Discriminant Embedding for Local Image Descriptors

Gang Hua
Microsoft Live Labs Research
ganghua@microsoft.com

Matthew Brown
Microsoft Research
brown@microsoft.com

Simon Winder
Microsoft Research
swinder@microsoft.com

Abstract

Invariant feature descriptors such as SIFT and GLOH have been demonstrated to be very robust for image matching and visual recognition. However, such descriptors are generally parameterised in very high dimensional spaces e.g. 128 dimensions in the case of SIFT. This limits the performance of feature matching techniques in terms of speed and scalability. Furthermore, these descriptors have traditionally been carefully hand crafted by manually tuning many parameters. In this paper, we tackle both of these problems by formulating descriptor design as a non-parametric dimensionality reduction problem. In contrast to previous approaches that use only the global statistics of the inputs, we adopt a discriminative approach. Starting from a large training set of labelled match/non-match pairs, we pursue lower dimensional embeddings that are optimised for their discriminative power. Extensive comparative experiments demonstrate that we can exceed the performance of the current state of the art techniques such as SIFT with far fewer dimensions, and with virtually no parameters to be tuned by hand.

1. Introduction

Recent years have seen great advances in the area of local feature matching. Various combinations of region detectors and local image descriptors have been employed in many compelling applications, for example content based image/video retrieval [12, 14], object categorization and recognition [6] and 3D scene reconstruction [15]. The local features used are typically of high dimensionality, e.g. 128 dimensions in the case of SIFT. This can cause problems for matching in large collections of images in terms of speed and scalability. For example, large-scale object recognition [12] and Photo Tourism [15] can both involve matching to millions of local features, requiring many distance computations and large amounts of storage space. A natural question is "can we reduce the dimensionality of the descriptors while maintaining their discriminative powers?"

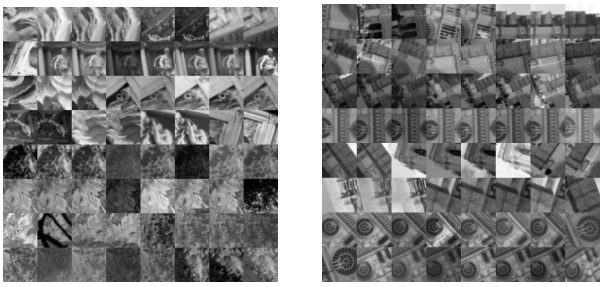
Until recently, most local descriptor designs were carefully crafted by hand, see [11] for a comparative study. Authors have controlled the dimensionality of their descriptors by tuning parameters such as the size of the sampling grid or the number of spatial pooling regions.

An alternative, data-driven approach to local feature matching was suggested by Lepetit and Fua [9]. In this work, the authors learn probability distributions for the keypoint class over a quantisation of the input space. The feature space is hierarchically quantised by thresholding on randomly chosen pixel differences. Another approach to classification using simple features (boxlets rather than pixel differences) was the Viola and Jones face detector [16]. Such techniques are attractive because of their simplicity, but have difficulty scaling to large multi-class problems because of the large number of simple features needed to accurately represent each class. Nearest neighbour classifiers are well suited to such applications, and thus are popular in indexing and recognition. In this work, we adopt this paradigm, attempting to find an optimal data-driven dimension reduction before nearest-neighbour classification proceeds.

A first attempt at data-driven dimension reduction for local features was PCA-SIFT [8]. Instead of performing spatial pooling of the gradient vectors using fixed histogram bins as in the original SIFT design [10], Ke performs a principal component analysis on the gradient patches. Whilst this provides some benefits in reducing the high frequency noise in the descriptors, PCA is not tuned to obtain a subspace that will be discriminative for matching.

In contrast, discriminative techniques such as Fisher analysis (LDA) directly pursue a set of projections that best separate data of different classes. Such techniques have been intensively studied in related areas such as face recognition, for example "Fisher Faces" [1], Locality Preserving Projections [7] and Local Discriminant Embedding [4].

Given recent advances in automatic multi-view matching [15], it is now possible to generate large databases of corresponding image patches, mimicking the large datasets of face images used in the face recognition community. This was exploited by the authors in [17], who tuned the parameters of highly structured local feature descriptors based on



(a) Train (b) Test

Figure 1: Typical input patches from our (a) training and (b) test datasets. The training set patches are taken from Photo Tourism [15] reconstructions of Trevi Fountain and Half Dome. The test set is from Notre Dame. All input patches are 64×64 grayscale. We typically use 10,000 pairs for training, and 100,000 for testing.

