11-2014

Deep learning for content-based image retrieval: A comprehensive study

Ji WAN
Chinese Academy of Sciences

Dayong WANG
Nanyang Technological University

Steven C. H. HOI
Singapore Management University, CHHOI@smu.edu.sg

Pengcheng WU
Nanyang Technological University

Jianke ZHU
Zhejiang University

See next page for additional authors

DOI: https://doi.org/10.1145/2647868.2654948

Published in MM '14: Proceedings of the ACM International Conference on Multimedia: November 3-7, 2014, Orlando. pp. 157-166.
http://doi.org/10.1145/2647868.2654948

Deep Learning for Content-Based Image Retrieval: A Comprehensive Study

Ji Wan[1,2,5], Dayong Wang[3], Steven C.H. Hoi[2], Pengcheng Wu[3],
Jianke Zhu[4], Yongdong Zhang[1], Jintao Li[1]
[1]Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing, China
[2]School of Information Systems, Singapore Management University, Singapore
[3]School of Computer Engineering, Nanyang Technological University, Singapore
[4]College of Computer Science, Zhejiang University, Hangzhou, China
[5]University of Chinese Academy of Sciences, Beijing, China
chhoi@smu.edu.sg; {dywang,wupe0003}@ntu.edu.sg; {wanji,zhyd,jtli}@ict.ac.cn; jkzhu@zju.edu.cn

ABSTRACT

Learning effective feature representations and similarity measures are crucial to the retrieval performance of a content-based image retrieval (CBIR) system. Despite extensive research efforts for decades, it remains one of the most challenging open problems that considerably hinders the successes of real-world CBIR systems. The key challenge has been attributed to the well-known "semantic gap" issue that exists between low-level image pixels captured by machines and high-level semantic concepts perceived by human. Among various techniques, machine learning has been actively investigated as a possible direction to bridge the semantic gap in the long term. Inspired by recent successes of deep learning techniques for computer vision and other applications, in this paper, we attempt to address an open problem: if deep learning is a hope for bridging the semantic gap in CBIR and how much improvements in CBIR tasks can be achieved by exploring the state-of-the-art deep learning techniques for learning feature representations and similarity measures. Specifically, we investigate a framework of deep learning with application to CBIR tasks with an extensive set of empirical studies by examining a state-of-the-art deep learning method (Convolutional Neural Networks) for CBIR tasks under varied settings. From our empirical studies, we find some encouraging results and summarize some important insights for future research.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; I.2.6 [**Artificial Intelligence**]: Learning; I.4.7 [**Image Processing and Computer Vision**]: Feature Measurement

General Terms

Algorithm; Experimentation

Keywords

Deep Learning; Content-Based Image Retrieval; Convolutional Neural Networks; Feature Representation

1. INTRODUCTION

The retrieval performance of a content-based image retrieval system crucially depends on the feature representation and similarity measurement, which have been extensively studied by multimedia researchers for decades. Although a variety of techniques have been proposed, it remains one of the most challenging problems in current content-based image retrieval (CBIR) research, which is mainly due to the well-known "semantic gap" issue that exists between low-level image pixels captured by machines and high-level semantic concepts perceived by human. From a high-level perspective, such challenge can be rooted to the fundamental challenge of Artificial Intelligence (AI), that is, how to build and train intelligent machines like human to tackle real-world tasks. Machine learning is one promising technique that attempts to address this grand challenge in the long term.

Recent years have witnessed some important advances of new techniques in machine learning. One important breakthrough technique is known as "deep learning", which includes a family of machine learning algorithms that attempt to model high-level abstractions in data by employing deep architectures composed of multiple non-linear transformations [5, 11]. Unlike conventional machine learning methods that are often using "shallow" architectures, deep learning mimics the human brain that is organized in a deep architecture and processes information through multiple stages of transformation and representation. By exploring deep architectures to learn features at multiple level of abstracts from data automatically, deep learning methods allow a system to learn complex functions that directly map raw sensory input data to the output, without relying on human-crafted features using domain knowledge. Many recent studies have reported encouraging results for applying deep learning techniques to a variety of applications, including speech recognition [16, 55], object recognition [26, 56], and natural language processing [19, 34], among others.

Inspired by the successes of deep learning, in this paper, we attempt to explore deep learning techniques with application to CBIR tasks. Despite much research attention of applying deep learning for image classification and recognition in computer vision, there is still limited amount of attention focusing on the CBIR applications. In this paper, we investigate deep learning methods for learning feature representations from images and their similarity measures towards CBIR tasks. In particular, we aim to address the following open research questions:

(i) Are deep learning methods effective for learning good feature representations from images to tackle CBIR tasks?

(ii) How much improvements can be achieved by deep learning techniques when compared with traditional features crafted by experts in multimedia and computer vision?

(iii) How to apply and adapt an existing deep learning model trained in one domain to a new CBIR task in another domain effectively?

In order to answer the above questions, we investigate a framework of deep learning for content-based image retrieval (CBIR) by applying a state-of-the-art deep learning method, that is, convolutional neural networks (CNNs) for learning feature representations from image data, and conduct an extensive set of empirical studies for a variety of CBIR tasks. From the empirical studies, we obtain some encouraging results and reveal several important insights for addressing the open questions. As a summary, we make the following major contributions in this work:

- We introduce a deep learning framework for CBIR by training large-scale deep convolutional neural networks for learning effective feature representations of images;
- We conduct an extensive set of empirical studies for comprehensive evaluations of deep convolutional neural networks with application to learn feature representations for a variety of CBIR tasks under varied settings.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 briefly introduces the framework of deep learning for CBIR. Section 4 introduces all the evaluated datasets and the adopted performance measurement. Section 5 shows the experimental results of our empirical studies. Finally, Section 6 discusses the limitations and concludes this paper.

2. RELATED WORK

Our research lies in the interplay of *content-based image retrieval*, *distance metric learning* and *deep neural network learning*. We briefly review each group of related work below.

2.1 Content-Based Image Retrieval

Content-based image retrieval (CBIR) is one of the fundamental research challenges extensively studied in multimedia community for decades [30, 25, 45]. CBIR aims to search for images through analyzing their visual contents, and thus image representation is the crux of CBIR. Over the past decades, a variety of low-level feature descriptors have been proposed for image representation [21], ranging from global features, such as color features [21], edge features [21], texture features [32], GIST [36, 37], and CENTRIST [49], and recent local feature representations, such as the bag-of-words (BoW) models [44, 54, 50, 51] using local feature descriptors (e.g. SIFT [31], and SURF [3], etc.). Conventional CBIR approaches usually choose rigid distance functions on some extracted low-level features for multimedia similarity search, such as Euclidean distance or cosine similarity. However, the fixed rigid similarity/distance function may not be always optimal to the complex visual image retrieval tasks due to the grand challenge of the semantic gap between low-level visual features extracted by computers and high-level human perceptions.

