

Binary Online Learned Descriptors

Vassileios Balntas, *Member, IEEE*, Lilian Tang, *Member, IEEE*, and Krystian Mikolajczyk, *Senior Member, IEEE*

Abstract—We propose a novel approach to generate a binary descriptor optimized for each image patch independently. The approach is inspired by the linear discriminant embedding that simultaneously increases inter and decreases intra class distances. A set of discriminative and uncorrelated binary tests is established from all possible tests in an offline training process. The patch adapted descriptors are then efficiently built online from a subset of features which lead to lower intra-class distances and thus, to a more robust descriptor. We perform experiments on three widely used benchmarks and demonstrate improvements in matching performance, and illustrate that per-patch optimization outperforms global optimization.

Index Terms—Learning feature descriptors, binary descriptors, feature matching, image matching.

1 INTRODUCTION

Significant progress has been made in developing feature descriptors that are either based on floating point arithmetic, such as SIFT [8], SURF [1] and GLOH [10] or on binary strings and hamming distances like BRIEF [3], ORB [13] and BRISK [7].

The various descriptors proposed in the literature differ in design, theory and implementation, but a common approach is the computation of the final feature vector from a fixed set of measurements that are extracted from every described patch. It follows that the measurements are not varied depending on the content of the patch. This is based on important practical considerations which primarily include convenience in using various distance metrics and efficient matching techniques for large scale problems. Moreover, learning based components are trained offline as they are typically too computationally intensive for any online processing. In BRIEF descriptor [3], four different arbitrarily designed configurations of binary tests were evaluated on an entire training set and the best performing configuration was selected. However, intuitively different patch appearances can be best represented by different measurements. For example, the results from [17] show that recognition performance can be improved by adapting the spatial structure of SIFT-based descriptors to each class.

In this paper we propose an approach which combines the advantages of efficient binary descriptors with the improved performance of learning-based descriptors. We demonstrate that there is no single set of measurements that is globally optimal for all patches in a dataset and significant improvement can be gained by adapting the binary tests to the content of each patch. The measurements are first designed to maximize the inter-class distances and then a subset is selected online for each patch to minimize the intra-class distances. This concept is illustrated in Figure 1.

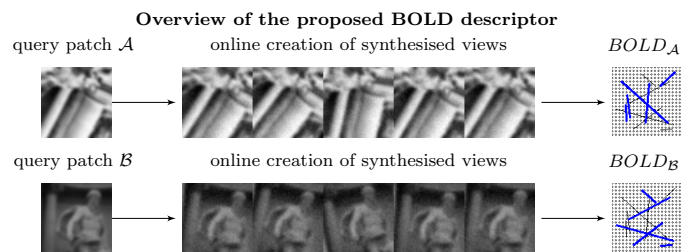


Fig. 1: In contrast to typical approaches that use the same measurements for every patch, we adapt the descriptor online to each patch. The blue line ends indicate the selected binary tests from a common superset based on the measurements from the synthesized views of each patch. Note that although the final descriptor is different for each patch, it consists of a subset of a fixed set of dimensions. This allows efficient sequential matching and common database storage.

The selection is done efficiently in such a way that the extraction time is comparable to other binary descriptors. We evaluate the proposed descriptor on different benchmarks and demonstrate performance that is on par with SIFT, and computational efficiency of BRIEF.

Our approach has been successfully applied for masking unstable features in scale-invariant descriptors [27], motion-invariant descriptors [25] and descriptors for images captured with wide-angle cameras [37], which further validates the benefits of online selection of stable features per patch.

2 RELATED WORK

Large datasets with correspondence ground truth enabled learning methods to be used to improve the descriptor performance [18]. One such approach consists of optimally learning descriptor parameters [20]. Another research direction is learning discriminative projections from high dimensional feature space to subspaces with better discriminating power. In [2], [9] the descriptor optimization is similar to the

- Vassileios Balntas & Krystian Mikolajczyk are with the Department of Electrical and Electronic Engineering, Imperial College London, London, UK
E-mail: v.balntas@imperial.ac.uk
- Lilian Tang is with the Department of Computer Science, University of Surrey, Guildford, UK.

Manuscript received Sep 5, 2016; revised 20 Dec, 2016.










		mAP	success rate %
		0.831	61
		0.835	59
		0.811	65
		0.842	58

Fig. 2: The effect of the randomness on BRIEF intensity tests on the performance. The top row of patches represents the query images, and the 4 bottom rows show the retrieval results from the dataset, where in each row the respective BRIEF descriptor is used. The true positives are shown in green, and the false positives are shown in red. Note that different BRIEF descriptors created with different random seeds, exhibit non-consistent behaviour.

LDA based projections, which simultaneously minimizes intra-class and maximizes inter-class distances, where each patch is consider a class. A similar idea was exploited in [19] where LDA like projections were learnt and applied to gradient based features and optimized thresholds were then used to binarize the dimensions resulting in a binary descriptor. Similarly, the authors of [14] propose a convex optimization for descriptor learning. However, in all these methods, the intra-class is formed by positive examples of correctly matched patch pairs while in LDA by various instances of the same image category / content. LDA projections cannot be learned for each patch independently due to practical complexity issues, *e.g.* inefficient distance calculation and matching. Thus in the case of image patches, discriminant projections are learned globally which leads to limited improvements. Local discriminant projections are expected to give better results if adapted to each class independently.

In the context of binary descriptors, BRIEF was improved in [13] by selecting uncorrelated tests that maximize the variance across training patches. Learning of discriminant and low dimensional spaces has also been applied to binary descriptors. DBRIEF [16] is built by using the inter to intra class distance objective adapted to a binary descriptor. A set of discriminative projections is computed and approximated with a set of predefined dictionaries in order to generate a binary feature vector. The recently proposed BINBOOST descriptor [15] applies boosting to learn a set of binary hash functions that achieve the performance comparable to real-valued descriptors. Both DBRIEF and BINBOOST are not based on binary intensity tests therefore the extraction process is less efficient. A different research direction is to use coding methods to make the descriptors representation compact [4].

Weighted Hamming distance ranking algorithm is proposed in [28] to improve the ranking performance of binary hashing methods. It shows that by assigning different bit-level weights to different hash bits, it is possible to rank two binary codes with the same Hamming distance to a query, at a fine-grained level of the binary codes. It also gives binary hashing methods the ability to distinguish between

the relative importance of different bits. This adaptation was applied to a query code online which is related to our idea of online learnt descriptor. We discuss the relations to this approach in more detail in section 5.1.

The rise of convolutional neural networks as optimisation and representation methods gave remarkable boost to many areas of computer vision including local descriptors. Compared to shallow descriptors considered in this work, CNNs differ in terms of the applied learning techniques, volume of training data and computational efficiency therefore direct comparison shows significant differences in performance and speed. The interest in CNNs based descriptors started from results shown in [29] that the features from the last layer of a convolutional deep network trained on ImageNet can outperform SIFT even though the networks were not specifically optimized for such local representations. End-to-end learning of patch descriptors using Siamese networks and the hinge contrastive loss [30], [31], [32] has recently been re-attempted in several works [29], [33], [34], [35], [36] and consistent improvements were reported over the state of the art descriptors in terms of matching performance. However, their efficiency is still far behind the traditional engineered descriptors and further progress has to be made to make their applications possible. In contrast, this work is focused on both learning and efficiency of local descriptors.

3 INSTABILITY OF RANDOM TESTS

In this section, we present several experimental results that motivate our approach to developing a new locally adapted descriptor. We first show that the subset of intensity tests that are included in a binary descriptor can greatly alter its discriminating ability for specific queries. Secondly, we illustrate the instability of the intensity tests and show that it is related to the internal patch structure. Lastly, we present similar instability results in a tracking by detection based method, that is based on a classifier built on such intensity tests.

3.1 Performance of BRIEF descriptors

We illustrate the potential of local adaptation of the descriptors per patch by manually selecting their optimal sampling patterns. Here we rely on the prior knowledge of the ground truth that we do not have in a typical application but in the following sections we propose an approach to address that issue. Using the BRIEF code available from `OpenCV`, we create 4 different BRIEF descriptors, by changing the random seed used in the random number generation process. This has the effect of altering the locations of the intensity tests that are measured in each descriptor. However, the underlying BRIEF approach and sampling process is not changed.

