Descriptor Learning Using Convex Optimisation

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman

Visual Geometry Group, University of Oxford

Abstract. The objective of this work is to learn descriptors suitable for the sparse feature detectors used in viewpoint invariant matching. We make a number of novel contributions towards this goal: first, it is shown that learning the pooling regions for the descriptor can be formulated as a *convex* optimisation problem selecting the regions using sparsity; second, it is shown that dimensionality reduction can also be formulated as a *convex* optimisation problem, using the nuclear norm to reduce dimensionality. Both of these problems use large margin discriminative learning methods. The third contribution is a new method of obtaining the positive and negative training data in a weakly supervised manner. And, finally, we employ a state-of-the-art stochastic optimizer that is efficient and well matched to the non-smooth cost functions proposed here. It is demonstrated that the new learning methods improve over the state of the art in descriptor learning for large scale matching, Brown *et al.* [2], and large scale object retrieval, Philbin *et al.* [10].

1 Introduction

Feature descriptors are an important component of many computer vision algorithms. In large scale matching, such as the Photo Tourism project [12], and large scale image retrieval [9], the discriminative power of descriptors and their robustness to image distortions is a key factor in the performance. During the last two decades a plethora of descriptors have been developed, with SIFT [6] certainly being the most widely used. Most of these methods are hand-crafted, though recently machine learning techniques have been applied to learning descriptors for wide-baseline matching [2] and image retrieval [10]. However, although these methods succeed in improving over the performance of SIFT, they use non-convex learning formulations and this can result in sub-optimal models being learnt.

In this paper we propose a new framework that, by leveraging on recent powerful methods for large scale learning of sparse models, can learn descriptors much more efficiently and effectively than previous techniques. First, we reformulate the learning of the *shape* of the spatial pooling regions of a descriptor as the problem of selecting a few optimal shapes among a large set of candidate ones (Sect. 3). The significant advantage compared to previous approaches is that selection can be performed by optimising a sparsity-inducing L^1 regulariser, yielding a *convex* problem and ultimately a globally-optimal solution. Second, we propose to *compress* the resulting descriptors as well as *improve discrimination* by learning a low-rank metric by penalising the nuclear norm of the corresponding matrix (Sect. 4). The nuclear norm is the equivalent of an L^1 regulariser for subspaces. The advantage on standard techniques such as Principal Component Analysis (PCA) is the fact that the low-rank subspace is learnt discriminatively to optimise the matching quality, while still yielding a convex problem and a globally optimal solution.

In our framework the learning of the pooling regions and of the discriminative projections are thus formulated as large-scale max-margin learning problems with non-smooth but convex regularisation terms. In order to optimise such objectives efficiently, we employ (for the first time for this purpose as far as we know) a very effective stochastic learning technique [16] (Sect. 5). There are two additional, more minor, technical contributions. First, we show that, contrary to the common approach, descriptors need not be normalised *a-posterori* after they are computed; instead, a normalisation factor can be computed directly from the image patch once for all (Sect. 2.1). This fact is instrumental to the convex learning of the descriptors as it removes the non-linear normalisation step that affects standard pipelines. Second, we develop a new method for generating examples of matching and mismatching descriptors for the purpose of discriminative learning which is more robust than the one of [10] (Sect. 6.2).

The result is that we have a principled, flexible, and convex framework for descriptor learning that, as we demonstrate in the experiments of Sect. 6, outperforms the descriptor learning of previous work [2, 10] using the authors' own, quite challenging, datasets. Furthermore, the descriptor learning is very efficient (in time and memory requirements) and is able to complete within a few hours on a single core for very large scale problems.

Related work. The proposed framework consists of two independent algorithms for learning descriptor pooling regions and discriminative dimensionality reduction. Most conventional feature descriptors are hand-crafted and use a fixed configuration of pooling regions, e.g. SIFT [6] uses rectangular regions organised in a grid, while DAISY [13] employs a set of multi-size circular regions grouped into rings. In [2] the Powell minimisation technique was employed to optimise the parameters of a DAISY-like descriptor. The corresponding objective is not convex, making the optimisation prone to local extrema.

Discriminative dimensionality reduction can also be related to metric learning, on which a vast literature exists. Of particular relevance here are the large margin formulations designed for nearest-neighbour classification, such as [15], the reason being that feature matching is usually performed by nearest-neighbour search in the descriptor space. While our ranking constraints are similar to those of [15], the authors themselves do not consider simultaneous dimensionality reduction. One approach to reduce dimension is to optimise directly over the projection matrix of the required size [14], but this leads to non-convex objectives. A similar formulation with application to learning descriptors for image retrieval was used in [10]. Another off-the-shelf metric learning technique is Linear Discriminant Analysis (LDA); in [2] it was shown that PCA outperforms LDA if applied to descriptors with already optimised pooling region configuration.

To encourage dimensionality reduction, we utilise the matrix nuclear norm as a convex surrogate of matrix rank. The resulting learning objective is convex, but non-smooth. In [11] the nuclear norm was used for max-margin matrix factorisation, but the implementation resorted to smooth surrogates to simplify the optimisation. We tackle the optimisation problem in principled way and perform large-scale optimisation of the non-smooth objective using the recently developed Regularised Dual Averaging (RDA) method [8,16], which we employ for both L^1-regularised learning of pooling regions and nuclear norm regularised learning of discriminative dimensionality reduction.

