Image Retrieval using Sketch+Text description

Revant Teotia (rt2819@columbia.edu)

Deep Learning for Computer Vision course project report

Abstract

In this project, I have developed a simple contrastive learning based image retrieval method which takes sketch and text description as a query and retrieves its corresponding image from an image database. For training and validation of the model, I have created a dataset of 90k query-target pairs using attribute annotations of the Visual Genome dataset. The model gives impressive retrieval results with 93.8% Recall@5 on 1k image database and 70.74% Recall@5 on 5k image database from dev set. Code at: https://github.com/revantteotia/sketch-text-image-retrieval

1 Introduction

Sketches are quite natural and effective way to explain a visual object. Recently sketch based image retrieval methods have gained some focus [5, 3, 1, 7] but none of these methods combine text with sketch and they only do sketch-to-image retrieval. Many properties of objects, like color(red, green), material (glass, leather) and texture (shiny, matte), are difficult to express with sketch alone. Adding text description along with sketch can enable more fine grained retrieval and improve image search. This work aims to develop a contrastive learning based deep learning retrieval model which can support sketch+text as query and is able to retrieve corresponding images from large image database.

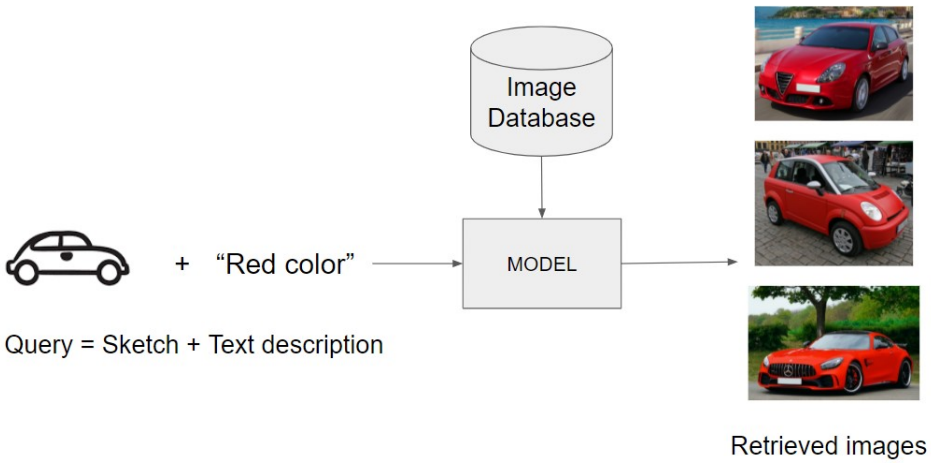

Figure 1: **Problem Statement.** Given a query consisting of a sketch and a text-description, the model aims to retrieve images from an image database.**[Best viewed in color].**

2 Problem statement

Given a Sketch of an object s and its text description d, the task of the retrieval model is to retrieve its corresponding target image t from an image database. For example in figure 1, a sketch of a car is given and its text description is provided as "Red color". The model has to retrieve image of red colored car from the image database containing many diverse images.