Figure 2 shows the four BRIEF sampling patterns as well as positive and negative matches obtained with these descriptors. Although the descriptors have 512 intensity tests, only the first 50 are plotted for clarity. We form a set of 500 query patches, together with a true positive matching patch for each of the query patches, totalling a set of 1000 patches. For each patch, we find the nearest neighbour by Hamming distance brute-force search and compare with the ground truth.

The figure shows mean Average Precision and the success rate of matching 1000 patch pairs. The results vary by up to a few percent for different sets of binary tests. This demonstrates that careful sampling can lead to low distance for the positive matches and high distance for the negative ones thus improving the correct matching rate.

3.2 Instability of pairwise intensity features

In this experiment, we illustrate the instability of the intensity test features under very small rotations.

Table 1 presents some example cases of robustness and instability for patch pairs, where the patches differ by a small rotation of (1°). Surprisingly even for such minor rotation which presents no significant change in the view of the patch there are many patches where up to 10% of binary tests flip signs. The patches in pairs are visually very similar, therefore the examples with 10% of bit flips demonstrate the sensitive nature of the binarized intensity tests. In general, we observe that the patch pairs where the intensity tests do not perform well are patches rich in texture and edges. Intuitively, pixels sampled from such patches come from very small uniform regions or their boundaries and even a minor geometric deformation can move the pixel locations across the region boundary thus change the intensity.

TABLE 1: Example cases of test (in)stability. Top row contains patch pairs with 1° rotation where no bit flips occur. Bottom row contains pairs where 10% of intensity tests flip after 1° rotation.

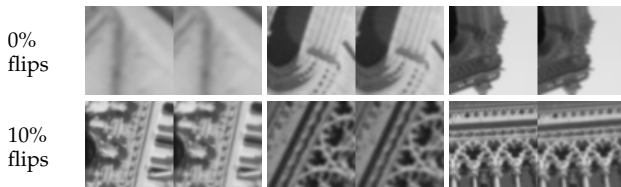


Figure 3 shows the distribution of Hamming distances for a number of pairwise intensity tests. We present three

distributions of intra-class distances, one for real views of matching patches and two for synthetically rotated i.e. for each patch, we create rotated versions of 5° and 10° degrees and compare the binary strings with the original one. As expected, the distributions for the synthetically rotated patches are much more compact than for the real views, which has a long tail of distances due to noise from varying viewing conditions. The overlap with the inter-class distances is significantly increasing with 10° rotation and real views. However, even with 5 degree rotation, we find cases where 40% of the binary tests flip signs. We identify this instability as the main problem of the BRIEF like binary descriptors, since such small transformations are frequent in real world applications (e.g. due to noise introduced by keypoint detectors).

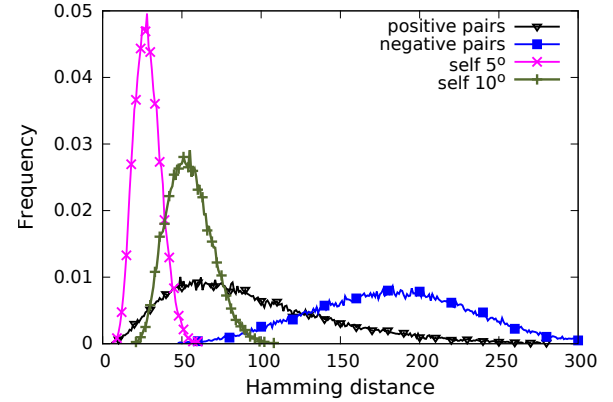


Fig. 3: Distribution of pairwise intensity test that flip their sign under 5° and 10° rotation of a patch, estimated for 10k patches. For comparison, we also plot positive (intra) and negative (inter-class) distance distributions for real patch pairs.

3.3 Tracking performance with intensity features

Due to their efficiency, pairwise intensity tests are often exploited in the context of real time tracking or object detection. In a video with a moving object, small affine transformations are very frequent. We consider this application to further demonstrate the stability issue of the intensity tests. Tracking-Learning-Detection approach [6] used an online learnt object detector based on randomized fern classifier with a set of intensity tests as measurements. The classifier is an essential part of the system that allows to re-detect the object in case the tracker drifts or object temporarily disappears.

In Figure 4 we show how the changes in the intensity tests used to form the classifier significantly impact the tracking results. To evaluate that, we used different seeds in the random test initialization¹. The results are surprising, as they show 5% change between the original code which uses the seed 0, and a different seed e.g. 1. This indicates the potential gain in choosing different locations for the pairwise tests. Better than a random selection should be possible if the stability of the tests can be evaluated prior

1. We used the original TLD implementation <https://github.com/zk00006/OpenTLD>

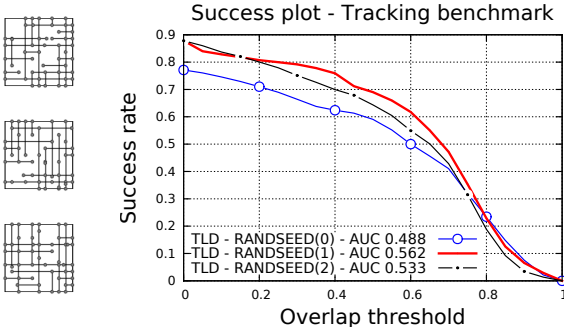


Fig. 4: The effect of altering the intensity tests used in the online learned detector of TLD [6]. (Left) The three different fern classifiers we used in our experiments. (Right) Results using the benchmark of [21]. Note the varying performance of the fern classifiers based on different intensity tests.

to the selection. Furthermore, since the patterns in patches differ, the selection will be more effective if the locations are adapted individually to every patch rather than globally to all of them.

4 LOCAL ADAPTATION OF BINARY DESCRIPTORS

In this section, we discuss the theoretical justifications of our work, and we propose two different approaches to adapt a set of binary features locally to each individual patch. The first method is inspired by the optimization of the ratio of intra to inter-class distances, and the second approach is based on utilizing a specific subset of features for each patch that remain stable under view deformations.

4.1 Locally adapted descriptors

Let $\mathbf{f}_L, \mathbf{f}_R \in \{0, 1\}^D$ represent binary descriptor extracted from patches L, R using D binary tests. Patches $\mathbf{f}_L, \mathbf{f}_R$ are from the same class (e.g. they represent the same interest point from two different views). The hamming distance is then defined as

$$\mathbb{H}(\mathbf{f}_L, \mathbf{f}_R) = \frac{1}{D} \sum_{i=1}^D |\mathbf{f}_{L,i} - \mathbf{f}_{R,i}| \quad (1)$$

Our goal is to identify the unstable bits in \mathbf{f}_L and \mathbf{f}_R . Once this is done we can associate binary masks $\mathbf{m}_L, \mathbf{m}_R \in \{0, 1\}^D$ with $\mathbf{f}_L, \mathbf{f}_R$ respectively, to suppress the contribution from unstable bits during Hamming distance calculation

$$\mathbb{H}_m(\mathbf{f}_L, \mathbf{f}_R, \mathbf{m}_L, \mathbf{m}_R) = \sum_{i=1}^D \mathbf{m}_{L,i} \wedge |\mathbf{f}_{L,i} - \mathbf{f}_{R,i}| + \sum_{i=1}^D \mathbf{m}_{R,i} \wedge |\mathbf{f}_{L,i} - \mathbf{f}_{R,i}| \quad (2)$$

The dimensions that are suppressed in both masks do not contribute to the final Hamming distance. Subsequently, the ℓ_0 -“norm” of the combined masks $\|\mathbf{f}\|_0 = \sum_{n=1}^D (\mathbf{m}_{L,i} \vee$

$\mathbf{m}_{R,i}$) indicates the final dimensionality of the masked descriptors for patches L and R . Note that the masks are adapted independently to each patch hence the dimensionality can differ for different pairs. We term the dimensions that are included in the mask \mathbf{m}_P for a patch P as *stable dimensions*.

To identify the stable dimensions of a given patch P we first explore a technique inspired by LDA, based on covariance of inter and intra class features.