2 Descriptor computation

This section describes the computation of the descriptors used in our framework, which closely follows that of [2]. The input is an image patch $\mathbf{x}$ which is assumed to be pre-rectified with respect to affine deformation and dominant orientation. The descriptor $\Psi(\mathbf{x})$ of the patch is a compressed statistics of the local gradient orientations obtained from the following steps:

Smoothing, binning, and normalisation. First, Gaussian smoothing is applied to the patch $\mathbf{x}$. Then the intensity gradient is computed at each pixel and soft-assigned to the two closest orientation bins, weighted by the gradient magnitude as in [2,6,13]. This results in p feature channels for the patch, where p is the number of orientation bins (we used $p = 8$). Finally, a normalisation factor $T(\mathbf{x})$ proportional to the gradient magnitude is computed (Sect. 2.1).

Spatial pooling. The oriented gradients computed at the previous step are spatially aggregated via convolution with a set of kernels (*e.g.,* Gaussians or box filters normalised to unitary mass) with different location and spatial support (Sect. 3); we refer to them as descriptor *Pooling Regions* (PR). Pooling is applied separately to each feature channel, which results in the descriptor vector $\widetilde{\phi}(\mathbf{x})$ with dimensionality pq, where q is the number of PRs. The output of each filter is divided by the pre-computed normalisation factor $T(\mathbf{x})$ and thresholded to obtain responses $\phi(\mathbf{x})$ invariant to intensity changes and robust to outliers.

Discriminative dimensionality reduction. After pooling, the descriptor $\phi(\mathbf{x})$ is compressed into a lower-dimensional vector $W\phi(\mathbf{x})$ by projection through a matrix W learnt to improve descriptor matching (Sect. 4).

The resulting descriptor can be used in feature matching directly or vector-quantised to compute visual words in retrieval applications [10].

2.1 Descriptor normalisation and cropping

After the spatial pooling step, the un-normalised descriptor $\widetilde{\phi}(\mathbf{x})$ is essentially a spatial convolution of gradient magnitudes distributed across p orientation bins. While the descriptor is invariant to an additive intensity change, it does

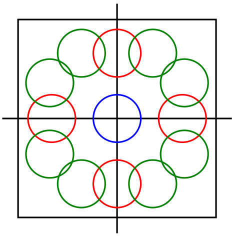

Fig. 1. Pooling region candidate rings. The blue circle shows a ring of a single PR, the red circles – four PRs, the green circle – eight PRs. Each PR is defined by the Gaussian kernel σ and polar coordinates (ρ, α) of its centre relative to the patch centre. The candidate rings are obtained by sampling these parameters in the ranges: $\rho \in [0; \rho_0]$ (one-pixel step), $\alpha \in \{0, \pi/12, \pi/8, \pi/6, \pi/4\}$, $\sigma \in [0.5; \rho_0]$ (half-pixel step), and then reflecting the resulting PRs (ρ_0 is the patch radius).

vary with intensity scaling. This is usually addressed by normalising (and cropping) the descriptor vector *a-posteriori*. Unfortunately, this method introduces a non-linear step involving the aggregate responses of the PRs which makes their learning complicated. Instead, a suitable normalisation factor $T(\mathbf{x})$ can be computed from the patch directly, independently of the particular PR configuration selected by learning.

In particular, we define $T(\mathbf{x})$ as a weighted combination of the mean and standard deviation of gradient magnitude $g(\mathbf{x})$ over the patch:

$$T(\mathbf{x}) = \left(\operatorname{mean}(g(\mathbf{x})) + \nu \operatorname{std}(g(\mathbf{x}))\right)/p \tag{1}$$

where $\nu > 0$ is a parameter which defines how aggressive the cropping (2) is. Given $T(\mathbf{x})$, the response of each PR is normalised and cropped to 1 for each PR independently as follows:

$$\phi_i(\mathbf{x}) = \min\left\{\widetilde{\phi}_i(\mathbf{x})/T(\mathbf{x}), 1\right\} \ \forall i. \tag{2}$$

Based on the experiments on a small hold-out set, ν was set to 1 in all experiments. The threshold (1) should be compared to standard definitions such as [2,6] for which $T(\mathbf{x}) \sim \|\widetilde{\phi}(\mathbf{x})\|_2$ depends on the L^2 norm of the overall descriptor, involving all the PRs. If $\nu = 0$, it is easy to check that the two definitions approximately match if the L^1 norm is used in place of L^2 and the non-overlapping PRs cover the whole patch.

3 Learning pooling regions

In this section we present a framework for learning pooling region configurations. First, a large pool of putative PRs is created, and then sparse learning techniques are used to select an optimal configuration of a few PRs from this pool.

The candidates PRs are generated by sampling a large number of PRs of different size and location within the feature patch. In this paper we consider only reflection-symmetric PR configurations, with each PR being an isotropic Gaussian smoothing kernel. Due to the symmetry, PRs are organised into rings with 8, 4, or 1 PRs (the latter corresponding to a PR centred on the patch). As shown in Fig. 1, a large set of candidate rings $\{\Omega_i\}_{i=1}^{N}$ is obtained by varying the PR geometry. The number of candidate rings N is essentially the number of triplets (ρ, α, σ) (e.g. $N = 4200$ for 41×41 pixel patches as in Sect. 6.2).

