SOLAR: Second-Order Loss and Attention for Image Retrieval

Tony Ng[1], Vassileios Balntas[2], Yurun Tian[1], and Krystian Mikolajczyk[1]

[1] MatchLab, Imperial College London
[2] Facebook Reality Labs
{tony.ng14, y.tian, k.mikolajczyk}@imperial.ac.uk
vassileios@fb.com

Abstract. Recent works in deep-learning have shown that second-order information is beneficial in many computer-vision tasks. Second-order information can be enforced both in the spatial context and the abstract feature dimensions. In this work, we explore two second-order components. One is focused on second-order spatial information to increase the performance of image descriptors, both local and global. It is used to re-weight feature maps, and thus emphasise salient image locations that are subsequently used for description. The second component is concerned with a second-order similarity (SOS) loss, that we extend to global descriptors for image retrieval, and is used to enhance the triplet loss with hard-negative mining. We validate our approach on two different tasks and datasets for image retrieval and image matching. The results show that our two second-order components complement each other, bringing significant performance improvements in both tasks and lead to state-of-the-art results across the public benchmarks. Code available at: http://github.com/tonyngjichun/SOLAR

Keywords: Image Retrieval, Descriptors, Features

1 Introduction

Second-order information is receiving increasing attention in computer-vision. It can be exploited in image retrieval in form of spatial auto-correlation of features, or by second-order similarities in a metric space. Bilinear features [10,13,24] compute second-order correlation, but significantly expand feature dimensions, requiring subsequent dimensionality reduction. Second-order (self) attention, successful in natural-language processing (NLP) [52], tackles the dimensionality problem with a multi-headed approach and is hence studied extensively in various vision areas [53,55,58,59]. Although recent deep-learning based global descriptors provide effective ways to aggregate features into a compact global vector, they have not explored the correlations between features within a feature map. Meanwhile, second-order similarity [47] has recently been shown to improve patch descriptors for image matching, and has been widely adopted in different vision tasks. In this work, we exploit the second-order relations between

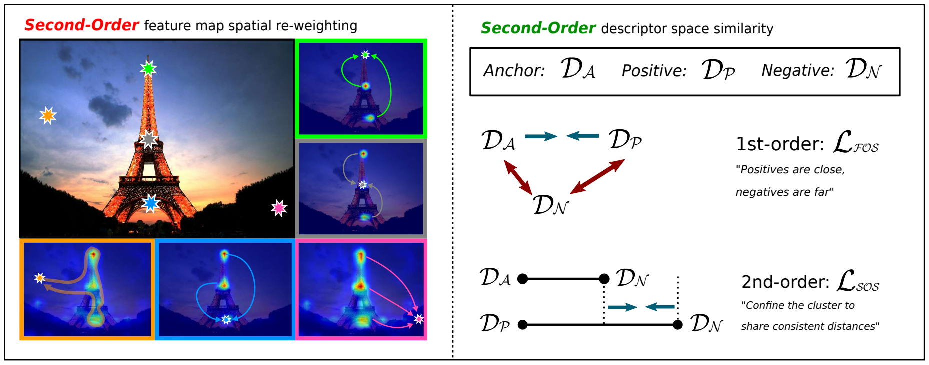
Second-Order feature map spatial re-weighting
Second-Order descriptor space similarity
Anchor: $\mathcal{D}_\mathcal{A}$ Positive: $\mathcal{D}_\mathcal{P}$ Negative: $\mathcal{D}_\mathcal{N}$
$\mathcal{D}_\mathcal{A}$
$\mathcal{D}_\mathcal{P}$
$\mathcal{D}_\mathcal{N}$
1st-order: $\mathcal{L}_{\mathcal{FOS}}$
"Positives are close, negatives are far"
$\mathcal{D}_\mathcal{A}$
$\mathcal{D}_\mathcal{N}$
$\mathcal{D}_\mathcal{P}$
$\mathcal{D}_\mathcal{N}$
2nd-order: $\mathcal{L}_{\mathcal{SOS}}$
"Confine the cluster to share consistent distances"

Fig. 1: Illustration of our SOLAR (**S**econd-**O**rder **L**oss and **A**ttention for image **R**etrieval) descriptor. **Left.** We exploit second-order spatial relations, re-weighting the feature maps to give a better global representation of the image. **Right.** We also apply second-order similarity of learning discriptor distances during training of SOLAR.

features at different spatial locations and combine with second-order descriptor similarity to improve feature descriptors for image retrieval and matching. This is illustrated in Fig. 1. On the left, we learn optimal relative feature contribution spatially (colours of the stars correspond to the frame borders showing the attention for that location). On the right, we use second-order similarity in the descriptor space to make the distance between clusters consistent.

Our main contributions are the following:

a) We combine the second-order spatial attention and the second-order descriptor loss to improve image features for retrieval and matching.

b) We show how to combine second-order attention for consecutive feature maps at different resolution to improve the descriptors and we perform a thorough ablation study on its effects.

c) We demonstrate that the combination of second-order spatial information and similarity loss generalises well in the context of local and global descriptor learning.

d) We validate our method with extensive evaluation on two public benchmarks for image retrieval and matching, showing significant improvements compared to the state-of-the-art.

2 Related Work

Methods for image retrieval [2,18,35,36,37] and place recognition [3,11,31] can be divided into two broad categories: *local aggregation* and *global single-pass*. Most methods prior to deep-learning were based on *local aggregation*, *e.g.* Bag-of-Words (BoW) [43] which aggregates a set of handcrafted, SIFT-like [9,25] local features into a single global vector [17,18,19,20,35,36,43,48,50]. While many of the *local aggregation* methods carried-over into the deep-learning era [31,44,45], the CNNs [16,23,42] with highly expressive feature maps [12] provided an effective approach for global descriptor encoding. Early attempts were mostly hybrid methods, exploring CNN features as direct analogies to local descriptors and aggregating them with similar techniques [1,4,44]. Later works showed that CNN feature maps can be embedded into a descriptor with a *single-pass* of a pooling operation [15,38,39,51], while matching the level of performance from *local aggregation* methods. We group these methods into *global single-pass*.

Local Aggregation methods generally consist of two steps. First, local features are detected and described by hand-crafted operators such as SIFT [25] and SURF [9], or CNN-based local descriptors [4,31]. Second, the descriptors are combined into a compact vector. Early works on BoW assigned local descriptors to visual words through various size codebooks [43]. They were then encoded with matching techniques *e.g.* Hamming Embedding [18], Fisher Kernels [33,34] and Selective Match Kernels [48]; or with aggregation techniques *e.g.* k-means [30,35] and VLAD [19,20]. With the advent of CNN descriptors [46,47,57], learnt features [4,14,29,31] led to substantial improvements in challenging, large-scale retrieval benchmarks [31,37]. Some hybrid methods also learn local-to-global encoding [1,5]. A recent state-of-the-art *local aggregation* system [45] considers features only from regions-of-interest [40], filtering out the irrelevant ones such as the sky, background and moving objects.