4.2 Learning discriminative descriptors

It has been frequently demonstrated that descriptors perform better when the ratio of the intra- and inter-class distances is maximized. Given a set of labelled matching and non-matching image patches, methods such as [2], [9] seek to find a projection \mathbf{w}^* s.t. $\mathbf{w}^* = \arg \max_{\mathbf{w}} (\mathbf{w}^T \mathbf{A} \mathbf{w}) / (\mathbf{w}^T \mathbf{B} \mathbf{w})$ which is the ratio of the inter \mathbf{A} to intra-class covariance \mathbf{B} along the direction \mathbf{w} . Intuitively, such methods minimize the expected distance between patches annotated as similar and maximize the expected distance between patches annotated as dissimilar. This has been done globally for real-valued descriptors in [2], [9], [16] with the use of a large set of negative and positive pairs of patches in an offline learning process.

In the following we propose an approach that exploits this idea to optimize a binary descriptor for each patch independently.

4.3 Properties of binary tests

Features (dimensions) $\mathbf{f}_{m,i} = \{I(\mathbf{t}_1) > I(\mathbf{t}_2)\}_i$ are binary tests that consist of comparing pixel intensities in pairs of locations \mathbf{t}_1 and \mathbf{t}_2 within the patch. For a grid of $G \times G$ locations within a patch the total number of tests is $M = \binom{G^2}{2}$. The locations are typically generated randomly but further constraints on how test are generated can be exploited. These may include only horizontal and vertical pairs or exclude locations on patch boundaries, large distances between \mathbf{t}_1 and \mathbf{t}_2 etc.

Let $\{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_N\}$ denote a set of binary descriptors of dimensionality D , extracted from N patches which can be arranged in matrix \mathbf{F} of size $N \times D$. Each column c_i with $i \in [1, \dots, D]$ represents a test (dimension) of the binary descriptors and can be viewed as a binary string of length N that follows a Bernoulli distribution with a certain probability of values 1 or 0. Matrix \mathbf{F} can then be expressed as the outcome of N trials of D Bernoulli distributions \mathcal{B}_i . If the mean value of \mathcal{B}_i is ρ_i , then the variance is $\sigma_i = \rho_i(1 - \rho_i)$ where ρ_i is the fraction of 1s and $(1 - \rho_i)$ is the fraction of 0s in column c_i . Variance σ_i of the i^{th} dimension has a direct relation with the Shannon entropy of the binary string of the corresponding column c_i i.e. $\mathcal{E}_i = -\rho_i \cdot \log_2 \rho_i - (1 - \rho_i) \log_2 (1 - \rho_i)$.

A required characteristic of such binary strings is to exhibit a high variance–entropy values if descriptors \mathbf{f}_n belong to different classes and a low variance–entropy values if descriptors belong to the same class. For the former, the discriminative dimensions are the ones where the variance reaches the maximum possible value of 0.25 (entropy reaches 1, $\rho_i = 0.5$). The latter implies that the process that

generates the values for this specific descriptor dimension, is stable and robust to noise, deformations, illumination changes etc. In an ideal case, with a perfect descriptor all columns of intra class descriptors \mathbf{f}_n would have entropy and variance equal to zero. Given \mathbf{F} and Bernoulli distributions $\mathcal{B}_i(\rho_i, \sigma_i)$ associated with dimension (column) i of \mathbf{F} , the expected average distance $\mathbb{E}[\Delta]$ between descriptors in \mathbf{F} is related to the sum of variances σ_i . This can be derived from:

$$\mathbb{E}[\Delta_{intra}] = \frac{1}{D} \sum_{i=1}^D \mathbb{E}[\Delta_i] \quad (3)$$

where $\mathbb{E}[\Delta_i]$ is the expected intra-class distance value for dimension i :

$$\mathbb{E}[\Delta_i] = \frac{1}{N^2} \sum_{m=1}^N \sum_{n=1}^N |\mathbf{f}_{m,i} - \mathbf{f}_{n,i}|_{\oplus} \quad (4)$$

and $|\mathbf{f}_{m,i} - \mathbf{f}_{n,i}|_{\oplus}$ is the Hamming distance between two binary values. Since $|\mathbf{f}_{m,i} - \mathbf{f}_{n,i}|_{\oplus} = (\mathbf{f}_{m,i} - \mathbf{f}_{n,i})^2$ we obtain:

$$\begin{aligned} \mathbb{E}[\Delta_i] &= \frac{1}{N^2} \sum_{m=1}^N \sum_{n=1}^N \mathbf{f}_{m,i}^2 - 2 \frac{1}{N^2} \sum_{m=1}^N \sum_{n=1}^N \mathbf{f}_{m,i} \mathbf{f}_{n,i} \\ &\quad + \frac{1}{N^2} \sum_{m=1}^N \sum_{n=1}^N \mathbf{f}_{n,i}^2 = 2\mathbb{E}[\mathbf{f}_i^2] - 2\mathbb{E}[\mathbf{f}_i]^2 \end{aligned} \quad (5)$$

The variance of dimension i is therefore directly reflected by the fraction of 1s in column i of matrix \mathbf{F} . From the above it is clear that dimensions with high variance increase the intra-class distances, and dimensions with low variance decrease it. Low variance is required for descriptors from the same class (positive patches) and high variance for descriptors from different classes (negative patches).

It was demonstrated in [2] that discriminant projections of SIFT dimensions can be achieved in a two stage process which first diagonalizes the intra-class covariance and then performs a global PCA. Thus the dimensions are decorrelated and oriented along dominant directions in the real-valued space. This process can be adapted to learning of discriminative binary descriptors by first selecting uncorrelated tests that maximize the inter-class distances globally and then by short-listing tests that minimize the intra-class distances locally. Correlation \mathbb{C}_{ij} between tests i and j can be measured on inter-class patches by the Hamming distance between the corresponding columns i and j :

$$\mathbb{C}_{ij} = \left| \frac{2}{N} \sum_{m=1}^N |\mathbf{f}_{m,i} - \mathbf{f}_{m,j}|_{\oplus} - 1 \right| \quad (6)$$

Thus the value of \mathbb{C}_{ij} varies between 0 and 1, with 1 for perfectly correlated tests. Suitable dimensions can be chosen by thresholding this measure.

The first two steps of the process, the global selection of discriminative dimensions and the decorrelation can be done offline from a large set of possible binary tests and random patches. The final selection of dimensions that minimize the intra-class variance has to be done per patch and online, which requires efficient implementation.

4.4 Efficient extraction of online learned descriptors

In this section we present the technical details of our online learned descriptor. This is done in two steps, namely inter-class offline optimization and intra-class online selection of tests.

4.4.1 Global optimization

In global optimization the goal is to identify the subset of discriminative features leading to maximization of inter-class distances. This can be done offline on a large set of N diverse image patches different from the test data. In the case of binary tests, it consists of finding features that give a large variance across inter-class examples as discussed in section 4.3.

It requires calculation of all test responses in each of the N patches. This results in a set of N binary strings of dimensionality M with \mathbf{f}_n representing the bitstring of patch n . \mathbf{F} is a matrix with descriptors \mathbf{f}_n as rows. We calculate the fraction of 1s in column i of \mathbf{F} and sort the columns according to that measure. This ranks high the discriminative tests, which exhibit a high variance across a random selection of patches.

The next step is to select a subset of uncorrelated features. We follow the greedy approach from [13] which starts by selecting the first high variance tests from the ranked list and then searches for another high variance test with the correlation score $\mathbb{C}_{ij} < \tau_C$ (e.g. $\tau_C = 0.2$). The process continues by verifying at each iteration the correlation between the candidate and all selected tests. The selection stops when a defined number G of tests has been found (e.g. $G = 512$).

Note that the global optimization is done offline as it concerns the whole set of possible tests and diverse image patches that represent negative examples in section 4.3.

4.4.2 Local online learning

As demonstrated in [13], [15] a set of globally optimized tests outperform a set of random tests in terms of matching error rates. However, to fully benefit from the LDA-like optimization, intra-class distances have to be minimized. As we show in Figure 2, different subsets of tests minimize the intra-class distances for individual classes of patches and can achieve superior performance compared to the globally optimized features.

We consider each patch as a separate class, therefore in many applications this optimization has to be performed online during descriptor extraction. Given that a patch is a single instance from a class, additional examples have to be synthetically generated to estimate intra-class variance $\mathbb{E}[\Delta_i]$. This approach proved successful in many applications, in particular in the context of local image patches where affine projections are typically applied [2], [12].