Hence, recent years have witnessed a surge of active research efforts in the design of various distance/similarity measures on some low-level features by exploring machine learning techniques [35, 7, 6]. Among these techniques, some works have focused on learning to hashing or compact codes [41, 35, 23, 57, 58]. For example, Norouzi et al [35] proposed a mapping learning scheme for large-scale multimedia applications from high-dimensional data to binary codes that preserve semantic similarity. Jegou et al [23] adopted the fisher kernel to aggregate local descriptors and adopted a joint dimension reduction in order to reduce an image to a few dozen bytes while preserving high accuracy. Another way to enhance the feature representation is distance metric learning (DML), as discussed in detail as follows.

2.2 Distance metric Learning

Distance metric learning for image retrieval has been extensively studied in both machine learning and multimedia retrieval communities [12, 2, 48, 29, 15, 47, 33, 46]. In the following, we briefly discuss different groups of existing work for distance metric learning organized by different learning settings and principles.

In terms of training data formats, most existing DML studies often work with two types of data (a.k.a. side information): *pairwise constraints* where must-link constraints and cannot-link constraints are given and *triplet constraints* that contains a similar pair and a dissimilar pair. There are also studies that directly use the class labels for DML by following a typical machine learning scheme, such as the Large Margin Nearest Neighbor (LMNN) algorithm [48], which however is not essentially different.

In terms of different learning approaches, distance metric learning techniques are typically categorized into two groups: the *global supervised approaches* [2, 18] that learn a metric on a global setting by satisfying all the constraints simultaneously, the *local supervised approaches* [48, 12] that learn a metric on the local sense by only satisfying the given local constraints from neighboring information.

In terms of learning methodology, most existing DML studies generally employ batch learning methods which often assume the whole collection of training data must be given before the learning task and train a model from scratch. Unlike the batch learning methods, in order to handle large-scale data, online DML algorithms have been actively studied recently [22, 24].

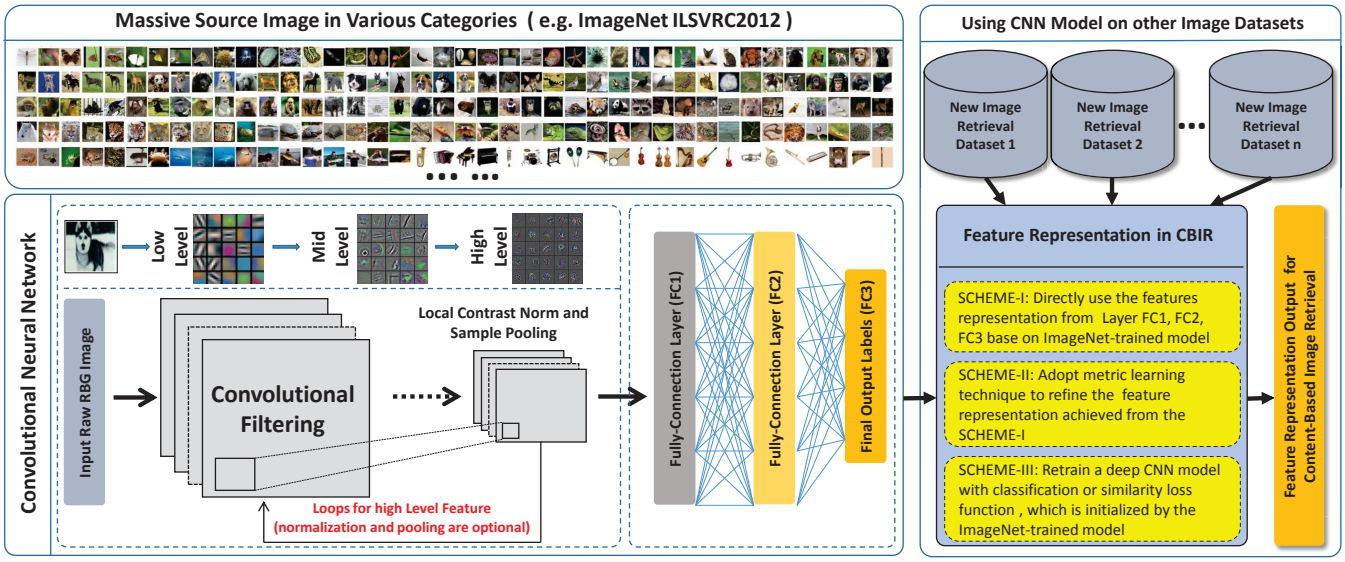
The key idea of distance metric learning is to learn an optimal metric which minimizes the distance between similar images and simultaneously maximizes the distance between dissimilar images. In this condition, another technique named similarity learning is closely related to distance metric learning. For example, Chechik et al. proposed an online algorithm for scalable image similarity (OASIS) [7] for improving image retrieval performance.

2.3 Deep Learning

Deep learning refers to a class of machine learning techniques, where many layers of information processing stages in hierarchical architectures are exploited for pattern classification and for feature or representation learning. It lies in the intersections of several research areas, including neural networks, graphical modeling, optimization, pattern recognition, and signal processing, etc.

Deep learning has a long history, and its basic concept is originated from artificial neural network research. The feed-forward neural networks with many hidden layers are indeed a good example of the models with a deep architecture. Back-propagation, popularized in 1980's, has been a well-known algorithm for learning the weights of these networks. For example, LeCun et al. [28] successfully adopt the deep supervised back-propagation convolutional network for digit recognition. Recently, it has become a hot research topic in both computer vision and machine learning, where deep learning techniques achieve stat-of-the art performance for various tasks. The deep convolutional neural networks (CNNs) proposed in [26] came out first in the image classification task of ILSVRC-2012 [1]. The model was trained on more than one million

[1] http://www.image-net.org/challenges/LSVRC/2012/



(a) Training deep CNN models in an existing domain (ImageNet) (b) Adopting trained model for CBIR in a new domain

Figure 1: A Framework of Deep Learning with Application to Content-based Image Retrieval.

images, and has achieved a winning top-5 test error rate of 15.3% over $1,000$ classes. After that, some recent works got better results by improving CNN models. The top-5 test error rate decreased to 13.24% in [43] by training the model to simultaneously classify, locate and detect objects. Besides image classification, the object detection task can also benefit from the CNN model, as reported in [14]. Generally speaking, three important reasons for the popularity of deep learning today are drastically increased chip processing abilities (e.g., GPU units), the significantly lower cost of computing hardware, and recent advances in machine learning and signal/information processing research.