Selecting pooling regions. This paragraph shows how to select a few PR rings from the N available candidates such that the resulting descriptor separates *positive* (correctly matched) and *negative* (incorrectly matched) feature pairs. More formally, let ϕ be the descriptor defined by PRs pool subset encoded by the w vector:

$$\phi_{i,j,c}(\mathbf{x}) = \sqrt{w_i}\varPhi_{i,j,c}(\mathbf{x}) \tag{3}$$

where $\varPhi_{i,j,c}(\mathbf{x})$ is the "full" descriptor induced by **all** PRs from the pool $\{\varOmega_i\}$, i indexes over PR rings $\varOmega_i$, j is a PR index within the ring $\varOmega_i$, and c is the feature channel number. The elements of w are non-negative, with non-zero elements acting as weights for the PR rings selected from the pool (and zero weights corresponding to PR rings that are not selected). Due to the symmetry of PR configuration, a single weight w_i is used for all PRs in a ring $\varOmega_i$.

We put the following margin-based constraints on the distance between feature pairs in the descriptor space [15]:

$$d(\mathbf{x},\mathbf{y}) + 1 < d(\mathbf{u},\mathbf{v}) \quad \forall(\mathbf{x},\mathbf{y}) \in \mathcal{P}, (\mathbf{u},\mathbf{v}) \in \mathcal{N} \tag{4}$$

where $\mathcal{P}$ and $\mathcal{N}$ are the training sets of positive and negative feature pairs, and $d(\mathbf{x},\mathbf{y})$ is the distance between descriptors of features $\mathbf{x}$ and $\mathbf{y}$. To measure the distance, the squared L^2 distance is used (at this point we do not consider descriptor dimensionality reduction):

$$\begin{aligned} d(\mathbf{x},\mathbf{y}) =& \|\phi(\mathbf{x}) - \phi(\mathbf{y})\|_2^2 = \sum_{i,j,c} \left(\sqrt{w_i}\varPhi_{i,j,c}(\mathbf{x}) - \sqrt{w_i}\varPhi_{i,j,c}(\mathbf{y})\right)^2 = \\ & \sum_i w_i \sum_{j,c} \left(\varPhi_{i,j,c}(\mathbf{x}) - \varPhi_{i,j,c}(\mathbf{y})\right)^2 = \sum_i w_i \psi_i(\mathbf{x},\mathbf{y}) = w^T \psi(\mathbf{x},\mathbf{y}), \end{aligned} \tag{5}$$

where $\psi(\mathbf{x},\mathbf{y})$ is an N-dimensional vector storing in the i-th element sums of squared differences of descriptor components corresponding to the ring $\varOmega_i$:

$$\psi_i(\mathbf{x},\mathbf{y}) = \sum_{j,c} \left(\varPhi_{i,j,c}(\mathbf{x}) - \varPhi_{i,j,c}(\mathbf{y})\right)^2 \;\; \forall i = 1 \dots N \tag{6}$$

Now we are set to define the learning objective for PR configuration learning. Substituting (5) into (4) and using the soft formulation of the constraints, we derive the following non-smooth *convex* optimisation problem:

$$\underset{w \geq 0}{\operatorname{argmin}} \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{P}, (\mathbf{u},\mathbf{v}) \in \mathcal{N}} \max\left\{w^T\left(\psi(\mathbf{x},\mathbf{y}) - \psi(\mathbf{u},\mathbf{v})\right) + 1, 0\right\} + \mu_1 \|w\|_1 \tag{7}$$

where the L^1 norm $\|w\|_1$ is a sparsity-inducing regulariser which encourages the elements of w to be zero, thus performing PR selection. The parameter $\mu_1 > 0$ sets a trade-off between the empirical ranking loss and sparsity. We note that "sparsity" here refers to the number of PRs, not their location within the image patch, where they are free to overlap. The formulation (7) can be seen as an instance of SVM-rank [4] with L^1 regularisation and non-negativity constraints. It maximises the area under ROC curve corresponding to thresholding the descriptor distance (5). The large-scale optimisation of the objective (7) is described in Sect. 5.

During training, all PRs from the candidate rings are used to compute the vectors $\psi(\mathbf{x}, \mathbf{y})$ for training feature pairs $(\mathbf{x}, \mathbf{y})$. While storing the full descriptor $\varPhi$ is not feasible for large training sets due to its high dimensionality (which equals $p \sum_{i=1}^{N} |\varOmega_i|$, i.e. the number of channels times the number of PRs in the pool) the vector ψ is just N-dimensional.

Once a sparse w is learnt, at test time only PRs corresponding to the non-zero elements of w are used to compute the descriptor. The descriptor normalisation procedure (Sect. 2.1) uses the normaliser (1) which does not depend on the elements of the unnormalised descriptor $\widetilde{\phi}$ (unlike conventional normalisation by the norm of $\widetilde{\phi}$). This ensures that in both training and testing the same normalisation is applied, even though different sets of PRs are used (the whole PR pool during training and the few selected PRs during testing).