Generating various geometric views of the same patch can be done easily (e.g. with affine matrices and bilinear interpolation), but in large datasets or real time applications the computational complexity would grow significantly. However, given the globally optimized set of binary tests, which is of a limited size, instead of bilinear patch warping we can apply the geometric transformations directly to the pixel locations $(\mathbf{t}_1, \mathbf{t}_2)_i$ of each test \mathbf{f}_i . For each test a new set of test can be created, which consist of its affine-transformed

versions. Furthermore, since the set of tests is fixed, the locations of tests under various affine transformations can be stored in a lookup table rather than calculated online. Thus, our set of tests is extended $f_{ia} = \{I(t_1) > I(t_2)\}_{ia}$ where a indicates an affine transformation of test f_i .

We examined a range of parameters of affine transformations to generate intra-class variances and to identify stable tests. We report and discuss the results in Figure 9 in terms of 95% error rate for 100k patches from the YOSEMITE dataset [18]. Parameters of affine transformations to generate positive examples were extensively studied in [2] with the conclusion that small random transformations lead to better results. We make similar observations and notice that small affine projections with a maximum rotations of 10° to 20° are the ones that give the best results. It is also worth noting that as few as 2 transformations are sufficient to identify tests that minimize the intra-class variance. This is an important observation as a small number of transformations leads to few affine lookup tables that need to be created. This then leads to more efficient online evaluation of binary tests which consist only of sampling and comparing pixel values.

Given the binary strings generated by tests f_{ia} represented in intra-class matrix \mathbf{F} , a subset of tests that minimizes the variance along dimension a is selected. In our implementation we select only the tests for which the variance is 0. However more sophisticated methods can be applied, such as variance sorting and thresholding.

Having identified the sets that are to be included in the per-patch adapted descriptor, each patch is represented by the results \mathbf{f}_n of the binary tests and a second binary string \mathbf{m}_n of length D where 1s indicate which tests are stable dimensions for patch n . Thus, the number of 1s (e.g. $\sum_{i=1}^D \mathbf{m}_{n,i}$) may differ for every patch.

5 ANALYSIS OF ONLINE LEARNT DESCRIPTOR

In this section we analyse the properties of the proposed descriptor and investigate various implementation options. We first discuss the relations to some hashing methods. We then investigate the parameters of transformations suitable for generating intraclass examples as well as alternative implementations of the descriptor. The experiments are done, unless stated otherwise, by using the 100k TREVI data for globally optimizing the tests and the 100k LIBERTY data for testing the descriptors.

5.1 Binary codes in related areas

Biometrics. We can relate our approach with previous works in the field of biometrics, particularly iris recognition. A similar binarisation technique is used to encode the image of iris as a string of bits. It has been known from the research in this area that *not all bits in the iris code are equally likely to flip* [23]. In the context of biometrics, several images from the same eye are used in order to identify the *fragile bits*. These bits are likely to change value across the training dataset and they may be different for every individual. The distance measure is therefore weighted in terms of how fragile a bit is for a particular individual from the training data. Note that as demonstrated in Figure 2, stable feature dimensions change per query, similarly to fragile bits in the iris codes. It

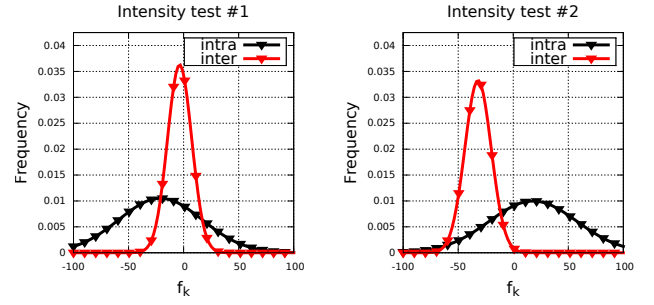
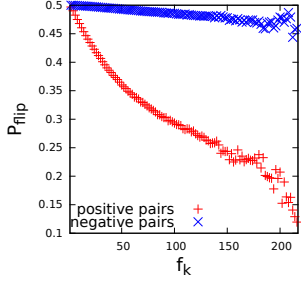


Fig. 5: Distribution of intensity differences f_k for two intensity tests for intra and inter class variations. Intra class is generated by affine transformation of patches and interclass is estimated on large number of different patches. Left distribution represents a less stable test with low f_k value. Interclass distribution is not centred on zero due to rotation alignment of the dominant gradient.

was demonstrated in [22] that the false negative recognition rate improves by splitting the iris code into two groups, one that includes the bits that flip with high probability and the other group with low probability flip. The bits from the latter are then used to model the iris. Our approach acts in a similar manner and by creating synthetic positive examples thus empirically identifying the bits that flip.

Binary hashing. The feature extraction in BRIEF and similar methods can be considered as closely related to binary hashing functions [28] where given \mathbf{x} as the input observation, and D intensity tests, each test can be considered a hash function $f_k(\mathbf{x}, \mathbf{i}_k, \mathbf{j}_k) = |\mathbf{x}(\mathbf{i}_k) - \mathbf{x}(\mathbf{j}_k)|$. The thresholded binarization of f_k is then performed according to $f_k < T_k$, which provides a cut-off threshold to make it more robust to noise. However, all BRIEF like methods use $T_k = 0$. Figure 5 shows distribution of f_k values for two different pairwise tests. Intra class is estimated on synthetically generated warped patches and inter class on various non-matching patches. One would expect inter class to be centred at zero, however, due to rotation alignment of patches in the dataset the pairwise intensity test may fall across the dominant gradient boundary thus the mode of their distribution can be biased. Ideally, the intra class distribution should be narrow and away from zero. Figure 5 (left) shows an example of less stable test and (right) shows a stable one.

Based on the above formulation, we can examine if the hashing method proposed in [28] that correlates the value of $|f_k - T_k|$ with the probability of hash bit flip holds in the binary feature descriptors. Note that the hash functions are typically projections from higher dimensional space in contrast to our simple intensity tests. The binary hash functions f_k map different real valued data points to different bits (0/1), which is their discriminating power or inversely, the the functions should map similar data points to the same bit with a high probability (> 0.5). The bits in Hamming distance are then weighted according to this probability. However, in the case of image intensity tests, and for typical deformations such as rotations and translations, their discriminating power is much lower and the probability of a flip for a specific patch and its bits is less reliable. Figure 6 shows the probability of a bit flip w.r.t.



method	FPR95
low- T	61.17
high- T	43.81
random	43.86
affine	28.87

Fig. 6: (left) Probability of bit flip $P_k(flip)$ w.r.t. intensity difference f_k . The larger the difference the less likely to flip. (right) Error rates for different methods of choosing subsets of binary intensity tests i.e. low and high intensity differences compared to random as well as our proposed online selection based on affine transformed patches.

intensity difference f_k between the the pair of pixels. As the intensity difference increases the probability of a flip decreases since the signal to noise ratio is much stronger. The probability estimate for large differences is less accurate as there are few tests with such difference.

To evaluate the impact of the probability on improving the stability of the tests, we perform the following experiment. Following the formulation of [28], we hypothesise that for a given patch, intensity tests with low values of f_k are less stable than tests with high values of f_k . To that end, we split the original D – *dimensional* binary descriptor to two groups, according to their f_k values. First we identify their median T_m , and then we create two masks termed low- T and high- T . A test f_k belongs to the low- T or high- T group if $f_k < T_m$ or $f_k > T_m$ respectively. We therefore expect that the high- T group performs better than the low- T group, since according to the results from Figure 6, the low difference intensity tests would be less stable. Table in Figure 6 reports the matching results in the 100K patches from the NOTREDAME dataset, in terms of FPR95 values, which is the false positive rate at 95% of true positive rate. Our first observation is that indeed using only the low- T tests for computing the descriptors results in a much less discriminative representation than using all the binary tests at random. However, we note unlike what is reported for the commonly used high-dimensional hashing methods in [28], there is little improvement by using only high- T tests. This is observed for real patch deformations such as the ones found in NOTREDAME dataset, since the descriptor computed with the high- T tests does not outperform the global parent descriptor. In contrast, when the masking is based on affine deformations, such as the ones commonly found in the NOTREDAME dataset, we see that the performance is significantly better, resulting in up to 33% improvement in terms of discriminative power.

