Image Retrieval by Content

Discussion Proposal

Mengqing Jiang
2014013443
jmq14@mails.tsinghua.edu.cn

ZiZhao Zhang
2014013430
zzz_14@126.com

Pei Ye
2014013456
yep14@mails.tsinghua.edu.cn

1. INTRODUCTION

This is a discussion proposal about Image Retrieval by Content. We first present our review about Content Based Image Retrieval (CBIR). We focused most on the combination of deep learning and image retrieval. Then we found supervised binary hashing code transformed from the learning networks is newly proposed and practically proved to be efficient and effective. Finally we proposed the general design of our system and our key ideas.

2. RELATED WORK

According to our chosen task, we searched for and reviewed lots of works related to image retrieval. Some of them concluded the current situation of researches about image retrieval, and some of them raised new methods for more precise and effective content based image retrieval.

2.1 Current Situation of CBIRs

Nowadays, invention of the digital camera has given the common man the privilege to capture his world in pictures, and conveniently share them with others. One can today generate volumes of images with content as diverse as family get-togethers and national park visits. Low-cost storage and easy Web hosting has fuelled the metamorphosis of common man from a passive consumer of photography in the past to a current-day active producer. Today, searchable image data exists with extremely diverse visual and semantic content, spanning geographically disparate locations, and is rapidly growing in size. All these factors have created innumerable possibilities and hence considerations for real-world image search system designers.

In [2], Datta et al. pointed out that image search is much more difficult compared to text retrieval. One reason which causes this distinction is that text is mans creation, while typical images are a mere replica of what man has seen since birth, concrete descriptions of which are relatively elusive. They also concluded that usually there are five steps of image retrieval: Extraction of Visual Signature, Image Similarity Using Visual Signature, Clustering and Classification, Relevance Feedback-Based Search Paradigms, Multimodal Fusion and Retrieval. Advances have been made in both the derivation of new features (e.g., shape) and the construction of signatures based on these features, with the latter type of progress being more pronounced.

The term *signature* drew our attention because we are familiar with that signature is a common indexing method in text retrieval. How can we learn from signature in text retrieval and apply it to image retrieval to reduce the large memory and time consumption? Besides, what is the most advances feature extracting method nowadays?

We kept on our research to find the answer.

2.2 Image Retrieval Based on Deep Learning

Deep learning method gets more and more popular recently, and we can hardly ignore this novel and efficient technology. We also found that researches that bloomed in the past years suggest that the convolutional neural network (CNN) be in a leading position on feature extraction & representation for CBIRs.

The paper *Neural codes for image retrieval* [1], they provided a quantitative evaluation of the image retrieval performance of the features that emerge within the convolutional neural network trained to recognise Image-Ne classes. Then they focused evaluation on the performance of the compact versions of the neural codes, such as PCA compression.

In a word, they tested and evaluated the performance of the deep neural codes within the image retrieval application. They finally drew the conclusion that neural codes, namely deep learning methods, perform well, especially in terms of retrieval accuracy.

Interestingly, they also found some unexpected results. For instance, the best performance is observed not on the very top of the network, but rather at the layer that is two levels below the outputs. Their speculated that the reason is the excessive tuning of the very top layer.

Besides, they reached the conclusion that PCA compression works better for neural codes than than the one of VLADs, Fisher Vectors, or triangulation embedding. One possible

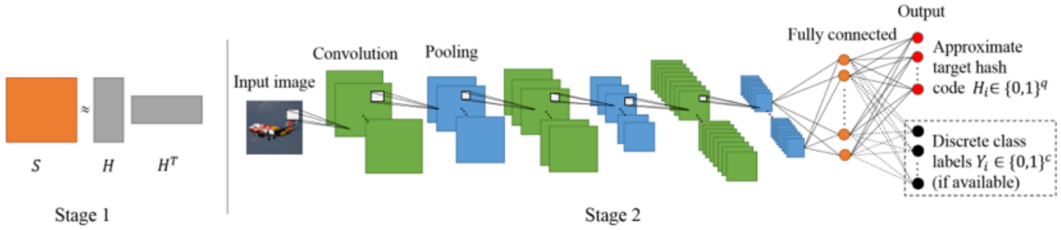
S
≈
H
H^T
Stage 1
Input image
Convolution
Pooling
Fully connected
Output
Approximate target hash code $H_i \in \{0,1\}^q$
Discrete class labels $Y_i \in \{0,1\}^c$ (if available)
Stage 2

Figure 1: Overview of the two-stage method of CNNH.

explanation is that passing an image through the network discards much of the information that is irrelevant for classification (and for retrieval).