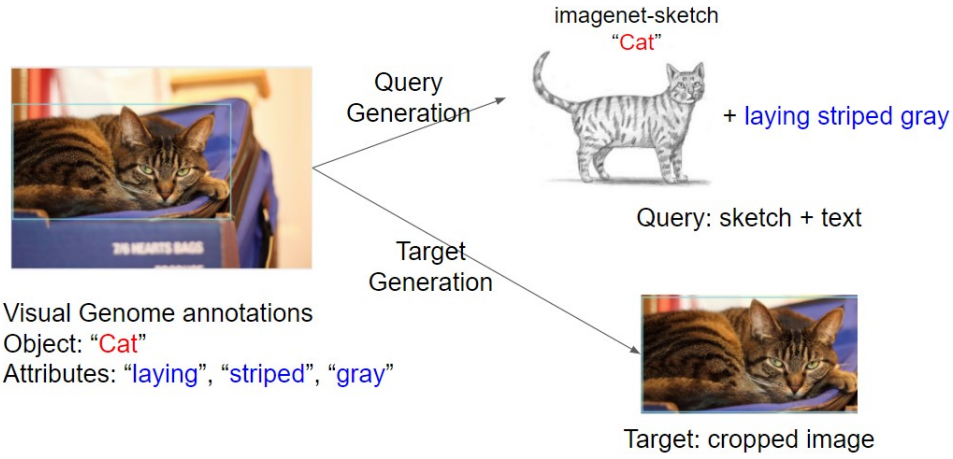

Figure 2: **Dataset creation.** Using attribute annotations in Visual Genome [4], query-target pairs are generated for training and evaluation. Cropped image is used as target, sketch from Imagenet-sketch [9] for the object category is used as sketch query and concatenated attributes is used as text query.**[Best viewed in color].**

3 Dataset Creation

To create the dataset for the reqtieval problem, I used attribute annotations of the Visual Genome [4] dataset. In Visual Genome attribute annotations, we are given a bounding box around the concerned object in the image and a list of its attributes. For example in figure 2, we are given a bounding box (in green) around the visual object cat and a list of attributes "laying", "striped" and "gray". To create query, I first got a sketch image of a cat from the Imagenet-sketch [9] dataset and then created its text description by concatenating all the attributes in a single sentence as "laying striped gray". To create the target image, I cropped the bounding box around the cat in the image and used the cropped image as target. This way I create one query-target pair from each attribute annotation. I used 20 kinds of object categories (cat, car, dog, umbrella, ...) and used 50 different sketches for each category from Imagenet-sketch. Used Visual Genome attribute annotations for more than 100k+ images. Thus

generated total of around 90,000 query-target samples after filtering those samples which had small cropped target image. Out of 90,000 samples, I used **5,000 for validation and remaining (around 85,000) for training.**

4 Method

I am approaching the retrieval as a ranking problem. The goal is to learn query feature and target feature in such a way that matching query-target are closer in the embedding space and non-matching pairs are farther. Once we learn to generate such embedding, during inference time when we get a query, we first get the query embedding and then find its similarity with all the target image embeddings in the image database and returned the most similar target images as the retrieved images . To encode the query sketch, I use a CNN (Resnet18) and to encode the query-text we use an LSTM. To get the whole query feature, I compose the sketch and text feature together by first concatenating them both and then using a MLP to encode the concatenated features. (just like in [8]). To encode the target image, I used the same CNN(Resnet18).

