leftarrow Desc[\underset{i}{argmin}(Sim[i, y_i = y])]$ ▷ positive sample
  $neg_i \leftarrow Desc[\underset{i}{argmax}(Sim[i, y_i \neq y])]$ ▷ negative sample
  $Z.append((Desc_i, pos_i))$ ▷ form positive pair
  $Z.append((Desc_i, neg_i))$ ▷ form negative pair
  $L.append(1)$ ▷ label of positive pair
  $L.append(0)$ ▷ label of negative pair
  $Seq.append(\langle\langle \mathrm{CLS}\rangle, Desc_i, \langle \mathrm{SEP}\rangle, pos_i\rangle)$ ▷ generate positive pair sequence
  $Seq.append(\langle\langle \mathrm{CLS}\rangle, Desc_i, \langle \mathrm{SEP}\rangle, neg_i\rangle)$ ▷ generate negative pair sequence
end for
$Z_C^{<CLS>} \leftarrow RRT(Seq)$ ▷ correlation vectors
$q \leftarrow Binary\_Classifier(Z_C^{<CLS>})$ ▷ similarity scores **return** q

Our local feature contrastive method (LFCM) loss can be summarized as:

$$\mathcal{L}_{lfcm} = \sum_{i=1}^{|Z|} \left[L_i \cdot \log \sigma\left(q_i\right) + \left(1 - L_i\right) \cdot \log\left(1 - \sigma\left(q_i\right)\right)\right] \tag{2}$$

where for each sample pair $Z_i \in Z$, there is a corresponding label L_i, the values of positive and negative pairs are 1 and 0 respectively. And q is the similarity scores of Z, $|Z|$ represents the cardinality of sample pairs Z. σ is the activation function sigmoid.

Cross-modal Contrastive Method

To reduce the domain gap in image-level representations, we utilize a cross-modal contrastive method proposed in [32]. Specifically, the feature vectors extracted from augmentations $\{\hat{I}, \tilde{I}, \hat{S}, \tilde{S}\}$ can be written as $\{\hat{v}_{im}, \tilde{v}_{im}, \hat{v}_{sk}, \tilde{v}_{sk}\} \subseteq V_s$. For illustration, we represent $\hat{v}_{im}$ as an anchor. it is worth noting that $\hat{v}_{im}$ and $\tilde{v}_{im}$ share the same label even though they are in different views. Hence $\tilde{v}_{im}$ is an **intra-modal** positive sample of $\hat{v}_{im}$. In contrast, $\hat{v}_{sk}$ and $\tilde{v}_{sk}$ are **cross-modal** samples of $\hat{v}_{im}$. Whether the sample label is positive or not depends on whether its category label is consistent with $\hat{v}_{im}$. In the cross-modal scenario, The contrastive method is used to cluster the anchors and positive samples, while the anchors and negative samples are separated.