4 Learning discriminative dimensionality reduction

This section proposes a framework for learning discriminative dimensionality reduction. The aim is to learn a linear projection matrix W such that (i) W projects descriptors into a lower dimensional space; and, (ii) positive and negative descriptor pairs are separated by a margin in that space.

The first requirement can be formally written as $W \in \mathbb{R}^{m \times n}, m < n$ where m is the dimensionality of the projected space and n is the descriptor dimensionality before projection. The second requirement can be formalised using a set of constraints similar to (4):

$$d_W(\mathbf{x}, \mathbf{y}) + 1 < d_W(\mathbf{u}, \mathbf{v}) \quad \forall (\mathbf{x}, \mathbf{y}) \in \mathcal{P}, (\mathbf{u}, \mathbf{v}) \in \mathcal{N} \tag{8}$$

where d_W is the squared L^2 distance in the projected space:

$$\begin{aligned} d_W(\mathbf{x}, \mathbf{y}) = & \| W\phi(\mathbf{x}) - W\phi(\mathbf{y}) \|_2^2 = (\phi(\mathbf{x}) - \phi(\mathbf{y}))^T W^T W (\phi(\mathbf{x}) - \phi(\mathbf{y})) = \\ & \theta(\mathbf{x}, \mathbf{y})^T A \theta(\mathbf{x}, \mathbf{y}), \end{aligned} \tag{9}$$

with $\theta(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x}) - \phi(\mathbf{y})$, and $A = W^T W$ is the Mahalanobis matrix.

The constraints (8), (9) are not convex in W, but are convex in A. Therefore, optimisation is performed over the convex cone of positive semi-definite matrices [15]: $A \in \mathbb{R}^{n \times n}$, $A \succeq 0$. The positive semidefiniteness constraint ensures that optimising over A is equivalent to optimising over W, i.e. for the learnt matrix A there exists a projection matrix W such that $A = W^T W$. If $\operatorname{rank}(A) = m$, then an $m \times n$ matrix W can be obtained from the eigen-decomposition $A = VDV^T$, where diagonal matrix $D \in \mathbb{R}^{n \times n}$ has m non-zero elements (positive eigenvalues). Let $D_r \in \mathbb{R}^{m \times n}$ be the matrix obtained by removing the zero rows from D. Then W can be constructed as $W = \sqrt{D_r} V^T$. Conversely, if $W \in \mathbb{R}^{m \times n}$ and $\mathrm{rank}(W) = m$, then $\mathrm{rank}(A) = \mathrm{rank}(W^T W) = \mathrm{rank}(W) = m$. Thus, a dimensionality reduction constraint on W can be equivalently transformed into a rank constraint on A. However, the direct optimisation of $\mathrm{rank}(A)$ is not tractable due to its non-convexity. The convex relaxation of matrix rank is described next.

Nuclear norm regularisation. The nuclear norm $\|A\|_*$ of matrix A (also referred to as the trace norm) is defined as the sum of singular values of A. For positive semi-definite matrices the nuclear norm equals the trace. The nuclear norm performs a similar function to the L^1 norm of a vector – in the case of a vector the L^1 norm is a convex surrogate of its L^0 norm, while in the case of a matrix the nuclear norm is a convex surrogate of its rank [3].

Using the soft formulation of the constraints (8), (9) and the nuclear norm in place of rank, we obtain the non-smooth *convex* objective for learning A:

$$\underset{A \succeq 0}{\mathrm{argmin}} \sum_{\substack{(\mathbf{x},\mathbf{y}) \in \mathcal{P} \\ (\mathbf{u},\mathbf{v}) \in \mathcal{N}}} \max\left\{\theta(\mathbf{x},\mathbf{y})^T A\, \theta(\mathbf{x},\mathbf{y}) - \theta(\mathbf{u},\mathbf{v})^T A\, \theta(\mathbf{u},\mathbf{v}) + 1, 0\right\} + \mu_* \|A\|_* \tag{10}$$

where the parameter $\mu_* > 0$ trades off the empirical ranking loss versus the dimensionality of the projected space: the larger μ_*, the smaller the dimensionality. We note that this formulation gives no *direct* control over the projected space dimensionality. Instead, the dimension can be tuned by running the optimisation with different values of μ_*.

5 Regularised stochastic learning

In sections 3 and 4 we proposed convex optimisation problems for learning the descriptor PRs as well as the discriminative dimensionality reduction. However, the corresponding objectives (7) and (10) yield very large problems as the number of summands is $|\mathcal{P}|\,|\mathcal{N}|$, where typically the number of positive and negative matches is in the order of $10^5 - 10^6$ (Sect. 6). This makes using conventional interior point methods infeasible.

To handle such very large training sets, we propose to use *Regularised Dual Averaging* (RDA), the recent method by [8, 16]. To the best of our knowledge, RDA has not yet been applied in the computer vision field, where, we believe, it could be used in a variety of applications beyond the one presented here. RDA is a stochastic proximal gradient method effective for problems of the form

$$\min_w \frac{1}{T} \sum_{t=1}^{T} f(w, z_t) + R(w) \tag{11}$$

where w is the weight vector to be learnt, z_t is the t-th training (sample, label) pair, $f(w, z)$ is a convex loss, and $R(w)$ is a convex regularisation term. Compared to proximal methods for optimisation of smooth losses with non-smooth regularisers (e.g. FISTA), RDA is more generic and applicable to *non-smooth* losses, such as the hinge loss employed in our framework. As opposed to other stochastic proximal methods (e.g. FOBOS), RDA uses more aggressive thresholding, thus producing solutions with higher sparsity. A detailed description of RDA can be found in [16]; here we provide a brief overview.

