Quartet-net Learning for Visual Instance Retrieval

Jiewei Cao, Zi Huang, Peng Wang, Chao Li, Xiaoshuai Sun, and Heng Tao Shen
The University of Queensland, Australia
{j.cao3, p.wang6, c.li1, x.sun1}@uq.edu.au, {huang, shenht}@itee.uq.edu.au

ABSTRACT

Recently, neuron activations extracted from a pre-trained convolutional neural network (CNN) show promising performance in various visual tasks. However, due to the domain and task bias, using the features generated from the model pre-trained for image classification as image representations for instance retrieval is problematic. In this paper, we propose quartet-net learning to improve the discriminative power of CNN features for instance retrieval. The general idea is to map the features into a space where the image similarity can be better evaluated. Our network differs from the traditional Siamese-net in two ways. First, we adopt a double-margin contrastive loss with a dynamic margin tuning strategy to train the network which leads to more robust performance. Second, we introduce in the mimic learning regularization to improve the generalization ability of the network by preventing it from overfitting to the training data. Catering for the network learning, we collect a large-scale dataset, namely GeoPair[1] , which consists of $68k$ matching image pairs and $63k$ non-matching pairs. Experiments on several standard instance retrieval datasets demonstrate the effectiveness of our method.

Keywords

Convolutional Neural Networks; Feature Learning

1. INTRODUCTION

The neuron activations of a convolutional neural network (CNN), serving as image features, are used in various visual tasks [5, 18]. For visual instance retrieval, i.e., finding images containing the same object or scene as in a query image, mounting evidences [3, 19] demonstrate that the CNN features show superior performance compared to the traditional handcrafted features (e.g., SIFT), especially in the case of low dimensionality. However, due to the domain and task bias (e.g., most CNNs are trained on ImageNet [20] for classification), directly using the features generated from the pre-trained model as image representations for instance retrieval is not an ideal option. One treatment to this problem is to fine-tune the deep model in a target domain.

In this paper, we propose a novel feature learning strategy called *quartet-net* learning. The general idea is to map the image features into an embedding space where the similarity can be better evaluated. Siamese-net [4] can be regarded as a special case of our method and our framework differs from it in two ways: 1) Instead of using single-margin loss as in the standard Siamese-net, we adopt double-margin loss [14, 21] with the margins being dynamically tuned during training. Comparing to single-margin loss, double-margin loss pushes the feature distances between images of the same object under a threshold rather than to be zero. This idea is similar to the triplet loss [25, 22] which allows the learned features of the same object to live on a manifold instead of being projected onto a single point in the embedding space. The rationality behind is that we empirically find it difficult to map the features of positive pairs onto the same point especially in complex scenes and the double-margin relaxation turns out to be crucial in the feature learning. Different from [14, 21], we propose to tune the margins progressively which enhances the discriminative power of network during training; 2) Another advantage of our network is that we introduce in the mimic learning regularization [2, 7]. Specifically, we use two CNNs (i.e., teacher CNN) with fixed parameters to regularize the networks (i.e., student CNN) in order to prevent them from overfitting to the training data and consequently improve the generalization ability of the learned features.

Catering for the network training, we collect a large-scale geo-related dataset, namely *GeoPair*, which consists of $68k$ visually matching image pairs and $63k$ non-matching pairs. Extensive experiments conducted on several standard instance retrieval benchmarks demonstrate the effectiveness of our feature learning method.

2. QUARTET-NET LEARNING

The instance retrieval framework is illustrated in Fig. 1. During training, both matching and non-matching image pairs are fed into the network. After training, the activations of a specific layer are used as image representations to perform instance retrieval.

2.1 Network Structure

Given a pre-trained CNN, we duplicate its network structure four times within a quartet-net (see Fig. 1). The middle

[1] https://goo.gl/reg8M9

MM '16, October 15–19, 2016, Amsterdam, The Netherlands.