Over the past several years, a rich family of deep learning techniques has been proposed and extensively studied, e.g., Deep Belief Network (DBN) [17], Boltzmann Machines (BM) [1], Restricted Boltzmann Machines (RBM) [42], Deep Boltzmann Machine (DBM) [40], Deep Neural Networks (DNN) [16], etc. More detailed survey of latest deep learning studies can be found in [11]. Among various techniques, the deep convolutional neural networks, which is a discriminative deep architecture and belongs to the DNN category, has found state-of-the-art performance on various tasks and competitions in computer vision and image recognition [28, 8, 10, 26]. Specifically, the CNN model consists of several convolutional layers and pooling layers, which are stacked up with one on top of another. The convolutional layer shares many weights, and the pooling layer sub-samples the output of the convolutional layer and reduces the data rate from the layer below. The weight sharing in the convolutional layer, together with appropriately chosen pooling schemes, endows the CNN with some "invariance" properties (e.g., translation invariance). We will discuss more details of the deep convolutional architecture in Section 3.1.

Our work is also related to some recent works in [13, 56]. Donahue et al. [13] evaluated whether features extracted from the activation of a deep convolutional network trained in a fully supervised fashion on a large, fixed set of object recognition tasks can be re-purposed to novel generic recognition tasks. For feature representation, they directly use the activation of the layer DeCAF_5, DeCAF_6 ("FC1" layer in our framework), and DeCAF_7 ("FC2" layer in our framework), and adopt LogReg or SVM to train a new classification model over the new dataset. Zeiler and Fergus [56] introduced a novel visualization technique that gives insight into the function of intermediate feature layers and the operation of the classifier. For feature representation, they keep the top $1-7$ layers of the ImageNet-trained model fixed and retrain a new softmax classifier on top using the training images in the new database. Our work is fundamentally different from these studies in that we focus on evaluating feature representation performance on CBIR tasks, where we aim to learn an effective distance measure for retrieval tasks instead of classifiers in recognition tasks. Finally, we note that our work is also very different from another recent study in [52] which aims to address multimodal image retrieval using deep learning and their raw input still rely on human-crafted features. By contrast, we aim to learn features directly from images without domain knowledge.

3. DEEP LEARNING FOR CBIR

In this section, we introduce the proposed deep learning framework for CBIR, which consists of two stages: (i) training a deep learning model from a large collection of training data; and (ii) applying the trained deep model for learning feature representations of CBIR tasks in a new domain. Specifically, for the first stage, we adopt the deep architecture of Convolutional Neural Networks (CNNs) as proposed in [26]. In the following, we first briefly introduce the basics of CNNs, and then focus on discussing how to generalize the trained deep models for feature representations in a new CBIR task.

3.1 Deep Convolutional Neural Networks

Figure 1 gives an overall view of the proposed framework by applying deep learning for CBIR tasks. For the implementation of deep CNNs learning, we follow the similar framework as discussed in [26] by adapting their publicly released C++ implementation [2]. This model has been successfully trained on the "ILSVRC-2012" dataset from ImageNet and found state-of-the-art performance with $1,000$ categories and more than 1 million training images.

[2] https://code.google.com/p/cuda-convnet/

In general, the deep convolutional network, as shown in Figure 1 (a), consists of two parts: 1) the convolution layers and max-pooling layers, and 2) the fully connection layers and the output layers. Specifically, the first layer is the input layer which adopts the mean-centered raw RGB pixels in intensity value. To reduce overfitting, two data augmentation tricks are performed: first, the input images are generated with translation and horizontal reflections by extracting random 224×224 patches from the original 256×256 images and our network is trained on these extracted patches; second, to capture the invariance in illumination and color, they add random multiples of the principle components of the RGB pixel values throughout the dataset. According to the authors [26], this scheme reduced their models' test set error by over 1%.

Following the input layers, there are five convolutional layers. The first and the second convolution layers are following with a response normalization layers and a max pooling layers, while the third, fourth, and fifth convolution layers are connected to one another without any intervening pooling or normalization. There are several novel or unusual features in Krizhevsky's convolutional neural network, which makes it work better than previous convolutional neural networks. First, the neuron output function f is the nonlinear function: Rectified Linear units (ReLUs), which can reduce the training time of the deep convolutional neural networks several times than the equivalents with "$tanh$"' units. Second, they adopt the "local response normalization", which is helpful for generalization. Last but not least, they adopt the "overlapping pooling" scheme. Max pooling layers are very common in general convolutional neural networks, which summarize the outputs of neighboring groups of neurons in the same kernel map. The max pooling step can enhance the transformation invariance of the feature mapping. Traditionally, the neighborhoods summarized by adjacent pooling units do not overlap. By adopting overlapped neighborhoods, they can reduce the top-1 and top-5 error rates by 0.4% and 0.3%, respectively.

Following the convolutional layers, there are two more fully-connected layers with $4,096$ neurons, denoted as "FC1" and "FC2". The last output layer, which is fed by the "FC2" layer, is a 1000-way softmax layer which produces a distribution over the $1,000$ class labels in ImageNet. In the whole deep convolutional neural network, there are about 60 million parameters in total. We train our deep convolutional neural network based on the ImageNet's ILSVRC-2012 training set, which contains about 1.2 million images. It takes about 200 hours to train a model with an error rate of 0.424 over the validation set ($50,000$ images), which is close to the error rate 0.407 reported in [26]. All the experiments were conducted on a Linux server with NVIDIA Tesla K20 GPUs. K20 is featuring 13 active SMXes along with 5 memory controllers and 1.25MB of L2 cache, attached to 5GB of GDDR5. In our experiments, the CNN model was trained on only one K20 GPU which has enough memory for our training tasks.

3.2 Feature Representation for CBIR

Although CNNs have been shown with promising results for classification tasks, it remains unknown how it can perform for CBIR tasks. In this paper, our goal is to investigate effective techniques by extending the trained models for learning feature representations in CBIR tasks. In particular, we want to address two open issues: (i) how to apply the trained CNNs from classification to CBIR tasks in ImageNet? and (ii) how to generalize the trained CNNs model in learning feature representation for CBIR tasks in a new domain, which may or may not have enough training data?

Specifically, to apply a trained CNNs model for direct feature representation, we take the activations of the last three fully connected layers (FC1, FC2, and FC3) as the feature representations for CBIR tasks. In our experiments, we denote the feature vector of this direct feature generalization as "DF.FC1", "DF.FC2", and "DF.FC3", respectively. DF.FC3 is the feature taken from the final output layer, DF.FC2 is the features taken from the final hidden layer, and DF.FC1 is the activations of the layer before DF.FC2. We do not evaluate features from lower convolutional layers in the network since the lower layers are unlikely to contain richer semantic representations than the later features which form higher-level hypothesis from the low-level to mid-level local information [13]. We note that the similar approach was also used for retraining classification models using other techniques (e.g., SVM) in some previous studies [13, 56].

The above direct feature generalization may work on the dataset used for training the CNNs model, but may not work well for CBIR tasks on a new dataset, as shown Figure 1 (b), which may be very different from the original training data set. In the following, we discuss three kinds of feature generalization schemes in detail.