At iteration t RDA uses the loss subgradient $g_t \in \delta_w f(w, z_t)$ to perform the update:

$$w_{t+1} = \operatorname*{argmin}_w \left(\langle \bar{g}_t, w \rangle + R(w) + \frac{\beta_t}{t} h(w) \right) \tag{12}$$

where $\bar{g}_t = \frac{1}{t} \sum_{i=1}^{t} g_i$ is the average subgradient, $h(w)$ is a strongly convex function such that $\arg\min_w h(w)$ also minimises $R(w)$, and β_t is a specially chosen non-negative non-decreasing sequence. We point out that $\bar{g}_t$ is computed by averaging subgradients across iterations, not samples. If the regularisation $R(w)$ is not strongly convex (as in the case of L^1 and nuclear norms), one can set $h(w) = \frac{1}{2}\|w\|_2^2$, $\beta_t = \gamma\sqrt{t}$, $\gamma > 0$ to obtain the convergence rate of $O(1/\sqrt{t})$.

It is easy to derive the specific form of the RDA update step for the objectives (7) and (10):

$$w_{t+1}^{(i)} = \max\left\{ -\frac{\sqrt{t}}{\gamma} \left(\bar{g}^{(i)} + \mu_1 \right), 0 \right\}, \quad A_{t+1} = \Pi\left(-\frac{\sqrt{t}}{\gamma} \left(\bar{g} + \mu_* \mathrm{I} \right) \right). \tag{13}$$

where $\bar{g}$ is the average sub-gradient of the corresponding hinge loss, I is the identity matrix and Π is the projection onto the cone of positive semi-definite matrices, computed by cropping negative eigenvalues in the eigen-decomposition.

6 Experiments

6.1 Local Image Patches Dataset

In this section we evaluate the proposed descriptor learning framework on the publicly available local image patches dataset [2].

Dataset and evaluation protocol. The dataset consists of three subsets, Yosemite, Notre Dame, and Liberty, each of which contains more than 450,000 image patches (64 × 64 pixels) sampled around Difference of Gaussians (DoG) feature points. The patches are rectified with respect to the scale and dominant orientation. Each of the subsets was generated from a scene for which 3D reconstruction was carried out using multiview stereo algorithms. The resulting depth maps were used to generate 500,000 ground-truth feature pairs for each dataset, with equal number of positive (correct) and negative (incorrect) matches.

To evaluate the performance of feature descriptors, we follow the evaluation protocol of [2] and generate ROC curves based on the distance between feature pairs in the descriptor space. We report false positive rate at 95% recall (FPR95) for the same combinations of training and test sets as in [2]. Note that training and test sets were generated from images of different scenes. Following [2], for training we used 500,000 feature matches of one set, and tested on 100,000 matches of the other (the subsets were made available by the authors).

Table 1. False positive rate (%) (at 95% recall) for learnt pooling regions.

Train set	Test set	Learnt PR, 576-D	Learnt PR, low-dim.	Brown et al. [2]
Yosemite	Notre Dame	**9.71 (576-D)**	11.32 (384-D)	14.43 (400-D)
Yosemite	Liberty	**18.47 (576-D)**	19.78 (384-D)	20.48 (400-D)
Notre Dame	Yosemite	10.65 (576-D)	**10.43 (512-D)**	15.91 (544-D)
Notre Dame	Liberty	**17.81 (576-D)**	18.53 (384-D)	21.85 (400-D)

Results. We compare our learnt descriptors with those of [2] in two scenarios: (i) learning pooling regions (Sect. 3) and (ii) learning discriminative dimensionality reduction on top of learnt PRs (Sect. 4). In both cases the proposed framework significantly outperforms the state of the art [2], reducing the error rate by up to 40%. It is worth noting that the non-linear feature transform we used (Sect. 2) corresponds to the T1b block in [2]. According to their experiments, it is outperformed by more advanced (and computationally complex) steerable filters, which they employed to obtain their best results. This means that we achieve better performance with simpler feature transform, but more sophisticated learning framework.

To learn the descriptors, we randomly split the set of 500,000 feature matches into 400,000 training and 100,000 validation. Training is performed on the training set for different values of μ_1, μ_* and γ, which results in a set of models with different dimensionality-accuracy tradeoff. Given the desired dimensionality of the descriptor, we pick the model with the best performance on the validation set among the ones whose dimensionality is not higher than the requested one. For a fixed training set, the same descriptor (selected on the validation part of the training set) is used for both test sets.

Learning pooling regions. Table 1 compares the error rates reported in [2] (5-th column) with those of the descriptors learnt using our method. The 4-th column corresponds to the descriptors with dimensionality not higher than the one used in [2]; in the 3rd column dimensionality was limited by 576 (arbitrary threshold corresponding to ≤ 9 PR rings selected). In Fig. 2 (left) we plot the error rate of the learnt descriptors as a function of their dimensionality.