DOI: http://dx.doi.org/10.1145/2964284.2967262

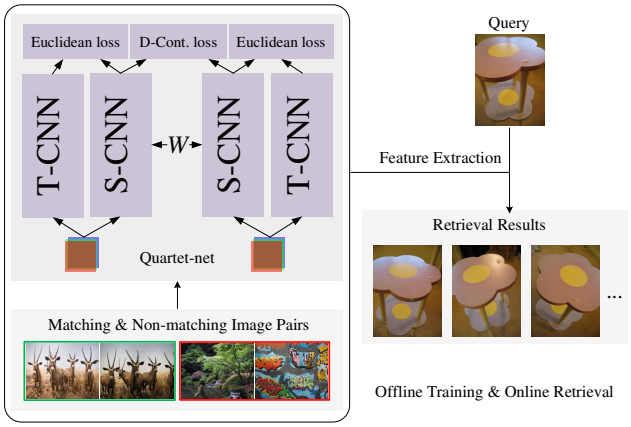

Figure 1: The framework of the proposed retrieval system.

two CNNs, similar to Siamese-net, share the same learned parameters W. We name these two CNNs "student CNNs" (S-CNNs) because during training they gradually learn the image similarities from the input image pairs. Different from [4] where single-margin contrastive loss is employed, we adopt double-margin contrastive loss [14] on the S-CNNs' top layers (e.g., the first fully connected layer). Inspired by the mimic learning in [2], the other two outside CNNs with fixed parameters act as "teacher CNNs" (T-CNNs) where their outputs are used as synthetic labels for "guiding" the S-CNNs. The advantages of these settings will be explained later. The inputs of a quartet-net are image pairs. The parameters are updated using standard back propagation algorithm and stochastic gradient descent.

2.2 Double-margin Contrastive Loss

Given a pair of images $(\mathbf{x}_\alpha^0, \mathbf{x}_\beta^0)$, the single-margin contrastive loss S_l for layer l is defined as:

$$S_l(\mathbf{x}_\alpha^0, \mathbf{x}_\beta^0) = y||\mathbf{x}_\alpha^l - \mathbf{x}_\beta^l||_2^2 + (1-y)\max(m - ||\mathbf{x}_\alpha^l - \mathbf{x}_\beta^l||_2^2, 0), \quad (1)$$

where $y = 1$ if $(\mathbf{x}_\alpha^0, \mathbf{x}_\beta^0)$ is a matching pair or $y = 0$ otherwise, $\mathbf{x}_\alpha^l$ and $\mathbf{x}_\beta^l$ are image $\mathbf{x}_\alpha^0$'s and $\mathbf{x}_\beta^0$'s feature representations in layer l respectively, and $m > 0$ is a margin parameter affecting non-matching pairs. This loss function can be interpreted as applying a contractive force between elements of any matching image pairs and a repulsive force between elements of non-matching pairs whose feature distances are smaller than margin $\sqrt{m}$. In contrast, the double-margin contrastive loss D_l adds another margin parameter to affect matching pairs:

$$D_l(\mathbf{x}_\alpha^0, \mathbf{x}_\beta^0) = y\max(||\mathbf{x}_\alpha^l - \mathbf{x}_\beta^l||_2^2 - m_1, 0) + (1-y)\max(m_2 - ||\mathbf{x}_\alpha^l - \mathbf{x}_\beta^l||_2^2, 0), \quad (2)$$

where $m_1 > 0$ and $m_2 > 0$ are margins affecting matching and non-matching pairs respectively. Therefore, double-margin contrastive loss only applies a contractive force between elements of matching pairs whose feature distances are larger than $\sqrt{m_1}$. The reason why double-margin contrastive loss is preferred rather than the single-margin one in instance retrieval is as follow: Given two matching images, they are probably far apart in the high-dimensional feature space [14]. Hence, we modify the loss to only penalize matching pairs whose distances are larger than a certain threshold. Similar mechanisms are involved in large margin nearest neighbor classification [26] and triplet loss [25, 22], but here we use two different margins to control how image pairs should be separated.