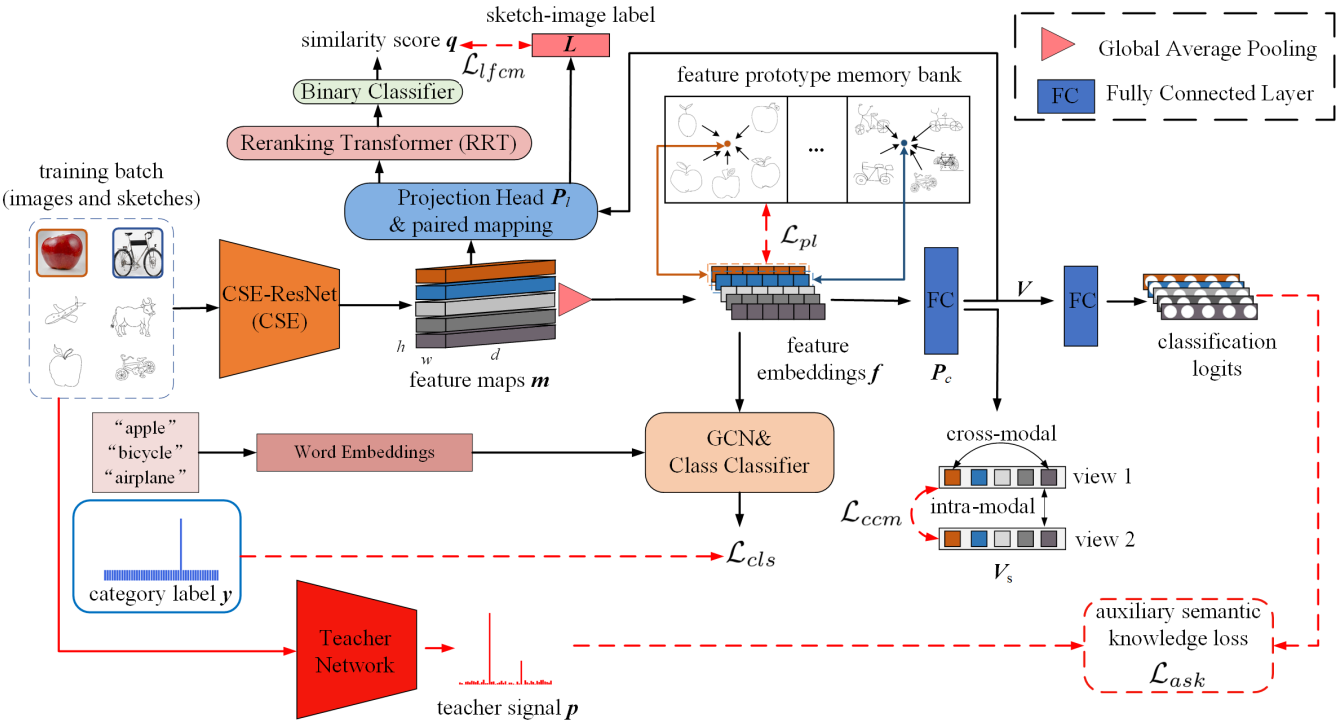

Figure 3: An overview of our model. A CSE-ResNet-50 is used to learn a general embedding for the sketches and images. After getting the feature maps m, a branch employs local feature contrastive method to obtain the similarity score q of sketch-image pairs with reranking transformer (RRT) after a projection head P_l. Image-level feature embeddings f are obtained after pooling, and then input f to four module branches. The first is to use a cross-modal contrastive method in different views after a projection layer P_c. The second is to force the image embedding to come closer to the prototype of sketch representations in the feature prototype memory bank. The third one introduces a graph convolutional neural network (GCN) as a regularizer and align the image-level embeddings with labels y by a class classifier. The last one is to utilize the teacher signal p to preserve knowledge and train the whole student model.

The cross-modal contrastive method (CCM) loss is written as:

$$\mathcal{L}_{ccm} = \sum_{i=1}^{|V_s|} \left[\frac{-1}{2N_i - 1} \sum_{j=1}^{2|V_s|} \mathbf{1}_{y_i = y_j} \log \frac{\exp(v_i \cdot v_j / \tau)}{\sum_{k=1}^{2|V_s|} \mathbf{1}_{i \neq k} \exp(v_i \cdot v_k / \tau)} \right] \quad (3)$$

where for a vector $v \in V_s$, there is a corresponding category label y. $|V_s|$ is the cardinality of feature set V_s, $N_i = \left|\{y_j \mid y_j = y_i, j \neq i, v_i \in V_s, v_j \in V_s\}\right|$. $\mathbf{1}_{\text{condition}} \in \{0, 1\}$ is an indicator function that evaluates to be 1 if the condition is true. τ is a hyper-parameter for contrastive learning.

Feature Prototype Memory Bank

To address the inherent in-class diversity problem in the sketch domain, we design a feature prototype memory bank to learn prototype representations of sketches in the same class. Algorithm. 2 illustrates the whole process of updating the feature prototype memory bank, where $topK(1 - Sm)$ represents to choose the new top-k sketch representations with the largest cosine distances. In detail, for each batch of training, we can get a batch of features $\{\mathcal{F}_{im}, \mathcal{F}_{sk}\}$, where $\mathcal{F}_{\text{im}} = \{f_{\text{im}} \mid y_{\text{im}} \in C_{\text{known}}\}$ and $\mathcal{F}_{\text{sk}} = \{f_{\text{sk}} \mid y_{\text{sk}} \in C_{\text{known}}\}$. To utilize the historical knowledge and update the memory bank at the same time, we adaptively maintain a stack of k sketch representations for each class. First, for each feature f_{sk} and corresponding label y_{sk}, we hunt for image features set $\mathcal{F}'_{im}$ with the same label in a batch, where $\mathcal{F}'_{im} = \{f_{im} \mid y_{im} = y_{sk}\}$. The average of image representations f'_{im} corresponding to f_{sk} can be obtained:

$$f'_{im} = \frac{1}{|\mathcal{F}'_{im}|} \sum_{f_{im} \in \mathcal{F}'_{im}} f_{im} \quad (4)$$

