Partially Does It: Towards Scene-Level FG-SBIR with Partial Input. Supplemental material

Pinaki Nath Chowdhury[1,2] Ayan Kumar Bhunia[1] Viswanatha Reddy Gajjala*
Aneeshan Sain[1,2] Tao Xiang[1,2] Yi-Zhe Song[1,2]
[1]SketchX, CVSSP, University of Surrey, United Kingdom.
[2]iFlyTek-Surrey Joint Research Centre on Artificial Intelligence.
{p.chowdhury, a.bhunia, a.sain, t.xiang, y.song}@surrey.ac.uk

A. PyTorch-like pseudo-code to solve the linear programming problem using QPTH .

Algorithm 1: PyTorch code to compute flow $\hat{\mathcal{X}}$

```
import torch
# A differentiable QP solver for PyTorch
from qpth.qp import QPFunction

def compute_flow(u, v):
    # u:  Tensor of shape [nbatch,c,m]
    # v:  Tensor of shape [nbatch,c,n]
    nbatch, _, m = u.shape
    n = v.shape[2]

    # Objective Function in Eq.  4
    Q = 1e-3 * torch.eye(m*n).float()
    Q = Q.unsqueeze(0).repeat(nbatch,1,1)
    p = torch.bmm(u.permute(0,2,1), v)
    p = p.view(nbatch, m*n)

    # Inequality Constraint x_{i,j} ≥ 0
    G = -torch.eye(m*n).float()
    G = G.unsqueeze(0).repeat(nbatch,1,1)
    h = torch.zeros(nbatch, m*n)

    # Equality Constraint in Eq.  5
    A = torch.zeros(nbatch, m+n, m*n)
    for i in range(m):
        A[:, i, n*i:n*(i+1)] = 1
    for j in range(n):
        A[:, m+j, j::n] = 1
    s = get_weights(u, v) # (nbatch, m)
    d = get_weights(v, u) # (nbatch, n)
    b = torch.cat([s, d], dim=1)

    # flow X̂ shape:(nbatch,m,n) in Eq.  7
    flow = QPFunction()(Q,p,G,h,A,b)
    return flow.view(nbatch, m, n)

# Utility function for Eq.  5
def get_weights(a, b):
    node = a.shape[2]
    w = a*b.sum(dim=2).repeat(1,1,node)
    w = torch.relu(w.sum(dim=1)) + 1e-3
    return w
```

*Interned with SketchX

B. Additional Discussion

B.1 Why our proposed method outperform SceneSketcher, a method that uses bounding box annotation for scene graph matching?

Performance of graph based methods depend significantly on (1) graph construction step [5], and (2) graph matching loss used for a downstream task [3]. This hints at the bottleneck of graph based approaches – a sub-optimal graph, that is often constructed based on some heuristics (e.g., computing cosine distance of selected foreground regions [6]), might lead to sub-optimal performance. The graph matching metric used in SceneSketcher [6] has a remarkable similarity to that of Multiple Instance Learning [2], that computes a loss between the most similar pairs, but leaves the other pairs unconstrained. While one could adapt SceneSketcher using Gromov-Wasserstein distance [4], in this work, we advocate for a graph-free approach that do not need expensive bounding box annotations.

B.2 Why not train on partial sketches?

While training on partial sketches can *artificially* inflate retrieval performance during evaluation, the objective of this paper is to study robustness of scene-level FG-SBIR methods for partial or incomplete sketches – especially for scenes where the problem is most relevant, as shown in our pilot study in Sec. 1. In addition, the strategy used to mask local sketch regions can have significant effect on performance of the model [1]. Hence, instead of relying on tricks based on heuristics to improve performance, our objective is to propose a distance function which is implicitly robust to partial sketches with a well studied theoretical background that popular in the research community.

B.3 Understanding the dilemma between *fast*- and *slow*-retrieval:

There can be two major approaches to fine-grained image retrieval, a *fast*, and a *slow* retrieval: (i) In *fast* retrieval,

photos and sketches are embedded independently into a joint embedding space and then their similarities are compared. We pre-compute the feature vectors for each photo in the gallery independently, prior to having access of any query sketch. During inference, a single pass through the encoder is performed to embed the input sketch query to the joint sketch-photo embedding space. The resulting feature vector is then matched to its semantically similar photo using some distance function (usually euclidean or cosine distance [7,9]). Given n photos in the gallery set, one would spend $\mathcal{O}(1)$ forward pass through encoder network. (ii) On contrary, *slow*-retrieval models trade off compute time for accuracy gains. They explore the interactions between photos and query sketch *before* calculating similarities in the joint sketch-photo embedding space. Existing methods like Wang *et al*. [8], propose to adaptively control the information flow for message passing across modalities. However, a key limitation to adaptively updating sketch and photo features is that we can only compute *paired*-feature embedding that jointly represents similarity of a sketch-photo pair. Considering n photos in our gallery during inference, we have to compute the paired embedding of a given query sketch with each photo that needs $\mathcal{O}(n)$ forward pass through the network. For practical applications where n can be millions of photos, $\mathcal{O}(n)$ forward pass through a heavy neural network is intractable.

We propose a *mid-ground* between *fast*- and *slow*- retrieval. Instead of computing paired sketch-photo embedding, we propose to independently compute local-level feature maps for each sketch and photo. Our novel distance function, then *adaptively* computes region-wise features from sketch and photo using region-wise associativity that gives greater weightage to semantically similar local patches. Since we independently compute local-level features, during inference, our approach needs $\mathcal{O}(1)$ forward pass through a neural network. Although our simple trick can result in competitive performance to *slow*-retrieval models, storing local-level features increase the space complexity. Effective approaches in annealing space complexity could be an interesting direction of future research.

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 *CVPR*, 2020. 1

[2] Razvan C. Bunescu and Raymond J. Mooney. Multiple instance learning for sparse positive bags. In *ICML*, 2007. 1

[3] Liqun Chen, Zhe Gan, Yu Cheng, Linjie Li, Lawrence Carin, and Jingjing Liu. Graph optimal transport for cross-domain alignment. In *ICML*, 2020. 1

[4] Samir Chowdhury and Facundo Mémoli. The gromov–wasserstein distance between networks and stable network invariants. *Information and Inference: A Journal of the IMA*, 2019. 1

[5] Yihe Dong and Will Sawin. Copt: Coordinated optimal transport on graphs. In *NeurIPS*, 2020. 1

[6] Fang Liu, Changqing Zhou, Xiaoming Deng, Ran Zuo, Yu-Kun Lai, Cuixia Ma, Yong-Jin Liu, and Hongan Wang. Scenesketcher: Fine-grained image retrieval with scene sketches. In *ECCV*, 2020. 1

[7] Jifei Song, Qian Yu, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Deep spatial-semantic attention for fine-grained sketch-based image retrieval. In *ICCV*, 2017. 2

[8] Zihao Wang, Xihui Liu, Hongsheng Li, Lu Sheng, Junjie Yan, Xiaogang Wang, and Jing Shao. Camp: Cross-modal adaptive message passing for text-image retrieval. In *ICCV*, 2019. 2

[9] Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M Hospedales, and Chen Change Loy. Sketch me that shoe. In *CVPR*, 2016. 2