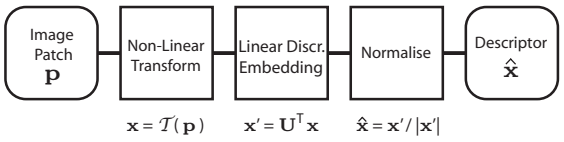


Figure 2: Our descriptor extraction procedure. After performing a non-linear transform $\mathcal{T}(\mathbf{p})$ (section 4.1), we apply a learnt discriminant projection $\mathbf{U}$. Post-normalisation is applied, before nearest neighbour classification of the descriptors $\hat{\mathbf{x}}$.

this data. In this paper we attempt to learn feature descriptors in a more unstructured fashion using linear discriminant embedding (LDE). We present three main contributions:

1. We are the first to exploit non-parametric dimension reduction techniques to learn invariant feature descriptors.

2. We propose a novel algorithm that uses power regularisation to enable stable linear discriminant embedding in high dimensional spaces.

3. We exceed the state of the art in feature matching performance, whilst using far fewer dimensions than previous approaches.

2. Linear Discriminant Embedding

The input to our method is a set of labelled matching and non-matching image patches

$$\mathcal{S} = \{\mathbf{p}_i, \mathbf{p}_j, l_{ij}\} \tag{1}$$

where $\mathbf{p}_i$, $\mathbf{p}_j$ are the input image patches, and l_{ij} is a label equal to 1 if $\mathbf{p}_i$, $\mathbf{p}_j$ constitute a match pair, and 0 otherwise. Following the approach in [17], we apply a set of non-linear transformations to these input patches.

$$\mathbf{x}_i = \mathcal{T}(\mathbf{p}_i) \tag{2}$$

The actual transformations that we use are described in section 3. We have used a range of lifted inputs $\mathbf{x}_i$ including normalised image patches, filter bank outputs and even other feature descriptors such as SIFT or GLOH.

We view the descriptor design problem as one of finding discriminative projections in the space of lifted image patches. We choose a simple objective function

$$J_1(\mathbf{w}) = \frac{\sum_{l_{ij}=0}(\mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_j))^2}{\sum_{l_{ij}=1}(\mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_j))^2} \tag{3}$$

which is the ratio of variance between the non-match and match differences along the direction $\mathbf{w}$. We seek projections

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} J(\mathbf{w}) \tag{4}$$

that maximise this ratio. Writing equation 3 in terms of the covariance matrices gives

$$J_1(\mathbf{w}) = \frac{\mathbf{w}^T\mathbf{A}\mathbf{w}}{\mathbf{w}^T\mathbf{B}\mathbf{w}} \tag{5}$$

where

$$\mathbf{A} = \sum_{l_{ij}=0}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \tag{6}$$

$$\mathbf{B} = \sum_{l_{ij}=1}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \tag{7}$$

Note that since the matches are symmetric ($l_{ij} = l_{ji}$), the means of the match/non-match classes are equal to 0. This motivates our choice of variance ratio in our objective function, instead of techniques such as Fisher LDA [1] that require the means of the two classes to have different values.

It is easy to show that the solution of equation 5 is the largest eigenvector of the generalised eigensystem

$$\mathbf{A}\mathbf{w} = \lambda\mathbf{B}\mathbf{w} \tag{8}$$

To form a linear embedding, we identify the k eigenvectors associated with the largest k generalized eigenvalues λ. This is equivalent to Local Discriminant Embedding [4], but without the local weighting functions.

An alternative objective function is suggested by [7]

$$J_2(\mathbf{w}) = \frac{\sum_{l_{ij}=1}(\mathbf{w}^T\mathbf{x}_i)^2}{\sum_{l_{ij}=1}(\mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_j))^2} \tag{9}$$

(again we have ignored the local weighting functions suggested in [7]). This is equivalent to replacing the non-match covariance $\mathbf{A}$ with a weighted data variance for matches

$$\hat{\mathbf{A}} = \mathbf{X}^T\mathbf{D}\mathbf{X} \tag{10}$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$ is the set of all input vectors. $\mathbf{D} = \text{diag}(k_i)$ is a diagonal matrix where $k_i = \sum_j l_{ij}$ is the number of matches to patch i in the training set. We will name the linear discriminant embedding obtained by optimising $J_1(\mathbf{w})$ (equation 3) as LDE-I, and the embedding obtained from $J_2(\mathbf{w})$ (equation 9) as LDE-II. For clarity of presentation we will use $\mathbf{A}$ and $\hat{\mathbf{A}}$ interchangeably in the remainder of this paper.

2.1. Power Regularisation

A common practical concern with the linear discriminant formulation described above is that it is prone to overfitting. This can occur for projections $\mathbf{w}$ that are essentially in the noise components of the signals, but appear to be discriminative in the absence of sufficient data. This is particularly relevant given the high dimensional inputs that result from the lifted input image patches. Our inputs $\mathbf{x}_i \in \mathbb{R}^n$ typically have $n > 1024$ or more dimensions (see section 3). To tackle this problem we propose a modified cost function

$$J_r(\mathbf{w}) = \frac{\mathbf{w}^T\mathbf{A}\mathbf{w}}{\mathbf{w}^T\mathbf{B}'\mathbf{w}} \tag{11}$$

where $\mathbf{B}' = \mathbf{U}\mathbf{\Lambda}'\mathbf{U}^T$ is a regularised version of $\mathbf{B} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ with its eigenvalues clipped against a minimum value. More specifically, if $\mathbf{\Lambda}_{ii} = \lambda_i$ then $\mathbf{\Lambda}'_{ii} = \lambda'_i$ where

$$\lambda'_i = \max(\lambda_i, \lambda_r) \tag{12}$$

and r is set to the maximum value for which $\{\lambda_i\},\ i \geq r$ accounts for a fraction α of the signal power, i.e.