Then, we compute the cosine similarity between f'_{im} and each image feature in set $\mathcal{F}'_{im}$ to select the feature $\hat{f}_{im}$ with the highest similarity to f'_{im}. Then calculate the cosine similarity between $\hat{f}_{im}$ and each feature in set $\mathcal{F}'_{sk} \cup \{f_{sk}\}$ to select the new top-k sketch features with the largest cosine distances and put them in the memory bank, where $\mathcal{F}'_{sk}$ denote the previously top-k sketch representations. The average value of the new sketch features can be calculated after the memory bank is updated:

$$f'_{sk} = \frac{1}{|\mathcal{M}_{sk}|} \sum_{f_{sk} \in \mathcal{M}_{sk}} f_{sk} \quad (5)$$

where $\mathcal{M}_{sk}$ represents the new top-k sketch representations. Then compute the cosine similarity between f'_{sk} and each feature in $\mathcal{M}_{sk}$

to select the feature $\hat{f}_{sk}$ with the highest similarity to f'_{sk} as the prototype of new top-k sketch representations. And the prototype loss (PL) can be written as:

$$\mathcal{L}_{pl} = \left\| |f_{im}| - |\hat{f}_{sk}| \right\|^2 \tag{6}$$

where $|\cdot|$ represents the normalized eigenvector and $\|\cdot\|^2$ denotes the quadratic of Euclidean distance.

Algorithm 2 Pseudocode of updating the feature prototype memory bank and getting the sketch prototype.

Require: image-level features of original sketches $\mathcal{F}_{sk} = \{f_i : i \in (1, \ldots, N)\}$, image-level features of original images $\mathcal{F}_{im} = \{f_j : j \in (1, \ldots, M)\}$, category labels of original sketches $y = \{y_i : i \in (1, \ldots, N)\}$, previous top-$k$ sketch features $\mathcal{F}'_{sk}$

Ensure: new top-k sketch features $\mathcal{M}_{sk}$, feature prototype $\hat{f}_{sk}$

for all $i \in (1, \ldots, N)$ **do**
  $f_{sk} \leftarrow f_i$
  $\mathcal{F}'_{im} \leftarrow \{f_{im} \mid y_{im} = y_i, f_{im} \in \mathcal{F}_{im}\}$ ▷ select image features.
  compute f'_{im} by Equation (4)
  $Sg \leftarrow cosine\_similarity(\mathcal{F}'_{im} \cdot f'_{im})$ ▷ similarity list
  $\hat{f}_{im} \leftarrow \mathcal{F}'_{im}[\underset{i}{argmax}(Sg)]$ ▷ select nearest feature
  $Sm \leftarrow cosine\_similarity(\{\mathcal{F}'_{sk} \cup \{f_{sk}\}\} \cdot \hat{f}_{im})$
  $\mathcal{M}_{sk} \leftarrow topK(1 - Sm)$ ▷ top-k sketch features
  compute f'_{sk} by Equation (5)
  $Sk \leftarrow cosine\_similarity(\mathcal{M}_{sk} \cdot f'_{sk})$
  $\hat{f}_{sk} \leftarrow \mathcal{M}_{sk}[\underset{i}{argmax}(Sk)]$ ▷ sketch prototype
end for

Semantic Knowledge Learning

To obtain discriminative representations, we employ a graph convolutional neural network (GCN) to obtain the semantic information in word embedding [28]. The class classifier is then used to calculate the cross entropy loss so that the image-level embeddings is aligned with the category labels y.