3.2.1 *Scheme I: Direct Representation*

This is the direct feature representation as discussed above. We assume the retrieval domain is similar to the original dataset for training the CNN model. In this scenario, we will simply adopt one of the activation features DF.FC1, DF.FC2, and DF.FC3, directly. To obtain the feature representation, we directly feed the images in new datasets into the input layer of the pre-trained CNN model, and then take the activation values from the last three layers. Since we only need to compute the feedforward network based on the matrix multiplication for one time, the whole scheme will be very efficient. In our experiment, we also normalize the feature representation with l_2-norm.

3.2.2 *Scheme II: Refining by Similarity Learning*

Instead of directly using the features extracted by the pre-trained deep model, we attempt to explore similarity learning (SL) algorithms to refine the features in scheme I. Various distance metric learning or similarity learning algorithms can be used, as discussed in Section 2.2, according to the training data available in the new CBIR tasks. In our experiment, we adopt the stat-of-the-art online similarity learning algorithm: Online Algorithm for Scalable Image Similarity Learning ("OASIS") [7] for SL, which aims to learn a bilinear similarity measure over sparse representation.

Specifically, consider a triplet constraint set $\mathcal{X}$ given as follows:

$$\mathcal{X} = \{(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-) | (\mathbf{x}_i, \mathbf{x}_i^+) \in \mathcal{S}; (\mathbf{x}_i, \mathbf{x}_i^-) \in \mathcal{D}, i = 1, \dots, T\}$$

where $\mathcal{S}$ contains relevant pairs and $\mathcal{D}$ includes irrelevant pairs, and T denotes the cardinality of the entire triplet set. We denote the similarity function of two samples as a bilinear form:

$$S_W(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^\top W \mathbf{x}_j$$

where $W \in \mathbb{R}^{d \times d}$. We define the hinge loss for a triplet:

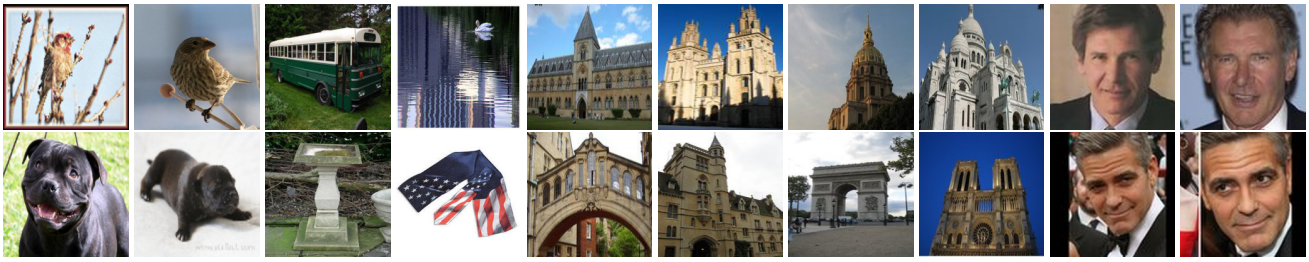
$$l_W(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-) = \max\{0, 1 - S_W(\mathbf{x}_i, \mathbf{x}_i^+) + S_W(\mathbf{x}_i, \mathbf{x}_i^-)\}$$

Hence the global loss L_W over all possible triplets in the training set can be computed as:

$$L_W = \sum_{(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-) \in \mathcal{X}} l_W(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-)$$

The OASIS algorithm aims to minimize the overall loss L_W using the same idea of the online Passive-Aggressive (PA) algorithm [9].

In our framework, we construct the triplets by simply considering relationships of instances in same class as relevant (positive),



(a) ImageNet (b) Caltech256 (c) Oxford (d) Paris (e) Pubfig83LFW

Figure 2: Example images of experimental datasets used in our experiments.

and relationships of instances belonging to different classes as irrelevant (negative). Scheme II is similar to the ones in [13]. However, they adopt the LogReg or SVM algorithm for classifier training and focus on object recognition. In the following experiments, we denote the feature representation of scheme II as "DF.FC1+SL", "DF.FC2+SL", and "DF.FC3+SL", respectively.

3.2.3 *Scheme III: Refining by Model Retraining*

Scheme III will retrain the deep convolutional neural networks on the new image dataset for different CBIR tasks by initializing the CNN model with the parameters of the ImageNet-trained models. Depend on the available label information, there are two ways to retrain the CNN model.

1) Refining with class labels. For datasets with class labels, we can retrain the model by optimizing the classification objective function. In this case, all layers of the new model will be initialized based on our ImageNet-trained model except the last output layer, which is adapted to the number of class labels of the new dataset and initialized randomly. Then, we update the whole convolutional neural networks by training on images from the new dataset. In our experiments, we denote the feature vector of this scheme as "ReCLS.FC1", "ReCLS.FC2", and "ReCLS.FC3", respectively.

2) Refining with side information. In some special real-world applications, obtaining the class information directly is expensive, but gathering the side information is easier. We can retrain the CNN model with similarity learning objective function, like what we do in scheme II, and back-propagate the errors to previous layers in order to refine the entire model on the new dataset. In our experiment, we adopt the online distance metric learning algorithm with cosine similarity proposed in [52]. In particular, denote $\mathbf{y} = \phi(\mathbf{x})$ the output of CNN model on input image $\mathbf{x}$, the cosine similarity of two input images $\mathbf{x}_1, \mathbf{x}_2$ is defined as:

$$S_{cos}(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{y}_1^\top \mathbf{y}_2 / (\|\mathbf{y}_1\| \times \|\mathbf{y}_2\|)$$

Given a training triplet input $(\mathbf{x}, \mathbf{x}^+, \mathbf{x}^-)$, we could define the hinge loss as follow:

$$l((\mathbf{x}, \mathbf{x}^+, \mathbf{x}^-); S_{cos}) = \max\{0, \gamma - S_{cos}(\mathbf{x}, \mathbf{x}^+) + S_{cos}(\mathbf{x}, \mathbf{x}^-)\}$$

where γ is the parameter of margin. Given $(\mathbf{y}, \mathbf{y}^+, \mathbf{y}^-)$ as the output achieved by the CNN network, the derivation of the l with respect to $\mathbf{y}, \mathbf{y}^+, \mathbf{y}^-$ can be computed individually. The entire CNN model can be updated by back-propagation algorithm following the same scheme in [52]. In our experiments, we denote the feature vector of this scheme as "ReDSL.FC1", "ReDSL.FC2", and "ReDSL.FC3", respectively.

Remark. We set a small learning rate for lower convolutional layers in order to preserve the original CNN model in lower-level feature layers. The original ImageNet-trained model has found a good starting point for the deep CNN model by optimizing a 1000-way classification problem, thus the good initialization will make the new model converge fast. Scheme III is similar to the ones in [56], which shows that the initialized CNN model could significantly outperform the one with random initializations.

4. IMAGE DATASETS

Our empirical studies aim to evaluate the performance of the three feature generalization schemes based on different image datasets, including the general image database "ImageNet", the object image database "Caltech256", the landmark image datasets "Oxford" and "Paris", and the facial image dataset "Pubfig83LFW". We briefly introduce each of them as follows.