Instead of fixed margins, we use a dynamic margin tuning strategy, namely multistage margins control (MMC), during training. The process is to decrease the margin m_1 for matching pairs and increase m_2 for non-matching pairs in Eq.2 after certain learning epochs. This adjustment can be performed multiple times during training. Each time we change the margins in this manner, it will impose a contractive force between elements of any matching pairs whose feature distances are larger than the new $\sqrt{m_1}$, and a repulsive force between elements of non-matching pairs whose feature distances are smaller than the new $\sqrt{m_2}$. Double-margin loss with MMC can better separate the distribution of matching and non-matching image pairs in the feature space by progressively adjusting the margins, and as a result the retrieval accuracy is improved (see Sec. 4.3).

2.3 Mimic Learning Regularization

In [2], the authors show that it is possible to train a shallow net with high accuracy via mimic learning by training the shallow net (student net) to mimic a deep net (teacher net) with high fidelity. The process is first to train a state-of-the-art deep net using the original training data, and then pass the unlabeled data through this net to collect the produced outputs as labels. These synthetically labeled data is then used to train the shallow net. Experiments in [2] show that the student net can achieve comparable accuracy while it requires fewer parameters and shorter training time. A more general version of this kind of learning, called "knowledge distillation", is proposed in [7].

In this paper, we show that mimic learning can also be used as an effective regularization for quartet-net learning to avoid overfitting. Specifically, given an image $\mathbf{x}$, we denote its probability predictions produced by a CNN as $\mathbf{p} \in \mathbb{R}^K$, where K is the number of output units in the last layer. Normally, $\mathbf{p}$ are the outputs of the last softmax layer, i.e. $p_k = e^{z_k} / \sum_j^K e^{z_j}$. Here the log probability values $\mathbf{z}$, also called logits, are the probability values before the softmax layer. The teacher CNNs consume the same input image pairs as student CNNs do and conduct high-level "guidance" via regressing logits with the Euclidean loss. For example, given the same input image $\mathbf{x}$, the mimic learning loss between a teacher and student CNNs pair is:

$$E(\mathbf{x}) = \frac{1}{2}||\mathbf{z}_S - \mathbf{z}_T||_2^2, \quad (3)$$

where $\mathbf{z}_S$ and $\mathbf{z}_T$ are the logits generated by S-CNN and T-CNN respectively. In [2], regressing logits are preferred, rather than the output probability predictions, in order to avoid the information loss that occurs after passing through the logits to probability space.

Finally, given a pair of images $(\mathbf{x}_\alpha^0, \mathbf{x}_\beta^0)$, the loss function $\mathcal{L}$ for quartet-net learning is defined as:

$$\mathcal{L}(\mathbf{x}_\alpha^0, \mathbf{x}_\beta^0) = \sum_{l \in L} D_l(\mathbf{x}_\alpha^0, \mathbf{x}_\beta^0) + E(\mathbf{x}_\alpha^0) + E(\mathbf{x}_\beta^0), \quad (4)$$

where L is the set of layers that double-margin contrastive loss is applied on. With this loss function, we impose constrains that both S-CNNs would mimic the behaviors of their

Figure 2: GeoPair dataset samples. Images in each row are taken at the same spot within 100 meters by different users and share high visual similarities.

T-CNNs while learning discriminative features for matching and non-matching image pairs. This behavior is similar to the idea of "learning without forgetting" recently proposed in [13]. Experiments in section 4.3 verify that the extended mimic learning regularization can help prevent overfitting.

3. GEOPAIR DATASET

Images of the same object/scene with various visual variations are desired for our network training. Driven by the proliferation of GPS-enabled devices and media-sharing platforms, people have been creating large collections of images with geo-coordinates [24]. We use the images crawled from Flickr to construct our image pair dataset, namely GeoPair. Note that the geo-locations and user information allow us to apply an automatic method (described below) to construct a large collection of image dataset where images of the same object/scene are captured at different viewpoints and share high visual similarities.