$$r = \min_{r'}\ s.t.\ \frac{\sum_{i=r'}^{n}\lambda_i}{\sum_{i=1}^{n}\lambda_i} \geq \alpha \tag{13}$$

Note that λ_i has units of $\mathbf{x}^2$ and thus the above equation defines a threshold on the signal to noise power ratio. In our experiments, we use a value of $\alpha = 20\%$.

Note that our power regularisation approach is preferable to simply taking the PCA of $\mathbf{B}$, which would effectively introduce infinite penalties for discriminative directions lying outside the signal subspace of $\mathbf{B}$. Figure 3 shows the effect of power regularisation on the projections learnt via LDE on normalised image patches.

2.2. Orthogonality Constraints

Another issue of interest with the generalized eigen solution to equation 5 is that the pursued projections are not necessarily orthogonal to one another. Previous work [5] has suggested potential benefits to maintaining orthogonality of the projections. This is easily achieved in practice by adding linear constraints to the optimisation criterion of 5. To pursue the k^{th} orthogonal projection $\mathbf{w}_k$, we solve the following constrained optimisation problem

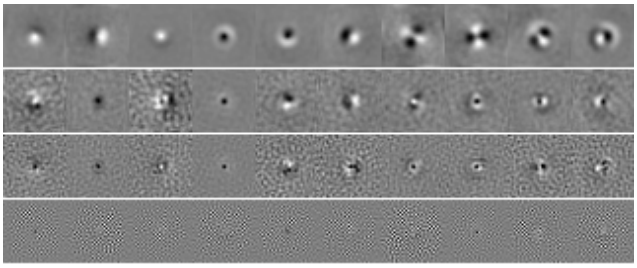


Figure 3: The first 10 projections found using LDE-I on match/non-match normalised patches. From top to bottom $\alpha = 20\%, 10\%, 2\%, 0\%$. Note that the eigenvectors become progressively noisy as the power regularisation is reduced.

$$\begin{aligned} \mathbf{w}^* = \arg\max_{\mathbf{w}} \quad & \frac{\mathbf{w}^T\mathbf{A}\mathbf{w}}{\mathbf{w}^T\mathbf{B}\mathbf{w}} \\ s.t. \quad & \mathbf{w}^T\mathbf{w}_1 = 0 \\ & \cdots \\ & \mathbf{w}^T\mathbf{w}_{k-1} = 0 \end{aligned} \tag{14}$$

where $\mathbf{w}_1, \ldots, \mathbf{w}_{k-1}$ are the set of orthogonal projections we already obtained. By formulating the Lagrangian, it can be shown that the solution can be found by solving another eigenvalue problem

$$\hat{\mathbf{M}}\mathbf{w} = \left((\mathbf{I} - \mathbf{B}^{-1}\mathbf{W}_k\mathbf{Q}_k^{-1}\mathbf{W}_k^T)\mathbf{B}^{-1}\mathbf{A}\right)\mathbf{W} = \lambda\mathbf{w} \tag{15}$$

where

$$\mathbf{W}_k = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{k-1}] \tag{16}$$

$$\mathbf{Q}_k = \mathbf{W}_k^T\mathbf{B}^{-1}\mathbf{W}_k \tag{17}$$

The optimal solution to the k^{th} projection is then the eigenvector associated with the largest eigenvalue of $\hat{\mathbf{M}}$. We omit details of this derivation but refer the interested reader to [5]. Orthogonal projections are attractive for two reasons. Firstly, the extra constraints on the projections may help to avoid overfitting. Secondly, the lack of linear dependence may avoid redundancy in representing the subspace. We use the terms OLDE-I and OLDE-II to refer to the two different orthogonal linear discriminant embeddings obtained using $J_1(\mathbf{w})$ and $J_2(\mathbf{w})$ respectively, subject to orthogonality constraints on $\mathbf{w}_i$.

3. Description of the Datasets

Our training dataset consists of several hundred thousand image patches, sampled by back-projecting 3D points from Photo Tourism reconstructions [15]. To establish a consistent scale and orientation we projected a virtual reference point, slightly offset from the original 3D point, into each image. See [17] for full details. Examples of image patches from our training and test datasets are given in figure 1.

In our experiments, we have used $10,000$ patch pairs (50% matches and 50% non-matches) for training, unless specified otherwise. For testing we use $100,000$ patch pairs from a separate dataset (again 50% matches and 50% non-matches). We use another small dataset of $1,000$ match/non-match pairs as a validation set to choose the optimal number of dimensions for the embedding space.

