Fast content-based image retrieval using Convolutional Neural Network and hash function

Domonkos Varga*†, Tamás Szirányi*‡
*MTA SZTAKI, Institute for Computer Science and Control
{varga.domonkos, sziranyi.tamas}@sztaki.mta.hu
†Budapest University of Technology, Department of Networked Systems and Services
‡Budapest University of Technology, Department of Material Handling and Logistics Systems

***Abstract*—Due to the explosive increase of online images, content-based image retrieval has gained a lot of attention. The success of deep learning techniques such as convolutional neural networks have motivated us to explore its applications in our context. The main contribution of our work is a novel end-to-end supervised learning framework that learns probability-based semantic-level similarity and feature-level similarity simultaneously. The main advantage of our novel hashing scheme that it is able to reduce the computational cost of retrieval significantly at the state-of-the-art efficiency level. We report on comprehensive experiments using public available datasets such as Oxford, Holidays and ImageNet 2012 retrieval datasets.**

I. Introduction

The field of content-based image retrieval consists of a lot of different indexing structures, schemes, and methods aiding the related retrieval tasks. The goal of the indexing structures is to take an available dataset, and produce a concise and easier to handle index which can be used to search for similar content. In content-based image retrieval, both image representations and computational cost play an important and unavoidable role. Binary hashing has attracted a lot of attention due to computational and storage efficiencies of binary hash codes. It tries to map high-dimensional image data to compact binary codes in a Hamming-space while keeping several notion.

Due to the explosive increase of online images, rapid similarity search is critical for large-scale image retrieval. Benefiting from the compact binary codes, fast image search can be measured via Hamming distance and binary pattern matching, which significantly reduces the computational overhead and further optimizes the efficiency of the search. The success of deep learning techniques such as convolutional neural network (CNN) motivated us to explore its applications in providing compact binary codes directly. To address this problem, a deep semantic hashing algorithm is proposed in this paper, which based on CNN to learn semantic information and binary representations simultaneously.

The main contributions of our work are as following. An efficient end-to-end supervised learning framework is presented for fast image retrieval that learns probability-based semantic-level similarity and feature-level similarity simultaneously. Unlike previous methods [10], [11], the semantic from the last fully-connected layer is derived directly, instead of hash layer. Different from other supervised methods that learn an explicit hash function directly to map binary code features from images, our method learns hashing codes and image representations in an implicit manner. The main advantage of our novel hashing scheme that it is able to reduce the computational cost of retrieval significantly at the state-of-the-art efficiency level.

The rest of this paper is organized as follows. In Section II, the related and previous works are reviewed. We describe the proposed fast image-retrieval architecture in Section III. Section IV shows experimental results and analysis. We draw the conclusions in Section V.

II. Related works

The existing hash methods can be roughly divided into two categories: data-independent and data-dependent. In this section, we will mainly focus on data-dependent hash methods which are related to our algorithm.

A typical example of data-independent methods is Locality Sensitive Hashing (LSH) [1] that uses random projection to construct hash functions. LSH hashes the input items so that similar items map to the same cluster with high probability.

Havasi et al. [2] proposed the Local Hash-indexing tree which is similar to M-index [3] where base points are chosen randomly to reduce the high-dimensional feature vectors. Unlike M-index, during random point selection a quasi orthogonality criteria is forced.

Convolutional neural networks have achieved amazing success in modeling large-scale data recently. It have been demonstrated to be very effective in various computer vision and image processing tasks including pedestrian detection [4], face detection [5], image classification [6], image super-resolution [7], automatic image colorization [8] etc. CNN-based methods have been applied on the task of image retrieval recently.

The work of Babenko et al. [9] focuses on exploring the features of different layers, and to improve the retrieval performance with dimensional reduction. Xia et al. [10] presented an image retrieval method that uses a two-stage framework in order to accurately preserve the semantic similarities of image pairs. In the first stage, given the pairwise similarity matrix over training images a scalable coordinate descent method is proposed to decompose the similarity matrix. In the second stage, a feature representation is learned simultaneously for the input images as well as a set of hash functions. Zhao et al. [11] proposed a deep semantic ranking based method for learning

hash functions in order to preserve multilevel semantic similarity between multi-label images. CNNs were incorporated into hash functions to jointly learn feature representations and mappings from them to hash codes. A ranking list was applied to encode the multilevel similarity information. Wang et al. [12] proposed a fine-grained image similarity learning method that captures efficiently between-class and within-class image differences. A ranking loss based CNN architecture was proposed on triplet sampling to learn image similarity metric. Developing the convolutional architecture, Gong et al. [13] integrated into the CNN a warp approximate ranking algorithm.