ImageNet: is a large-scale dataset with over 15 million labeled high-resolution images belonging to roughly $22,000$ categories. The images were collected from the web and labeled by human labelers using Amazon's Mechanical Turk crowd-sourcing tool. Starting in 2010, as part of the Pascal Visual Object Challenge, an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) has been held. ILSVRC uses a subset of ImageNet with roughly $1,000$ images in each of $1,000$ categories. In all, there are roughly 1.2 million training images, $50,000$ validation images, and $150,000$ testing images. The Deep CNN model in our framework is trained based on ILSVRC2012.

Caltech256: contains $30,607$ images of objects, which were obtained from Google image search and from `PicSearch.com`. Images were assigned to 257 categories and evaluated by humans in order to ensure image quality and relevance.

Oxford: contains $5,063$ high resolution images automatically downloaded from Flickr. It defines 55 queries used for evaluation, which consists of 5 for each of the 11 chosen Oxford landmarks. It is quite challenging due to substantial variations in scale, viewpoint and lighting conditions.

Paris: is analogously to the previous "Oxford" datasets. Its $6,412$ images were obtained from Flickr, and there are also 55 queries for evaluation. As it contains images of Paris it is considered to be an independent dataset from "Oxford".

Pubfig83LFW: is an open-universe web facial image dataset [4], which combines two widely used face databases: PubFig83 [27] and LFW [20]. The 83 individuals from PubFig represent the test images and training gallery, and all the remaining individuals from LFW represent the distractor gallery or background faces. All the faces from each individual in PubFig83 are randomly divided into two-third training faces and one third test faces. All overlapping individuals from LFW are manually removed, and the left LFW dataset is used as distractors to PubFig83. All the facial images are resized to 250×250, and only the facial images that can be detected by a series commercial software are remained. In summary, the PubFig83+LFW dataset has 83 individuals with $8,720$ faces for training and $4,282$ faces for testing and over $5,000$ individuals from LFW with $12,066$ faces for background and distractor faces.

Table 1: Image Retrieval Performance on ImageNet

Feature	mAP	P@K=1	P@K=10	P@K=50	P@K=100	R@K=1	R@K=10	R@K=50	R@K=100
BoW.1200*	0.0007	0.0472	0.0234	0.0144	0.0118	0.0000	0.0002	0.0006	0.0009
BoW.4800*	0.0008	0.0487	0.0243	0.0148	0.0121	0.0000	0.0002	0.0006	0.0009
BoW.1K	0.0012	0.0243	0.0170	0.0124	0.0108	0.0000	0.0001	0.0005	0.0008
BoW.10K	0.0013	0.0212	0.0142	0.0106	0.0094	0.0000	0.0001	0.0004	0.0007
BoW.100K	0.0015	0.0216	0.0140	0.0110	0.0099	0.0000	0.0001	0.0004	0.0008
BoW.1M	0.0016	0.0286	0.0174	0.0132	0.0117	0.0000	0.0001	0.0005	0.0009
GIST	0.0002	0.0137	0.0080	0.0056	0.0049	0.0000	0.0001	0.0002	0.0004
DF.FC1	0.0748	0.4427	0.3650	0.2967	0.2637	0.0003	0.0028	0.0116	0.0205
DF.FC2	0.1218	0.4711	0.4105	0.3547	0.3254	0.0004	0.0032	0.0138	0.0254
DF.FC3	0.1007	0.4282	0.3726	0.3183	0.2898	0.0003	0.0029	0.0124	0.0226

* BoW.1200 and BoW.4800 are extracted by other groups, which are publicly available [3].

Table 2: Image Retrieval Performance on Caltech256

		Euclidean				Various Metric Learning on BoW*				OASIS on Deep Feature		
	Features	BoW*	DF.FC1	DF.FC2	DF.FC3	OASIS	MCML	LEGO	LMNN	DF.FC1	DF.FC2	DF.FC3
10 classes	mAP	0.2300	0.6424	0.7695	0.7109	0.3300	0.2900	0.2700	0.2400	0.8617	0.9153	0.8512
	P@K=1	0.3700	0.8800	0.9200	0.8800	0.4300	0.3900	0.3900	0.3800	0.9360	0.9560	0.9440
	P@K=10	0.2700	0.7748	0.8624	0.7928	0.3800	0.3300	0.3200	0.2900	0.9084	0.9400	0.8944
	P@K=50	0.1800	0.3614	0.4188	0.3930	0.2300	0.2200	0.2000	0.1800	0.4457	0.4614	0.4498
20 classes	mAP	0.1400	0.4984	0.5609	0.5493	0.2100	0.1700	0.1600	0.1400	0.6962	0.7388	0.6399
	P@K=1	0.2500	0.7840	0.7960	0.8020	0.2900	0.2600	0.2600	0.2600	0.8320	0.8780	0.7860
	P@K=10	0.1800	0.6228	0.6772	0.6588	0.2400	0.2100	0.2000	0.1900	0.7768	0.8130	0.7186
	P@K=50	0.1200	0.2998	0.3315	0.3290	0.1500	0.1400	0.1300	0.1100	0.3905	0.4084	0.3764
50 classes	mAP	0.0900	0.4011	0.4624	0.4286	0.1200	-	0.0900	0.0800	0.5186	0.5410	0.4603
	P@K=1	0.1700	0.6792	0.7240	0.6912	0.2100	-	0.1800	0.1800	0.7288	0.7320	0.6840
	P@K=10	0.1300	0.5333	0.5913	0.5519	0.1600	-	0.1300	0.1200	0.6194	0.6394	0.5600
	P@K=50	0.0800	0.2509	0.2836	0.2700	0.1000	-	0.0800	0.0700	0.3155	0.3308	0.2980

The results marked with $\star$ are taken directly from the study in [7].

5. EXPERIMENTS

In this section, we design an extensive set of experiments to evaluate the performance of deep learning techniques for CBIR tasks. Specifically, the first experiment is to examine how the deep CNN model performs for CBIR tasks on the same dataset that was used to train the model, and the rest experiments aim to test the generalization of the pre-trained deep model to CBIR tasks on other new domains, which may be very different from the training data used for training the original CNN models.

For performance evaluation metrics, we use three standard evaluation measures widely used in CBIR tasks, including the mean average precision (mAP), the precision at particular ranks ("P@K"), and the recall at particular ranks ("R@K").

5.1 Experiment on ImageNet

In this experiment, we aim to evaluate the CBIR performance using scheme I. We evaluate the retrieval performance on the ILSRVC 2012 dataset. We use the $50,000$ validation images as query set, and search on the 1.2-million training image set. We compare scheme I with several bag-of-words (BoW) feature representations, which are widely used for large-scale image retrieval. Among these BoW features, "BoW.1200" and "BoW.4800" are extracted by other groups [3]. The experimental results are shown in Table 1. Several image retrieval results on ImageNet dataset are shown in Figure 3.