4. Descriptor Extraction Procedure

Figure 2 shows our descriptor extraction procedure. We begin by applying a variety of non-linear transformations such as bias-gain normalisation, rectified gradients etc. to the input patches. Next we learn discriminative projections of the lifted inputs using the techniques described in section 2. Finally, we normalise the projected descriptor vector to unit length. Nearest neighbour classification is applied in the resulting descriptor space to identify matches and non-matches.

4.1. Non-Linear Feature Transformations

We perform experiments with a variety of non-linear transformations applied to the input patches. This is analogous to the T-blocks in [17].

Normalised Patches we compute bias-gain normalised patches by subtracting the mean and dividing by the standard deviation of the input patch.

Normalised Gradient we compute the x and y gradients of a bias-gain normalised input patch.

T1 we evaluate the gradient at each location in the input patch, and linearly interpolate the gradient magnitude into 4 orientation bins. The interpolation is performed in the orientation of the gradient θ, similar to the scheme used in SIFT.

T2 we evaluate the gradient at each location and transform it into a positive valued 4 vector whose elements are the positive and negative components of the x and y gradients.

T3 we compute 2^{nd} order steerable filter responses at each location using 4 orientations. For each orientation, we compute a 4 vector containing the rectified components of the quadrature pair.

T4 we compute isotropic difference of Gaussian responses at each location for 2 different scales. The outputs are rectified to give 2×2 components for each location.

There are a small number of parameters associated with each of these T-blocks, for example, the pre-smoothing scale and filter widths. We have manually set fixed values for these parameters based on the work of [17]. The non-linear transform outputs $\mathbf{x} = \mathcal{T}(\mathbf{p})$ are normalised to unit length before identification of the linear discriminant subspace.

4.2. Linear Discriminant Embedding

We identify a linear projection $\mathbf{U}$ using the techniques described in section 2, and project the lifted vector $\mathbf{x}$ to this subspace $\mathbf{x}' = \mathbf{U}^T\mathbf{x}$.

4.3. Post Normalisation

We have found that normalising the descriptors to unit length ($\hat{\mathbf{x}} = \mathbf{x}'/|\mathbf{x}'|$) after projection to the discriminant subspace substantially improves the results in most cases. See the results in section 5.3.1.

5. Experiments

We present four main sets of experiments. First we explore dimensionality reduction on the normalised image patches directly. Second, we perform the same experiments using gradients of the patches (analogous to PCA-SIFT). Thirdly, we attempt to find linear embeddings after applying non-linear T-block transformations to the image patches. Finally, we test the ability of our algorithms to reduce the dimensionality of existing descriptors.

In all our experiments, we have applied a synthetic jitter to the patches in both the training and test sets to simulate errors in the interest point localisation process. Such errors have been suppressed in our database since we use projections of bundle adjusted 3D point positions. We apply Gaussian noise to the position, orientation and scale of the patches with standard deviations of 0.25 pixels in position, 11 degrees in orientation and 12% in scale respectively.

To summarise the results of our experiments we plot ROC curves and quote the error rate as the false positive rate at 95% true positives. This is a sensible operating point for our intended application of large scale object recognition, where verifying a potential match is easy but missing a match may be problematic. Results from our four proposed algorithms LDE-I, LDE-II, OLDE-I and OLDE-II are presented. For comparison purposes we also include results for SSD and PCA on $\mathbf{x}$, as well as the baseline results for our own implementation of SIFT applied to the input patches $\mathbf{p}$.

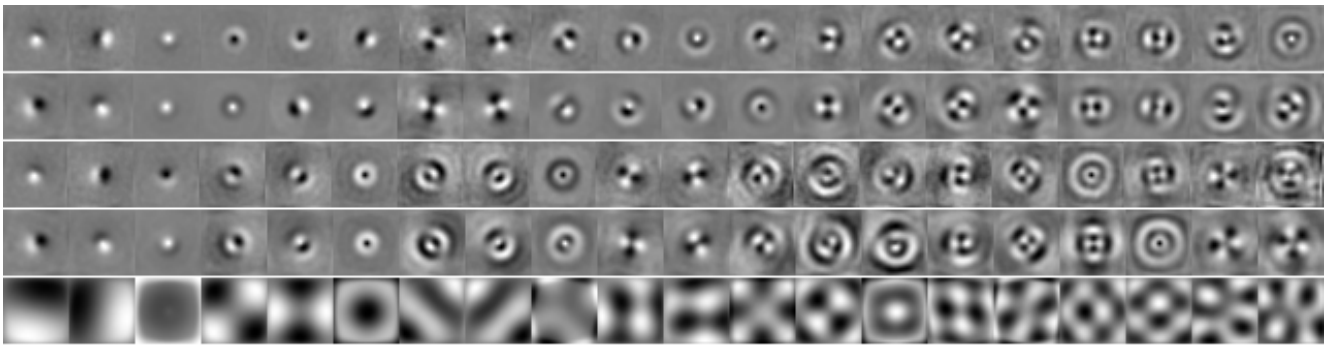


Figure 4: The top 20 projections from each dimensionality reduction method using normalised patches. From top to bottom: LDE-I, LDE-II, OLDE-I, OLDE-II, PCA. These projections were learnt from 500,000 training examples without imposing any prior knowledge on the nature of the projections, other than the power regularisation criterion of Section 2.1. Using only 18 and 14 projections from LDE-I and LDE-II respectively we were able to outperform our own implementation of SIFT (128 dimensions) on our test dataset. The PCA projections gave much worse performance.

