2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Siamese Network Features for Image Matching

Iaroslav Melekhov
Department of Computer Science
Aalto University, Finland
Email: iaroslav.melekhov@aalto.fi

Juho Kannala
Department of Computer Science
Aalto University, Finland
Email: juho.kannala@aalto.fi

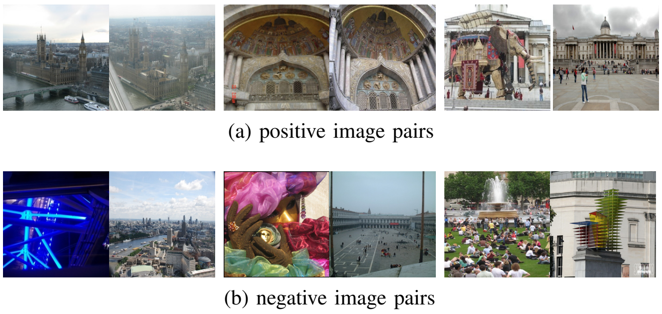
Esa Rahtu
Center for Machine Vision Research
University of Oulu
Email: esa.rahtu@ee.oulu.fi

***Abstract*—Finding matching images across large datasets plays a key role in many computer vision applications such as structure-from-motion (SfM), multi-view 3D reconstruction, image retrieval, and image-based localisation. In this paper, we propose finding matching and non-matching pairs of images by representing them with neural network based feature vectors, whose similarity is measured by Euclidean distance. The feature vectors are obtained with convolutional neural networks which are learnt from labeled examples of matching and non-matching image pairs by using a contrastive loss function in a Siamese network architecture. Previously Siamese architecture has been utilised in facial image verification and in matching local image patches, but not yet in generic image retrieval or whole-image matching. Our experimental results show that the proposed features improve matching performance compared to baseline features obtained with networks which are trained for image classification task. The features generalize well and improve matching of images of new landmarks which are not seen at training time. This is despite the fact that the labeling of matching and non-matching pairs is imperfect in our training data. The results are promising considering image retrieval applications, and there is potential for further improvement by utilising more training image pairs with more accurate ground truth labels.**

I. Introduction

Nowadays, finding similar images in a large, unstructured image collection is a common problem in computer vision systems. It may be very time-consuming procedure involving testing many images to find a correspondence between matched pairs. In recent years, a number of algorithms for image matching have been proposed aiming to improve accuracy and performance of the algorithms. In general, these methods can be split into two categories. The first category includes hand-crafted representation designed to predict whether image pair is positive (similar) or not (i.e. whether both images represent pairs of the same scene). Such methods like bag-of-visual-words (BoW) [1] have good results of predicting a smaller set of candidate image pairs. There are also approaches which use discriminative learning of BoW models to predict which pairs of images in an input dataset match, and which do not [2].

The second group is based on deep learning models, particularly deep convolutional neural networks (CNNs), which have been successfully used in various visual tasks such as image classification, object detection and image retrieval. Image representation and similarity measure become critical to image retrieval task which aims to find matched images in a big dataset. The recent works [3], [4] proposed an idea of utilizing a pretrained CNN from a related image classification problem in image retrieval and showed very promising results.


(a) positive image pairs

(b) negative image pairs

Fig. 1. Randomly picked positive and negative image pairs of evaluation datasets [2]. The images are partially occluded and taken under different lightning and weather conditions, with changes in viewpoint and appearance. All these factors make generic image matching task more challenging.

In this paper, instead of trying to learn classification of individual images, our aim is to directly learn a CNN for the matching task. That is, we utilize labeled training image pairs to learn an image-level feature representation so that similar images are mapped close to each other in the feature space, and dissimilar image pairs are mapped far from each other. This is analogous to face-verification problem where Siamese networks [5] have been utilized to predict whether the persons illustrated in an input image pair are the same or not [6], [7]. Another application using somewhat similar techniques is a system proposed by Lin *et al.* [8] which can successfully match street-level and aerial view images. In addition, similar methods have been used for matching small local image patches [9], [10], [11] but not yet for generic image retrieval or whole-image matching.

Figure 1 shows examples of image pairs randomly picked from evaluation databases and used in our experiments. The images represent indoor and outdoor views of 5 landmarks across the world and were captured under different lightning and weather conditions with some occlusions and changes in appearance and viewpoint. All of these negative factors make finding generic image similarity more challenging problem.