$$\mathcal{L}_{cls} = -\sum_{i=1}^{N} \log \frac{\exp\left(\alpha_{y_i}^{\top} f_i\right)}{\sum_{j \in C^{known}} \exp\left(\alpha_j^{\top} f_i\right)} \tag{7}$$

where α denotes the weight of class classifier, N represents the amount of samples in the training batch. Introducing auxiliary semantic knowledge is an effective way to bridge the known and unknown categories in zero-shot setting. Inspired by [18], we utilize the same teacher model pre-trained on ImageNet to guide our student LFCN model in learning semantic embedding. Specifically, we use the predictive information $p \in \mathbb{R}^{|C^T|}$ of the teacher model as the supervisory signal for samples and their augmentations. where C^T represents the label space for teacher model and satisfies that $C^T \cap C^{\text{un}} = \varnothing$ under the zero-shot setting. Therefore, the auxiliary semantic knowledge (ASK) loss can be written as:

$$\mathcal{L}_{ask} = -\sum_{i=1}^{3N} \sum_{k \in C_T} p'_{i,k} \log \frac{\exp\left(\gamma_k^{\top} f_i\right)}{\sum_{j \in C^T} \exp\left(\gamma_j^{\top} f_i\right)} \tag{8}$$

where p' indicates the weighted teacher signal $\eta \cdot p$ with semantic information proposed in [18],where η is the weight coefficient. γ is the weight of semantic classifier. The total semantic knowledge learning loss can be written as:

$$\mathcal{L}_{skl} = \mathcal{L}_{cls} + \mathcal{L}_{ask} \tag{9}$$

3.3 Objective and Inference

Overall, the complete loss function of the proposed method is:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{lfcm} + \lambda_2 \mathcal{L}_{ccm} + \lambda_3 \mathcal{L}_{pl} + \lambda_4 \mathcal{L}_{skl} \tag{10}$$

where $\lambda_1, \lambda_2, \lambda_3$ and λ_4 are weight coefficients for balancing the overall performance.

In the testing phase, we first obtain the feature maps m of original sketches and images extracted by CSE, then we acquire the image-level representations f through the global average pooling layer. After that, real-value retrieval features $\{v_{im}, v_{sk}\}$ are generated through a projection head P_c with a fully connected layer, and we employ the ITQ algorithm [12, 33] to obtain binary hash codes. Finally, we calculate the distance between sketches and images according to the real-value retrieval features or binary hash codes to achieve retrieval.

4 EXPERIMENTS

4.1 Datasets

There are two popular medium-size datasets Sketchy [23] and TU-Berlin [10] to evaluate the high efficiency of our LFCN model for the ZS-SBIR task.

Sketchy dataset is composed of 75,471 sketches and 12,500 images from 125 different categories. The research [17] expanded the image subset with additional 60,502 images, so the expanded Sketchy dataset contains 73,002 images.

TU-Berlin dataset is composed of 20,000 sketches, which distributed evenly across 250 different categories. The research [17] expanded this dataset with additional 204,489 natural images.

For comparison, we randomly selected 25 and the remaining 100 categories according to the previous division [24] for testing and training in Sketchy. For the TU-Berlin dataset, 30 and the remaining 220 categories were selected for testing and training. Furthermore, following the previous approach, we selected at least 400 images for each class in the test set [8, 24].

4.2 Implementation

Implementation Details. Our LFCN algorithm is implemented by PyTorch [20] on two Tesla P100 16G GPUs. In our model, the image resolution is 224×224, a pretrained CSE-ResNet-50 [18] model is utilized to initialize the backbone. Inspired by [18], we employ a pretrained backbone to provide supervisory signals as auxiliary semantic knowledge in the same way. During the training phase, we use Adam optimizer [13] with initial learning rate of 1e-4 and decay exponentially to 1e-5 during training. In our default setting, the weight decay is set to 1e-4, the batch size is 114 and the training epochs is 30. Unless otherwise stated, we set coefficients $\lambda_1 = 1, \lambda_2 = 0.1, \lambda_3 = 1$ and $\lambda_4 = 1$. For local feature contrastive method, the dimension of descriptor is 128. For cross-modal contrastive method, the hyper-parameter $\tau = 0.07$. For feature prototype memory bank, we set $k = 10$. For semantic knowledge learning, we set $\eta = 0.1$.