[3] http://cloudcv.org/objdetect/

Several observations can be achieved from the results. Firstly, we can observe that this is a very challenging CBIR task. The best BoW feature representation based on a codebook with the vocabulary size $1,000,000$ can only achieve the mAP of 0.0016, and the performance of global GIST feature is even much worse. The results of "BoW.1200" and "BoW.4800" are generated based on the features released in other works, which is similar to the other BoW representations generated by ourselves.

Secondly, the activations based feature vector from the fully-connected layer FC1/FC2/FC3 achieve significantly much better results, among which the "DF.FC2" (the last hidden layer) achieved the best performance with top-1 precision of 47.11%. Although the last output layer DF.FC3 is the classification output of the ImageNet-trained CNN model, which contains the best sematic information, it seems not a good feature representation for CBIR tasks. By examining the standard evaluation measures: *precision* and *recall*, we can find the same observations. When $K = 1$, the P@K=1 of DF.FC2 is about 0.4711. It means the error rate of the nearest neighbor classification with $K = 1$ is 0.529, which is very close to the classification error rate of the ImageNet trained model (0.424). Finally, we note that our current experiments did not add extra post processing step to improve the CBIR performance, although some techniques (such as "geometric constraint based reranking" [38] or "query expansion" [53]) often can further boost the image retrieval performance, which however is out of the scope of this study.

Figure 3: Qualitative evaluation of image retrieval results on ImageNet. For each row, the image on the left-hand side is the query, and the images on the right are the top-5 returned results by each retrieval scheme.

5.2 Experiments on Various CBIR Tasks

In this section, we aim to evaluate the performance of feature representation schemes in Figure 1 (b) on new diverse CBIR tasks. Specifically, we examine the performance of the models for feature representations on three different CBIR tasks: (i) object retrieval tasks using the "Caltech256" dataset, which is an object-based image dataset, and different categories are quite distinct; (ii) landmark retrieval tasks using the "Oxford" and "Paris" datasets, which consist of landmark photos where all the images are captured under various conditions (scale, viewpoint and lighting conditions); and (iii) facial image retrieval tasks using the "Pubfig83LFW" dataset, which is challenging as intra class difference sometimes could be even larger than inter class difference.

5.2.1 *Object Image Retrieval*

In this experiment, we evaluate the retrieval performance of scheme I and scheme II on the Caltech256 dataset. Following exactly the same experimental setting in [7], we use subsets of sizes 10, 20, 50 classes, and images from each class were randomly split into a training set of 40 images and a test set of 25 image. Several kinds of distance metric learning algorithms are used to refine the original BoW feature vector (note that these experimental results were taken directly from the study in [7]). For scheme II, we adopt the OASIS algorithm for similarity learning. All the experimental results are shown in Table 2.

Several observations can be drawn from the results. Firstly, considering scheme I, the performances of DF.FC1/2/3 are consistently better than the general BoW representation. Similar to the previous experiment, the feature vector of the last hidden layer DF.FC2 achieved the best performance.

Second, similar to the previous experiment, based on the Caltech256 dataset, the output layer feature DF.FC3 generally performs better than the feature DF.FC1. Since the Caltech256 dataset is similar to the ImageNet dataset, and the difference between different categories is significant, the last layer feature that represents the final semantics information is also a good representation.

Thirdly, by adopting the similarity learning algorithm, the OASIS algorithm achieves the best performance, which can boost the mAP value from 0.23 to 0.33 for the 10-classes case. However, the performance of the refined BoW feature representation is still much lower than DF.FC2 in scheme I (0.7695). If we consider scheme II by refining the first feature generalization scheme with OASIS algorithm, the performance can be further improved, as shown in the last three columns in Table 2. Last but not least, we observe that after using similarity learning, the performance of DF.FC1 is better than that of the DF.FC3, especially for the harder problems with more categories (e.g. 50 classes). This shows that with DML, one can explore more useful information from the high-dimensional hidden layers than the final class output layers.

The overall encouraging results show that by adapting the pre-trained deep model on a new dataset using similarity learning, we can achieve significantly better performance than the original features.

5.2.2 *Landmark Image Retrieval*

In this experiment, we evaluate the retrieval performance of the three different feature representation schemes for landmark retrieval tasks on the "Oxford" and "Paris" datasets. Following the previous studies [38], we construct the BoW feature using a codebook with vocabulary size of $100K$ and $1M$, respectively. In additional, we take the experimental results on "Oxford" and "Paris" from [39] for comparison, which tried to use the features extracted from a CNN model in a different way. The experimental results are shown in Table 3 and Table 4. Since there are no explicit class label available in these datasets, we adopt the SL objective function in scheme III.

Several observations can be drawn from the results. Firstly, we observe that the BoW feature representation works well on both

Table 3: Image Retrieval Performance on "Oxford"

Feature	mAP	P@K=1	P@K=10	R@K=10
BoW-100K	0.5193	0.7636	0.6127	0.2764
BoW-1M	0.6044	0.8000	0.7073	0.3352
CNNaug-ss [39]	0.6800	-	-	-
DF.FC1	0.4170	0.8364	0.5273	0.2091
DF.FC2	0.3875	0.8364	0.4873	0.1810
DF.FC3	0.3347	0.7091	0.4309	0.1495
DF.FC1+SL	0.4658	0.4909	0.4909	0.1044
DF.FC2+SL	0.4441	0.5273	0.4836	0.1013
DF.FC3+SL	0.3019	0.4364	0.3818	0.0646
ReDSL.FC1	0.7834	0.8727	0.7291	0.2948
ReDSL.FC2	0.6770	0.7455	0.6309	0.2169
ReDSL.FC3	0.7332	0.8727	0.6927	0.2574

Table 4: Image Retrieval Performance on "Paris"

Feature	mAP	P@K=1	P@K=10	R@K=10
BoW-100K	0.5841	0.9455	0.8709	0.0711
BoW-1M	0.6298	1.0000	0.9491	0.0766
CNNaug-ss [39]	0.7950	-	-	-
DF.FC1	0.5808	0.9818	0.9200	0.0740
DF.FC2	0.6009	0.9636	0.9145	0.0739
DF.FC3	0.5168	0.9455	0.8836	0.0716
DF.FC1+SL	0.8683	0.9455	0.9582	0.0775
DF.FC2+SL	0.8479	0.9818	0.9600	0.0770
DF.FC3+SL	0.7007	0.8545	0.8873	0.0718
ReDSL.FC1	0.9474	0.9818	0.9727	0.0782
ReDSL.FC2	0.9122	1.0000	0.9655	0.0775
ReDSL.FC3	0.9233	1.0000	0.9473	0.0760

datasets, which is consistent with the previous studies. For example, based on a vocabulary with size 1 million, the mAP performances of BoW on the "Oxford" and "Paris" datasets are 0.6044 and 0.6298, respectively. Generally, the landmark images contain lots of useful interesting points, which are corresponding to the corner or edge information of the inside buildings. Hence, the BoW feature representation is a suitable solution for such kinds of datasets. The performances on the Paris dataset are generally better than those on the Oxford dataset.