The goal of this paper is to address image similarity problem. We aim to get an image similarity function without attempting to use any manually designed features but instead directly learn this function from annotated pairs of raw image pairs. Inspiring from the advancement of deep learning we decided to choose this technique in our experiments.

Our contributions of this paper are two-fold. First, we

978-1-5090-4847-2/16/$31.00 ©2016 IEEE

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on December 16,2024 at 14:24:42 UTC from IEEE Xplore. Restrictions apply.

present a method to predict the similarity of an image pair based on deep neural network using whole-image similarity measure; second, we apply our approach on unseen data to examine the generalization characteristics, showing that it outperforms a state-of-the-art CNN trained for the classification task.

The paper is organized as follows. Section II describes related papers focusing on patch matching problem and image retrieval. Section III describes the proposed method of finding similar images, discusses CNN architecture, objective function and details of evaluation datasets. Section IV presents the experimental pipeline and results on constructed database. In the end of this paper we summarize our results and point some directions of future work.

II. Related work

One important application area for good image similarity metrics is image retrieval. Babenko *et al.* [3] show that fine-tuning a pretrained CNN with domain specific data can improve retrieval performance on relevant data sets. Chandrasekhar *et al.* [4] propose a systematic evaluation of Fisher Vectors [12] and CNN pipelines for image retrieval and show that their combination has better performance on some datasets than separately. Like our approach, both methods [3], [12] are based on CNN but using different objective and network structures which are not able to handle image pairs directly. In fact, in the conclusion section of [3] the authors suggest utilizing Siamese networks for directly learning features for generic image matching and retrieval but no one has studied it yet.

A very interesting idea and implementation were proposed in [8]. The authors utilize deep networks to geolocalize a photo by directly learning to match street-level images to aerial images without using ground-level reference imagery. This work is close to our approach but evaluating on quite different and specific input data.

A related problem to computing the image similarity is finding matches between image *patches*. Brown *et al.* [13] proposed an original method for learning patch descriptors and a general framework for evaluation descriptors performance. As opposed to the hand-crafted descriptors, the recent approach in this field utilizes deep neural networks [9], [10], [11] which can significantly outperform state-of-the-art on several benchmarks. Although these methods are related to this work, proposed network architectures and objective functions differ from ours. Moreover, image similarity is more challenging task than patch matching due to bigger size of images and potential distortions, for example, occlusions and changes in viewpoint, appearance and lightning.

Recent work by Žbontar and LeCun [14] mainly focused on a method of comparing image patches for extracting depth information. The proposed method is based on using convolutional networks minimizing a hinge loss function and showed the best performance on KITTI stereo evaluation dataset [15]. However, as that approach operates on very small patches (9×9 pixels), it restricts the area of applicability. In contrast, in this work we aim to consider a broader set of changes in images that allows to apply our method to a wide range of applications, *e.g.* structure-from-motion or image retrieval.

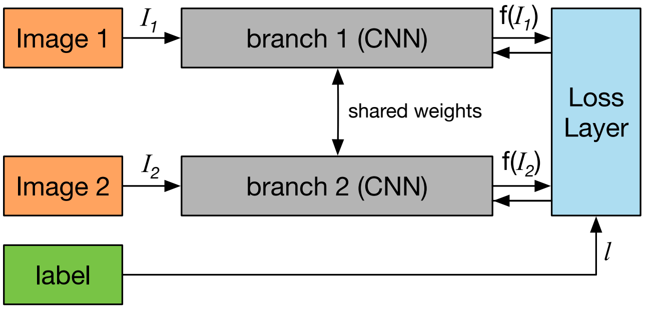


Fig. 2. Model structure. Proposed network architecture (sHybridNet) for image matching. Branch 1 and 2 have the same HybridCNN structure presented in [16]

Schönberger *et al.* [17] proposed a method to predict scene overlapping for SfM task. This method can be considered as an application for applying our approach which can efficiently find similar image pairs and improve the accuracy of 3D reconstruction as a result. The common part of both approaches is the same image data.

III. Method

Our goal is to learn a general similarity function for image pairs. To encode such function, we propose a method based on a deep convolutional neural network. Inspired by the recent success in image classification we use HybridCNN [16] as a core element of our network.