Evaluation Metrics. Mean Average Precision (mAP) and Precision ($Prec$) are two mainstream indicators used to evaluate ZS-SBIR

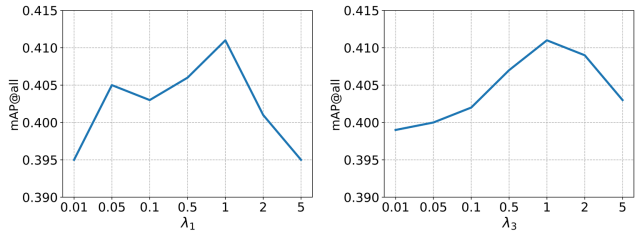

Figure 4: The 64-d binary hash retrieval performances with diverse λ_1 for local feature contrastive method loss and λ_3 for prototype loss in TU-Berlin.

tasks [24]. *Prec* is used to calculate top-k (for example, 100, 200), and *mAP* is utilized to compute top-k or all ranked results. The higher the *Prec* and *mAP*, the better the retrieval performance.

4.3 Comparing with Previous Approaches

We compare our LFCN model with previous ZS-SBIR approaches, all experiments are conducted under the same setting as others for fairness. Hamming distance and cosine distance are used in binary hash code and real - valued feature retrieval respectively. The performances of our LFCN method on ZS-SBIR are shown in Table 1.

As Table 1 shows, our method significantly outperforms most of the previous methods especially in the Sketchy dataset with 64-d embedding. In detail, in 64-d real-valued feature retrieval, our LFCN is superior to the state-of-the-art (SOTA) methods [5, 32] in performance metrics mAP@all and (Prec@100) by 8.3% (8.5%) in Sketchy and 3.7% (2.7%) in TU-Berlin, respectively. In 512-d real-valued feature retrieval, our LFCN outperforms the SOTA method [32] on mAP@all by 3.0% and 0.9% on Sketchy and TU-Berlin, respectively. For binary hash retrieval, we employ ITQ algorithm [12] on the real value feature following the same setting with [18] and the results are better than the SOTA method in performance indicators mAP@all and (Prec@100), and exceed 2.2% (3.9%) and 2.6% (3.7%) respectively on Sketchy and TU-Berlin at 64-d setting. Besides, we show the retrieval performance of 512-d binary hash codes in Table 1, which is similar to the real-valued feature results of 512-d. It proves that LFCN not only products accurate real-valued representations, but also operates stably under the binary hash setting of ZS-SBIR. The above results demonstrate that our LFCN model not only effectively reduces the domain gap between sketch and image, but also significantly addresses the problem of in-class diversity in sketch field.

4.4 Experiment Analysis

Quantitative Analysis

Inspired by previous studies [18, 32], we employ the cross-modal contrastive method to reduce the domain gap in image-level representations and utilize a pretrained teacher model to incorporate auxiliary semantic knowledge into the training process. So in the experimental setting, we set $\lambda_2 = 0.1$ and $\lambda_4 = 1$, respectively, and explore different values of λ_1 and λ_3 on retrieval performance.

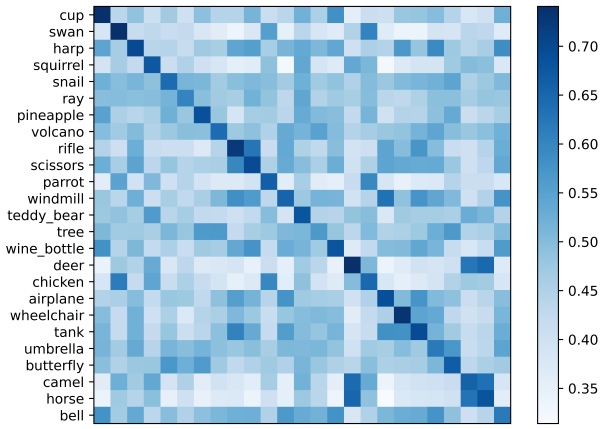

Figure 5: The similarity matrix between the sketches and images represented by 512-d real-valued retrieval representations from the Sketchy dataset. The horizontal and vertical axes denote sketch and image categories, respectively.

Figure 4 shows mAP@all results with diverse coefficients. When setting $\lambda_1 = 1$ and $\lambda_3 = 1$, we can achieve the best retrieval performance using 64-d binary hash codes in TU-Berlin. It is worth noting that this setting was adopted in all our experiments unless otherwise stated.