The above experiment also indicates that the random noise is not the main issue in patch matching. Intuitively, patches are blurred before sampling therefore Gaussian noise is minimized and the probability of bit flip is less related to the magnitude of the intensity difference. The view-point change, rotation and other geometric transformations are the main factors affecting the matching performance.

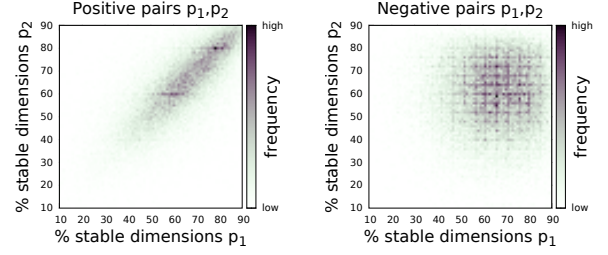


Fig. 7: Histogram of numbers of stable dimensions for positive (left) and negative (right) pairs of patches p_1 and p_2 . The distribution is more compact and the numbers of stable dimensions are similar for positive patch pairs in contrast to the negative ones.

5.2 Intra class adaptation

Modelling intra-class distribution is crucial for successful selection of stable dimensions. The intra-class patches are generated with common affine deformations such as scaling, translation & rotation. We therefore investigate the effect of different settings for intra-class optimization.

Number of stable dimensions. Our proposed online adaptation approach relies on the assumption that patches that should match have a very similar number of stable dimensions that are selected by the masking process. Furthermore, since the number of selected dimensions may vary we investigate the extent of this variation. Figure 7 shows histograms of numbers of stable dimensions for positive pairs (left) and negative pairs (right). Majority of the positive pairs have similar number of stable tests ranging from 60 to 80 while patches in negative pairs have a much broader distribution. i.e. the number of dimensions is significantly different. This further validates the stability of the proposed online selection process.

Robustness to affine transformations. Intuitively, since descriptors based on intensity tests such as BRIEF, BRISK, ORB are not scale, translation or rotation invariant, the percentage of binary tests that remain stable is inversely proportional to the extent of the transformations. We experimentally quantify this by creating transformed views of 10k patches from NOTREDAME and measuring the number of dimensions that change bits after patch deformation while increasing transformation parameters. The results are presented in Figure 8. We observe that 90% of tests are stable for very small transformations i.e. up to 5 degree rotation, 1.05 scaling or 2 pixel translation. These are minor deformations that are easily exceeded in real applications. Typical detectors introduce larger error in keypoint location and scale estimation e.g. the orientation estimation methods often use quantization bins of 10 degrees. The results show that nearly 50% of tests fail with translation by 5 pixels, scaling of 1.15 or rotation of 30 degree. Descriptors such as SIFT were engineered to be robust to such deformations but pairwise intensity tests are more sensitive.

Online learning with patch transformations. Figure 9 shows the FPR95 for descriptors with selected stable dimensions based on 2, 4, 8, or 16 synthetically generated patches with different rotation angles including the original patch. The smallest error is given by tests selected with

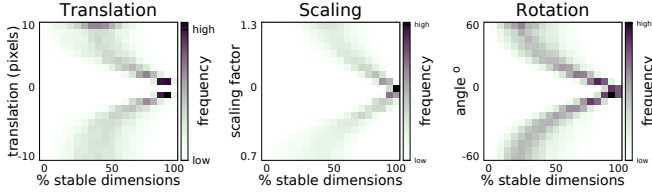


Fig. 8: Histogram of numbers of stable dimensions under various geometric transformations of patches including translations, scaling and rotations. We observe that 90% of tests are stable only for very small transformations i.e. up to 5 degree rotation, 1.05 scaling or 2 pixel translation, but nearly 50% of tests fail with translation by 5 pixels, scaling of 1.15 or rotation of 30 degree, which shows high sensitivity of binary intensity tests to geometric patch deformations.

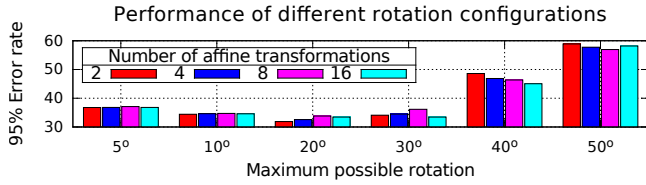


Fig. 9: Matching error FPR95 for YOSEMITE 100k with respect to various affine examples in intra-class optimization of binary tests. Small affine transformations and few examples are sufficient to achieve low error rate.

patches transformed with up to 20 degree rotation. This may be related to the error that is typically introduced with orientation estimation within patch rectification. Second observation is that the error is relatively independent of the number of patches used to model intra class variations. The results show that the identification of the stable dimensions can be done with as few as two examples. In that case, the mask is defined as

$$\mathbf{m} = \neg(\mathbf{f} \oplus \mathbf{f}') \quad (7)$$

with \mathbf{f}' being the transformed patch of query \mathbf{f} . The results show that the masks produced with this method lead to nearly the same performance as masks produced with more synthetic examples. This is an important observation since it shows that even a single perturbed version of the input patch can help in identifying the stable feature dimensions, thus significantly reduces the complexity of online extraction of the descriptor.

5.3 Descriptor variants

Our binary online learnt descriptor consist of a descriptor string \mathbf{f} and mask \mathbf{m} . This doubles the number of bits a patch is represented with. To demonstrate that the improvement results from suppressed dimensions and not from increased number of bits we perform additional experiments. We increase the size of the original BRIEF to 1024 bits and compare with our descriptor that consists of 512+512 bits including the mask. Figure 10 (left) shows the ROC curves for matching the 100k LIBERTY data. The descriptor combined

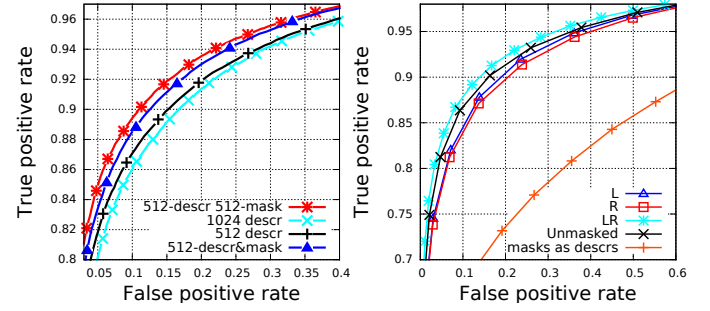


Fig. 10: (left) Matching performance for variants of descriptor and distance measures, (right) symmetric and asymmetric masking variants.

with the mask improves upon 1024 bit BRIEF by up to 5% in terms of FPR95. This shows that masking out unstable bits reduces the intraclass variations, and the extra gain in performance does not come from the extra information used in the hamming distance computation. It is also interesting to note that the 1024 bit BRIEF performs slightly worse than its 512 version, which indicates that increasing the number of the randomly sampled intensity tests does not always lead to improved results as the descriptor may become too discriminative and less robust to noise.

Another approach to suppress the noisy dimensions is to zero the unstable bits in the descriptor instead of using a mask. This results in the variant denoted with 512-descr&mask in Figure 10 (left). Note that this configuration also performs better than both the 512 and 1024 versions of BRIEF, although it does not need to store any extra information for the mask bits. Interestingly it is only slightly worse in terms of performance than the full mask equivalent, which needs to store double amount of information. This variant can be used when the memory saving is more important than the discriminative power.

When comparing two descriptors \mathbf{f}_L and \mathbf{f}_R with their respective masks \mathbf{m}_L and \mathbf{m}_R , there are three possible ways of generating a masked distance: $\{ \text{with } \mathbf{m}_L, \text{ with } \mathbf{m}_R, \text{ with both } \mathbf{m}_L \text{ and } \mathbf{m}_R \}$. Note that only the last option is using online adaptation for both patches, the two unilateral options are asymmetric. The comparison for these approaches including the unmasked descriptor from globally decorrelated dimensions, is presented in Figure 10 (right). The asymmetric distances with single mask give noticeably lower scores than the unmasked descriptor, with the one using both masks obtaining the top score.