The PR configuration of a 576-D descriptor learnt on the Yosemite set is depicted in Fig. 3 (left). Pooling regions are shown as circles with the radius equal to their Gaussian σ (the actual size of the Gaussian kernel is 3σ). The pooling regions' weights are colour-coded. Note that σ increases with the distance from the patch center, which is also specific to certain hand-crafted descriptors, e.g. DAISY [13]. In our case, no prior has been put on the pooling region location and size: they were sampled uniformly, and the optimal configuration was automatically discovered by learning. Also, the PR weights near the patch center are mostly small, which can be explained by the fact that the central part of the feature region usually contains less discriminative information than the outer part [7].

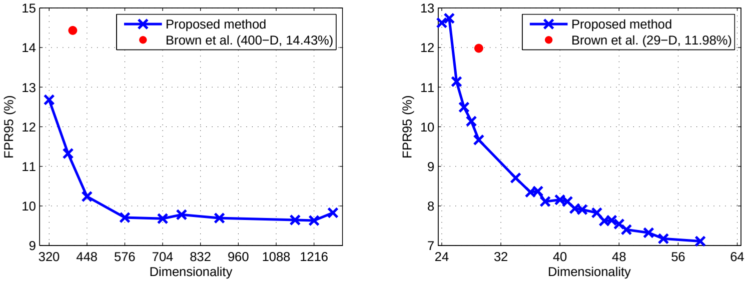

Fig. 2. Dimensionality vs error rate, training on Yosemite, testing on Notre Dame. *Left*: learnt pooling regions. *Right*: learnt projections for 576-D descriptor on the left.

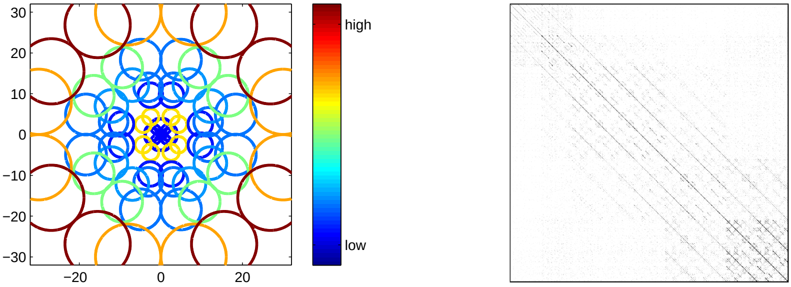

Fig. 3. *Left:* learnt pooling regions' configuration in a 64 × 64 feature region. *Right:* learnt Mahalanobis matrix A corresponding to projection from 576-D to 59-D space (darker pixels correspond to larger values).

Learning discriminative dimensionality reduction. For experiments with dimensionality reduction, we utilised 576-D descriptors evaluated above, and learnt linear projections into lower-dimensional spaces as described in Sect. 4. In Table 2 we compare our results with the best results presented in [2] (5-th column). The proposed algorithm exhibits lower error rates with the same descriptor dimensionality (4-th column) and even lower with increased (but still reasonably low) dimensionality (3-rd column). It should be noted that we obtain projection matrices by discriminative supervised learning, while in [2] the best results were achieved using PCA, which outperformed LDA in their experiments. In Fig. 2 (right) we show the dependency of the error rate on the projected space dimensionality. As can be seen, the learnt projections allow for significant (order of magnitude) dimensionality reduction, while lowering the error at the same time. In Fig. 3 (right) we visualise the learnt Mahalanobis matrix A (Sect. 4) corresponding to discriminative dimensionality reduction. It has a clear diagonal structure, with each diagonal encoding dependencies between pooling regions within the same ring and across the rings. Optimal weights for neighbouring orientation bins in PRs are also learnt.

Table 2. False positive rate (%) (at 95% recall) for learnt projections.

Train set	Test set	Learnt proj., ≤64-D	Learnt proj., low-dim.	Brown *et al.* [2]
Yosemite	Notre Dame	**7.11 (59-D)**	9.67 (29-D)	11.98 (29-D)
Yosemite	Liberty	**16.27 (59-D)**	17.44 (29-D)	18.27 (29-D)
Notre Dame	Yosemite	**10.36 (61-D)**	12.54 (36-D)	13.55 (36-D)
Notre Dame	Liberty	**13.63 (61-D)**	14.51 (36-D)	16.85 (36-D)

6.2 Oxford Buildings and Paris Buildings Datasets

In this section the proposed learning framework is evaluated on challenging Oxford Buildings (Oxford5K) and Paris Buildings (Paris6K) datasets and compared against the SIFT baseline as well as state of the art methods [1, 10].

Dataset and evaluation protocol. The Oxford Buildings dataset consists of 5062 images capturing various Oxford landmarks. It was originally collected for the evaluation of large-scale image retrieval methods [9]. The only available annotation is the set of queries and ground-truth image labels, which define relevant images for each of the queries. The Paris Buildings dataset includes 6412 images of Paris landmarks and is also annotated with queries and labels. Both datasets exhibit a high variation in viewpoint and illumination.