Global Single-Pass methods, in contrast, do not separate the extraction and aggregation steps. Instead, the global descriptor is generated by a single forward-pass through a CNN. Notice that even though hybrid methods use CNN features as local descriptors followed by *local aggregations* [1,29], thus generating the global descriptor through a forward-pass of a CNN, we do not consider them to be strictly *global single-pass*, as an individual local representation is still required and aggregated with a handcrafted encoding technique. In order to aggregate a feature map from a CNN, either a general [12] one or fine-tuned on retrieval-specific datasets [39], a global pooling operation must be applied. Various *global single-pass* methods differ mostly by the pooling operations, which include Max-pooling [51], SPoC [4], CroW [21], R-MAC [51] and GeM [39]. GeM pooling has been shown to give excellent results in a recent work that optimises a differentiable approximation of the average-precision metric [41].

Second-Order Attention mechanisms proved successful in NLP [52]. It has since gained popularity in various computer-vision tasks, including video classification [53], GANs [58], semantics segmentation [53,59] and person reID [55]. However, it has not been employed for visual representation and descriptor learning, in particular for image retrieval and matching tasks. On the other hand, **Second-Order Similarity** has only recently been introduced to representation learning [47] on local patches by confining the second-order distance in clusters to be similar and distributing them in the area of the unit hypersphere of the descriptor space. Our work is the first to exploit the second-order spatial attention in descriptor learning and to combine it with second-order descriptor loss for learning global image representation for retrieval.

3 Method

In this section, we first present the state-of-the-art Generalised-Mean (GeM) pooling [39] which we then extend with our second-order spatial pooling, followed by second-order similarity loss, whitening and descriptor normalisation.

3.1 Preliminaries

From an input image $I \in \mathbb{R}^{H,W,3}$ processed through a Fully-Convolutional Network (FCN) denoted by θ, we obtain a feature map $\mathbf{f} = \theta(I) \in \mathbb{R}^{h,w,d}$ where h, w and d are height, width and feature dimensionality, respectively. For $h, w > 1$, Generalised-Mean (GeM) pooling was proposed in [39] as a flexible way to aggregate the feature map into a single descriptor vector $\mathbf{D} = \mathrm{GeM}(\mathbf{f}, p)$. The GeM pooling with learnable parameter p is defined as

$$\mathrm{GeM}\left(\mathbf{f},\ p\right) = \left(\frac{1}{N}\sum_{i=0}^{N} f_i^p\right)^{\frac{1}{p}}. \tag{1}$$

3.2 Second-Order Spatial Pooling

Motivation. There are two main motivations for using spatial second-order attention specifically for image retrieval. First, p in Equation 1 is able to adjust each local contribution from $\mathbf{f}$ to the global descriptor $\mathbf{D}$ according the their corresponding feature activation, *i.e. absolute* magnitude of a feature vector, which is considered a first-order measurement. Thus, it assumes the independence of various locations in the map and does not include any *relative* contribution of each spatial feature with respect to the other features.

This is followed closely by the second motivation, where in the case of FCNs such as VGG [42] and ResNet [16], each local feature that contributes to the global descriptor $\mathbf{D}$ has a limited receptive field covering pixels from the input image. Thus, in Equation 1, for a specific f_i, GeM pooling lacks information on its relation to other features $\{f_k : k \neq i\}$ in $\mathbf{f}$.

Therefore we propose to generate a map $\mathbf{f}^{so}$ with local features $f_{i,j}^{so}$ that reflect the correlations between all spatial locations from within $\mathbf{f}^{so}$, hence the 'second-order'. Ideally, this will allow the model to learn the optimal *relative* contribution of each spatial feature to the final descriptor $\mathbf{D}$.

Formulation. Let each location (i, j) in map $\mathbf{f}$ correspond to (i_I, j_I) when projected onto the input image I. Assuming a rectangular receptive field $R = [R_x, R_y]$ each vector $f_{i,j} \in \mathbf{f}$ is a function of the input pixels $I_{\mathcal{R}}$ included in the receptive field R.

To incorporate second-order spatial information into the feature pooling, we adopt the non-local block [53]. A visualisation of the concept is shown in the top left of Fig. 2. First, we generate two projections of feature map $\mathbf{f}$ termed *query*

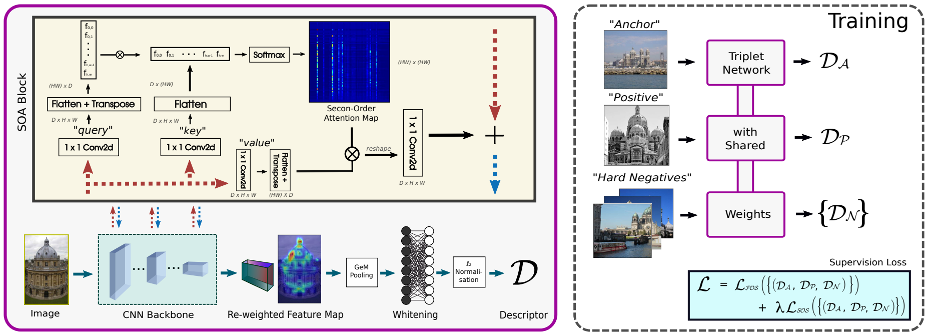

Fig. 2: Pipeline for our proposed global descriptor, SOLAR. We insert a number of **S**econd-**O**rder **A**ttention (SOA) blocks at different levels of a CNN backbone, followed by GeM [39] pooling, whitening and ℓ_2 normalisation. We train SOLAR using a triplet network combining first and second-order descriptor loss.