III. Our approach

In general, a hash function h: $\mathbb{R}^D \rightarrow \{0,1\}$ is treated as a mapping that projects a D-dimensional input onto a binary code. Let's assume that we are given a set of images and their labels $I = \{(\mathbf{x}_n, \mathcal{Y}_n)\}$, where each image $\mathbf{x} \in \mathbb{R}^D$ is associated with a subset of possible labels $\mathcal{Y} \subseteq \mathcal{L}$. In previous works, the main objective was to learn a similarity function which mapped low-level image representation to a similarity value. Unlike previous methods [11], [12], [13], we obtain first the mid-level features from raw image data, then these mid-level features are used to estimate the probabilites of the semantic labels and to compute the hash codes of CNN feature vectors.

As shown in Figure 1, our algorithm consists of two main steps. In the first step, we obtain the image representations via a CNN which is supervised pre-trained on the `ImageNet` dataset [14] and fine-tuned on `target` dataset. The CNN model of Krizhevsky et al. [14] contains five convolutional layers, two fully-connected layers, and a softmax classifier. This model incorporates a huge amount of semantic information, since it was trained on more than 1 million images.

Unlike [11], the semantic from the last fully-connected layer is derived directly, instead of hash layer. The output of the last fully-connected layer is splitted into two branches. One branch leads to a n-ways softmax classifier where n stands for the number of categories of the target dataset. The other branch is a hash-like function to make the CNN features map to hash codes.

The mid-level features are extracted from the last fully-connected layer, and softmax classifiers are trained for each semantic simultaneously. We interpret the output of the classifiers as probabilites of semantic. The layers $FC6$ and $FC7$ are connected to the deep hash layer in order to encode a wide variety of information of visual appearance. In the following subsections we will define our feature vector and the computation of the hash function.

A. Probability-based Semantic-level Similarity

In case of hard assignment of semantic categories, we are given the semantic labels $\mathcal{L} = \{1, ..., C\}$ and the similarity between two images a and b are measured how much their indicator functions match. Let $\delta_i(a) \in \{0,1\}$ be the indicator function of image a has semantic i, so the semantic of image a can be denoted as $L_i^C(a) = \{\delta_i(a) | i = (1, ..., C)\}$, s.t. there must be one and only one $\delta_i(a) = 1$. We define the similarity between image a and b as $\varsigma(a,b) = \sum_{(i,j)} \delta_i(a) \mathbf{S}_{ij} \delta_j(b)$, where $\mathbf{S} \in \mathbb{R}^{C \times C}$ and S_{ij} is a matching score between semantic i and j.

As pointed out in [15], natural semantic categories always overlap and inherently ambiguous, using hard-categories to recognize objects always leads to failure. On the other hand, perfect classification of semantic is unrealistic. In order to solve this problem and improve performance, a probability-based semantic-level similarity is proposed in this paper.

Lets assume that image a has semantic label i, we can use probability $P(\delta_i(a) = 1|a)$ to indicate that image a has semantic i. Obviously, the semantic label of the image will be: $i = max_i(P(\delta_i(a) = 1|a))$. As we mentioned $\mathcal{L} = \{1, ..., C\}$ stands for the set of semantic labels and I denotes the set of images. For each $m \in \mathcal{L}$, a_m^+ denotes the set of images that are relevant to the semantic i, and a_m^- stands for the set of irrelevant. The semantic relevance is defined by the matrix $\mathbf{R}_{IL} : I \times L \rightarrow \mathbb{R}^+$, and where $\mathbf{R}_{IL}(a_m^+, m) > 0$ and $\mathbf{R}_{IL}(a_m^-, m) = 0$ for all $m \in \mathcal{L}$. We treated images as being conditionally independent: $P(a, b|m) = P(a|m)P(b|m)$ for any given image a, b and semantic m. The joint image-image probability can be computed as a relevance measure

$$P(a,b) = \sum_{m \in \mathbb{L}} P(a,b|m)P(m) = \\ = \sum_{m \in \mathbb{L}} P(a|m)P(b|m)P(m). \quad (1)$$

To improve scalability, we considered two images to be related if their joint distribution exceeded a cutoff threshold t. In short, we write:

$$\mathbf{R}_{II}(a,b) = [P(a,b)]_t, \quad (2)$$

where t denotes the cutoff threshold that is $[z]_t = z$ if $z > t$ otherwise 0.