2.3 Supervised Hashing for Image Retrieval

Previously, we mentioned that we have paid attention to how to design an efficient and effective signature as index for image retrieval, because the relatively well-performed features extracted from an image through Gist, SWIFT and CNN etc. are generally high-dimensioned. Obviously using original features as index can lead to poor performance in terms of storage, memory usage, and computational capacity etc.

Hash is a classic and efficient index method. In [6], Xia, Rongkai et al. proposed a supervised hashing method for image retrieval, in which they automatically learn a good image representation tailored to hashing as well as a set of hash functions.

The proposed method has two stages:

In the first stage, given the pairwise similarity matrix S over training images, as shown in Figure 1(Stage 1), they decomposed S into a product of HHT where H is a matrix with each of its rows being the approximate hash code associated to a training image. Then they minimised the reconstruction errors by calculating the distance between S and HH^T, adjusting and updating H repeatedly.

In the second stage, they propose to simultaneously learn a good feature representation for the input images as well as a set of hash functions, via a deep convolutional network tailored to the learned hash codes in H and optionally the discrete class labels of the images.

Here they mainly focused on the design of the output layer, so as to explore how to train a network tailored to the hashing task. They designed the output layer of the network in two ways, depending on whether the discrete class labels of the training images are available. In the first way, given only the learned hash code matrix H with each of its rows being a q-bit hash code for a training image, they defined an output layer with q output units (the red nodes in the output layer in Figure 1(Stage 2)), each of which corresponds to one bit in the target hash code for an image. They only used the learned hash code matrix H to train the module. In the second way, they assumed the discrete class labels of the training images are available. For the output layer in our network, we added c output units (the black nodes in the output layer in Figure 1(Stage 2)) which correspond to the class labels of a training images. They trained the module with both matrix H and Y. The incorporated image class labels are expected to be helpful for learning a more accurate image representation.

They denote the proposed hashing method using a CNN with such an output layer as CNNH. It is a classic supervised hashing method based on deep learning framework. Empirical evaluations in image search according to [6] showed that the proposed method has encouraging performance gains over state-of-the-arts.

However, the performance of CNNH has then been beaten by other optimised method also using both idea of deep learning and idea of hashing code.

2.4 Deep Learning of Binary Hashing Code

Previous hashing methods take low-level features as input and use shallow models to generate the hash codes, based on the prerequisite that the visual similarity is somehow embedded in the low-level feature space. Although CNNH and CNNH+ gain a great performance boost via leveraging deep models to learn representation and hash codes, they break the learning process into two separate stages and thus may reduce the co-adaptation, which can be of great importance for a high performance of CNN.

Recent studies[4] have shown that the feature activations of layers F68 of CNN induced by the input image can serve as the visual signatures.

In [3], Guo, Jinma and Li, Jianmin proposed a straightforward CNN-based hashing method, i.e. binarizing the activations of a fully connected layer with threshold 0 and taking the binary result as hash codes. They denoted their method as CNNBH.

Specifically, the model they use is the classic CNN targeted at classification, and after analysing and trying on different layers, they finally selected the first fully connected for hashing evaluation. To control the length of hash codes, another fully connected layer with no nonlinear transformation was inserted between.

Unlike CNNH, this model is suitable for single labelled image dataset, which means the training process is point-wise, not pair-wise. The similarity/dissimilarity matrix constructed with two discrete values that indicate similar or dissimilar is actually not necessary.

Similarly, another paper *Deep Learning of Binary Hash Codes for Fast Image Retrieval* [4] proposed an effective deep learn-

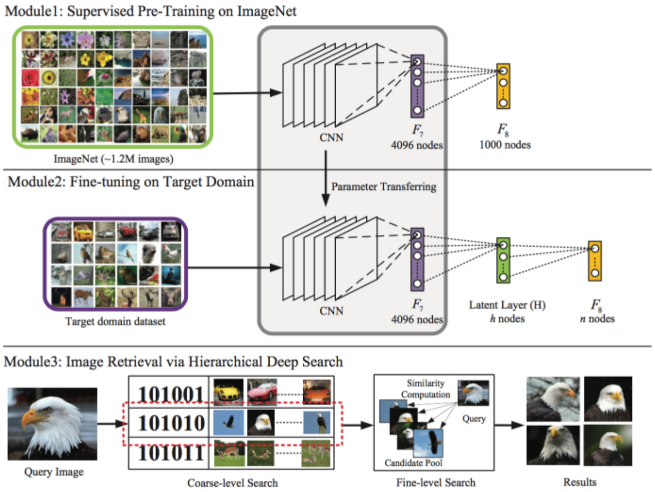

Figure 2: Image retrieval framework via hierarchical deep search.

ing framework to generate binary hash codes for fast image retrieval.

Their main idea is that when the data labels are available, binary codes can be learned by employing a hidden layer for representing the latent concepts that dominate the class labels, taking advantages of deep learning to achieve hashing, which is the same as the idea of [3].