Suppressing the model details, the model structure is illustrated in Figure 2. In general, a pair of images goes through a network consisting of two branches during training. The outputs of these branches are fed to a loss layer. The loss layer tries to minimize squared Euclidean distance between the features of positive image pairs ($f(I_1)$ and $f(I_2)$) and maximize it for negative pairs.

The following section describes the proposed objective loss function and how it can be used in our approach. The details of the network architecture are described in section III-B.

A. Contrastive loss

To optimize the proposed network, we utilise a cost function which is capable to make a distinguish between pairs. More precisely, it encourages similar examples to be close, and dissimilar ones to have Euclidean Distance of at least margin m from each other.

To implement this, we use margin-based contrastive loss function proposed in [18] which is defined as follows:

$$\mathcal{L} = \frac{1}{2} l D^2 + \frac{1}{2} (1 - l) \{\max (0, m - D)\}^2 \qquad (1)$$

where l is a binary label selecting whether the input pair consisting of image I_1 and I_2 is a positive ($l = 1$) or negative ($l = 0$), $m > 0$ is the margin for dissimilar pairs and $D = \|f(I_1) - f(I_2)\|_2$ is the Euclidean Distance between feature vectors $f(I_1)$ and $f(I_2)$ of input images I_1 and I_2.

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on December 16,2024 at 14:24:42 UTC from IEEE Xplore. Restrictions apply.

Dissimilar pairs contribute to the loss function only if their distance is within margin m. This loss function encourages matching pairs to be close together in feature space while pushing non-matching pairs apart. Moreover, it can be clearly seen that negative pairs with a distance which is bigger than margin would not contribute to the loss (second part of Equation 1).

The loss function penalizes positive pairs by the squared Euclidean distances and negative pairs by the squared difference between margin m and Euclidean distance for pairs which have a distance less than a margin m. We discuss the strategy of finding a suitable margin value in the following section.

B. Network architecture

In order to realize a learned similarity metric between images, we use a pair-based (Siamese) network structure in experiments. Our approach was influenced by recent Ground-to-Aerial geolocalization method [8] and a fundamental work [5]. The structure consists of two identical branches that share weights and parameters. Each branch poses a deep neural net and includes a set of convolutional layers, rectified linear units (ReLU) as non-linearity for the convolutional layers, and fully connected layers. Images I_1 and I_2 are fed into branches which are identical during training. The main goal of a proposed network structure is to learn optimal feature representations of the input pairs where matched images in a pair are pulled closer and unmatched images are pushed far away.

An architecture of our Siamese network (sHybridCNN) is based on HybridCNN [16] which was used for both object and scene image classification, outperforming state-of-the-art methods on the MIT Indoor67 dataset [19]. Specifically, HybridCNN is AlexNet [20] trained on a combination set of ImageNet and Places databases [16]. Moreover, HybridCNN outperforms pure AlexNet and OxfordNet [4] in image retrieval on Oxford Building dataset [21].

At the top of HybridCNN are three fully connected layers (fc6, fc7 and fc8) taking as an input the output of the previous layer. As the last layer (fc8) of the network was designed considering the number of classes in the original training dataset (1183 classes), we removed it and use fc7 layer as the feature representation. The network has in total 58 million parameters and it is 13 layers deep.

Corresponding to Equation 1 of the loss function, we have to specify a margin m to optimize the proposed network. Evaluating experiments, we found that to train the proposed network efficiently, the margin value should be twice the average Euclidean distance between features of training image pairs before learning.

In general, a key problem of applying deep convolutional networks in computer vision is to find a large, consistent dataset suitable for a specific task. To perform our experiments, it is necessary to have a database consisting of image pairs relevant to the landmark-type datasets. The collection of such set of image pairs is a non-trivial task and often involves testing many pairs by matching SIFT features and performing geometric verification. For our experiments, we utilize 5 crowd-sourced image collections downloaded from Flickr, each corresponding to a popular landmark (London Eye (LE) 6856 images, San Marco (SM) 7580 images, Tate Modern (TM) 4583 images, Times Square (TS) 6361 images, Trafalgar (T) 6802 images) [2]. Original datasets contained both color and grayscale images.

The ground truth labels for matching (positive) image pairs were provided by [2] and computed only considering the top 500 most similar images based on raw Bag-of-Words (BoW) similarity measured by the dot product of BoW vectors. Using information about positive pairs, we generate image ids corresponding to unmatched pairs.