B. Feature-level Similarity

Given an image a, we first extract the output of the fully-connected layers as image representation which is denoted by a D-dimensional feature vector $g(a)$, where $g(\cdot)$ is the convolution transformation over all of the previous layers. Then a q-bit binary code is obtained by a hashing function $h(\cdot)$. For each bit $i = 1, ..., q$, the output of the binary hash codes will be:

$$H_i = \begin{cases} 1 & \text{if } \sigma(x_i) - Mean(\sigma(x_i)) > 0 \\ 0 & \text{if } \sigma(x_i) - Mean(\sigma(x_i)) \leq 0, \end{cases} \quad (3)$$

$H_i \in \{0,1\}^q$ stands for the binary codes of each image I, $Mean(\mathbf{v})$ is the average value of vector $\mathbf{v}$. Given a query image x, we use its binary codes to identify the image from the images set I. We used the Euclidean distance to define feature-level similarity. The smaller the Euclidean distance is, the higher the similarity of the two images is, and top k ranked images are identified.

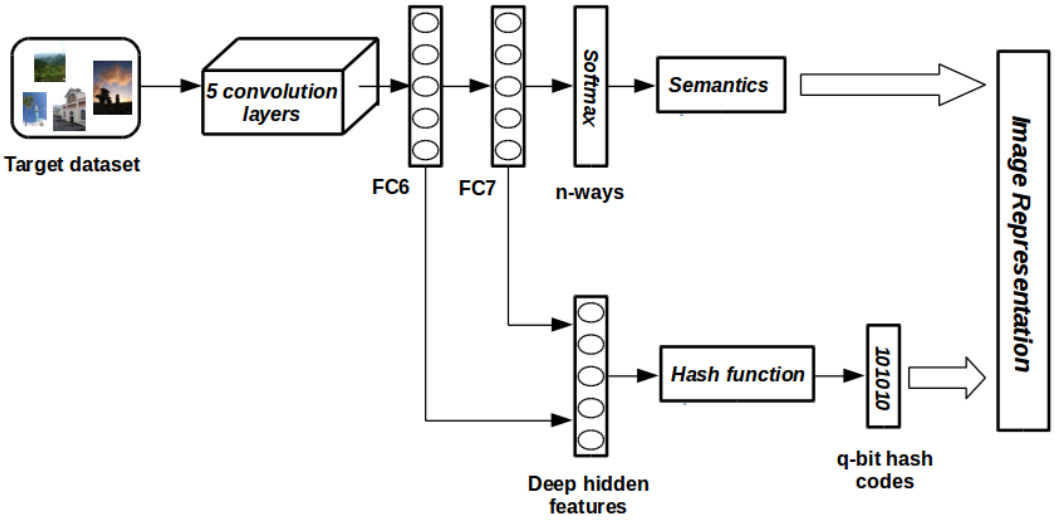

Fig. 1. The architecture of our proposed image retrieval algorithm.

TABLE II
THE DETAILED MEAN AVERAGE PRESCISION (MAP) RESULTS OF EACH CATEGORIES ON OXFORD DATASET. THE BEST RESULT OF EACH CATEGORIES IS TYPED BY **BOLD**, THE SECOND BEST IS TYPED BY *italic*. OUR ALGORITHM ACHIEVED BEST RESULT IN FOUR CATEGORIES AND SECOND BEST RESULT IN TWO CATEGORIES.