Interestingly, since the mask is learnt for every patch it can be considered a characteristic of the patch and used as a descriptor on its own. Figure 10 (right) shows the performance for matching masks. Masks are scoring below the full descriptor as they only carry information about which tests is stable and not the actual value of the test but still significantly higher than a random classifier which would perform along the diagonal line.

6 EXPERIMENTAL EVALUATION

In this section we evaluate our descriptor and compare to other state-of-the-art methods. We first evaluate the

proposed descriptor in the patch matching task using the dataset from [20] and the evaluation protocol from [9], [15], based on ROC curves and error rates. This dataset consists of three subsets LIBERTY, YOSEMITE and NOTREDAME containing more than 500k patch pairs extracted from feature points detected by DoG [8]. We use sets of 100k patches for our experiments, which are resized to 32×32 . For all the experiments, the offline training that selects the global pool of intensity tests is done on the TREVI dataset from [18], which is not used for testing.

We also evaluate the descriptors using the image matching benchmark from [10] as well as the tracking dataset from [21]. We then show that the proposed feature adaptation method can be successfully applied to other types of descriptors. Finally, we report the speed and compare to state-of-the-art descriptors.

Unless stated otherwise, in all the experiments we use the descriptor that consists of computing the symmetric distance between a pair of patches using both masks, which is the best performing variant as seen in Figure 10. We name our descriptor BOLD (Binary Online Learnt Descriptor).

6.1 Patches

In Figure 11 (top) we plot the ROC curves for the full set of the globally optimized binary features of 512 bits compared to the per-patch optimized subsets of the proposed BOLD descriptor on YOSEMITE, NOTREDAME and LIBERTY data. Our method outperforms the globally optimized set of features across all false positive rates. This is significant, since it shows the clear advantage of per-patch optimizations compared to global per-dataset optimizations. It has to be noted, that although the final BOLD descriptor has significantly less dimensions involved in the computation of the distances and it is always a subset of the globally optimized tests, it outperforms this superset of tests.

In Figure 11 (bottom), we present the results of the comparison between our descriptor and other widely used methods such as BINBOOST, SIFT, SURF, ORB, DBRIEF, and BRIEF. It is important to note that out of the best performing descriptors i.e. BINBOOST, SIFT and BOLD, our descriptor is the only one to use simple binary intensity tests. Both SIFT and BINBOOST use quantized gradient responses which capture significantly more information about the patch statistics. Recently, in [15] it was shown that intensity binary tests are less discriminative as descriptor dimensions compared to features based on quantized gradients when optimized globally with the same theoretical framework. Our results show however that their performance can be greatly improved by simply using the proposed online per-patch adaptation framework.

We observe slightly lower performance on YOSEMITE data for all the descriptors compared to the other datasets. We estimated the average number of stable dimensions, similar to the one in Figure 7, however there is no significant difference between the three sets in (average, standard deviation)% of unstable dimensions across all positive pairs for YOSEMITE (23.5,10.62), NOTREDAME (23.99,10.46), and LIBERTY (24.32,10.98).

The results of the BOLD descriptor compared directly with other descriptors that are based on simple intensity

tests such as BRIEF and ORB, show a indicate the significant performance boost that can be achieved by the proposed method.

6.2 Matching

In this section, we evaluate the proposed descriptor in image matching, following the benchmark introduced in [10]. The data consists of six sequences, each with a reference image and five corresponding images. Using the Harris-Laplace detector [11], we extract a set of keypoints from each of the images and normalize them under a canonical representation. We compute a set of descriptors from all those patches and compute the matching scores following the original protocol from [10]. The results are reported in terms of recall vs. 1-precision, based on varying matching thresholds.

In Figure 12 (top) we plot the results for a pair of images from each sequence from [10] that represents a significant transformation. Results of other image pairs are consistent. Interestingly, SIFT gives the best results overall. However, BOLD outperforms SIFT for the high precision part of the curves in Boat, Bikes and Bark sequences. It is worth noting that although BINBOOST performs well in the patch dataset, it is ranked third in this matching experiment behind SIFT and BOLD. This may be due to a different training data used to optimize BINBOOST and using different interest point detector.

In Figure 12 (bottom) we can also observe the improvement introduced by online selection of binary tests in the intra-class optimization. This advantage of per-patch vs. global optimization is significant and consistently observed in all our experiments on different datasets.

6.3 Tracking

In this section, we demonstrate the application of our method to the tracking by detection problem. Several works [5], [6] follow the tracking by detection approach in which a model is initialized in the first frame, and updated online in order to account for appearance changes.

We use the tracking-by-detection mechanism from [6], where the online learnt detector is based on random ferns [12]. Our goal is to show the impact our optimization of the binary tests adapted to the object to be tracked. We build a detector that is adapted in the first frame but it is not updated online to avoid the influence of various training examples that can be collected and alleviate the problems of weak binary tests. We demonstrate that the performance of the fern detector depends on the choice of the tests f_i . Full randomization in all stages is proposed in [12], but based on our results from matching the descriptors, we investigate if the per-object adaptation of the binary features that are included in the ferns, can have an effect on the tracking result.

Similarly to [12] we create a classification system based on a set of N simple binarized intensity differences, similar to the ones in BRIEF and ORB. Following a sliding window approach, which is common among the state of the art detectors, we classify each window candidate as the object or background. Since each of the f_i features is a simple intensity test, a number of those is required to achieve

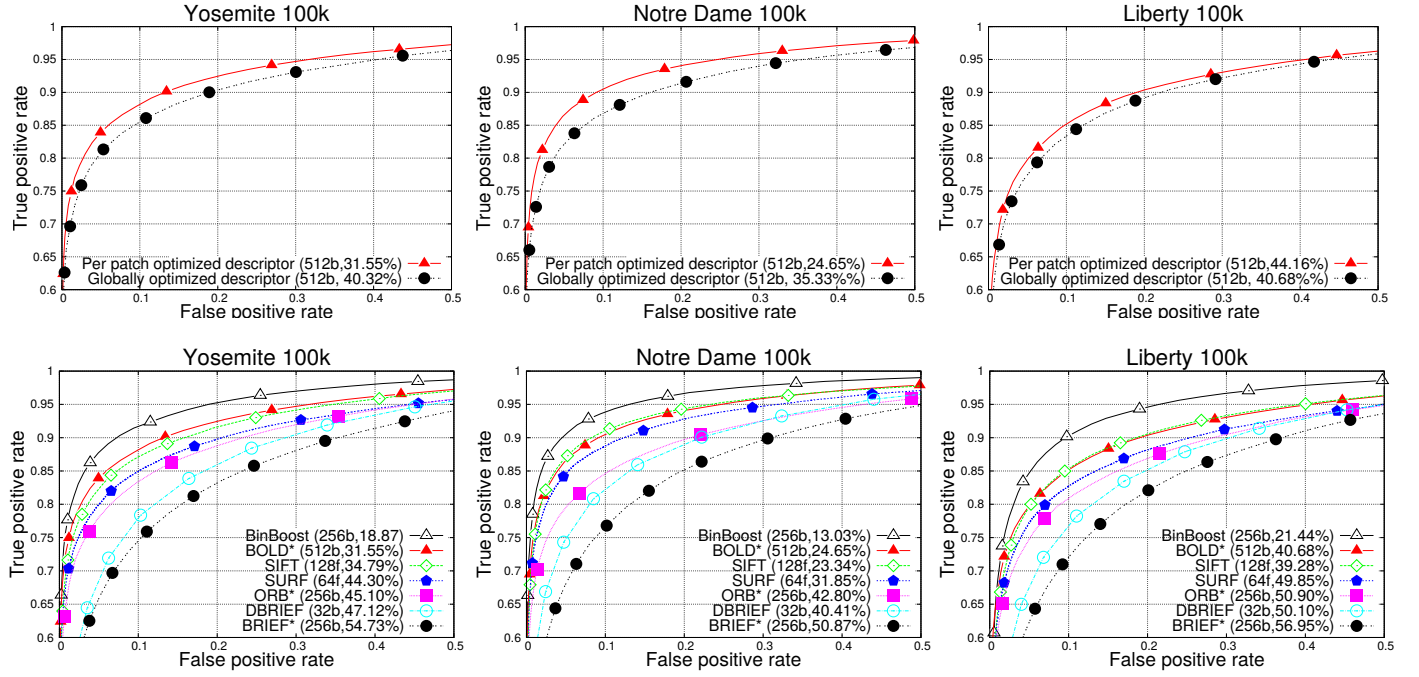


Fig. 11: Top: Globally vs. locally optimized features. Bottom: BOLD compared to several state of the art descriptors. Descriptors with * are based on simple intensity tests. Using our per-patch optimization framework, the performance of gradient based SIFT can be approached by descriptors based on simple intensity tests.

good detection performance. The authors of [12] use ≈ 300 while the fern classifier in [6] samples ≈ 130 . A complete representation of the posterior probabilities for each of the background and object classes is therefore impractical due to the large number of used binary tests. Thus in [12] N features are divided into M groups of size $\frac{N}{M}$. Each of those groups forms a fern. The conditional probability becomes $P(object|f_1, \dots, f_N) = \prod_{i=1}^M P(object|F_i)$. Following [6], we use a sum of the log likelihoods and a threshold. Thus, if $\sum_{i=1}^M \log P(F_i|object) \geq t_{object}$ we consider it a valid detection.