IV. Experiments

In this section, we present experimental results evaluating the proposed approach. We would like to find an answer for two questions: (a) whether the network is able to learn to better distinguish between similar and dissimilar pairs utilising this kind of training data; (b) if our network is learnable, do features extracted from it generalize to other datasets? In each experiments, we take the activations of fc7 layer of the network as feature vectors.

A. Image similarity metric

There are many ways to measure the accuracy of image similarity methods [22]. In our case, to evaluate performance we use the receiver operating characteristics (ROC curve) which are commonly used to analyse results for binary decision problems in machine learning and the area under ROC curve as the quantitative metric. In addition to ROC curves, we present a set of precision-recall (PR) characteristics which can better illustrate the success of our system in improving recall. In the following section we provide information about training and test data in details.

B. Dataset and data preprocessing

As described in section III-B, in our experiments we use a set of images of 5 different landmarks (London Eye, San Marco, Tate Modern, Times Square and Trafalgar) downloaded from Flickr. To analyse generalization performance of the proposed algorithm, we should evaluate sHybridCNN on an unseen data which is not presented in training data at all. Therefore, we construct a training dataset as a combination of image pairs of 4 landmarks and a test dataset as a set of image pairs of the remaining landmark. Following this procedure, we get 5 different test and training datasets. The images and the list of positive image pairs for each landmark were originally provided by [2]. In addition, we randomly generated negative pairs utilizing images of the same landmark, so the number of matched and unmatched pairs in test datasets is equal. In contrast to test data, all training datasets are unbalanced. Specifically, the number of dissimilar pairs in training data is $1.5\times$ larger than the number of similar image pairs.

To handle the network structure we have to operate a pair of images with 6 channels (a pair of RGB images). Therefore, if original pair consists of a grayscale and a color images

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on December 16,2024 at 14:24:42 UTC from IEEE Xplore. Restrictions apply.

TABLE I

DETAILED COMPARISON OF DIFFERENT DEEP LEARNING APPROACHES ON TEST DATASETS (AREA UNDER THE ROC CURVE)

Landmark	AlexNet	HybridNet	sHybridNet
Tate Modern	0.818	0.857	**0.926**
London Eye	0.837	**0.853**	0.849
San Marco	0.676	0.776	**0.826**
Times Square	0.808	0.804	**0.850**
Trafalgar	0.745	0.816	**0.857**

we convert RGB image into grayscale and then treat these grayscale pair as a color one. The ratio of grayscale and color images is 5% per object landmark. As proposed deep network was pretrained on the images in a normal orientation, we automatically rotate images in our training dataset to the normal orientation using EXIF information. Finally, we resize images to 227×227 pixels size without cropping and put it as input for a neural network that we propose.

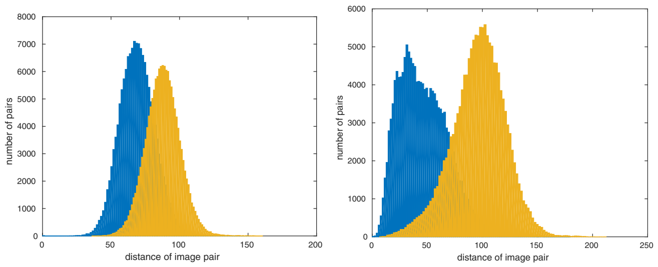
C. Experimental details

In all our experiments we train the network using Stochastic Gradient Descent (SGD) with a standard back-propagation method [23] and AdaDelta [24]. As mentioned in Subsec. III-B we use the weights from a deep network (HybridCNN) pretrained on Imagenet and Places database [16] as the initialization to train our Siamese approach. Specifically, we finetune a pretrained model using similar technique as in [8] *e.g.* setting the learning rate 10^{-5} for the last fully-connected layer (fc7) and 10^{-6} for other layers. The model was trained using publicly available deep learning framework Caffe [25] on one NVIDIA TITAN Z GPU. It took around 40 hours to finish 10 epochs of training.