Method	souls	ashm	ball	bodle	christ	corn	hert	keble	magd	pitt	red	**Average**
Neural Codes [9]	0.566	0.486	0.439	0.802	0.597	0.385	0.936	0.434	0.285	**0.68**	0.958	0.605
Compressed Fisher [19]	0.581	0.513	0.424	0.813	0.604	**0.938**	0.503	0.234	0.79	0.597	0.98	0.634
Compressed BoW [20]	0.667	0.621	0.225	0.506	0.537	0.084	0.704	0.143	0.617	0.112	0.713	0.448
CVLAD [21]	0.536	0.425	0.120	0.587	0.544	0.231	0.621	0.109	0.474	0.310	0.754	0.428
OASIS [22]	0.286	0.686	*0.615*	0.178	0.598	0.292	0.135	0.713	0.761	0.677	0.818	0.523
MCML [23]	0.321	0.756	**0.685**	0.143	0.563	0.327	0.096	**0.748**	0.831	0.642	0.853	0.542
LEGO [24]	0.697	0.321	0.338	0.463	0.217	0.267	0.287	*0.579*	0.734	0.112	0.180	0.381
LMNN [15]	0.845	0.7	0.324	0.727	0.832	*0.450*	0.631	0.503	0.811	0.206	0.351	0.58
Spatial Pooling [25]	**1.0**	**0.945**	0.483	*0.956*	*0.923*	0.246	*0.99*	0.425	**1.0**	0.295	**1.0**	**0.751**
CNNaug-ss [26]	0.905	0.794	0.375	**0.959**	0.913	0.245	*0.99*	0.476	0.843	*0.679*	*0.99*	0.742
MOP-CNN [27]	0.963	0.917	0.521	0.802	0.833	0.29	**1.0**	0.44	0.739	0.28	**1.0**	0.707
Ours (t=0.2)	*0.968*	*0.939*	0.496	0.907	**0.936**	0.277	**1.0**	0.439	**0.913**	0.287	**1.0**	*0.744*

TABLE III
COMPARISON OF RETRIEVAL TIMES (MS) ON THREE RETRIEVAL DATASETS. THE BEST RESULT IS TYPED BY **BOLD**, THE SECOND BEST IS TYPED BY *italic*.

Method	Holidays	Oxford	ImageNet 2012
Number of Categories	500	12	1000
Neural Codes [9]	*0.16*	67.8	230.45
Compressed Fisher [19]	0.21	73.4	242.1
Compressed BoW [20]	0.2	69.3	232.7
CVLAD [21]	**0.15**	64.32	214.5
OASIS [22]	*0.16*	24.2	68.8
MCML [23]	**0.15**	30.1	75.5
LEGO [24]	0.19	*27.45*	89.5
LMNN [15]	**0.15**	27.9	*55.2*
Spatial Pooling [25]	2.1	145.1	390.6
CNNaug-ss [26]	1.3	122.2	370.6
MOP-CNN [27]	1.8	165.4	387.3
Ours (t=0.2)	**0.15**	**27.64**	**53.92**

C. Fusion of Semantic-level and Feature-level Similarity

After defining the semantic-level and feature-level similarity, we define hierarchical similarity between image a and b by:

$$\varsigma(a,b) = (p(a), H_a)^T \mathbf{S}(p(b), H_b) = \\ = \mathbf{R}_{II}(a,b) \times (1 - d(q,i)), \quad (4)$$

where $1 - d(q,i)$ is the feature-level similarity and $\mathbf{R}_{II}(a,b)$ was defined by Eq. 1 and 2. First, the semantic relevance $\mathbf{R}_{II}(a,b)$ is determined between the target and the query image, if it equals to zero the target image will be disregarded. After the semantic relevance checking, we obtain a set of candidate images. The feature-level similarity will be computed over this set.

TABLE I
MEAN AVERAGE PRECISION (MAP) COMPARISON WITH STATE-OF-THE-ART METHODS ON ImageNet 2012 DATASET MEASURED ON 100 CATEGORIES. THE BEST RESULT IS TYPED BY **BOLD**, THE SECOND BEST IS TYPED BY *italic*.

Method	MAP on ImageNet 2012
Neural Codes [9]	0.247
Compressed Fisher [19]	0.324
Compressed BoW [20]	0.305
CVLAD [21]	0.273
OASIS [22]	0.342
MCML [23]	0.315
LEGO [24]	0.142
LMNN [15]	0.183
Spatial Pooling [25]	**0.605**
CNNaug-ss [26]	0.563
MOP-CNN [27]	0.493
Ours (t=0.2)	*0.580*

IV. EXPERIMENTAL RESULTS

We used the Oxford [16] and Holidays datasets [17] to compare with other state-of-the-art algorithms and ImageNet 2012 [18] was used for large-scale experiments. The Oxford retrieval dataset contains 5062 images which were collected from Flickr by searching for particular Oxford landmarks. The collection was manually annotated to generate a comprehensive ground truth for 12 different landmarks (souls, ashm, ball, bodle, christ, corn, hert, keble, magd, pitt, red), each represented by 5 possible labels (Good, Ok, Bad, Junk).