We first use k-means to cluster all the images into 500 clusters according to their geo-coordinates. For every image in a cluster, we find a set of visually similar images from the same cluster [2], which is considered as a finer spot in this coarse cluster. To refine the result, only the images uploaded by different users and geographically within 100 meters are selected. In this way, we filter out most redundant images uploaded by the same user and collect diverse viewpoints of the same object/scene from different users. We obtain 28,418 spots with 109,725 images in total. Each spot has two similar images at minimum, and five images on average. Fig. 2 gives some examples of the collected dataset.

We create the matching image pairs by selecting images from the same spot and non-matching pairs by randomly pickup images from different spots. As a result, we generate 68,248 matching pairs and 63,432 non-matching pairs from the whole corpus. We randomly shuffle all image pairs and select 80% (105,344 pairs) for training and the rest 20% (26,336 pairs) for validation. In the following experiments, GeoPair dataset is used for quartet-net learning.

[2]SIFT and RANSAC verification [16] are used in our work.

4. EXPERIMENTS

In this section, we evaluate the retrieval performance of CNN features before and after quartet-net learning.

4.1 Datasets & CNNs

We report results on four instance retrieval datasets: Oxford5k [16], Paris6k [17], INRIA Holidays [8], and UKBench [15]. The retrieval performance is evaluated by mean average precision (mAP).

To demonstrate our methods are applicable to different CNNs, we used two publicly available pre-trained networks implemented by Caffe [11]. The first one is the BVLC reference CaffeNet which is similar the one proposed in [12]. The second one is the OxfordNet [23] which has 16 layers in total and is much deeper than CaffeNet. Both of them are pre-trained on the ILSVRC2012 dataset [20].

4.2 Quartet-net Learning Settings

In quartet-net learning, we apply the double-margin contrastive loss on the first and second fully connected layers of the two S-CNNs, namely $FC6$ and $FC7$ layers. We set the two margin parameters m_1 and m_2 in Eq.2 equal and their values are the average median distance of matching and non-matching pairs calculated from the GeoPair validation set. For example, the margin for CaffeNet's $FC6$ layer is the average of matching pairs' median distance and non-matching pairs' median distance in the $FC6$ layer's feature space (i.e., $m_1 = m_2 = 72105.6$ in this case). Note that we do not apply dropout [12] on $FC6$ and $FC7$ layers since we need to calculate the squared ℓ_2 norm of their outputs (i.e., $||\mathbf{x}^l_\alpha - \mathbf{x}^l_\beta||_2^2$ in Eq.2). For mimic learning regularization, the Euclidean loss is applied on each teacher and student CNNs pair's last fully connected layers, namely $FC8$ layers, via regressing logits before the softmax activation.

During training, one can decide to tune all the layers within a S-CNN or only a partial of them, such as $FC6$, $FC7$ and $FC8$ on which the loss functions are applied and keep other layers' parameters fixed. In the following experiments, we compare the performance of different combinations of network settings. Specifically, we denote the baseline Siamese-net learning for all layers in CaffeNet as CaffeNet$_{\times 2}$. Similarly, we denote CaffeNet$_{\times 4}$ as quartet-net learning for all the layers in CaffeNet. Due to the size and time consumption of OxfordNet, we only perform fine-tuning for its fully connected layers (i.e., $FC6$, $FC7$, $FC8$ layers), which is denoted as OxfordNet$_{\times 4}$.

We set the initial learning rate to $1e^{-8}$. This value is chosen through trial and error by keep reducing the learning rate by a factor of 10 from its initial value 0.01 until we observe a steady decrease in the validation error rate at the beginning of learning steps. Follow the parameters settings as in [12, 6] for training, we set momentum to 0.9, weight decay to $5e^{-4}$, and train on the GeoPair dataset for a maximum 200 epochs[3]. After every 20 epochs, we decrease the learning rate by a factor of 10. And after every 50 epochs, we decrease m_1 (margin for matching pairs) and increase m_2 (margin for non-matching pairs) in Eq.2 by a factor of 10. Finally, we use the activations of $FC6$ (or $FC7$) layer in the S-CNN as global image descriptors. Given a query image, we extract the query feature for the whole image and

[3] The training time depends on the network size and computing hardware. For CaffeNet, it takes about 14 hours for 50 learning epochs on the GTX Titan Black.