$\mathbf{q}$ head, and *key* $\mathbf{k}$ head, each obtained through 1×1 convolutions[3]. Then, by flattening both tensors, we obtain $\mathbf{q}$ and $\mathbf{k}$ with shape $d \times hw$. The second-order attention map $\mathbf{z}$ is then computed through

$$\mathbf{z} = \mathrm{softmax}(\alpha \cdot \mathbf{q}^\intercal \mathbf{k}), \tag{2}$$

where α is a scaling factor and $\mathbf{z}$ has shape $hw \times hw$, enabling each $f_{i,j}$ to correlate with features from the whole map $\mathbf{f}$. A third projection of $\mathbf{f}$ is then obtained by *value* head $\mathbf{v}$, in a similar way to $\mathbf{q}$ and $\mathbf{k}$, but resulting in shape $hw \times d$. Finally, $\mathbf{f}^{so}$ map is obtained from the first-order features $\mathbf{f}$ by the second-order attention

$$\mathbf{f}^{so} = \mathbf{f} + \psi\left(\mathbf{z} \times \mathbf{v}\right), \tag{3}$$

where ψ is another 1×1 convolution[3] to control the influence of the attention. Thus, a new feature $f^{so}_{i,j}$ in the second-order map $\mathbf{f}^{so}$ (reshaped to $h \times w \times d$), is a function of features from all locations in $\mathbf{f}$

$$f^{so}_{i,j} = g(\mathbf{z}_{ij} \odot \mathbf{f}), \tag{4}$$

where g denotes the combination of all convolutional operations within the non-local block. We can express each feature $f^{so}_{i,j}$ as a function of the full input image $f^{so}_{i,j} = \phi\left(i, j, I\right)$, viewed from location (i, j), with ϕ as the new FCN with the non-local block(s). Finally, our extended GeM-pooling

$$\mathrm{GeM}\left(\mathbf{f}^{so},\ p\right) = \left(\frac{1}{N}\sum_{i=0}^{N} {f_i^{so}}^p\right)^{\frac{1}{p}} \tag{5}$$

incorporates second-order information from feature correlations. This is referred to as the **S**econd-**O**rder **A**ttention (SOA) block in the remainder of the paper.

[3] We omit Batch-Norm, ReLU and channel reduction for simplicity. Please refer to our code for the exact model details: http://github.com/tonyngjichun/SOLAR

3.3 Second-Order Similarity Loss

First-Order Similarity. The triplet loss is a standard formulation for learning first-order descriptors [8,27,46]. Given a set of triplets formed by anchor, positive and negative images, their corresponding global descriptors are denoted as $\{(\mathbf{D}_a, \mathbf{D}_p, \mathbf{D}_n)\}$. The triplet loss with margin m can be considered as first-order in the descriptor space

$$\mathcal{L}_{FOS} = \frac{1}{|\{(\mathbf{D}_a, \mathbf{D}_p, \mathbf{D}_n)\}|} \sum_{\{(\mathbf{D}_a, \mathbf{D}_p, \mathbf{D}_n)\}} \max\left(0, \|\mathbf{D}_a - \mathbf{D}_p\|^2 - \|\mathbf{D}_a - \mathbf{D}_n\|^2 + m\right) \tag{6}$$

Second-Order Similarity. Following SOSNet [47] in local features, a second-order similarity loss can also be applied to global descriptors. We hard-mine negative pairs as in [39] and calculate the SOS loss for our descriptors

$$\mathcal{L}_{SOS} = \frac{1}{|\{(\mathbf{D}_a, \mathbf{D}_p, \mathbf{D}_n)\}|} \sum_{\{(\mathbf{D}_a, \mathbf{D}_p, \mathbf{D}_n)\}} \left(\|\mathbf{D}_a - \mathbf{D}_n\|^2 - \|\mathbf{D}_p - \mathbf{D}_n\|^2\right)^{\frac{1}{2}} . \tag{7}$$

The final objective function is a combination of first and second-order loss for global descriptors obtained with second-order spatial attention balanced by λ

$$\mathcal{L} = \mathcal{L}_{FOS} + \lambda \mathcal{L}_{SOS}. \tag{8}$$

3.4 Descriptor Whitening

Whitening operation is crucial for obtaining well performing descriptors. While the original work in GeM [39] used a linear projection for descriptor whitening [26], recent experiments[4] show superior results from whitening operation learnt end-to-end. We follow this new approach, by inserting a bias-enabled fully-connected layer after GeM pooling with ℓ_2-norm, and train it end-to-end.

3.5 Network Architecture and Training

The pipeline of our proposed method is shown in Fig. 2. The SOA blocks are insert-able at any feature maps (including intermediate ones), as they serve as learnt feature attention mechanisms. During training all triplets are passed through shared weight networks. Hard-negative mining is also performed at the start of every epoch from a random pool of negatives and it is assured that no negatives from each triplet are from the same scene / landmark class. This is to provide high sample variability from within the mini-batch. Details are described in Section 6.

[4] http://github.com/filipradenovic/cnnimageretrieval-pytorch

4 Results on Large-Scale Image Retrieval

In this section, we present results of SOLAR on large-scale image retrieval tasks and compare to the existing methods, both *local aggregation* and *global single-pass.*

4.1 Datasets

Google Landmarks 18 (GL18) [45] is an extension to the original Kaggle challenge [31] dataset. It contains over 1.2 million photos from 15k landmarks around the world. These landmarks cover a wide-range of classes from historic cities to modern metropolitan areas to nature scenery. GL18 also contains over 80k bounding boxes singling out the most prominent landmark in each image. In this work it serves as a semi-automatically labelled training dataset.

Revisited Oxford and Paris [37] is the commonly used dataset for evaluating the performance of global descriptors on large-scale image retrieval tasks. Oxford [35] and Paris [36] datasets were recently revisited by removing annotation errors and adding new images. The Revisted-Oxford ($\mathcal{R}$Oxf) and Revisited-Paris ($\mathcal{R}$Par) datasets contain 4,993 and 6,322 images respectively, and each with 70 queries by a bounding box depicting the most prominent landmark in that query. The evaluation protocol is divided into three difficulty levels – *Easy*, *Medium* and *Hard.* The mean average precision (mAP) and mean precision at rank 10 (mP@10) are usually reported as performance metrics. The supplementary 1M-distractors ($\mathcal{R}$1M) database contains 1-million extra images to test the robustness of descriptors, using the same protocols and metrics as in $\mathcal{R}$Oxf-$\mathcal{R}$Par.

4.2 Comparison to the State-of-the-Art on Image Retrieval

SOTA. Recent works on large-scale image retrieval [41,45,56] select GeM [39] trained on the SfM120k dataset with the contrastive loss as the baseline for *global single-pass* methods. However, an update on the GitHub repo by GeM's authors[4] sets the new state-of-the-art results from GeM trained on the GL18 [45] dataset, with the triplet loss as in Equation 6. This setting outperforms the recent method that proposed the AP-loss [41] trained on GL18, when evaluated on $\mathcal{R}$Oxf-$\mathcal{R}$Par [37]. Therefore, unlike other recent papers, we select GeM [39] trained on GL18 with the triplet loss as our baseline, and we denote it ResNet101-GeM [SOTA] in Table 1. We also advocate the use of GL18 training dataset as the new standard protocol for large-scale image retrieval. The inconsistency of training sets that can be observed across different works makes it difficult to assess what performance gains can be attributed to the proposed methods, rather than the training sets.