	Error Rate $(\%)_{(dimensions)}$		
# Training Pairs	10^4	10^5	5×10^5
SSD	$31.90_{(1024)}$	$31.90_{(1024)}$	$31.90_{(1024)}$
PCA	$31.61_{(285)}$	$28.18_{(20)}$	$28.04_{(28)}$
LDE-I	$8.40_{(17)}$	$6.03_{(21)}$	$5.92_{(18)}$
LDE-II	$9.53_{(24)}$	$7.28_{(24)}$	$5.76_{(14)}$
OLDE-I	$13.962_{(35)}$	$13.05_{(32)}$	$11.80_{(25)}$
OLDE-II	$13.25_{(21)}$	$12.91_{(28)}$	$12.04_{(26)}$

Table 1: Effect of # of training examples for each subspace method applied to normalised patches (95% error rates).

	Error Rate $(\%)_{(dimensions)}$			
	T1	T2	T3	T4
SSD	$35.96_{(1024)}$	$34.92_{(1024)}$	$36.56_{(4096)}$	$52.93_{(1296)}$
PCA	$34.94_{(160)}$	$34.55_{(80)}$	$49.19_{(145)}$	$51.16_{(59)}$
LDE-I	$4.77_{(35)}$	$\mathbf{4.36}_{(29)}$	$\mathbf{4.15}_{(18)}$	$8.39_{(26)}$
LDE-II	$\mathbf{4.40}_{(32)}$	$4.54_{(32)}$	$\mathbf{4.00}_{(24)}$	$7.40_{(29)}$
OLDE-I	$4.58_{(34)}$	$\mathbf{4.45}_{(29)}$	$5.11_{(19)}$	$8.29_{(28)}$
OLDE-II	$4.98_{(37)}$	$4.81_{(35)}$	$4.77_{(23)}$	$8.31_{(25)}$

Table 2: LDE on T-block outputs using non-linear transforms T1-T4.

5.1. Experiments on Normalised Patches

We first present our results using normalised patches. For training we use $500,000$ example pairs drawn randomly from the Trevi-Half Dome dataset. The ROC curves for the different methods are presented in Figure 5.

Note that our linear discriminant algorithms produce substantial improvements over raw SSD (31.9% error rate) and PCA (28.0% error rate). PCA produces only a small improvement over SSD in this case. The LDE-I and LDE-II algorithms both give low error rates of $< 6\%$ false positives at 95% true positives, which is comparable to our implementation of SIFT on this dataset. However, our LDE algorithms use far fewer dimensions than SIFT (14 and 18 dimensions respectively compared to 128 in SIFT).

We visualise the projections learnt by our algorithms in figure 4. Note that the most discriminative projections found by LDE concentrate on the centre of the patch (the interest point location). This reflects the fact that image data further from the interest point is less likely to be reliable for matching. There are many practical reasons for this, for example geometric distortions and occlusions due to viewpoint change, and scale and rotation errors in the interest points. These factors all tend to cause distortions that are larger further away from the interest point location than at the centre.

We also note that our learnt projections strongly resemble the "Jets" of Schmid and Mohr [13], combined with the geometric blur of Berg and Malik [3]. Both of these techniques have been found to be very effective in practical matching problems, and thus it is unsurprising to see that they are learnt as discriminative projections by our technique.

The PCA subspace, whilst capturing some of the spatial integration characteristics of the discriminative projections, has no notion that the centre of the patch is more important than its edge. Also, it tends to focus on axis aligned projections rather than the circular integrations found by LDE. The fact that the latter gives dramatically better performance for matching is not unknown in the research community, where state of the art feature designs such as GLOH [11] and Shape Contexts [2] use log-polar instead of axis aligned summation regions.

A final point to note is that the results of LDE-I and LDE-

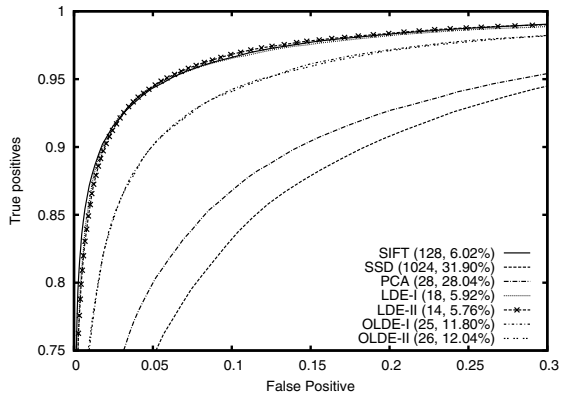


Figure 5: LDE and PCA using normalised patches *vs*. SSD and SIFT. In parenthesis: # of dimensions and 95% error rate.