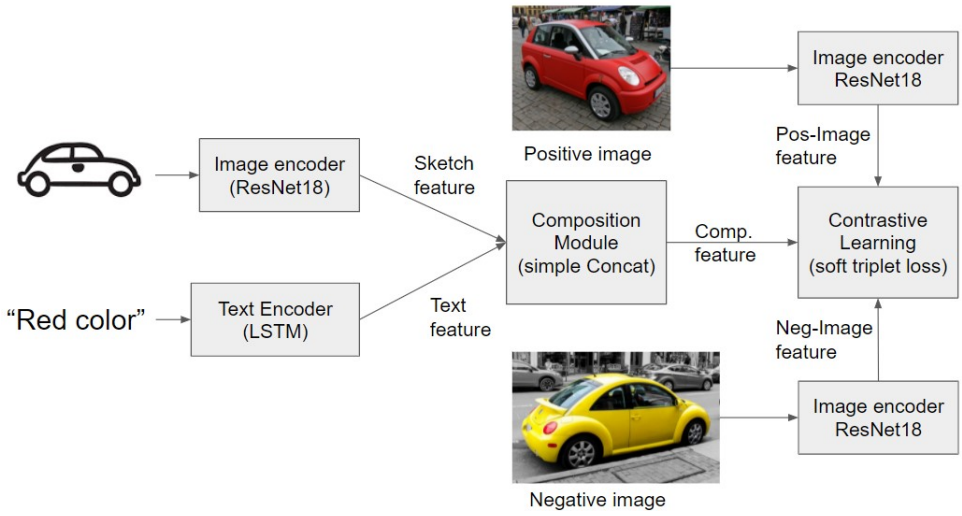

Figure 3: **Retrieval method using a contrastive learning approach.** For query feature, a CNN(ResNet18) is used to encode the sketch image and an LSTM is used to encode the text and finally are composed together to get query feature. For target feature, a CNN is used to encode the target images. Then Soft-triplet loss [6] is used to make the matching query-target features closer and non-matching ones farther in the embedding space.**[Best viewed in color].**

To make the model learn similarity, I used soft triplet loss [6], a contrastive learning approach, to make matching pairs closer and non-matching pairs far-

ther. To create triplets, I randomly created non-matching query-target pair for every matching query-target pair in a training batch and used negative of L_2 norm between normalized query-target embeddings as similarity. For example, in figure 3, model learns to bring the query embedding of car sketch + "red color" closer to the embedding of red colored car in positive-image and farther from the embedding of the yellow colored car in the negative image.

5 Results

The effectiveness of the model is measured in terms of Recall@k. Recall@k is defined as percentage of queries for which the ground-truth matching target in the gallery is in the top-k retrieved images. The dev set of 5k query-target pairs is used to measure the performance and a random subset of 1k out of 5k is also used to see how the model performs for a smaller gallery. In a dataset, all queries and targets are encoded using the learned model and then for each query all the dataset target images are ranked based on the similarity to calculate the recall measure. Smaller dataset of 1k is easier for the model than the larger dataset of 5k because the search space is less in the smaller dataset. The recall values are in table 1 and table 2

Model	Recall@1	Recall@5	Recall@10	Recall@50	Recall@100
Concat	63.7%	93.8%	98.7%	100%	100%

Table 1: **Retrieval performance on 1k subset of 5k dev set data.**

Model	Recall@1	Recall@5	Recall@10	Recall@50	Recall@100
Concat	35.52%	70.74%	82.66%	99.26%	99.98%

Table 2: **Retrieval performance on 5k dev set data.**

Qualitative results with example queries and the top-ranked images retrieved by the model are shown in figure 4 and figure 5.

6 Possible improvements

Although the model performs well on recall metrics and retrieves the target image in the top-ranked retrieved images, if we look at other images among the top retrieved ones, they sometime don't align with the query. For example in figure 4, the top two ranked images are of bears but 3rd rank image is of soup. To improve the quality of the overall retrieved image, few improvements can be done. First one is by increasing the diversity and size of the dataset. By increasing both the number of object categories (from 20 in current experiments) and the diversity of sketches, we can make the model generalize well on unseen test datasets. Secondly, we can increase the complexity of the model (in current experiments it is a simple Concat composition of sketch and text). And lastly, we can use different optimization methods (like SmoothAP [2]) which are meant for ranking/retrieval tasks.