Comparison of SOLAR against other state-of-the-art image retrieval methods on the $\mathcal{R}$Oxf-$\mathcal{R}$Par [37] data is presented in Table 1. By adding SOA blocks, we achieve state-of-the-art mAP and mP@10 performance, and improve by a large margin all other *global single-pass* methods, for both *Medium* and *Hard*

Table 1: Large-scale image retrieval results of our proposed second-order method against the state-of-the-art on $\mathcal{R}$Oxf-$\mathcal{R}$Par [37] and their respective $\mathcal{R}$1M-distractors sets. We evaluate against the *Medium* and *Hard* protocols with the mAP and mP@10 metrics. For *global single-pass* methods, the first term refers to the backbone CNN. [O] denotes results from off-the-shelf networks pretrained on Imagenet. Our method uses ResNet101 with SOA† denoting the best configuration described in Table 2. SOLAR† is the full proposed method including the **S**econd-**O**rder similarity **L**oss

	Method	Medium								Hard							
		$\mathcal{R}$Oxf		$\mathcal{R}$Oxf+$\mathcal{R}$1M		$\mathcal{R}$Par		$\mathcal{R}$Par+$\mathcal{R}$1M		$\mathcal{R}$Oxf		$\mathcal{R}$Oxf+$\mathcal{R}$1M		$\mathcal{R}$Par		$\mathcal{R}$Par+$\mathcal{R}$1M	
		mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10
Local Agg.	HesAff-rSIFT-ASMK* [49]	60.4	85.6	45.0	76.0	61.2	97.9	42.0	95.3	36.4	56.7	25.7	42.1	34.5	80.6	16.5	63.4
	DELF-ASMK* [45]	65.7	87.9	–	–	77.1	98.7	–	–	41.0	57.9	–	–	54.6	90.9	–	–
	DELF-D2R-R-ASMK* [45]	69.9	89.0	–	–	78.7	99.0	–	–	45.6	61.9	–	–	57.7	93.0	–	–
	— DELF [GL18] [45]	**73.3**	**90.0**	**61.0**	**84.6**	**80.7**	**99.1**	**60.2**	**97.9**	**47.6**	**64.3**	**33.6**	**53.7**	**61.3**	**93.4**	**29.9**	**82.4**
Global Single-Pass	AlexNet-GeM [39]	43.3	62.1	24.2	42.8	58.0	91.6	29.9	84.6	17.1	26.2	9.4	11.9	29.7	67.6	8.4	39.6
	VGG16-GeM [39]	61.9	82.7	42.6	68.1	69.3	97.9	45.4	94.1	33.7	51.0	19.0	29.4	44.3	83.7	19.1	64.9
	ResNet101-R-MAC [15]	60.9	78.1	39.3	62.1	78.9	96.9	54.8	93.9	32.4	50.0	12.5	24.9	59.4	86.1	28.0	70.0
	ResNet101-SPoC [4] [O]	39.8	61.0	21.5	40.4	69.2	96.7	41.6	92.0	12.4	23.8	2.8	5.6	44.7	78.0	15.3	54.4
	ResNet101-CroW [21]	41.4	58.8	22.5	40.5	62.9	94.4	34.1	87.1	13.9	25.7	3.0	6.6	36.9	77.9	10.3	45.1
	ResNet101-GeM [39] [O]	45.8	66.2	25.6	45.1	69.7	97.6	46.2	94.0	18.1	31.3	4.7	13.4	47.0	84.9	20.3	70.4
	ResNet101-GeM [39]	64.7	84.7	45.2	71.7	77.2	**98.1**	52.3	**95.3**	38.5	53.0	19.9	34.9	56.3	89.1	24.7	73.3
	ResNet101-GeM+DAME [56]	65.3	85.0	44.7	70.1	77.1	98.4	50.3	94.6	40.4	56.3	22.8	35.6	56.0	88.0	22.0	69.0
	ResNet101-GeM+AP [41]	67.5	–	47.5	–	80.1	–	52.5	–	42.8	–	23.2	–	60.5	–	25.1	–
	ResNet101-GeM [SOTA] [39]	67.3	84.7	49.5	–	80.6	96.7	57.3	–	44.3	59.7	25.7	–	61.5	90.7	29.8	–
	Ours																
	ResNet101-GeM+SOS	67.6	84.7	50.0	73.1	80.9	96.6	57.6	94.4	44.9	60.1	26.2	42.9	61.9	91.0	30.3	78.9
	ResNet101+SOA†	68.6	85.7	51.3	74.7	81.4	96.6	58.8	94.6	46.9	62.7	28.3	46.0	63.7	91.9	32.4	80.9
	ResNet101+SOLAR†	**69.9**	**86.7**	**53.5**	**76.7**	**81.6**	97.1	**59.2**	94.9	**47.9**	**63.0**	**29.9**	**48.9**	**64.5**	**93.0**	**33.4**	**81.6**

protocols. Adding the Second-Order Loss (denoted by SOLAR†), the results are further improved by 1%. SOLAR outperforms mAP of the baseline in the most challenging *Hard* protocol for $\mathcal{R}$Oxf and $\mathcal{R}$Par by significant 3.6% and 3.0% gains respectively, as well as 3.3% and 2.7% in mP@10. Our method also outperforms the state-of-the-art *local aggregation* method of DELF-D2R-R-ASMK* in mAP on $\mathcal{R}$Oxf-*Hard* by 0.3%, $\mathcal{R}$Par-*Medium* by 0.9% and $\mathcal{R}$Par-*Hard* by 3.2%.

For $\mathcal{R}$-1M, SOLAR also achieves the state-of-the-art performance across *global single-pass* methods, outperforming in mAP the SOTA by 4.0% on $\mathcal{R}$Oxf-*Medium*, 4.2% on $\mathcal{R}$Oxf-*Hard*; and by 1.9% on $\mathcal{R}$Par-*Medium*, 3.6% on $\mathcal{R}$Par-*Hard*. Compared to ResNet101-GeM+AP [41] the improvements are even higher (6.0%, 6.7%, 6.7% and 8.3%). As for *local aggregation*, SOLAR still achieves comparable results in the $\mathcal{R}$-1M set and even outperforms DELF-D2R-R-ASMK* by 3.5% in mAP for $\mathcal{R}$Par-*Hard*.