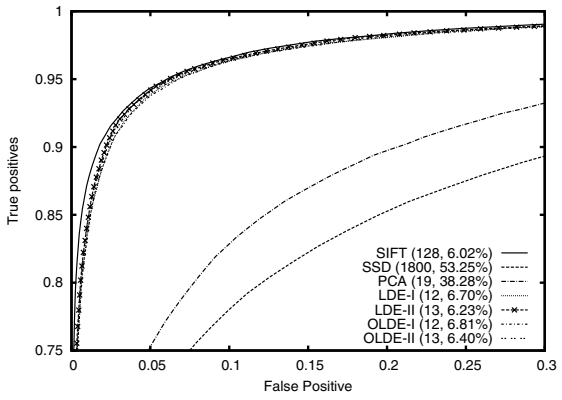


Figure 6: LDE and PCA using gradient patches *vs*. SSD and SIFT. In parenthesis: # of dimensions and 95% error rate.

II are quite similar to each other, as are the results of OLDE-I and OLDE-II. Indeed we find that this is true in general and we take up this issue in the appendix A.

5.1.1 Effect of Number of Training Examples

To understand the effect of the number of training examples on the performance of our proposed methods, we have performed experiments with 10^4, 10^5 and 5×10^5 training pairs using normalized image patches. Table 1 summarizes the 95% error rates. We see substantial gains moving from 10,000 to 100,000 training examples, with diminishing returns thereafter.

5.2. Experiments on Gradient Patches

We now perform a similar analysis with normalised gradient patches as input. This setup is directly comparable to the approach used in PCA-SIFT[1] [8]. The results using 10^5 training pairs are shown in figure 6. We were unable to reproduce the strong results reported in the PCA-SIFT paper. Although we did find small improvements to using PCA (41.5%) over SSD on the patch gradient (53.3%), this result was well below the SIFT baseline (6.0%). However, LDE was again able to significantly improve over PCA and SSD (6.23% error, LDE-II), giving comparable results to those found with normalised image patches 5.1.

5.3. Learning from Non-Linear Filter Responses

In this section, we present results for learning feature descriptors from the output of several different non-linear lifting methods presented in Section 4.1. We resized the training patches to 18×18 before applying the non-linear transformations due to the extra dimensions added by each filter band. Table 2 summarizes the 95% error rate of the different methods using the non-linear transformation methods of T1 to T4.

As we can observe, the proposed methods applied to T1, T2 and T3 all give excellent results with very low dimensionality. They all achieve error rates of around $4-5\%$ with less than 40 dimensions, and thus beat SIFT (6.02%) while using far fewer dimensions. In particular, LDE-II combined with T3 obtains the best error rate of 4.00% with 24 dimensions, followed by LDE-I combined with T3 with an error rate of 4.15% using only 18 dimensions. The top 5 results are highlighted in Table 2. Once again, PCA did not improve significantly over the SSD baseline. We present the ROC curves for the results on T2 and T3 in Figures 7 and 8.

We visualize the top four projections using T1 responses for each subspace learning method in Figure 9. Since the T1 output has lifted the image patch to 4 orientation bands, each projection contains four small blocks arranged from band 1 to band 4.

The projections of the T1 outputs share some similarities with those found for normalised patches (Figure 4) in that they are centrally focussed, but they are more horizontally and vertically oriented. This seems reasonable since the T1 transformation splits the gradient magnitude into axis-aligned orientation bins. Again, the energy of the top PCA projections is distributed evenly over the patch, ignoring the extra discriminative power of the central region.

5.3.1 Effect of Post Normalisation

We found that post normalisation of the descriptor after projection to the discriminative subspace gave a marked improvement in performance in the case of nonlinear feature transformations (T-Blocks in [17]) (Table 3). Post normali-

[1] Although in this experiment we used 32×32 input patches instead of the 41×41 patches used in PCA-SIFT.

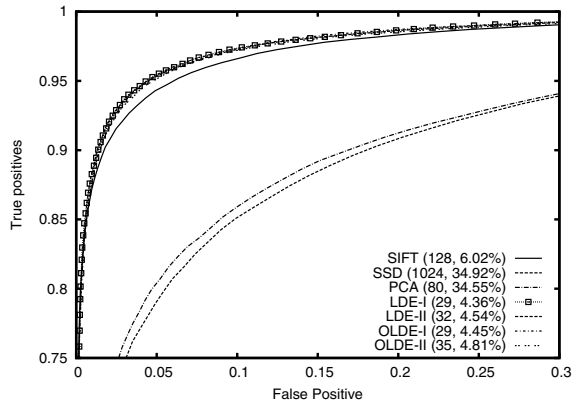


Figure 7: LDE and PCA using T2 outputs *vs*. SSD and SIFT.