V. RESULTS AND DISCUSSION

In this section we discuss the performance of deep features learned by sHybridCNN. We compare the results of the following approaches based on deep learning technique:

1) **AlexNet.** We use deep convolutional neural network (AlexNet) pretrained for classification on ImageNet as an image descriptor in feature space. To do that, we directly extracted $4,096$ dimensional output from the fc7 layer of the network structure as the feature for image matching. We consider AlexNet as a baseline in our experiments.
2) **HybridNet.** Another neural network used in our experiments is HybridNet [16]. It has exactly the same architecture as AlexNet but was trained on different data. More precisely, the training data is a combination of ImageNet and a scene-centric database called Places. According to [4], HybridNet has impressive performance in image retrieval on Oxford Building dataset. Therefore, it can generalize well on unseen data.
3) **sHybridNet.** We extract features from the Siamese deep convolutional network trained on our database. We initialize the parameters for sHybridNet in the training stage with the learned parameters (weights) from pretrained HybridNet. After training on landmarks pairs, we extract a $4,096$ dimensional feature vector from fc7 layer of the proposed network architecture.



(a) HybridCNN features

(b) sHybridCNN features

Fig. 3. Distribution of positive and negative pairs from (a) HybridCNN and (b) sHybridCNN on Tate Modern test set

In all cases, the ordering of image pairs for evaluation is based on the Euclidean distance between the feature vectors of the images.

To demonstrate what has been learned by sHybridCNN, we compute the histogram of pairwise Euclidean distances of HybridCNN and sHybridCNN on the test set (Tate Modern landmark) in Figure 3.

The blue bars represent pairwise distances of positive pairs and yellow bars represent pairwise distances of negative pairs. The pair distance distribution of HybridCNN shows the initial distance distribution of sHybridCNN without learning. It can clearly be seen that the training process of sHybridCNN on image pairs effectively pushes dissimilar pairs and pulls similar pairs together.

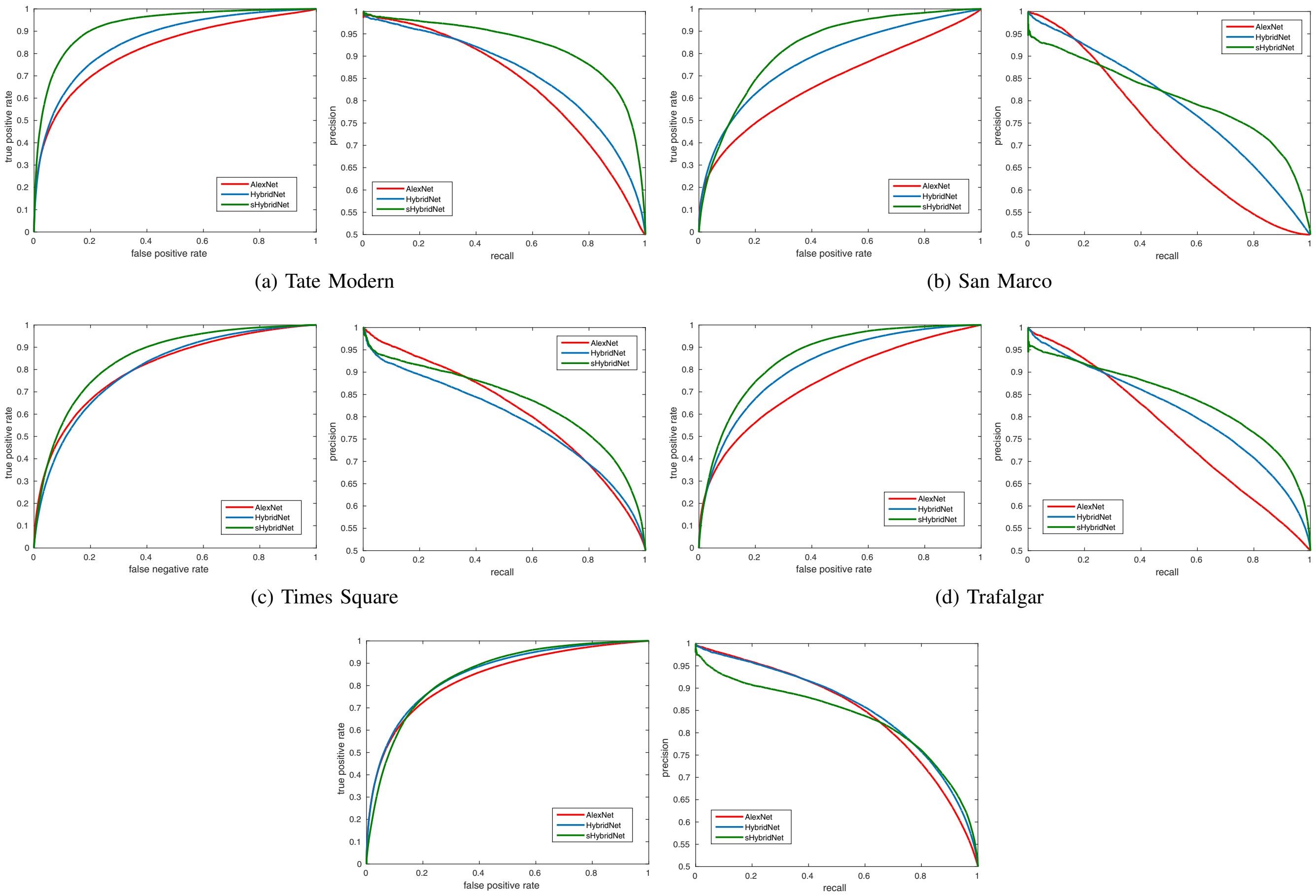
Table I summarizes classification accuracy of considered approaches on different test datasets in terms of area under ROC curve (AUC). One can see that the proposed method consistently outperforms other algorithms. More specifically, sHybridNet outperforms AlexNet and HybridNet by 11% and 5% respectively in average AUC. Training with a larger number of image pairs (particularly, “hard negative” pairs) could improve these results further.