Qualitative Analysis

Visualization of Similarity. In order to illustrate the superiority of this approach, we visualize the similarity of real-valued retrieval representations between sketches and images (see Figure 5). Here, we compute the cosine similarity for each class in Sketchy test set with the mean sketch representation and the mean image representation . The darkness of matrix color denotes the similarity score. As shown in Figure 5, the LFCN approach narrows the domain gap by establishing the cross-modal relations on local features, thereby improving the performance. However, the case of incorrect retrieval still exists. For instance, horse and deer have a great similarity in appearance, resulting in a higher score in the matrix relative to other category pairs.

Examples of Retrieval. We show the first 10 ZS-SBIR results obtained by our LFCN on the Sketchy and TU-Berlin datasets using 512-d real-valued retrieval representations in Figure 6 and Figure 7, respectively. Incorrectly retrieved images are highlighted in red border. From these examples, we can see that our proposed LFCN can effectively obtain the correct image based on the query sketch in most cases, and we provide two similar instances (the 1_{th} and 2_{th} row in Figure 6) from the same category "cup" that have the similar shape representations except for the difference at the bottom. Our LFCN method can distinguish these different representations and provide the corresponding required images. This indicates that our LFCN does establish the connection between the sketch and the image on local features. However, a high degree of matching between the retrieved image and the contour of the search sketch may lead to the retrieval of images with similar contour but different categories (the 5_{th} row in Figure 6). For instance, the retrieved

Table 1: The retrieval results comparing with previous ZS-SBIR methods. (The subscript b represents binary hash results and the rest represents real-valued results. Optimal and sub-optimal performance results are shown in bold and blue, respectively.)

Methods	Dimension	Sketchy		TU-Berlin	
		mAP@all	Prec@100	mAP@all	Prec@100
ZSIH$_b$ [24]	64	0.258	0.342	0.223	0.294
CVAE [14]	4096	0.196	0.284	0.005	0.001
SEM-PCYC [8]	64	0.349	0.463	0.297	0.426
SEM-PCYC$_b$ [8]	64	0.344	0.399	0.293	0.392
SAKE$_b$ [18]	64	0.364	0.487	0.359	0.481
CSDB [9]	64	0.376	0.484	0.254	0.355
PCMSN [5]	64	0.523	0.616	0.424	0.517
PCMSN$_b$ [5]	64	0.506	0.615	0.355	0.452
DSN [32]	64	0.484	0.610	0.442	0.538
DSN$_b$ [32]	64	0.436	0.553	0.385	0.497
LFCN(ours)	64	**0.606**	**0.701**	**0.479**	**0.565**
LFCN$_b$(ours)	64	**0.528**	**0.654**	**0.411**	**0.534**
OCEAN [36]	512	0.462	0.590	0.333	0.467
DSN [32]	512	0.583	0.704	0.481	0.586
DSN$_b$ [32]	512	0.581	0.700	0.484	0.591
SAKE [18]	512	0.547	0.692	0.475	**0.599**
LFCN(ours)	512	**0.613**	**0.721**	**0.490**	0.585
LFCN$_b$(ours)	512	**0.620**	**0.726**	**0.494**	**0.600**

Figure 6: Retrieval results on the Sketchy dataset.

Figure 7: Retrieval results on the TU-Berlin dataset.

results for 'bell' include four images of rifle. Since the local feature contrastive method of our LFCN still cannot establish a perfect correspondence between simple sketch and complex image, our method cannot deal with this situation well.

Table 2: The mAP@all results in our ablation studies under 64-d binary hash code setting. (Optimal performance results are in bold.)

Experiment #	Models	Sketchy	TU-Berlin
1	Baseline	0.499	0.383
2	Baseline+$\mathcal{L}_{lfcm}$	0.521	0.408
3	Baseline+$\mathcal{L}_{pl}$	0.520	0.405
4	Baseline+$\mathcal{L}_{lfcm}$ + $\mathcal{L}_{pl}$	**0.528**	**0.411**

4.5 Ablation Study

In order to verify the effectiveness of our LFCN method, we perform four ablation studies: 1) Baseline: A baseline [18] that uses cross-modal contrastive method and semantic knowledge learning module to learn a general embedding for sketch and image; 2) Baseline+$\mathcal{L}_{lfcm}$: Adding our local feature contrastive method to the baseline for mining cross-modal relations on local features of samples under the local feature contrastive loss $\mathcal{L}_{lfcm}$; 3) Baseline+$\mathcal{L}_{pl}$: Adding our feature prototype memory bank to the baseline to address the in-class diversity problem under the prototype loss $\mathcal{L}_{pl}$; 4) Baseline+$\mathcal{L}_{lfcm}$ + $\mathcal{L}_{pl}$: Full LFCN model. Moreover, all experiments were performed using 64-d binary hash codes for retrieval in ablation studies.