Speed & Memory Costs. It should be noted that the memory requirement for *local aggregation* descriptors is much higher than for *global single-pass* e.g. 27.6GB as reported in DELF-D2R-R-ASMK* [45] *vs.* 7.7GB for GeM [39] & SOLAR descriptors in the $\mathcal{R}$1M-distractors set. SOLAR also runs with a significantly faster speed compared to DELF-D2R-R-ASMK*, *i.e.* 0.15s processing time per image *vs.* >1.5s on a Titan Xp GPU. The SOAs in SOLAR only cause an extra 7.4% cost in inference time compared to GeM. For the $\mathcal{R}$-1M distractors set, the extraction time difference is a significant 1.5 days *vs.* weeks required for DELF-D2R-R-ASMK*. Hence, SOLAR is much more suitable for large-scale

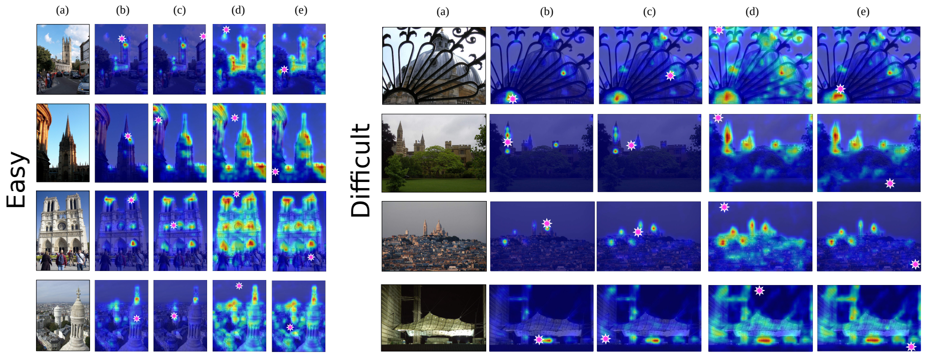

Fig. 3: Qualitative examples of second-order attention maps on the $\mathcal{R}$Oxf-$\mathcal{R}$Par dataset [37]. Each row depicts (a): the source image and four corresponding second-order attention maps obtained for specific spatial locations (marked by pink stars). For each example, four spatial pixel locations are selected – (b): on the dominant landmark, (c): on a secondary landmark, (d): on the sky and (e): on another background part other than the sky. **Left:** easy examples. **Right:** difficult examples.

retrieval tasks given its scalability when compared to *local aggregation* methods, as well as the performance when compared to *global single-pass* methods.

Moreover, we observe that during training the network converges faster and leads to higher performance on the benchmarks when training **only** the SOAs and the whitening layer, *i.e.* freezing backbone weights. Not only does this greatly reduce the training time, it also indicates that the SOAs are optimised for *re-weighting* the features, as will be described in the following section.

4.3 Qualitative Retrieval Results

We visualise the effects of second-order feature map re-weighting in Fig. 3. For locations in the background ((d) & (e)), the attention from that feature is sparsely distributed within the main landmark(s). On the other hand, when the feature is located within a landmark ((b) & (c)), the attention is then on highly distinctive regions including informative features from outside of its receptive field.

This is visible on both, easy examples (**left** in Fig. 3), where there is a clear landmark with distinctive features at similar scales located in the centre and occupies a significant portion of the image, as well as challenging examples (**right** in Fig. 3). For example, the top right example has significant occlusion; in the second and third row the landmark is far-away and a large portion of the image is background; and in the bottom row with night-time image. We can see that even for these hard examples, the second-order attention maps are consistent. This provides qualitative evidence that the spatial re-weighting of feature maps, through second-order attentions, is able to assist the network in learning relative contributions from various features into the final descriptor.

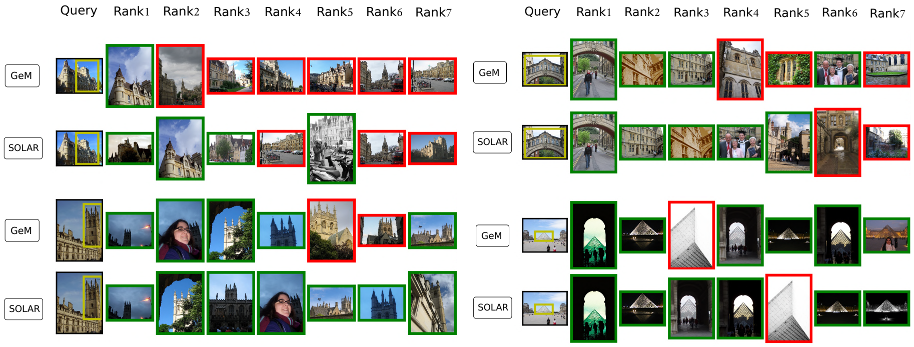

Fig. 4: Qualitative comparison between the baseline GeM (top) and SOLAR (bottom).

We also compare the results from image retrieval in Fig. 4 on very challenging examples in $\mathcal{R}$Oxf-*Hard* [37]. The rows for each example show the query bounding box in yellow, and the Top-7 ranked retrieved images by the baseline ResNet101+GeM [SOTA] [39] and our ResNet101+SOLAR†, with green and red borders denoting correct and incorrect retrievals. While GeM performs reasonably well on these examples, it has a tendency to rank high the images containing some similar features, resulting in more false positives. On the other hand, SOLAR is able to leverage the global correlation from the second-order attentions to increase, in the top few ranks, the number of correct (green) retrievals.

5 Ablation Study

In this section we evaluate the impact of SOLAR on descriptor performance. We first show how SOLAR leads to learning the optimal feature contribution for pooling a global descriptor from the feature map. Next, we break it down into the two second-order components. Lastly, we extend SOLAR to patch datasets to show that it generalises well to local descriptors for image matching task.

5.1 Optimal Feature Contribution

In Section 4.3, we have shown in Fig. 3, that SOAs are effectively *re-weighting* individual feature contributions into the global descriptor based on their uniqueness within the image. Fig. 4 shows examples of improved retrieval results by SOLAR compared to GeM. In this section, we conduct a detailed quantitative assessment on the advantages over GeM in optimal feature contributions.