The performance measure is specific to image retrieval tasks and is computed in the following way. For each of the queries, the ranked retrieval results (obtained using the framework of [9]) are compared to the ground-truth landmark labels, which gives a precision-recall curve. Area under the curve is a performance measure for a particular query; averaged across all queries, it gives an integral measure for the whole dataset, called mean Average Precision (mAP). We implemented three flavours of the visual search engine [9]: *tf-idf* uses the tf-idf index computed on quantised descriptors (500K visual words); *tf-idf+sp.* additionally re-ranks the top 200 images using RANSAC-based spatial verification. The third engine (*raw*) is based on nearest-neighbour matching of raw (non-quantised) descriptors and spatial verification.

Following [9], feature detection was performed using the Hessian-Affine detector [7]. The same feature regions were used for both SIFT and the proposed descriptor. To ensure that the proposed descriptor is computed over the same rectified patches as SIFT, we applied the conventional affine region rectification procedure [7], which resulted in 41 × 41 pixel feature patches.

Learning from image collections using latent variables. In this section we outline a novel formulation for learning feature descriptors from image datasets with extremely weak supervision, which can be seen as an application of the more generic learning frameworks of Sect. 3 and 4. In particular, the only information given to the algorithm is that *some* (but unknown) pairs of dataset images contain a common *part*, so that correspondences can be established between them. The assumption is valid for the datasets in question. Computing the correspondences by 3-D reconstruction [2] is not feasible on large scale. A more practical approach of [10] relies on the homography estimation by Nearest-Neighbour (NN) SIFT matching and RANSAC. Then, NN inlier matches can be used as positives, and NN outliers and non-NN as negatives. Unfortunately, this leads to positives that can already be matched by SIFT, while our goal is to learn a better descriptor. The less biased alternative of ignoring appearance and finding correspondences based on geometry only is also problematic as it may pick up occlusions and repetitive structure, which, being unmatchable based on appearance, would disrupt learning. We address these issues by the latent variables formalism described next.

Consider image pairs randomly sampled from the dataset, for which the homographies are automatically estimated as in [10]. For each feature $\mathbf{x}$ of one image, we compute the sets $P(\mathbf{x})$ and $N(\mathbf{x})$ of putative positive and negative matches in another image based on the homographies and the region overlap criterion [7]. We aim at learning a descriptor such that the NN of $\mathbf{x}$ is a positive match from $P(\mathbf{x})$. To account for the cases where $\mathbf{x}$ can not be matched based on its appearance, we introduce a binary latent variable $b(\mathbf{x})$ which equals 0 iff the match can not be established. This leads to the optimisation problem:

$$\arg\min_{\eta,b} \sum_{\mathbf{x}} b(\mathbf{x}) \max \left\{ \min_{\mathbf{y}\in P(\mathbf{x})} d_\eta(\mathbf{x},\mathbf{y}) - \min_{\mathbf{u}\in N(\mathbf{x})} d_\eta(\mathbf{x},\mathbf{u}) + 1, 0 \right\} + R(\eta) \quad (14)$$

$$\text{s.t. } b(\mathbf{x}) \in \{0,1\}, \ \sum_{\mathbf{x}} b(\mathbf{x}) = K$$

where η denotes descriptor parameters (w or A), $R(\eta)$ is the regulariser, and K is the number of samples to use in training, which prevents all $b(\mathbf{x})$ from being set to zero. The objective (14) is related to self-paced learning [5], and its local minimum can be found by alternation. The optimisation is repeated for different values of K, and the resulting model is selected on the validation set.

Results. In our first experiment, we learn the descriptors on the Oxford5K dataset, and then assess the image retrieval performance on it. We note that ground-truth matches are not used in training; instead, the training data is extracted *automatically* as described above. This corresponds to the use case of learning a descriptor for a particular image collection to allow for more accurate retrieval and/or lower memory footprint (if the raw descriptors are used). It should be noted, however, that in the case of more practical tf-idf retrieval the only benefit of lower dimensionality is faster visual words computation, as raw descriptors are not stored in the index.

The mAP values computed using different "descriptor - search engine" combinations are given in Table 3 and compared against the best results reported in [10], both linear and non-linear. We include the results of SIFT (as a baseline), learnt projections on top of SIFT, and learnt pooling regions with and without projection. As can be seen, even linear projections on top of SIFT result in significant (≈ 6%) improvement over SIFT, with smaller dimensionality. Learning

Table 3. mAP on Oxford5K and Paris6K for learnt descriptors, SIFT, and RootSIFT. The performance of our SIFT baseline is better than that reported in [10], making a direct comparison impossible. Therefore, we also show the mAP improvement relative to the corresponding baseline for our methods and [10]. The learnt projections shown are the ones with the best performance on the validation set among $\leq$ 128-D projections.