As shown in Table 2, our full model outperforms all baselines. The local feature contrastive method in experiment #2 improves baseline performance by 2.2% on Sketchy and 2.5% on TU-Berlin respectively, as this method is beneficial to reduce the domain gap. Add the feature prototype memory bank to the baseline can

boost baseline performance by 2.1% and 2.2% for each set, respectively. When combining all proposed modules, the full model obtains the highest mAP@all of 52.8% and 41.1% on Sketchy and TU-Berlin respectively. It is worth noting that the improvement with $\mathcal{L}_{lfcm} + \mathcal{L}_{pl}$ is not the simply sum of gains gained by $\mathcal{L}_{lfcm}$ and $\mathcal{L}_{pl}$. As far as we know, the reason may be that our local feature contrastive method has already partially addressed the in-class diversity problem in the sketch domain, which aggregates sketches with the same label in a training batch. The two approaches overlap in solving the in-class diversity problem, leading to the above phenomenon. Furthermore, the experiment results show that $\text{Imp}(\mathcal{L}_{lfcm}) > \text{Imp}(\mathcal{L}_{pl})$, where Imp represents the improvement from baseline, which proves our conjecture from another angle.

5 CONCLUSION

This paper presents an end-to-end framework, named Local Feature Contrastive Network (LFCN), to solve the zero-shot sketch-based image retrieval task more effectively. The framework improves performance by establishing the cross-modal relations to reduce the domain gap and learning sketch prototype to address the in-class diversity problem in the sketch domain. It creatively employs transformers to obtain similarity score on local features in cross-modal image retrieval scenario to narrow the domain gap. Additionally, our work reveals a new opportunity to transfer limited knowledge to unknown categories and reduce the potential risk of overfitting through assisted learning. Benefiting from the idea of assisted learning, we leverage the semantic information in word embeddings and the experience of teacher model in classification task to assist our model on ZS-SBIR. We stress that future efforts should be made to carry out further research in this direction.

REFERENCES

[1] Ayan Kumar Bhunia, Yongxin Yang, Timothy M Hospedales, Tao Xiang, and Yi-Zhe Song. 2020. Sketch less for more: On-the-fly fine-grained sketch-based image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9779–9788.

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European conference on computer vision*. Springer, 213–229.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.

[4] Kenneth Ward Church. 2017. Word2Vec. *Natural Language Engineering* 23, 1 (2017), 155–162.

[5] Cheng Deng, Xinxun Xu, Hao Wang, Muli Yang, and Dacheng Tao. 2020. Progressive cross-modal semantic network for zero-shot sketch-based image retrieval. *IEEE Transactions on Image Processing* 29 (2020), 8892–8902.

[6] Jia Deng. 2009. A large-scale hierarchical image database. *Proc. of IEEE Computer Vision and Pattern Recognition, 2009* (2009).

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 248–255. https://doi.org/10.1109/CVPR.2009.5206848

[8] Anjan Dutta and Zeynep Akata. 2019. Semantically tied paired cycle consistency for zero-shot sketch-based image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5089–5098.

[9] Titir Dutta and Soma Biswas. 2019. Style-Guided Zero-Shot Sketch-based Image Retrieval.. In *BMVC*, Vol. 2. 9.

[10] Mathias Eitz, James Hays, and Marc Alexa. 2012. How do humans sketch objects? *ACM Transactions on graphics (TOG)* 31, 4 (2012), 1–10.

[11] Weifeng Ge. 2018. Deep metric learning with hierarchical triplet loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 269–285.

[12] Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. 2012. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE transactions on pattern analysis and machine intelligence* 35, 12 (2012), 2916–2929.