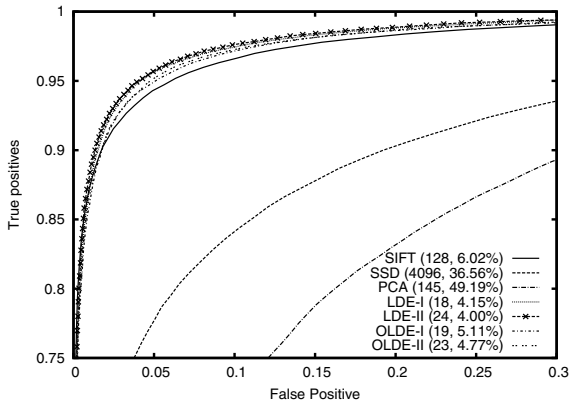


Figure 8: LDE and PCA using T3 outputs *vs*. SSD and SIFT.

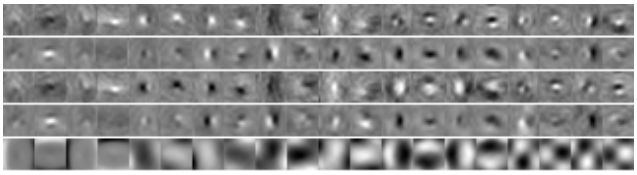


Figure 9: The top 4 projections from each method using T1 outputs. From top to bottom: LDE-I, LDE-II, OLDE-I, OLDE-II, and PCA. Each projection contains four consecutive blocks since the T1 output has 4 orientation bands.

sation also gave small improvements (1-2%) for normalised patches, but actually made things slightly worse in the case of S-blocks [17], as shown in 5.4.

5.4. Dimension Reduction on Feature Descriptors

In this set of experiments, we demonstrate that we can even apply the proposed methods to existing feature descriptors, achieving improved or comparable results with fewer dimensions. In particular, we present the results of applying the proposed methods to four of the best feature descriptors presented in [17]. The descriptors that we use are named T3h-S4-25, T3j-S2-17, T1b-S1-16 and T1c-S2-17 in that paper. We summarize the results in Table 4. Note that the baseline results now correspond to the best result achieved in [17].

In all cases we are able to improve the performance of the feature descriptors whilst using fewer dimensions by using one of our algorithms. Furthermore, the optimal number of dimensions is typically 5-10 times smaller than the original descriptors (i.e., around $30 - 50$ dimensions). Note that the results presented in Table 4 are obtained *without* performing post-normalisation. We found that post-normalisation on embeddings learned from these feature descriptors did not in fact improve performance. This may be because these descriptors have already been subject to SIFT like normalisation (i.e., threshold clipping and then normalization).

6. Conclusions

We have proposed a new discriminative framework for learning local image descriptors. In contrast to all previous work in this area, our scheme is almost parameter free. We demonstrate that our approach produces descriptors with equal or better performance than state of the art approaches, but with 5-10 times fewer dimensions.

A. Approximate Equivalence $J_1(\mathbf{w})$ and $J_2(\mathbf{w})$

We will show that under certain conditions, the objective functions $J_1(\mathbf{w})$ and $J_2(\mathbf{w})$ in Section 2 are approximately equivalent.

Assume a labelled dataset $\mathcal{D} = \{\mathbf{x}_i, g_i\}_{i=1}^n$, where g_i is the group ID. Samples with the same group ID are considered to be matched. Suppose that the data $\mathbf{x}$ has zero mean and covariance $\mathbf{C}$. Our training set $\mathcal{S} = (\mathbf{x}_i, \mathbf{x}_j, \mathbf{l}_{ij})$, is randomly sampled from $\mathcal{D}$ as follows. First, we randomly select $\mathbf{x}_i$, with replacement. To generate a non-match pair, we randomly select $\mathbf{x}_j$ with a different group ID, while to generate a match pair, we randomly draw $\mathbf{x}_j$ from the same group as $\mathbf{x}_i$. In our experiments, we generate an equal number of match and non-match pairs, and the cardinality of each group of matches g_i is variable. However, if instead the following conditions hold, we will see that $J_1(\mathbf{w})$ and $J_2(\mathbf{w})$ are approximately equivalent:

1. The number of matches k_i to input $\mathbf{x}_i$ is a constant k and $k << n$.

2. A large number of input pairs $\mathbf{x}_i, \mathbf{x}_j$ are sampled *independently* from the dataset.

	w/ Normalisation		w/o Normalisation	
	$T1_{(1024)}$	$T2_{(1024)}$	$T1_{(1024)}$	$T2_{(1024)}$
SSD	35.96	34.92	35.96	34.92
PCA	$34.94_{(160)}$	$34.55_{(80)}$	$36.04_{(405)}$	$34.69_{(444)}$
LDE-I	$4.77_{(35)}$	$4.36_{(29)}$	$14.12_{(19)}$	$13.18_{(26)}$
LDE-II	$4.40_{(32)}$	$4.54_{(32)}$	$16.49_{(14)}$	$14.24_{(19)}$
OLDE-I	$4.58_{(34)}$	$4.45_{(29)}$	$11.62_{(28)}$	$10.75_{(26)}$
OLDE-II	$4.98_{(37)}$	$4.81_{(35)}$	$14.30_{(16)}$	$14.94_{(12)}$

Table 3: Effects of post-normalization of descriptors on the performance of embeddings using T-blocks.