Descriptor	mAP			mAP improvement (%)		
	raw	tf-idf	tf-idf+sp.	raw	tf-idf	tf-idf+sp.
Oxford5K						
SIFT	0.784	0.636	0.667	-	-	-
RootSIFT	0.798	0.659	0.703	1.8	3.6	5.4
SIFT + Learnt proj., 120-D	0.802	0.673	0.706	2.3	5.8	5.8
Learnt PR, 256-D	0.819	0.664	0.702	4.5	4.4	5.2
Learnt PR + proj., 115-D	**0.841**	**0.709**	**0.749**	**7.3**	**11.5**	**12.3**
Philbin *et al.* [10], linear	N/A	0.636	0.665	N/A	3.8	2.8
Philbin *et al.* [10], non-linear	N/A	0.662	0.707	N/A	8	9.3
Paris6K						
SIFT	0.691	0.656	0.668	-	-	-
RootSIFT	0.706	0.701	0.710	2.2	6.9	6.3
Learnt PR + proj., 115-D	**0.732**	**0.711**	**0.722**	**5.9**	**8.4**	**8.1**
Philbin *et al.* [10], non-linear	N/A	0.678	0.689	N/A	3.5	3

optimal pooling regions leads to further increase of performance ($\approx$ 12%), surpassing that of non-linear SIFT embeddings [10]. This proves the importance of learning the complete descriptor pipeline. We also assess the *generalisation* of the learnt descriptor to different image collections by testing it on the Paris6K dataset (Table 3). Again, we outperform the non-linear projections of [10]; in our case, the drop of mAP improvement when moving to a different image set is smaller than that of [10], which means that our models generalise better.

We also include the results of our implementation of the recently proposed RootSIFT descriptor [1], which is a Hellinger kernel map of SIFT. While the results of our descriptor are better, the advantage of RootSIFT over SIFT underlines the importance of the non-linear mapping in the descriptor computation pipeline. We plan to address this in the future work.

7 Conclusion

In this paper we proposed a generic framework for learning two major components of feature descriptor computation: spatial pooling and discriminative dimensionality reduction. Rigorous evaluation showed that the proposed algorithm outperforms the state of the art on challenging datasets. This was achieved via the use of convex learning formulations coupled with large-scale regularised optimisation techniques. Each of the two presented learning frameworks can be used independently and applied to other computer vision tasks. The source code will be released at `http://www.robots.ox.ac.uk/~vgg/research/learn_desc/`

Acknowledgements. This work was supported by Microsoft Research PhD Scholarship Program and ERC grant VisRec no. 228180. A. Vedaldi was supported by the Violette and Samuel Glasstone Fellowship.

References

1. Arandjelović, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: IEEE Int. Conf. on Comp. Vis. and Pat. Rec. IEEE Press, New York (2012)
2. Brown, M., Hua, G., Winder, S.: Discriminative learning of local image descriptors. IEEE Trans. on Patt. Anal. and Mach. Intell. 33(1), 43–57 (2011)
3. Fazel, M., Hindi, H., Boyd, S.P.: A rank minimization heuristic with application to minimum order system approximation. In: IEEE Amer. Control Conf. pp. 4734–4739. IEEE Press, New York (2001)
4. Joachims, T.: Optimizing search engines using clickthrough data. In: ACM SIGKDD Int. Conf. on Knowl. Disc. and Data Mining. pp. 133–142. ACM Press, New York (2002)
5. Kumar, M.P., Packer, B., Koller, D.: Self-paced learning for latent variable models. In: Lafferty, J., Williams, C.K.I., Shawe-Taylor, J., Zemel, R., Culotta, A. (eds.) Adv. Neur. Inf. Proc. Sys. 23, pp. 1189–1197. Curran Associates, Inc., Red Hook (2010)
6. Lowe, D.: Distinctive image features from scale-invariant keypoints. Int. J. Comp. Vis. 60(2), 91–110 (2004)
7. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A comparison of affine region detectors. Int. J. Comp. Vis. 65(1/2), 43–72 (2005)
8. Nesterov, Y.: Primal-dual subgradient methods for convex problems. J. Math. Prog. 120(1), 221–259 (2009)
9. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: IEEE Int. Conf. on Comp. Vis. and Pat. Rec. IEEE Press, New York (2007)
10. Philbin, J., Isard, M., Sivic, J., Zisserman, A.: Descriptor learning for efficient retrieval. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV (3). LNCS, vol. 6313, pp. 677–691. Springer, Heidelberg (2010)
11. Rennie, J.D.M., Srebro, N.: Fast maximum margin matrix factorization for collaborative prediction. In: De Raedt, L., Wrobel, S. (eds.) Int. Conf. Mach. Learn. pp. 713–719. ACM Press, New York (2005)
12. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3D. ACM Trans. on Graph. 25(3), 835–846 (2006)
13. Tola, E., Lepetit, V., Fua, P.: A fast local descriptor for dense matching. In: IEEE Int. Conf. on Comp. Vis. and Pat. Rec. IEEE Press, New York (2008)
14. Torresani, L., Lee, K.: Large margin component analysis. In: Schölkopf, B., Platt, J., Hoffman, T. (eds.) Adv. Neur. Inf. Proc. Sys. 19, pp. 1385–1392. MIT Press, Cambridge (2007)
15. Weinberger, K., Blitzer, J., Saul, L.: Distance metric learning for large margin nearest neighbor classification. In: Weiss, Y., Schölkopf, B., Platt, J. (eds.) Adv. Neur. Inf. Proc. Sys. 18, pp. 1473–1480. MIT Press, Cambridge (2006)
16. Xiao, L.: Dual averaging methods for regularized stochastic learning and online optimization. J. Mach. Learn. Res. 11, 2543–2596 (2010)