Secondly, the feature representation of scheme I performs poorly on the two landmark datasets, and the performances of feature representations in different levels are different. In particular, the hidden layer features (DF.FC1 and DF.FC2) are quite comparable, and obviously better than the output layer feature (DF.FC3). By adopting the SL technique in scheme II, the retrieval performance can be improved on both datasets, especially for the Paris dataset.

Thirdly, for scheme III, the model retrained with SL objective achieves the best retrieval performance on both datasets. It indicates that the "deep" retraining step in scheme III is more effective to explore the hidden semantic information than the "shadow" similarity learning step in scheme II. This is reasonable since scheme III has the ability of refining the similarity measurements of the extracted feature and tuning the underlying feature representations simultaneously. However, the extra retraining cost of scheme III is also much higher than that of scheme II using similarity learning.

Finally, we compare our methods to the spatial search method with feature augmentation, named "CNNaug-ss", in [39]. From Table 3 and Table 4 we can see that "CNNaug-ss" is much better than scheme I, but worse than scheme III with SL objective. Compared with scheme II, "CNNaug-ss" is better on "Oxford" but worse on "Paris". Generally, "CNNaug-ss" has relatively better results on "Oxford" than on "Paris", since it explores sub-patches over multiple scales and the scale variation on Oxford dataset is larger. Exploring a large number of sub-patches over multiple scales makes "CNNaug-ss" computational expensive and limits its scalability.

5.2.3 *Facial Image Annotation*

In this experiment, we evaluate the search-based facial image annotation performance with the first and third feature generalization schemes on the Pubfig83LFW facial dataset. We conduct this experiment because the performance of search-based annotation scheme highly depends on the performance of content-based facial image retrieval. Specifically, we conduct the whole experiment based on the evaluation framework in [4], in which all the feature are evaluated based on the same settings. For feature representation, we use scheme I and scheme III with classification objective, since the class label information is available on this dataset. We compare them with a state-of-the-art feature representation, Hog-LBP-Gabor, which fuses three kinds of famous facial image representation features [4]. All the experimental results are shown in Table 5 and Fig. 4.

Several observations can be drawn from these results. First, similar to the previous experiments on the landmark datasets, scheme I does not work well on the facial image dataset, by comparing with the well-known facial image representation features. In particular, the best mAP value of scheme I (DF.FC1) is only 0.51.

Second, by using scheme III and retraining a new deep CNN model on the new facial image dataset, the performance of deep features can be significantly boosted. For example, the mAP of ReCLS.FC2 is about 0.81. The improvement over the regular Hog-LBP-Gabor feature is almost 23%, which indicate scheme III can considerably extract a set of suitable feature representation for new image retrieval task. In general, the intra-personal differences might be even larger than the inter-personal differences on facial image datasets, which also indicates that we need a higher level feature generalization scheme for facial image retrieval problem.

Finally, based on the KNN annotation method, we compare the Precision-Recall curves of scheme I and scheme III, as shown in Figure 4. We can see that the retrieval performance can be significantly boosted by adopting scheme III. This encouraging results again validate the good generalization performance of the CNN models for learning effective features in a new domain.

6. CONCLUSIONS

Inspired by recent successes of deep learning techniques, in this paper, we attempt to address the long-standing fundamental feature representation problem in Content-based Image Retrieval (CBIR). We aim to evaluate if deep learning is a hope for bridging the semantic gap in CBIR for the long term, and how much empirical improvements in CBIR tasks can be achieved by exploring the state-of-the-art deep learning techniques for learning feature representations and similarity measures. In particular, we investigate a framework of deep learning with application to CBIR tasks with an extensive set of empirical studies by examining a state-of-the-art deep learning method (convolutional neural networks) for CBIR tasks under varied settings.

Our encouraging results from the extensive empirical studies reveal that (i) deep CNN model pre-trained on large scale dataset

[4] http://goo.gl/QQUvjj

Table 5: The Recall (@Precision=95%) and mean Average Precision on the Pubfig83LFW Dataset.

	Hog-LBP-Gabor		DF.FC1(Best)		ReCLS.FC1		ReCLS.FC2		ReCLS.FC3	
Algorithm	Recall	mAP	Recall	mAP	Recall	mAP	Recall	mAP	Recall	mAP
Nearest Neighbor(NN)	0.249	0.655	0.330	0.514	0.434	0.790	**0.421**	**0.805**	0.259	0.733

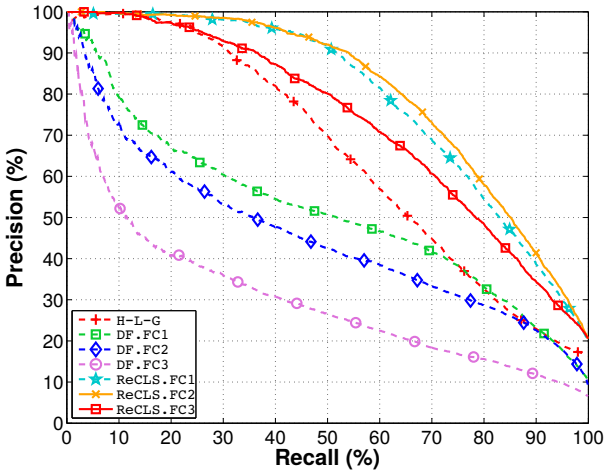


Figure 4: Precision-recall result on the Pubfig83LFW dataset.

can be directly used for features extraction in new CBIR tasks, the promising results on Caltech256 dataset demonstrate that the pre-trained model are able to capture high semantic information in the raw pixels; (ii) The features extracted by pre-trained CNN model may or may not be better than the traditional hand-crafted features, but with proper feature refining schemes, the deep learning feature representations consistently outperform conventional hand-crafted features on all datasets; (iii) When being applied for feature representation in a new domain, we found that similarity learning can further boost the retrieval performance of the direct feature output of pre-trained deep models; and (iv) Finally, by retraining the deep models with classification or similarity learning objective on the new domain, we found that the retrieval performance could be boosted significantly which is much better than the improvements made by "shallow" similarity learning. Despite encouraging results achieved, we believe this is just a beginning for deep learning with application to CBIR tasks, and there are still many open challenges. In future work, we will investigate more advanced deep learning techniques and evaluate more other diverse datasets for more in-depth empirical studies so as to give more insights for bringing the semantic gap of multimedia information retrieval in the long term.

7. ACKNOWLEDGEMENT

This work was supported by the National High Technology Research and Development Program of China (2014AA015202), the National Nature Science Foundation of China (61173054, 61428207), the National Key Technology Research and Development Program of China (2012BAH39B02).