In Fig. 5 we compare the performance of the baseline (ResNet101-GeM [SOTA]) *vs.* SOLAR for different values of p-norm in Equation 1. We show the mAP of both methods on the *Hard* and *Medium* protocols of $\mathcal{R}$Oxf-$\mathcal{R}$Par [37] for p ranging from $p = 1$ (*i.e.* equal contribution) to $p = 100$ (*i.e.* focused on the strongest features). Note that p is a learnable parameter, we therefore

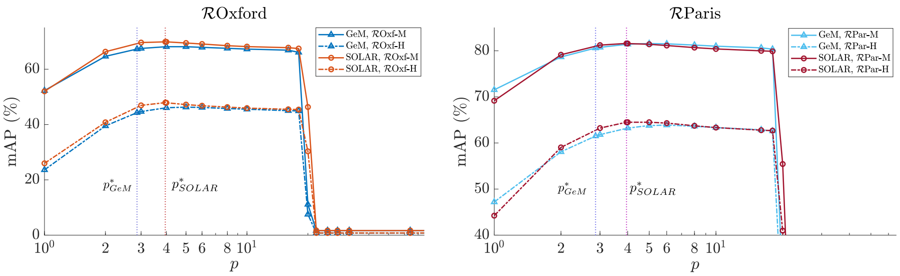

Fig. 5: Comparison of mAP against p on $\mathcal{R}$Oxf-$\mathcal{R}$Par between SOLAR *vs.* GeM.

mark the p learnt by each method with dotted-lines on the graphs. The mAP is clearly increasing as p is raised from 1 to the learnt value, then drops gradually up to $p \approx 20$, after which mAP rapidly decreases to a very weak performance. For high values of p, GeM-pooling approaches Max-pooling [51]. However, $\lim_{p \to \infty} f_i^p = 0 \; \forall \, |f_i| \leq 1$, causing numerical instabilities in Equation 1. Hence, in the implementation, feature magnitudes are clipped to a minimum of 10^{-6}, explaining why mAPs fall after a threshold of p and differ from Max-pooling [51].

We observe that SOLAR outperforms GeM across most values of p, especially in *Hard* examples of both $\mathcal{R}$Oxf and $\mathcal{R}$Par. More importantly, when comparing the values of p learnt by GeM (p^*_{GeM}) and SOLAR (p^*_{SOLAR}), p^*_{SOLAR} corresponds to the peak of each of SOLAR's mAP curve, while p^*_{GeM} is sub-optimal to the best mAPs. This further supports that our SOAs facilitate learning the optimal relative contributions of each feature to the global descriptor.

5.2 Impact of Second-Order Components on Image Retrieval

The results in Section 4.2 show that by simultaneously exploiting second-order spatial information through the SOA blocks and second-order descriptor similarity through the SOS loss, we greatly improve image retrieval performance. In this section, we perform an ablation study by gradually incorporating separate second-order components in SOLAR, and discuss the results on image retrieval.

In Table 2 we present the impact of adding the second-order loss (SOS) and spatial (SOAs) components, with ResNet101+GeM [SOTA] [39] as the baseline. Firstly, by adding SOS in training, the mAPs improved slightly for $< 1\%$. Then, we look at the effects of adding SOAs into ResNet101 [16], which contains 5 fully-convolutional blocks `conv1` to `conv5_x`. In retrieval, the input image typically has high resolution (1000+ pixels on longer side), inserting SOA blocks before `conv4_x` is computationally too expensive given the $\mathcal{O}(n^2)$ complexity of Equation 2. Table 2 shows that our proposed SOA insertions improve retrieval mAP for 0.93% with SOA$_4$, 1.15% with SOA$_5$ and 1.78% with SOA$_{4,5}$. This shows that fine-tuning SOAs alone are more effective than retraining the backbone with SOS. More importantly, we observe that addition of consecutive SOAs is beneficial and that the improvement brought by fine-tuning on SOA$_5$ is higher

Table 2: Ablation study of second-order components on $\mathcal{R}$Oxf-$\mathcal{R}$Par [37]. We use ResNet101-GeM [SOTA] [39] as baseline and incrementally add second-order loss and attention components. Results are in mAP for the *Medium* and *Hard* protocols.

Second-Order Component(s)		Medium		Hard	
		$\mathcal{R}$Oxf	$\mathcal{R}$Par	$\mathcal{R}$Oxf	$\mathcal{R}$Par
None (Baseline)	ResNet101-GeM [SOTA]	67.3	80.6	44.3	61.5
Loss (SOS)	ResNet101-GeM+SOS	67.6	80.9	44.9	61.9
Spatial (SOA)	ResNet101+SOA$_4$	68.2	81.0	45.7	62.3
	ResNet101+SOA$_5$	68.3	81.3	45.9	62.8
	ResNet101+SOA$_{4,5}$	68.6	81.4	46.9	63.7
Both (SOLAR)	ResNet101+SOLAR	**69.9**	**81.6**	**47.9**	**64.5**

than SOA$_4$. We believe that this is due to for large images, where the spatial second-order information is still rich and fine-grained even at the last feature map. As SOA$_5$ re-weights the last feature map before GeM pooling, it adds second-order spatial information directly into the global descriptor, resulting in a better performance.

Lastly, combining SOS and SOA (*i.e.* SOLAR) gives the best mAPs, and the gain by SOS on SOA ($> 1\%$) is more than that of SOS on baseline ($< 1\%$). This further supports that the two second-order components complement each other.

5.3 Generalisation to Image Matching with Local Descriptors

To validate the generalisation ability of SOLAR besides retrieval with global descriptors, we further test it on local descriptor learning. Local patches have different statistics than images, containing less semantic information. However, some degree of structure is still present in patches, thus spatial correlation is still informative [28]. Therefore, we train a local descriptor network with the proposed spatial SOAs. With the second-order similarity included in local SOSNet [47], it is straightforward to directly insert SOAs into SOSNet.

Datasets. In contrast to image retrieval, there are several tasks in different benchmarks to evaluate the performance of local descriptors. Most frequently used are the UBC Patches [54] and HPatches [7], as well as other localisation benchmarks that test both feature detectors and descriptors simultaneously.

UBC Patches [54], consists of three scenes (*liberty*, *notredame*, and *yosemite*) from which corresponding patches are extracted. Models are trained on one scene and tested on the other two for evaluation. Previous works [8,27,28,46,47] report the false positive rate at 95% recall (**FPR@95**) on the 100K test pairs. However, the performance on this dataset has saturated, and the limitations of the **FPR@95** metric have also been pointed out [6]. Moreover, the evaluation task for UBC is different in nature from retrieval. Therefore, we leave the results for UBC in the supplementary material and use UBC data only for training, which is a standard protocol for the HPatches benchmark.