The detailed evaluation of the algorithms on test datasets is presented in Figure 4. From this set of ROC and PR curves we can make the following observations:

- Features extracted from the proposed sHybridCNN have better performance in 4 cases out of 5 compared to original AlexNet and HybridNet respectively. It also confirms the results illustrated in Figure 3 and proves that sHybridCNN can efficiently distinguish positive and negative pairs. That is, sHybridCNN outperforms HybridCNN on images of unseen landmarks.
- In one case (London Eye) all three methods have almost similar ROC curves and the PR curve of sHybridNet shows lower precision among easy positive pairs (i.e. PR curve drops in the beginning). However, as explained in detail below, deeper analysis indicates that in the London Eye test set there seems to be particularly many image pairs with incorrect ground truth labels.

In order to illustrate the performance on London Eye test set, we visualize false positive and false negative image pairs of

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on December 16,2024 at 14:24:42 UTC from IEEE Xplore. Restrictions apply.



(a) Tate Modern

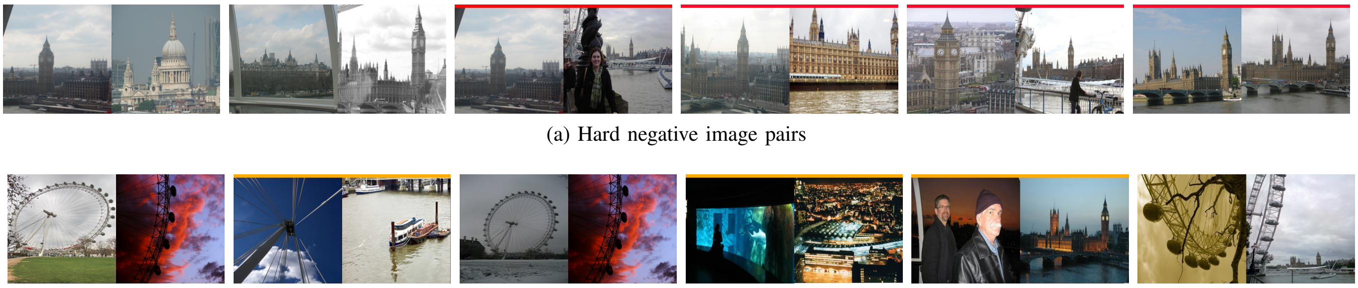
(b) San Marco

(c) Times Square

(d) Trafalgar

(e) London Eye

Fig. 4. A set of ROC and PR curves for 5 different test datasets: (a) Tate Modern, (b) San Marco, (c) Times Square, (d) Trafalgar, (e) London Eye. The experiment is discussed in section IV in details. The proposed method (sHybridNet) has better generalization performance than other approaches.