8. REFERENCES

[1] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski. A learning algorithm for boltzmann machines*. *Cognitive science*, 9(1):147–169, 1985.

[2] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning distance functions using equivalence relations. In *ICML*, pages 11–18, 2003.

[3] H. Bay, T. Tuytelaars, and L. J. V. Gool. Surf: Speeded up robust features. In *ECCV (1)*, pages 404–417, 2006.

[4] B. C. Becker and E. G. Ortiz. Evaluating open-universe face identification on the web. In *CVPR Workshops*, pages 904–911, 2013.

[5] Y. Bengio, A. C. Courville, and P. Vincent. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR*, abs/1206.5538, 2012.

[6] H. Chang and D.-Y. Yeung. Kernel-based distance metric learning for content-based image retrieval. *Image and Vision Computing*, 25(5):695–703, 2007.

[7] G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 11:1109–1135, 2010.

[8] D. C. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *NIPS*, pages 2852–2860, 2012.

[9] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585, 2006.

[10] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, Q. V. Le, M. Z. Mao, M. Ranzato, A. W. Senior, P. A. Tucker, K. Yang, and A. Y. Ng. Large scale distributed deep networks. In *NIPS*, pages 1232–1240, 2012.

[11] L. Deng. A tutorial survey of architectures, algorithms, and applications for deep learning. *APSIPA Transactions on Signal and Information Processing*, 3:e2, 2014.

[12] C. Domeniconi, J. Peng, and D. Gunopulos. Locally adaptive metric nearest-neighbor classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(9):1281–1285, 2002.

[13] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *CoRR*, abs/1310.1531, 2013.

[14] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, abs/1311.2524, 2013.

[15] M. Guillaumin, J. J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *ICCV*, pages 498–505, 2009.

[16] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97, 2012.

[17] G. E. Hinton, S. Osindero, and Y. W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006.

[18] S. C. H. Hoi, W. Liu, M. R. Lyu, and W.-Y. Ma. Learning distance metrics with contextual constraints for image retrieval. In *CVPR (2)*, pages 2072–2078, 2006.

[19] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng. Improving word representations via global context and multiple word prototypes. In *ACL (1)*, pages 873–882, 2012.

[20] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[21] A. K. Jain and A. Vailaya. Image retrieval using color and shape. *Pattern Recognition*, 29(8):1233–1244, 1996.

[22] P. Jain, B. Kulis, I. S. Dhillon, and K. Grauman. Online metric learning and fast similarity search. In *NIPS*, pages 761–768, 2008.

[23] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(9):1704–1716, 2012.

[24] R. Jin, S. Wang, and Y. Zhou. Regularized distance metric learning: Theory and algorithm. In *NIPS*, pages 862–870, 2009.

[25] Y. Jing and S. Baluja. Visualrank: Applying pagerank to large-scale image search. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(11):1877–1890, 2008.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012.

[27] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, pages 365–372, 2009.

[28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[29] J.-E. Lee, R. Jin, and A. K. Jain. Rank-based distance metric learning: An application to image retrieval. In *CVPR*, 2008.

[30] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain. Content-based multimedia information retrieval: State of the art and challenges. *TOMCCAP*, 2(1):1–19, 2006.

[31] D. G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1150–1157, 1999.

[32] B. S. Manjunath and W.-Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(8):837–842, 1996.

[33] A. S. Mian, Y. Hu, R. Hartley, and R. A. Owens. Image set based face recognition using self-regularized non-negative coding and adaptive distance metric learning. *IEEE Transactions on Image Processing*, 22(12):5252–5262, 2013.

[34] T. Mikolov, W. tau Yih, and G. Zweig. Linguistic regularities in continuous space word representations. In *HLT-NAACL*, pages 746–751, 2013.

[35] M. Norouzi, D. J. Fleet, and R. Salakhutdinov. Hamming distance metric learning. In *NIPS*, pages 1070–1078, 2012.

[36] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[37] A. Oliva and A. Torralba. Scene-centered description from spatial envelope properties. In *Biologically Motivated Computer Vision*, pages 263–272, 2002.

[38] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007.

[39] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. *CoRR*, abs/1403.6382, 2014.

[40] R. Salakhutdinov and G. E. Hinton. Deep boltzmann machines. In *AISTATS*, pages 448–455, 2009.

[41] R. Salakhutdinov and G. E. Hinton. Semantic hashing. *Int. J. Approx. Reasoning*, 50(7):969–978, 2009.

[42] R. Salakhutdinov, A. Mnih, and G. E. Hinton. Restricted boltzmann machines for collaborative filtering. In *ICML*, pages 791–798, 2007.

[43] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229, 2013.

[44] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their localization in images. In *ICCV*, pages 370–377, 2005.

[45] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(12):1349–1380, 2000.

[46] D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. He, and C. Miao. Learning to name faces: a multimodal learning scheme for search-based face annotation. In *SIGIR*, pages 443–452, 2013.

[47] Z. Wang, Y. Hu, and L.-T. Chia. Learning image-to-class distance metric for image classification. *ACM TIST*, 4(2):34, 2013.

[48] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *NIPS*, 2005.

[49] J. Wu and J. M. Rehg. Centrist: A visual descriptor for scene categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(8):1489–1501, 2011.

[50] L. Wu and S. C. H. Hoi. Enhancing bag-of-words models with semantics-preserving metric learning. *IEEE MultiMedia*, 18(1):24–37, 2011.

[51] L. Wu, S. C. H. Hoi, and N. Yu. Semantics-preserving bag-of-words models and applications. *IEEE Transactions on Image Processing*, 19(7):1908–1920, 2010.

[52] P. Wu, S. C. H. Hoi, H. Xia, P. Zhao, D. Wang, and C. Miao. Online multimodal deep similarity learning with application to image retrieval. In *ACM Multimedia*, pages 153–162, 2013.

[53] H. Xie, Y. Zhang, J. Tan, L. Guo, and J. Li. Contextual query expansion for image retrieval. *IEEE Transactions on Multimedia*, 16(4):1104–1114, 2014.

[54] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo. Evaluating bag-of-visual-words representations in scene classification. In *Multimedia Information Retrieval*, pages 197–206, 2007.

[55] D. Yu, M. L. Seltzer, J. Li, J.-T. Huang, and F. Seide. Feature learning in deep neural networks - a study on speech recognition tasks. *CoRR*, abs/1301.3605, 2013.

[56] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. *CoRR*, abs/1311.2901, 2013.

[57] L. Zhang, Y. Zhang, X. Gu, J. Tang, and Q. Tian. Scalable similarity search with topology preserving hashing. *IEEE Transactions on Image Processing*, 23(7):3025–3039, 2014.

[58] Y. Zhang, L. Zhang, and Q. Tian. A prior-free weighting scheme for binary code ranking. *IEEE Transactions on Multimedia*, 16(4):1127–1139, 2014.