The central difference is they embed the latent layer H between $F7$ and $F8$ as shown in the middle row of Figure 2. The latent layer H is a fully connected layer, and its neuron activities are regulated by the succeeding layer $F8$ that encodes semantics and achieves classification. The proposed latent layer H not only provides an abstraction of the rich features from $F7$, but also bridges the mid-level features and the high-level semantics. In our design, the neurons in the latent layer H are activated by sigmoid functions so the activations are approximated to $\{0, 1\}$.

Besides, this paper concretely described their image retrieval system based on a coarse-to-fine search strategy: firstly, retrieve a set of candidates with similar hidden binary activations from the latent layer. Then, to further filter the images with similar appearance, similarity ranking is performed based on the deepest mid-level image representations.

Experimental results show that, with only a simple modification of the deep CNN, their method improves the previous best retrieval results with 1% and 30% retrieval precision on the MNIST and CIFAR-10 datasets respectively.

2.5 What Works and What Doesn’t

Under the background that an industrial content-based image retrieval system (CBIRs) needs a fully consideration of feature extraction, feature processing and feature indexing, [5] raised several questions by observation in these three parts and pay most attention to the performance of convolutional neural network (CNN) in a real industrial CBIRs.

By experimental results and observations, they pointed out what works and what doesnt for using deep learning in image retrieval as follows:

1. When the level of features extracted by CNN is higher, they tend to be much more fitting the in-class data and as they appear to be with much less the generalization ability.

2. The cosine similarity is much better than the Euclidean similarity.

3. PCA can be applied to reduce the dimension of the extracted features, and there will be a very small loss in accuracy when a heavy reduction on the feature dimension applied.

4. The binarization is really effective for a real CBIR system with a small precision loss and the significantly reduced memory occupy and computing time. Besides, they found when the threshold is set to make the sparseness close to 50% after binarization, the loss of accuracy in CBIR tasks tend to be minimized correspondingly.

5. To build index on the data after binarization can be much more effective than to build index on the original data.

Their work and their approaches for experiment are worth understanding and using for reference when building a real CBIRs based on deep learning, especially CNNs. However, the methods mentioned and compared by experiments are limited and probably out-of-date.

3. GENERAL DESIGN OF OUR SYSTEM AND OUR KEY IDEAS

Our image retrieval system generally includes three main components.

The first component is the supervised pre-training CNN on a large-scale dataset, since the given training set is relatively small which the precision of the retrieval result may not be desired if starting bare-handed. According to research review, we found that ImageNet is a widely used large-scale dataset, and we can use it as pre-trained dataset.

The second component is fine-tuning the network with the latent layer to simultaneously learn domain-specific feature representation and transform the activation units into binary code as image signature. What we are not sure now is which layer to be chosen as the layer to transform the feature vector to binary code and how many dimension the hash code should have. Then such binary compact codes can be quickly compared using hashing or Hamming distance. However, since the database is relatively small and each category contains about 500 images in average, chances are that using classification information can raise the rate of recall and guarantee the accuracy of the results, with a little loss of efficiency. We will do experiments to find the best-performed method.

The third component is query and retrieval system. We will use the idea of progressive refinement. Each queried image will be extracted its high-dimensioned feature and binary hashing code. The coarse-level search is to compare Hamming distance of the binary hashing code or classification information between queried image and every image in database, with a threshold, which is to be decided after experiment. Then in the fine-level search, given the candidate pool by the coarse-level search, we use the features extracted from certain layer of CNN to identify the top k ranked images to form the candidate pool. The similarity level between the queried image and the i-th image in the pool can be defined in may ways such as the Euclidean distance and Cosine distance between their corresponding features vectors.

This is our preliminary design of our image retrieval system, and the key idea is extracting features by CNN and indexing by hashing code transformed by activation units of a certain layer.

4. REFERENCES

[1] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. Neural codes for image retrieval. In *Computer Vision–ECCV 2014*, pages 584–599. Springer, 2014.

[2] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)*, 40(2):5, 2008.

[3] J. Guo and J. Li. Cnn based hashing for image retrieval. *arXiv preprint arXiv:1509.01354*, 2015.

[4] K. Lin, H.-F. Yang, J.-H. Hsiao, and C.-S. Chen. Deep learning of binary hash codes for fast image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 27–35, 2015.

[5] H. Wang, Y. Cai, Y. Zhang, H. Pan, W. Lv, and H. Han. Deep learning for image retrieval: What works and what doesn't. In *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pages 1576–1583. IEEE, 2015.

[6] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan. Supervised hashing for image retrieval via image representation learning. In *AAAI*, volume 1, page 2, 2014.