LaTeX Author Guidelines for CVPR Proceedings

Anonymous CVPR submission

Paper ID ****

Abstract

pass

1. Introduction

The importance of our research area

Some progress in sketch based image retrieval

Difficulty in Zero-shot setup and some possible solutions

Our proposed methods and their advantages

itemize our contributions in this paper

2. Related Work

2.1. Sketch-based image retrieval

2.2. Zero-Shot Learning

2.3. Cross-modal domain translation

3. Methodology

There will be five parts in this section. Sec. 3.1 defines the our targeted problem and briefly introduce our framework. Sec. 3.2 introduce the feature extractor. Sec. 3.3 introduce the parallel VAEs. Sec. 3.4 introduce the semantic preservation module. Sec. 3.5 introduce the design of loss functions during training procedure.

3.1. Problem Definition

In this paper, we focus on solving the problem of hand-free sketch-based image retrieval under zero-shot setup, where only the sketches and images from seen class are used during training stage. Our proposed framework is expected to use the sketches to retrieve the images, the categories of which have never appeared during training.

We first provide a definition of the SBIR in zero-shot setting. Given a dataset $S = \{(x_i^{img}, x_i^{ske}, x_i^{sem}, y_i) | y_i \in \mathcal{Y}\}$, where x_i^{img}, x_i^{ske}, x_i^{sem} and y_i are corresponding to the image, sketch, semantic representation and class label. Following the zero-shot setting in [**?**], we split all classes $\mathcal{Y}$ into $\mathcal{Y}_{train}$ and $\mathcal{Y}_{test}$ according to whether the label exists in ImageNet[**?**], where no overlap exists between two label set, i.e. $\mathcal{Y}_{train} \cap \mathcal{Y}_{test} = \emptyset$. Based on the partition of label set $\mathcal{Y}$, we split dataset into S_{train} and S_{test}. Our model need to disentangle structure representations of image using data in S_{train}. During test, given x^{ske} from S_{test}, our model need to retrieve several images from test images candidate.

Our goal is to learn a two-way map between image feature domain to sketch feature domain. To this end, we propose a new deep network (shown in Figure **??**), which contains feature extractor $F(\cdot)$, two structure encoders $\{E_s^{img}(\cdot), E^{ske}(\cdot)\}$, one appearance encoder $E_a^{img}(\cdot)$, two feature decoder $\{D^{img}(\cdot,\cdot), D^{ske}(\cdot)\}$ and a semantic decoder $S(\cdot)$. Note that by combining the encoders and decoders, our model can be regarded as two VAEs working parallel, which target to reconstruct sketch features from image and reconstruct image features from both sketches and images. To better preserve the semantics information within the sketches and images, we also add a semantic decoder to preserve semantics information while reconstructing the image features.

3.2. Feature Extractor

Considering the abstractness and visual sparsity of sketch, it is challenging to extract feature from sketch. To alleviate this issue, multi-channel and multi-scale model was proposed to extract more saint features [**?**]. Motivated by the visualization in [**?**], where different layers in the backbone model capture visual features at different levels, we build our feature extractor by combining different layers' features together to enrich feature representation capacity without adding any additional parameters.

In detail, we first use a pre-trained backbone model on ImageNet [**?**] to process each sketch and image, where we use VGG-16 as our backbone model in this paper. Suppose

$f_i \in \mathbb{R}^{C_i \times H_i \times W_i}$ is the output of the i-th convolution module and $f_{fc} \in \mathbb{R}^N$ is the feature of the last fully connected layer, the extracted feature $f^{\cdot}$ can be formulated as:

$$f^{\cdot} = F(x_i^{\cdot}) = [f_{fc}, GAP(f_5), GAP(f_4), GAP(f_3)], \tag{1}$$

where $GAP(\cdot)$ means global spatial average pooling and $[\cdot, \cdot]$ means the concatenation operation between vectors. Following the instruction in [?], a Principal Component Analysis algorithm is adopted to reduce the dimension of the extracted feature, which is helpful to not only improve the computational effiency but remove some redundant information.