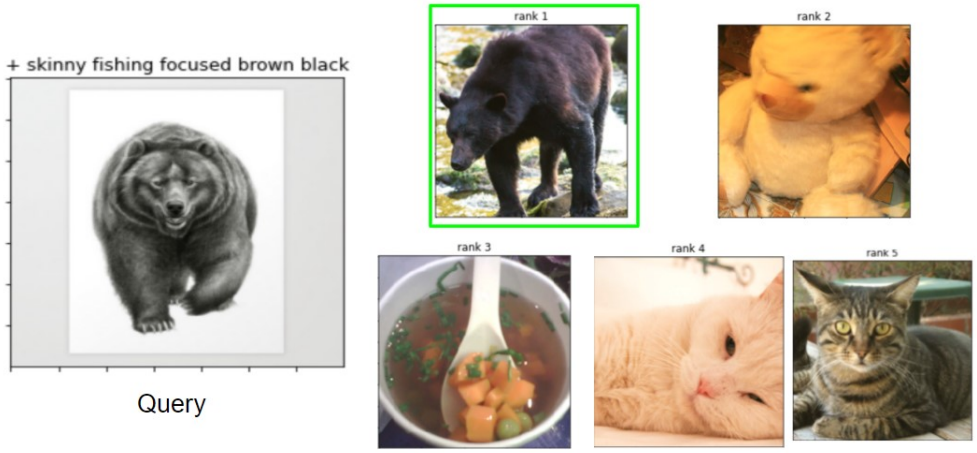

Figure 4: **An example of images retrieved** by mode from a gallery of 1000 dev set images. A bear sketch with text "skinny fishing focused brown black" is given on the left as query and the top ranked images are on the right hand side. Image with green bounding box is the ground truth target. **[Best viewed in color].**

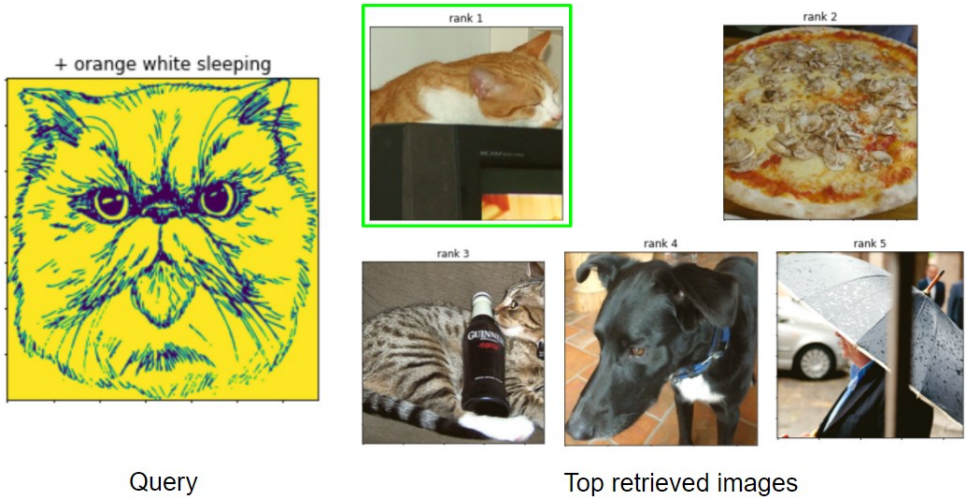

Figure 5: **An example of images retrieved** by mode from a gallery of 1000 dev set images. A cat sketch with text "orange white sleeping" is given on the left as query and the top ranked images are on the right hand side. Image with green bounding box is the ground truth target. **[Best viewed in color].**

References

[1] Ayan Kumar Bhunia, Yongxin Yang, Timothy M. Hospedales, Tao Xiang, and Yi-Zhe Song. 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 (CVPR)*, June 2020.

[2] Andrew Brown, Weidi Xie, Vicky Kalogeiton, and Andrew Zisserman. Smooth-ap: Smoothing the path towards large-scale image retrieval. In *European Conference on Computer Vision (ECCV)*, 2020.

[3] Sounak Dey, Pau Riba, Anjan Dutta, Josep Llados, and Yi-Zhe Song. Doodle to search: Practical zero-shot sketch-based image retrieval. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[4] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.*, 123(1):32–73, 2017.

[5] Li Liu, Fumin Shen, Yuming Shen, Xianglong Liu, and Ling Shao. Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

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

[7] Aneeshan Sain, Ayan Kumar Bhunia, Yongxin Yang, Tao Xiang, and Yi-Zhe Song. Stylemeup: Towards style-agnostic sketch-based image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8504–8513, June 2021.

[8] Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. Composing text and image for image retrieval-an empirical odyssey. In *CVPR*, 2019.

[9] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.