[13] DP Kingma and LJ Ba. 2015. Adam: A Method for Stochastic Optimization. (2015).

[14] Sasi Kiran Yelamarthi, Shiva Krishna Reddy, Ashish Mishra, and Anurag Mittal. 2018. A Zero-Shot Framework for Sketch based Image Retrieval. In *The European Conference on Computer Vision (ECCV)*.

[15] Wei-Lin Ku, Hung-Chun Chou, and Wen-Hsiao Peng. 2015. Discriminatively-learned global image representation using CNN as a local feature extractor for image retrieval. In *2015 Visual Communications and Image Processing (VCIP)*. IEEE, 1–4.

[16] Hao Liu, Xinghua Jiang, Xin Li, Zhimin Bao, Deqiang Jiang, and Bo Ren. 2022. NomMer: Nominate Synergistic Context in Vision Transformer for Visual Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12073–12082.

[17] Li Liu, Fumin Shen, Yuming Shen, Xianglong Liu, and Ling Shao. 2017. Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2862–2871.

[18] Qing Liu, Lingxi Xie, Huiyu Wang, and Alan L Yuille. 2019. Semantic-aware knowledge preservation for zero-shot sketch-based image retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3662–3671.

[19] Hoang D Nguyen, Xuan-Son Vu, and Duc-Trong Le. 2021. Modular graph transformer networks for multi-label image classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 9092–9100.

[20] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. (2017).

[21] Qi Qian, Lei Shang, Baigui Sun, Juhua Hu, Hao Li, and Rong Jin. 2019. Softtriple loss: Deep metric learning without triplet sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6450–6458.

[22] Muhammad Ramadiansyah and Laksmita Rahadianti. 2020. Proxy-based Losses and Pair-based Losses for Face Image Retrieval. In *2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. IEEE, 177–186.

[23] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. 2016. The sketchy database: learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–12.

[24] Yuming Shen, Li Liu, Fumin Shen, and Ling Shao. 2018. Zero-shot sketch-image hashing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3598–3607.

[25] Hyun-Je Song, A-Yeong Kim, and Seong-Bae Park. 2020. Learning translation-based knowledge graph embeddings by N-pair translation loss. *Applied Sciences* 10, 11 (2020), 3964.

[26] Kaikai Song, Feng Li, Fei Long, Junping Wang, and Qiang Ling. 2018. Discriminative deep feature learning for semantic-based image retrieval. *IEEE Access* 6 (2018), 44268–44280.

[27] Fuwen Tan, Jiangbo Yuan, and Vicente Ordonez. 2021. Instance-level image retrieval using reranking transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 12105–12115.

[28] Xuan-Son Vu, Duc-Trong Le, Christoffer Edlund, Lili Jiang, and Hoang D Nguyen. 2020. Privacy-preserving visual content tagging using graph transformer networks. In *Proceedings of the 28th ACM International Conference on Multimedia*. 2299–2307.

[29] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. 2018. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5265–5274.

[30] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. 2017. Deep metric learning with angular loss. In *Proceedings of the IEEE international conference on computer vision*. 2593–2601.

[31] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. 2019. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5022–5030.

[32] Zhipeng Wang, Hao Wang, Jiexi Yan, Aming Wu, and Cheng Deng. [n. d.]. Domain-Smoothing Network for Zero-Shot Sketch-Based Image Retrieval. ([n. d.]).

[33] Zhijian Wu, Jun Li, Jianhua Xu, and Wankou Yang. 2021. Beyond ITQ: Efficient binary multi-view subspace learning for instance retrieval. *Journal of Visual Communication and Image Representation* 79 (2021), 103234.

[34] Xinxun Xu, Muli Yang, Yanhua Yang, and Hao Wang. 2021. Progressive domain-independent feature decomposition network for zero-shot sketch-based image retrieval. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*. 984–990.

[35] Sasi Kiran Yelamarthi, Shiva Krishna Reddy, Ashish Mishra, and Anurag Mittal. 2018. A zero-shot framework for sketch based image retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 300–317.

[36] Jiawen Zhu, Xing Xu, Fumin Shen, Roy Ka-Wei Lee, Zheng Wang, and Heng Tao Shen. 2020. Ocean: A dual learning approach for generalized zero-shot sketch-based image retrieval. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1–6.