For the results in Table 2, we use the same detector configuration as in [6] with 10 ferns, each consisting of 13 binary intensity tests. The probability $P(F_i|object)$ for each fern is learned only from the first frame, using a set of 200 affine transformations of the original patch plus noise.

We generate a pool of 20 ferns, and compare two strategies for the selection of the final 10 that act as the classifier, one global and one adapted per object. In the first case, we follow the approach of [12] and [6] of randomly selecting a subset. For the second approach, we evaluate the posteriors of each fern in our set of 200 positive examples generated from the object, and we choose the 10 ferns that minimize the intra-class Hamming binary distance across the synthesized 200 positive examples.

We test this method in 10 sequences from the recently published tracking benchmark [21]. We report the recall, which is $\frac{\# \text{ of correct detections}}{\# \text{ frames}}$. We do not report the precision, since this simple detector/tracker does not update its

Sequence	Fixed Ferns	Ferns adapted per object
Subway	0.19	0.28
Jumping	0.26	0.46
Girl	0.44	0.58
Suv	0.25	0.42
Woman	0	0.1
Freeman1	0.07	0.13
Freeman4	0.09	0.16
Deer	0.04	0.18
Crossing	0.3	0.45
Couple	0.03	0.1
Average	0.17	0.29

TABLE 2: Recall results for 10 sequences of the recently published tracking evaluation benchmark [21]. We observe that selecting a subset of ferns per object outperforms a global set of ferns fixed for all objects.

model online, its precision is therefore 1 or very close to 1 in most cases.

The results reported in Table 2 compare the randomly generated tests to object-adapted ferns based on our approach. The per-object optimized ferns perform significantly better than the random tests. Similarly to per-patch online adaptation of descriptors, per-object adaptation of ferns improves the recall of the detectors. Object tracking is an excellent application for the proposed method, since due to the efficiency requirements the learning has to be done online therefore powerful machine learning methods that require large set of training examples are of limited use.

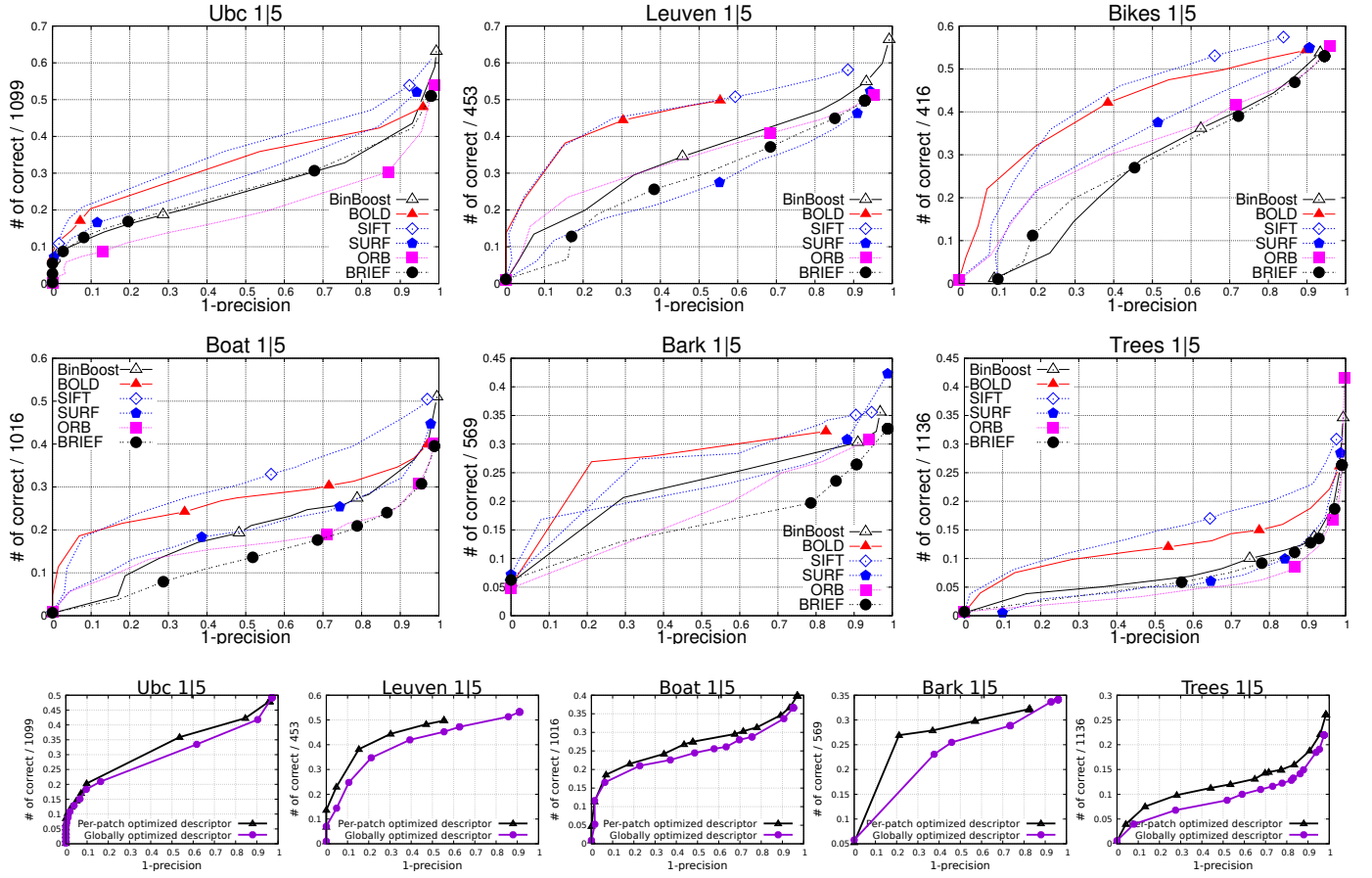


Fig. 12: Image matching experiment of planar homographies benchmark from [10]. Top six figures compare state of the art descriptors on various sequences that correspond to different image transformations. Bottom five figures compare our proposed local adaptation to globally optimized tests.

6.4 Low bit-rate locally adapted descriptors

We experiment with low dimensional locally adapted descriptors to examine how their performance may vary for memory constrained applications. Such descriptors are useful in various memory and computationally intensive environments such as embedded systems and tracking scenarios. In Figure 13, we plot several versions of our locally adapted descriptors, with dimensionality as low as 32 bits. Local adaptation shows much better performance than BRIEF across the wide range of dimensionality. Surprisingly, 32-bit dimensional descriptor performs better than the 128 dimensional BRIEF. Furthermore, we observed that the 32-dimensional BOLD is on par with a 300-dimensional BRIEF. We attribute this performance difference to the discriminative power of the tests selected with our approach in contrast to the random ones in BRIEF.