Table 1: The retrieval performance of CaffeNet's features with different learning settings. "Org." are the mAPs before fine-tuning. "CaffeNet$^{(S)}_{\times 2}$(CaffeNet$^{(D)}_{\times 2}$)" are the mAPs after Siamese-net learning with single(double)-margin loss, and CaffeNet$_{\times 4}$ are the mAPs after quartet-net learning using the GeoPair dataset.

		Org.	CaffeNet$^{(S)}_{\times 2}$	CaffeNet$^{(D)}_{\times 2}$	CaffeNet$_{\times 4}$
*FC*6	Ox5k	0.407	0.233	0.415	**0.426**
	Pa6k	0.584	0.372	0.586	**0.619**
	Hol.	0.681	0.393	**0.687**	0.679
	UKB.	0.849	0.549	0.837	**0.866**
*FC*7	Ox5k	0.387	0.220	0.407	**0.412**
	Pa6k	0.587	0.343	0.593	**0.604**
	Hol.	0.681	0.338	0.685	**0.701**
	UKB.	0.856	0.493	0.830	**0.867**

return a list of images ranked by the dot product of the ℓ_2-normalized feature in descending order.

4.3 The Performance of Quartet-net Learning

We evaluate the effects of different components of quartet-net learning and compare it with Siamese-net learning as shown in Tab. 1. 1) *Effects of double-margin contrastive loss*: The results of CaffeNet$^{(S)}_{\times 2}$ show that Siamese-net learning with single-margin contrastive loss significantly decreases retrieval accuracies, which means that the distribution of matching and non-matching image pairs is not distinguishable after fine-tuning. This result is consistent with the previous findings in [14]. In contrast, the mAPs are slightly improved when Siamese-net is equipped with double-margin contrastive loss (CaffeNet$^{(D)}_{\times 2}$), which confirms the effectiveness of this loss function. 2) *Effects of mimic learning regularization*: CaffeNet$_{\times 4}$ outperforms CaffeNet$^{(D)}_{\times 2}$ on all the datasets except for the case of *FC*6 on Holiday dataset. With the help of mimic learning regularization between student and teacher CNNs, quartet-net learning can maximize the learning capacity and representational power of the tuned CNN. These results demonstrate the effectiveness of our proposed network structure.

Next, we evaluate the effect of dynamic margin turning. We decrease m_1 (margin for matching pairs) and increase m_2 (margin for non-matching pairs) in Eq. 2 by a factor of 10 after 50 epochs before continuing training. Fig. 3 shows the retrieval accuracy of CaffeNet's *FC*6 features on Oxford5k after quartet-net learning with MMC. We observe that MMC provides consistent improvements for both Siamese-net and quartet-net learning. But it downgrades the discriminative power of the CNN features after certain epoch (e.g., the 200-th epoch). Therefore, our total training epochs is set to 200. "CaffeNet$_{\times 4}$(*nommc*)" uses fixed margins and trains the same amount of learning epochs but there is no significant improvement, which means the performance gains of S-CNN are not due to the longer training time. In addition, if we use the best margin values for fixed margins training, as demonstrated by "CaffeNet$_{\times 4}$(#200)", the mAP steadily increases. Therefore, MMC can also be used to determine the best margins for training.

4.4 Comparisons with Existing Work

According to the previous experiments, we use the *FC*6 layer's features as image descriptors. We report the results of CaffeNet$_{\times 4}$ and OxfordNet$_{\times 4}$. Tab. 2 shows the comparisons between our method with existing work. Note that

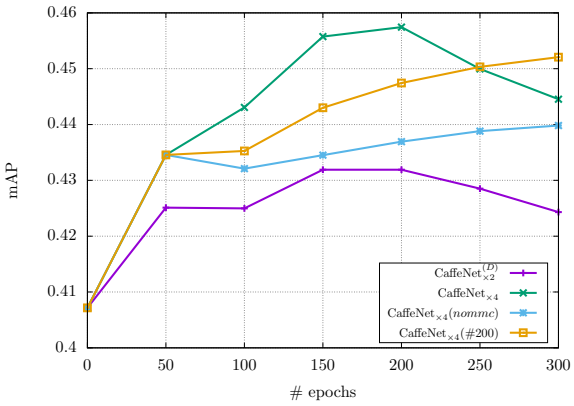

Figure 3: The performance of CaffeNet's *FC*6 features on Oxford5k dataset during different learning stages. CaffeNet$_{\times 4}$(*nommc*) stands for quartet-net learning without MMC, and CaffeNet$_{\times 4}$(#200) uses the fixed margin values of m_1 and m_2 at the 200-th epoch.