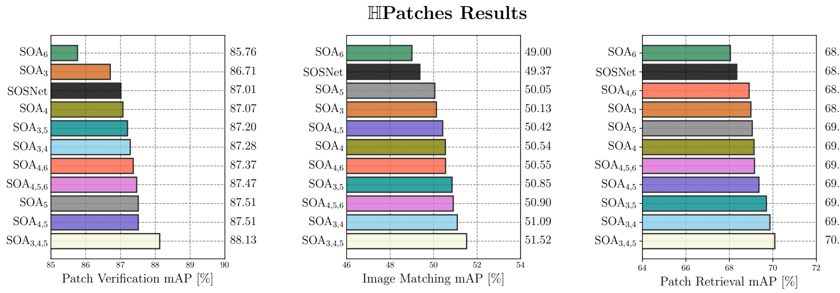

Fig. 6: Patch description performance on HPatches. Each of the configurations is denoted as SOA followed by the numbers indicating layers in SOSNet [47] backbone after which the blocks are inserted. We train all models with the *liberty* subset of UBC and select the model with the lowest average FPR@95. Patches are resized to 32×32.

HPatches [7] contains over 1.5 million patches extracted from 116 scenes with varying viewpoint and illumination. There are three evaluation tasks: *Patch Verification*, *Image Matching* and *Patch Retrieval*.

Impact of SOA at Different Layers. SOSNet [47] uses the L2-Net [46] architecture as the backbone. There are 7 convolutional layers in L2-Net which takes a 32×32 grayscale input patch and outputs a local descriptor with dimensionality of 128. The L2-Net architecture is presented in the supplementary material. The SOA block can be inserted at each intermediate feature map except for Layer-7, as the spatial dimension is reduced to 1×1 only. The earlier the SOA block(s) is inserted, the higher the resolution and more second-order information can be exploited. However, this comes at two costs. First, the complexity of Equation 2 is $\mathcal{O}(n^2)$, where n is the product of the two spatial dimensions. Second, the channel depth is shallower at early layers (32 in the first two *vs.* 128 in the final three layers), *i.e.* each spatial feature in the early layers is less informative.

The results on HPatches with our SOLAR patch descriptors are presented in Fig. 6. To investigate how second-order spatial information changes in patch description, we insert 1 to 3 SOA blocks from between Layers-3 to 7 of L2-Net (Layers-1 & 2 add too much computational cost), giving the set of results $\{\text{SOA}_3$, SOA_4, SOA_5, SOA_6, $\text{SOA}_{3,4}$, $\text{SOA}_{3,5}$, $\text{SOA}_{4,5}$, $\text{SOA}_{4,6}$, $\text{SOA}_{3,4,5}$, $\text{SOA}_{4,5,6}\}$.

Models are trained on the *liberty* subset of the UBC dataset [54] following standard protocols. We select the best model according to the average **FPR@95** on *notredame* and *yosemite* for each SOA configuration. Fig. 6 shows that SOAs generally improve *Patch Retrieval* mAP, up to 1.75% over SOSNet. The only exception is SOA_6 and is due to low spatial resolution of this feature map (only 8×8) compared to large images in Section 5.2, resulting in less informative second-order spatial correlation. This poses a more difficult optimisation task for the SOAs at the final feature levels. We notice that SOAs on consecutive levels ($\text{SOA}_{3,4} > \text{SOA}_{3,5}$ for 0.17%, $\text{SOA}_{4,5} > \text{SOA}_{4,6}$ for 0.45%), and across different scales ($\text{SOA}_{3,5} > \text{SOA}_{4,5}$ for 0.34% despite having fewer parameters) are both beneficial to retrieval, further validating the results from Section 5.2. The results on *Patch Verification* and *Image Matching* are consistent with *Patch Retrieval*, especially with the ordering *w.r.t.* different SOA configurations. This shows that our SOLAR descriptor also extends well to describing local patches, generalising well between tasks of image retrieval and matching.

6 Implementation Details

GeM+SOLAR. We start with ResNet101-GeM [39] pre-trained on GL18[5] and fine-tune the SOAs and the whitening layer with Equation 8. We train for a maximum of 50 epochs on the same GL18 [45] dataset using Adam [22] with an initial learning rate of $1e^{-6}$ ($1e^{-4}$ for p) and exponential decay rate of 0.01. For each epoch 2000 anchors are randomly selected. The triplets are formed, for every anchor, with 1 positive and 5 hard-negatives mined from 20,000 negative samples, each from a separate landmark, yielding 5 triplets $\{(\mathbf{D}_a, \mathbf{D}_p, \mathbf{D}_n)\}$ for Equations 6 and 7. The batch-size is 8. We use margin $m = 1.25$ for the triplet loss and $\lambda = 10$ for SOS loss. At test time, we follow [39] by passing 3 scales $[1, \sqrt{2}, \frac{1}{\sqrt{2}}]$ to the network and taking the average of the output descriptors.

SOSNet+SOAs. We re-implemented SOSNet [47] with the details in the original paper to serve as a baseline (100 epochs max). SOAs are inserted and trained with identical settings. All experiments are implemented in PyTorch [32]. For GeM+SOLAR†, fine-tuning takes roughly 12 hours across 4 1080Ti GPUs. For SOSNET+SOAs, each training takes roughly 5 hours on a single 1080Ti GPU.

7 Conclusion

In this work, we propose SOLAR, a global descriptor that utilises second-order information through both spatial attention and descriptor similarity for large-scale image retrieval. We conduct detailed quantitative and qualitative studies on the impact of incorporating second-order attention that learns to effectively re-weight feature maps, and combine with the second-order information from descriptors similarity to produce better representation for retrieval. We extend the SOLAR approach to local patch descriptors and show that it improves upon the current state-of-the-art without extra supervision, proving that such second-order combination generalises to different type of data. SOLAR achieves state-of-the-art image retrieval performance on the challenging $\mathcal{R}$Paris+1M benchmark compared to similar *global single-pass* methods by a large margin of 3.6% as well as outperforms *local aggregation* methods by 3.5%, while running at a fraction of both time and memory costs. Our approach also improves state-of-the-art for local descriptors in HPatches benchmark by 1.75%.

Acknowledgement. This work was supported by UK EPSRC EP/S032398/1 & EP/N007743/1 grants. We also thank Giorgos Tolias for providing $\mathcal{R}$-1M results of ResNet101-GeM [SOTA] in Table 1.