3.3. Parallel VAE

Generative approaches have shown their power in ZS-SBIR [?, ?]. But both of these two papers, only encode the image in one encoder, which makes the appearance and the structure features fused together. When training the VAE based model, the decoder may ignore the condition information in sketch feature. Motivated by [?], we can regard image and sketch as two domains, where image domain have richer information than sketch domain. Our model is expected to learn a two-way mapping between this two domains. To this end, we devise a parallel VAE model, where the sketch VAE maps the image feature to the sketch domain and the image VAE maps the sketch domain to image domain with additional image appearance to complement the missing information.

3.3.1 Overall structure

In general case, a Variational Autoencoder (VAE) [?] map a prior distribution on a hidden latent variable p(z) to the data distribution p(x). The intractable posterior p(z—x) is approximated by the variational distribution q(z—x) which is assumed to be Gaussian. In our work, the sketch VAE is expect to map the image domain to the sketch domain. The variational distribution $q(z_i^{ske}|f_i^{img})$ of the sketch VAE are estimated from f_{img} via the image structure encoder E_s^{img} which is a neural network parameterized by θ_s^{img}. And the conditional $p(\hat{f}_i^{ske}|z_i^{ske})$ is modeled by the decoder network parameterized by ϕ_{skt}. Compare with the image, the sketch are obvious lack of image information, which makes it hard to reconstruct image features from sketch only. So the variational distribution $q(z^{img}|f_i^{img}, f_i^{skt})$ of the sketch VAE are estimated from both f_{img} and f_{skt} via the image structure encoder E_a^{img} and sketch encoder E^{ske} which is a neural network parameterized by θ_a^{img} and θ^{ske} respectively. And the conditional distribution $p(\hat{f}_i^{img}|z^{img}, f_i^{ske})$ is modeled by the decoder network parameterized by ϕ_{skt}.

3.3.2 Encoder

When building the image encoder and sketch encoder, both of them are excepted to receive enough information to fully represent their corresponding domain and learn some cross-modal representation between image domain and sketch domain. As we have mentioned that sketch can be regarded as the abstraction of a corresponding image. So the sketch hidden representation is designed to be the output of the image structure encoder and the image hidden representation is designed to be the combination of the output of image appearance encoder and sketch encoder. In detail, the sketch and image hidden representation is formulated as :

$$z_i^{ske} = E_s^{img}(f_i^{img}), \tag{2}$$
$$z_i^{img} = [E_a^{img}(f_i^{img}), E^{skt}(f_i^{ske})], \tag{3}$$
$$\tag{4}$$

3.3.3 Decoder

In this part, the decoders are designed to reconstruct their corresponding domain features condition on the hidden state and additional information. While, in fact, the sketch decoder is designed to make sketch hidden representation, i.e. the output of image structure encoder, more feasible to the sketch distribution. And on the other hand, the image decoder is designed to map the sketch domain to image domain for all categories including seen ones and unseen ones. So the sketch decoder is design to be a conditional decoder only condition on sketch hidden representation. For the image decoder part, the sketch feature f_i^{ske} is also added to the condition.

In detail, the reconstructed sketch and image is formulated as:

$$\hat{f}_i^{ske} = D^{ske}(z_i^{ske}), \tag{5}$$
$$\hat{f}_i^{img} = D^{img}(z_i^{img}, f_i^{ske}), \tag{6}$$

3.4. Semantics Preservation

Due to the large intra-class variance of sketches, simply using the parallel VAEs cannot overcome this. To tackle this issue, we add a semantic preserve module to ensure the image VAE to not only preserve the instance-level information with each sketch-image pair but also the category-level information of the training categories. Given the reconstructed image features $\hat{f}_i^{img}$, the semantics loss can be formulated as:

$$\hat{s}_i = S(\hat{f}_i^{img}), \tag{7}$$
$$\mathcal{L}^{sem} = D_{L2}(\hat{s}_i, s_i), \tag{8}$$

where the $\hat{s}_i$ is the predicted semantic representation and s_i is the

3.5. Loss Function

3.5.1 VAE loss

For the image VAE and sketch VAE, the evidence lower bound (ELBo) of them are:

$$\begin{aligned}\mathcal{L}_{ske}^{ELBo}(\theta_s^{img}, \phi^{ske}, f_i^{img}, f_i^{ske}) =\\ - D_{KL}(q(z_i^{ske}|f_i^{img})|p(z_i^{ske}))\\ + \mathbb{E}[\log p(\hat{f}_i^{ske}|z_i^{ske})]\end{aligned} \tag{9}$$

$$\begin{aligned}\mathcal{L}_{img}^{ELBo}(\theta_a^{img}, \theta^{ske}, \phi^{img}, f_i^{img}, f_i^{ske}) =\\ - D_{KL}(q(z_i^{img}|f_i^{img}, f_i^{ske})|p(z_i^{img}|f_i^{ske}))\\ + \mathbb{E}[\log p(\hat{f}_i^{img}|z_i^{ske}, f_i^{ske})]\end{aligned} \tag{10}$$

Furthermore, to encourage the model to preserve the latent alignments of the sketch and image, we add the reconstruction regularization to the objective:

$$\mathcal{L}_{ske}^{recon} = D_{Euclidean}(\hat{f}_i^{ske}, f_i^{ske}) \tag{11}$$

$$\mathcal{L}_{img}^{recon} = D_{Euclidean}(\hat{f}_i^{img}, f_i^{img}) \tag{12}$$

3.5.2 Orthogonality loss

To ensure the image structure encoder and appearance encoder different features of the image feature, we add a orthogonality loss between the output of them, which is formulated as:

$$\mathcal{L}_{\perp} = 1 - D_{cosine}(E_s^{img}(f_i^{img}), E_a^{img}(f_i^{img})) \tag{13}$$

3.5.3 Sketch loss

To better capture the sketch feature between intra-class and inner-class sketch-image pair, we add a marginal loss between positive pair $(z_{i,s}^{img}, z_i^{ske})$ and negative pair $(z_{i,s}^{img}, z_j^{ske})$, this can be formulate as:

$$Sim = D_{L2}(E_s^{img}(f_i^{img}), E^{skt}(f_i^{ske})), \tag{14}$$

$$Dis = D_{L2}(E_s^{img}(f_j^{img}), E^{skt}(f_i^{ske})), \tag{15}$$

$$\mathcal{L}_{dis} = \frac{1}{2}Sim + \frac{1}{2}\max(0, m - Dis), \tag{16}$$

where m is the margin.

So the overall loss is:

$$\begin{aligned}\mathcal{L} =& \lambda_1\mathcal{L}_{ske}^{ELBo} + \lambda_2\mathcal{L}_{img}^{ELBo} + \lambda_3\mathcal{L}_{ske}^{recon} + \quad (17)\\ & \lambda_4\mathcal{L}_{img}^{recon} + \lambda_5\mathcal{L}_{\perp} + \lambda_6\mathcal{L}_{dis} + \lambda_7\mathcal{L}^{sem}, \quad (18)\end{aligned}$$

where the λ_i are the weight for all the loss functions.

3.6. Training and Inference

3.6.1 Training

pass

3.6.2 Inference

At test stage, we divided our process into two parts, 1) the conditional decoder is used to synthesize a number of corresponding natural image features conditioned on test sketch features. Then the average of the synthesized features is obtained to represent the final synthesized image features $\hat{f}_{img}$; 2) the image structure encoder and the sketch encoder will map the image and sketch into a same space.

While retrieving the image, we computer the fusion cosine distance between image and sketch. The fusion cosine distance is defined as:

$$\begin{aligned}Dis_{fusion} =& \omega D_{cosine}(\hat{f}_i^{img}, f_i^{img}) +\\ & (1-\omega)D_{cosine}(h_i^{ske}, h_i^{img}),\end{aligned} \tag{19}$$

where h_i^{ske} and h_i^{img} are the output of $E^{ske}(\cdot)$ and $E_s^{img}(\cdot)$.

4. Experiment

4.1. Experiment Setup

4.1.1 Dataset

4.1.2 Implementation Details

4.2. Comparison

4.3. Ablation Study

4.4. Case study

5. Conclusion

6. To Discuss

- Whether to generator the whole image/sketch.
- If the poses between the image and the sketch are different, can the model learn the sketch information between image and sketch.
- Where to add the semantics information to further supervise the model's training.