Table 2: Comparisons with existing work.

	Dim.	Oxford5k	Paris6k	Holidays	UKBench
KTH [18]	4096	32.2	49.5	64.2	76.0
VLAD [9]	4096	37.8	-	55.6	-
FV [9]	4096	41.8	-	59.9	-
CaffeNet$_{\times 4}$	4096	45.7	69.2	70.2	86.6
OxfordNet$_{\times 4}$	4096	**48.3**	**71.5**	**71.5**	**88.2**
Inria [10]	256	47.2	-	65.7	86.3
CaffeNet$_{\times 4}$	256	47.3	47.4	71.6	86.3
OxfordNet$_{\times 4}$	256	**49.7**	**48.8**	**72.5**	**88.3**
VLAD-intra [1]	128	44.8	-	62.5	-
CaffeNet$_{\times 4}$	128	46.9	47.7	**71.6**	85.6
OxfordNet$_{\times 4}$	128	**48.5**	**48.8**	71.2	**87.5**

for a fair comparison, we only report results on features with the same dimension. We also use PCA to reduce the original feature dimensionality to much lower ones to further compare the performance in different dimensional feature spaces. Note that KTH [18] also uses the *FC*6 features as image descriptors, but their results are not directly comparable since their CNN is different from us. When comparing with other optimized handcraft features [9, 10, 1], our method outperforms them under different dimensionalities on all datasets.

5. CONCLUSION

In this paper, we present quartet-net learning to improve the discriminative power of CNN features for visual instance retrieval. By incorporating the double-margin contrastive loss and mimic learning regularization, quartet-net learning can help avoid overfitting when training on large image pairs dataset. Our proposed method can be applied to different CNNs and outperforms existing methods on various datasets. Besides, we release the GeoPair dataset, a large-scale dataset consisting of matching/non-matching image pairs, to facilitate future studies.

6. REFERENCES

[1] R. Arandjelovic and A. Zisserman. All about VLAD. In *CVPR*, 2013.

[2] J. Ba and R. Caruana. Do deep nets really need to be deep? In *NIPS*, 2014.

[3] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. Neural codes for image retrieval. In *ECCV*. 2014.

[4] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005.

[5] 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. In *ICML*, 2014.

[6] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[7] G. E. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

[8] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, 2008.

[9] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *TPAMI*, 2012.

[10] H. Jégou and A. Zisserman. Triangulation embedding and democratic aggregation for image search. In *CVPR*, 2014.

[11] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *MM*, 2014.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[13] Z. Li and D. Hoiem. Learning without forgetting. *CoRR*, abs/1606.09282, 2016.

[14] J. Lin, O. Morere, V. Chandrasekhar, A. Veillard, and H. Goh. Deephash: Getting regularization, depth and fine-tuning right. *CoRR*, abs/1501.04711, 2015.

[15] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

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

[17] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *CVPR*, 2008.

[18] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: an astounding baseline for recognition. In *CVPRW*, 2014.

[19] A. S. Razavian, J. Sullivan, A. Maki, and S. Carlsson. A baseline for visual instance retrieval with deep convolutional networks. In *ICLRW*, 2015.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 2015.

[21] F. Sadeghi, C. L. Zitnick, and A. Farhadi. Visalogy: Answering visual analogy questions. In *NIPS*, 2015.

[22] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.

[23] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[24] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L. Li. YFCC100M: the new data in multimedia research. *Commun. ACM*, 2016.

[25] 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. In *CVPR*, 2014.

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