[5] `http://cmp.felk.cvut.cz/cnnimageretrieval/data/networks/gl18/`

References

1. Arandjelović, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: NetVLAD: CNN architecture for weakly supervised place recognition. In: CVPR (2016)
2. Arandjelović, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: CVPR (2012)
3. Arandjelović, R., Zisserman, A.: DisLocation: Scalable descriptor distinctiveness for location recognition. In: ACCV (2014)
4. Babenko, A., Lempitsky, V.: Aggregating deep convolutional features for image retrieval. In: ICCV (2015)
5. Babenko, A., Slesarev, A., Chigorin, A., Lempitsky, V.: Neural codes for image retrieval. In: ECCV (2014)
6. Balntas, V., Lenc, K., Vedaldi, A., Tuytelaars, T., Matas, J., Mikolajczyk, K.: Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. TPAMI (2019)
7. Balntas, V., Lenc, K., Vedaldi, A., Mikolajczyk, K.: Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In: CVPR (2017)
8. Balntas, V., Riba, E., Ponsa, D., Mikolajczyk, K.: Learning local feature descriptors with triplets and shallow convolutional neural networks. In: BMVC (2016)
9. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded up robust features. In: ECCV (2006)
10. Carreira, J., Batista, J., Sminchisescu, C.: Semantic segmentation with second-order pooling. In: In ECCV (2012)
11. Chen, D.M., Baatz, G., Köeser, K., Tsai, S.S., Vedantham, R., Pylvänäinen, T., Roimela, K., Chen, X., Bach, J., Pollefeys, M., Girod, B., Grzeszczuk, R.: City-scale landmark identification on mobile devices. In: CVPR (2011)
12. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Li, F.F.: ImageNet: A large-scale hierarchical image database. In: CVPR (2009)
13. Gao, Y., Beijbom, O., Zhang, N., Darrell, T.: Compact bilinear pooling. In: CVPR (2016)
14. Gong, Y., Wang, L., Guo, R., Lazebnik, S.: Multi-scale orderless pooling of deep convolutional activation features. In: ECCV (2014)
15. Gordo, A., Almazán, J., Revaud, J., Diane, L.: Deep image retrieval: Learning global representations for image search. In: ECCV (2016)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
17. Jégou, H., Chum, O.: Negative evidences and co-occurrences in image retrieval: the benefit of PCA and whitening. In: ECCV (2012)
18. Jégou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometry consistency for large scale image search. In: ECCV (2008)
19. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: CVPR (2010)
20. Jégou, H., Perronnin, F., Douze, M., Sánchez, J., Pérez, P., Schmid, C.: Aggregating local images descriptors into compact codes. TPAMI (2012)
21. Kalantidis, Y., Mellina, C., Osindero, S.: Crossdimensional weighting for aggregated deep convolutional features. In: ECCV Workshops (2016)
22. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: NeurIPS (2012)

24. Lin, T., RoyChowdhury, A., Maji, S.: Bilinear CNN models for fine-grained visual recognition. In: ICCV (2015)
25. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. In: IJCV (2004)
26. Mikolajczyk, K., Matas, J.: Improving descriptors for fast tree matching by optimal linear projection. In: ICCV (2007)
27. Mishchuk, A., Mishkin, D., Radenović, F., Matas, J.: Working hard to know your neighbor's margins: Local descriptor learning loss. In: NeurIPS (2017)
28. Mukundan, A., Tolias, G., Chum, O.: Explicit spatial encoding for deep local descriptors. In: CVPR (2019)
29. Ng, J.Y.H., Yang, F., Davis, L.S.: Exploiting local features from deep networks for image retrieval. In: CVPR Workshops (2015)
30. Nistér, D., Stewénius, H.: Scalable recognition with a vocabulary tree. In: CVPR (2006)
31. Noh, H., Araujo, A., Sim, J., Weyand, T., Han, B.: Image retrieval with deep local features and attention-based keypoints. In: ICCV (2017)
32. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: An imperative style, high-performance deep learning library. In: NeurIPS (2019)
33. Perronnin, F., Liu, Y., , Sánchez, J., Poirier, H.: Large-scale image retrieval with compressed fisher vectors. In: CVPR (2010)
34. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: ECCV (2010)
35. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: CVPR (2007)
36. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: CVPR (2008)
37. Radenović, F., Iscen, A., Tolias, G., Avrithis, Y., Chum, O.: Revisiting oxford and paris: Large-scale image retrieval benchmarking. In: CVPR (2018)
38. Radenović, F., Tolias, G., Chum, O.: CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples. In: ECCV (2016)
39. Radenović, F., Tolias, G., Chum, O.: Fine-tuning CNN image retrieval with no human annotation. TPAMI (2018)
40. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: NeurIPS (2015)
41. Revaud, J., Almazán, J., Sampaio de Rezende, R., Roberto de Souza, C.: Learning with average precision: Training image retrieval with a listwise loss. In: ICCV (2019)
42. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
43. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: ICCV (2003)
44. Sydorov, V., Sakurada, M., Lampert, C.H.: Deep fisher kernels – end to end learning of the fisher kernel GMM parameters. In: CVPR (2014)
45. Teichmann, M., Araujo, A., Zhu, M., Sim, J.: Detect-to-Retrieve: Efficient regional aggregation for image search. In: CVPR (2019)
46. Tian, Y., Fan, B., Wu, F.: L2-Net: Deep learning of discriminative patch descriptor in euclidean space. In: CVPR (2017)

47. Tian, Y., Yu, X., Fan, B., Fuchao, W., Heijnen, H., Balntas, V.: SOSNet: Second order similarity regularization for local descriptor learning. In: CVPR (2019)
48. Tolias, G., Avrithis, Y., Jégou, H.: To aggregate or not to aggregate: Selective match kernels for image search. In: ICCV (2013)
49. Tolias, G., Avrithis, Y., Jégou, H.: Image search with selective match kernels: Aggregation across single and multiple images. In: IJCV (2015)
50. Tolias, G., Furon, T., Jégou, H.: Orientation covariant aggregation of local descriptors with embeddings. In: ECCV (2014)
51. Tolias, G., Sicre, R., Jégou, H.: Particular object retrieval with integral max-pooling of CNN activations. In: ICLR (2016)
52. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
53. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: CVPR (2018)
54. Winder, S.A., Brown, M.: Learning local image descriptors. In: CVPR (2007)
55. Xia, B.N., Gong, Y., Zhang, Y., Poellabauer, C.: Second-order non-local attention networks for person re-identification. In: ICCV (2019)
56. Yang, T.Y., Nguyen, D.K., Heijnen, H., Balntas, V.: DAME WEB: DynAmic MEan with Whitening Ensemble Binarization for landmark retrieval without human annotation. In: ICCV Workshops (2019)
57. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: LIFT: Learned invariant feature transform. In: ECCV (2016)
58. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. In: ICML (2019)
59. Zhu, Z., Xu, M., Bai, S., Huang, T., Bain, X.: Asymmetric non-local neural networks for semantic segmentation. In: ICCV (2019)