(a) Hard negative image pairs

(b) Hard positive image pairs

Fig. 5. Top-ranking hard negatives (i.e. negative image pairs with smallest Euclidean distances of feature vectors) and hard positives (i.e. positive image pairs with largest Euclidean distances) are shown for sHybridCNN on London Eye test set. Red color marks pairs which represent the same object but are erroneously labeled as negative in the ground truth, brown color marks pairs which show dissimilar objects but are erroneously labeled as positive in the ground truth. It can be clearly seen that the ground truth labels are imperfect. That is, our network has correctly assigned similar pairs close to each other and dissimilar pair far from each other.

test data. To achieve that, we extract feature vectors from layer fc7 of the two branches and calculate Euclidean distance between them over test data and sort them in ascending order for negative pairs and in descending order for positive pairs. The visualization is presented in Figure 5. It shows hard negatives and hard positives image pairs encountered by sHybridNet on London Eye test data. Hard positives are examples of positive pairs with largest pairwise feature distances returned on test data by using sHybridCNN feature. Similarly, hard negatives are examples of negative pairs with smallest distances. By looking at the examples in Figure 5, we observe that ground truth labels are imperfect and actually most of the negative pairs with smallest distances represent the same scene and should have been labeled as positive (i.e. matching).

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on December 16,2024 at 14:24:42 UTC from IEEE Xplore. Restrictions apply.

Furthermore, also the positive pairs with largest distances seem to have incorrect labels. Thus, we infer that the original algorithm [2] (based on bag-of-visual-words) for computing ground truth labels is not perfect and could be a likely reason for many misclassified test pairs and for the slump in the beginning of PR curves. However, despite the errors in the labels of training and test data, we may conclude that the network has improved in distinguishing between similar and dissimilar pairs, as it is still realistic to assume that most of the positive/negative labels provided by [2] are correct.

VI. Conclusion

We have evaluated the performance of Siamese network features for image matching on landmark datasets. There are several conclusions that we can get from our experiments. First, this network architecture is able to learn from such data when one uses pretrained CNN from a related image classification problem as a starting point. We also showed that our approach has promising results of generalization on unseen landmark datasets. We also observed that potentially the imperfect ground truth labels during training are preventing the network to learn and generalize optimally. Nevertheless, it allows to suggest that Siamese architecture together with contrastive loss objective is a good choice for learning features for image matching and retrieval tasks. Moreover, using additional relevant datasets [26], [27] during training might further enhance the accuracy and performance of the approach.

References

[1] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. ICCV*, 2003.

[2] S. Cao and N. Snavely, "Learning to match images in large-scale collections," *Proc. ECCV*, 2012.

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

[4] V. Chandrasekhar, J. Lin, O. Morère, H. Goh, and A. Veillard, "A practical guide to cnns and fisher vectors for image instance retrieval," *CoRR*, vol. abs/1508.02496, 2015.

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

[6] 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," 2007.

[7] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *CVPR*, 2014.

[8] T.-Y. Lin, Y. Cui, S. Belongie, and J. Hays, "Learning deep representations for ground-to-aerial geolocalization," in *CVPR*, 2015.

[9] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *CVPR*, 2015.

[10] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg, "Matchnet: Unifying feature and metric learning for patch-based matching," in *CVPR*, 2015.

[11] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, and F. Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *ICCV*, 2015.

[12] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. ECCV*, 2010.

[13] G. Hua, M. Brown, and S. Winder, "Discriminant learning of local image descriptors," in *IEEE Transactions on PAMI*, 2010.

[14] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," in *CVPR*, 2015.

[15] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *IJRR*, 2013.

[16] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," *NIPS*, 2014.

[17] J. L. Schönberger, A. C. Berg, and J.-M. Frahm, "Paige: Pairwise image geometry encoding for improved efficiency in structure-from-motion," in *CVPR*, 2015.

[18] R. Hadsell, C. Sumit, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," *CVPR*, 2006.

[19] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *CVPR*, 2009.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*. Curran Associates, Inc., 2012, pp. 1097–1105.

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

[22] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proc. ICML*. New York, NY, USA: ACM, 2006, pp. 233–240.

[23] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[24] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012.

[25] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[26] T. Weyand and B. Leibe, "Visual landmark recognition from internet photo collections: A large-scale evaluation," *CoRR*, vol. abs/1409.5400, 2014.

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

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on December 16,2024 at 14:24:42 UTC from IEEE Xplore. Restrictions apply.