	Error Rate $(\%)_{(dimensions)}$			
	T3h-S4-25	T3j-S2-17	T1b-S1-16	T1c-S2-17
Base	$1.99_{(400)}$	$2.51_{(544)}$	$5.50_{(128)}$	$2.98_{(272)}$
PCA	$2.70_{(28)}$	$2.76_{(39)}$	$5.40_{(99)}$	$2.69_{(108)}$
LDE-I	$\mathbf{1.97}_{(31)}$	$\mathbf{1.89}_{(39)}$	$\mathbf{3.76}_{(47)}$	$2.19_{(98)}$
LDE-II	$1.89_{(142)}$	$2.39_{(27)}$	$3.96_{(28)}$	$2.28_{(54)}$
OLDE-I	$2.31_{(37)}$	$2.31_{(41)}$	$4.77_{(27)}$	$\mathbf{2.05}_{(63)}$
OLDE-II	$2.52_{(28)}$	$2.61_{(31)}$	$5.36_{(20)}$	$2.19_{(52)}$

Table 4: Dimension reduction of the descriptors of [17]. Note although the performance is only slightly improved, the number of dimensions is significantly reduced.

For $J_1(\mathbf{w})$, condition 2 implies that $\mathbf{A} = \sum_{l_{ij}=0}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \approx \sum_{i=1}^{n}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$. This is because the number of possible non-match pairs is much greater than the number of possible match pairs, i.e. $\sum_{j=1}^{n} l_{ij} << n$. Furthermore, by the properties of independent random variables, and the central limit theorem, we have $\mathbf{A} \approx \text{covar}(\mathbf{x}_i - \mathbf{x}_j) = 2 \times \text{covar}(\mathbf{x}_i) = 2 \times \mathbf{C}$. Now in the case of $J_2(\mathbf{w})$, we can write $\hat{\mathbf{A}} = \sum_{i=1}^{n} k_i \mathbf{x}_i \mathbf{x}_i^T$. But by condition 1, $k_i = k$ and thus $\hat{\mathbf{A}} = k \times \mathbf{C}$. Since the scaling on $\mathbf{C}$ is arbitrary, we find that $J_1(\mathbf{w})$ and $J_2(\mathbf{w})$ are both equivalent to the following objective, if conditions 1 and 2 are met

$$J_3(\mathbf{w}) = \frac{\sum_{i=1}^{n}(\mathbf{w}^T\mathbf{x}_i)^2}{\sum_{l_{ij}=1}(\mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_j))^2} = \frac{\mathbf{w}^T\mathbf{C}\mathbf{w}}{\mathbf{w}^T\mathbf{B}\mathbf{w}} \qquad (18)$$

where $\mathbf{C}$ is the covariance of all the data $\mathbf{C} = \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T$. This suggests that $J_3(\mathbf{w})$ itself should be a suitable objective, an assertion that we leave for future work. ∎

References

[1] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 19(7):711–720, 1997. Special Issue on Face Recognition.

[2] S. Belongie, J. Malik, and J. Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *Advances in Neural Information Processing Systems 13*. MIT Press, Cambridge, MA, 2000.

[3] A. C. Berg and J. Malik. Geometric blur for template matching. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pages 607–614, 2001.

[4] H.-T. Chen, H.-W. Chang, and T.-L. Liu. Local discriminant embedding and its variants. In *Proc. of IEEE Conf. on Computer Vision and Patter Recognition*, volume 2, pages 846–853, San Diego, CA, June 2005.

[5] J. Duchene and S. Leclercq. An optimal transformation for discriminant and principal component analysis. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 10(6):978–983, November 1988.

[6] R. Fergus, F.-F. Li, P. Perona, and A. Zisserman. Learning object categories from google's image search. In *Proc. of IEEE International Conf. on Computer Vision*, volume 2, pages 1816–1823, Beijing, China, Octobor 2005.

[7] X. He, S. Yan, Y. Hu, P. Niyogi, and H. Zhang. Face recognition using Laplacianfaces. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 27(3):328–340, March 2005.

[8] Y. Ke and R. Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. In *Proc. of IEEE Conf. on Computer Vision and Patter Recognition*, volume 2, pages 506–513, Washington, DC, June 2004.

[9] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 775–781, June 2005.

[10] D. G. Lowe. Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[11] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(0):1615–1630, 2005.

[12] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, New York City, NY, June 2006.

[13] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, May 1997.

[14] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proc. of IEEE International Conf. on Computer Vision*, volume 2, pages 1470–1477, Nice, France, Octobor 2003.

[15] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3D. In *ACM Transactions on Graphics (SIGGRAPH Proceedings)*, volume 25, pages 835–846, 2006.

[16] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, December 2001.

[17] S. A. J. Winder and M. Brown. Learning local image descriptors. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Minneapolis, MN, June 2007.