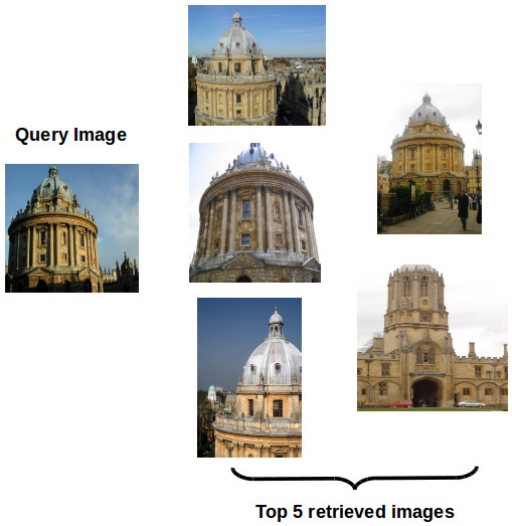

Fig. 2. Images retrieved by the proposed architecture.

We have used 1670 images as our validation dataset and the other 3392 images as training dataset. The amount of training images is very important for training a CNN. That is why we were forced to extend the number of images of each category to 1000 using among others horizontal flipping, rotation, and brightness transformations.

Figure 2 and Figure 3 illustrate some query results. In these figures, the query image can be seen on the left and we give the top 5 retrieved images to each query image on the right. In the following, we give qualitative results in order to prove that our architecture is good for preserving the semantics which makes the target images similar to the query image.

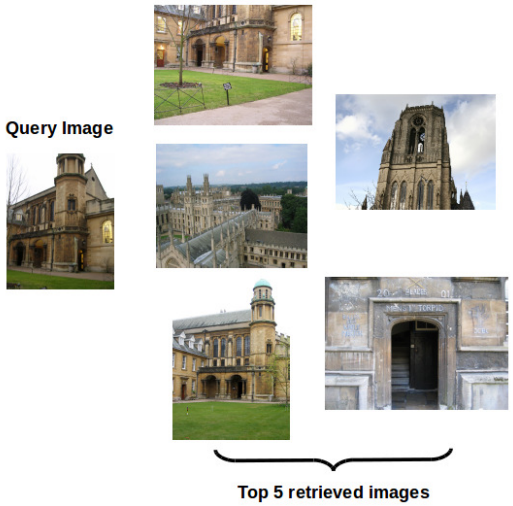

Fig. 3. Images retrieved by the proposed architecture.

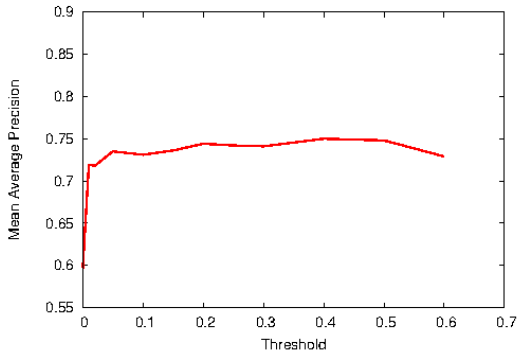

Fig. 4. Combining Probability-based Semantic-level Similarity with different thresholds. Measured on Oxford dataset.

Figure 4 shows the mean average precision curves on Oxford dataset with different threshold t. As we described above, higher thresholds will filter out more semantic dissimilarities, but it may exclude similar categories, especially for complex scenes. The published curve shows that $t = 0.2$ is a good choice. That is why we used everywhere $t = 0.2$ threshold during the evaluation of our algorithm.

Table II summarizes the comparison with other state-of-the-art methods with respect to Mean Average Prescision measured on Oxford dataset. It can be seen that our method produces competitive results in comparison to other CNN-based approaches (Spatial Pooling, CNNaug-ss, MOP-CNN) and significantly outperforms most of the SIFT-based approaches (Compressed Fisher, Compressed BoW, CVLAD, OASIS, LEGO). Due to the proposed architecture and special localization of hash function, we outperform in retrieval times the other state-of-the-art algorithms (Table III).

Finally, we measured the presision on ImageNet 2012

dataset in order to demonstrate the efficiency and scalability of our algorithm. Table I summarizes the obtained results. It can be seen that our method gives competitive results in comparison to other CNN-based approaches (Spatial Pooling, CNNaug-ss, MOP-CNN).

V. Conclusion

We have introduced a novel end-to-end supervised learning framework that learns probability-based semantic-level similarity and feature-level similarity simultaneously. We reported on competitive results using public available datasets including `Oxford`, `Holidays` and `ImageNet 2012` retrieval datasets. Semantic-level similarity and feature-level similarity were combined combined in the retrieval process and this layout provided strong priors to determine similarity distance effectively. We showed if efficieny and speed considered together our method outperforms the state-of-the-art results.

Acknowledgment

The research was supported by the Hungarian Scientific Research Fund. We are very thankful to Levente Kovács for helping us with professional advices in high-performance computing.

References

[1] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. *VLDB*, **99**(6): 518–529, 1999.

[2] L. Havasi, D. Varga, and T. Szirányi. LHI-tree: An efficient disk-based image search application. *International Workshop on Computational Intelligence for Multimedia Understanding*, 1–5, 2014

[3] G. Amato and P. Savino. Approximate similarity search in metric spaces using inverted files. *Proceedings of the 3rd International Conference on Scalable Information Systems*, 1–10, 2008

[4] P. Luo, Y. Tian, X. Wang, and X. Tang. Switchable deep network for pedestrian detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 899–906, 2014

[5] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua. A convolutional neural network cascade for face detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5325–5334, 2015

[6] D. Ciresan, U. Meier, X. Shen, and J. Schmidhuber. Multi-column deep neural networks for image classification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3642–3649, 2012

[7] C. Dong, C. Loy, X. Shen, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. *Proceedings of the European Conference on Computer Vision*, 184–199, 2014

[8] Z. Cheng, Q. Yang, and B. Sheng. Deep Colorization. *Proceedings of the IEEE International Conference on Computer Vision*, 415–423, 2015

[9] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. Neural codes for image retrieval. *Proceedings of the European Conference on Computer Vision*, 584–599, 2014

[10] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan. Supervised Hashing for Image Retrieval via Image Representation Learning. *AAAI*, **1**: 2156–2162, 2014.

[11] F. Zhao, Y. Huang, L. Wang, and T. Tan. Deep Semantic Ranking Based Hashing for Multi-Label Image Retrieval. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1556–1564, 2015.

[12] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1386–1393, 2014.

[13] Y. Gong, Y. Jia, T. Leung, A. Toshev, and S. Ioffe. Deep convolutional ranking for multilabel image annotation. *CoRR*, abs/1312.4894, 2013.

[14] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 1097–1105, 2012.

[15] K. Weinberger and L. Saul. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, **10**: 207–244, 2009.

[16] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–8, 2007.

[17] H. Jegou, M. Douze, and C. Schmid. Hamming Embedding and Weak geometry consistency for large scale image search. *Proceedings of the European Conference on Computer Vision*, 304–317, 2008.

[18] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, and M. Bernstein. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, **115**(3): 211–252, 2015.

[19] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier. Large-scale image retrieval with compressed fisher vectors. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3384–3391, 2010.

[20] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3304–3311, 2010.

[21] W. Zhao, H. Jégou, and G. Gravier. Oriented pooling for dense and non-dense rotation-invariant features. *Proceedings of the British Machine Vision Conference*, 2013

[22] G. Chechik, U. Shalit, S. Bengio, S. Sonnenburg, V. Franc, E. Yom-tov, and M. Sebag. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 1109–1135, 2010

[23] A. Globerson and S. Roweis. Metric learning by collapsing classes. *Advances in neural information processing systems*, 451–458, 2005

[24] P. Jain, B. Kulis, I. Dhillon, and K. Grauman. Online metric learning and fast similarity search. *Advances in neural information processing systems*, 761–768, 2009

[25] A. Razavian, J. Sullivan, A. Maki, and S. Carlsson. Visual Instance Retrieval with Deep Convolutional Networks. *CoRR*, abs/1412.6574, 2014

[26] A. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: an astounding baseline for recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 806–813, 2014

[27] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. *Proceedings of the European Conference on Computer Vision*, 392–407, 2014