6.5 Extension to other binary descriptors

In this section we apply the proposed approach for selecting stable dimensions to other binary descriptors such as BRIEF and BINBOOST. BRIEF and BINBOOST are first extracted with the original methods from every patch with its geometric views. Following the method proposed in Section 4.4.2, we generate a single synthetic view of a patch

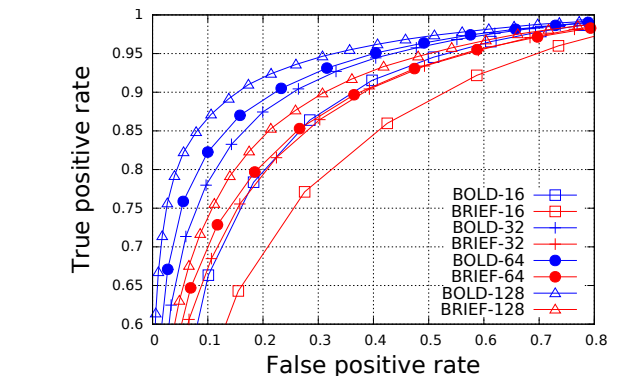


Fig. 13: Low bit-rate versions of our locally adapted descriptor, compared with low bit-rate BRIEF descriptors, on the NOTREDAME dataset. Note that the 32-bit version of BOLD, outperforms the 128-bit version of BRIEF.

by randomly rotating it between -20 and 20 degrees. We then use Equation 7 to generate the mask. We finally use this mask to compute a patch-adapted distance, that includes only the stable dimensions for each patch.

The results in Figure 14 show that the local adaptation generalizes to other types of binary descriptors. We refer to

Distance (512 dimensions)	μS
$\mathbf{f}_L \oplus \mathbf{f}_R$	220
$(\mathbf{f}_L \oplus \mathbf{f}_R \wedge \mathbf{m}_L) + (\mathbf{f}_L \oplus \mathbf{f}_R \wedge \mathbf{m}_R)$	340

TABLE 3: Performance of the masked Hamming distance, for 1000 pairs of patches. Our proposed masked Hamming distance presents similar efficiency to the original Hamming distance.

Descriptor	extraction	matching	total
BINBOOST	713	0.11	713.11
SIFT	417	10	427
SURF	48.2	5	53.2
BOLD	10.5	0.34	10.84
DBRIEF	6.8	0.02	6.82
ORB	2.7	0.11	2.88
BRIEF	2.7	0.11	2.88

TABLE 4: Comparison of efficiency per operation for various feature descriptors. Time is reported in μS per descriptor.

the original descriptor as X and $X-o$ as its online adapted version (e.g. ORB and ORB-o). The adaptation leads to significant improvements in all the descriptors that are based on binary features. In contrast, it provides a modest improvement to BINBOOST. This can be explained by the fact that BINBOOST is computed from averaged gradient maps, which are more discriminative than simple intensity tests, and the discriminative power varies less across the dimensions. Local adaptation has therefore less effect and masking of *unstable dimensions* provides limited improvement. On the contrary, due to the extremely fragile nature of the simple intensity tests, the local adaptation helps to significantly improve the results in BRIEF and ORB. BOLD outperforms both descriptors in various image sequences in Figure 14.

6.6 Speed

One of the main advantages of BOLD descriptor is its extraction and matching speed. We therefore discuss the computational efficiency of the proposed masked Hamming distance (cf. Section 4.1). The results are averaged on a set of 100k patches from the LIBERTY dataset. All the experiments were done on an Intel i7-Haswell processor with the avx-2 instruction set enabled, and all the possible SIMD optimizations were used (i.e. `popcount`).

In Table 3, we compare the calculation time of our masked distance to the regular Hamming distance when matching two binary descriptors. Despite the introduction of the symmetric masked Hamming distance thus longer binary strings, the computational efficiency remains high i.e. only 340 μs , and comparable to the regular Hamming distance of 220 μs . The only additional operation is the logical AND with the masks otherwise the optimized instructions compensate for longer strings.

In Table 4, we report the running times for extraction and matching for several of the descriptors reported in the results. We show that BOLD remains competitive with BRIEF in terms of both extraction and matching speed yet presents much better results in terms of 95% error rate. Real

valued descriptors such as SIFT and SURF have a long extraction and matching time i.e. 5-40 times slower than BOLD, with BINBOOST being the slowest in this set. ORB and BRIEF are still three times faster as no optimization is applied during extraction.

Furthermore, Figure 15 presents the performance of each descriptor w.r.t. its computational requirements. With the proposed framework, we achieve error rates similar to the SIFT descriptor, with extraction times on the level of BRIEF descriptor. The top performance is with BINBOOST, however it is 70 times slower than BOLD.

7 STATE-OF-THE ART DESCRIPTORS

Recent advances in local descriptor matching involve deep neural network architectures. These approaches significantly differ from the ones considered in this paper, however for completeness, we report performance of two recent methods that are based on deep learning, DEEPBIT [24] and DEEPCOMPARE [33].

Table 5 compares FPR95 (error rate for 95% correct matches on the ROC curve) of several so called hand crafted or engineered descriptors to end-to-end deep learning based descriptors. We observe that DEEPCOMPARE descriptors lead to error rates that are significantly lower than the other methods. DEEPCOMPARE includes a complex convolutional neural network with two stream and two branches each, as well as fully connected distance metric layers. These are trained on GPUs from large number of patches using data augmentation. The execution time is then at least 2 times slower than SIFT. In addition, DEEPCOMPARE descriptor is represented with a 512 dimensional floating point vector, which leads to significantly higher storage requirements, especially when compared to the memory-efficient binary descriptors.

DEEPBIT [24] is similarly based on deep convolutional neural networks but the final layer converts real value features into binary strings. Binary strings allow for fast matching however the performance is lower than for some hand crafted descriptors. Our results show that Yosemite is more challenging for any descriptor although DeepBit seems to be more affected. It is observed in [24] that higher similarity between negative patch pairs in YOSEMITE is due to smooth texture of nature scenes, in contrast to structure content of man made buildings in NOTREDAME and LIBERTY. In addition, DeepBit is trained on synthetically rotated versions of patches which is not the main challenge in these datasets.

It is important to note that deep convolutional neural network descriptors are extremely slow to compute and involve a lengthy training process. This is likely to be improved with further progress in the field however, for many practical applications the engineered binary descriptors will still be the preferred choice.

8 CONCLUSION

We have proposed a novel approach for generating descriptors that are adapted independently per-patch. Our method relies on binary tests that can be efficiently extracted, evaluated and selected. We presented a full inter- and intra-class

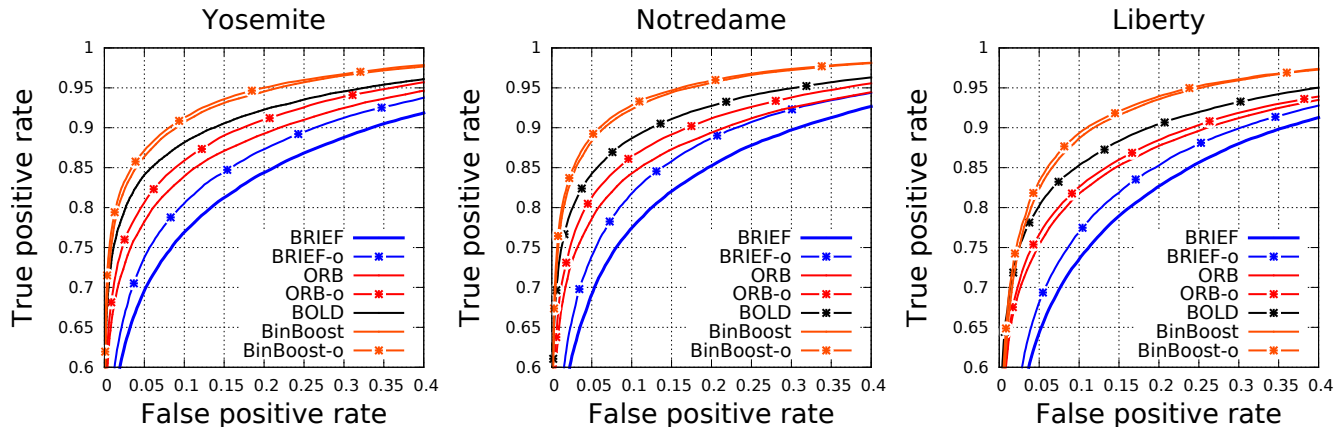


Fig. 14: Performance of other binary descriptors extended with the local adaptation of binary features. That there is a consistent improvement for all the descriptors, with significant performance boost for descriptors based on intensity tests such as BRIEF and ORB.

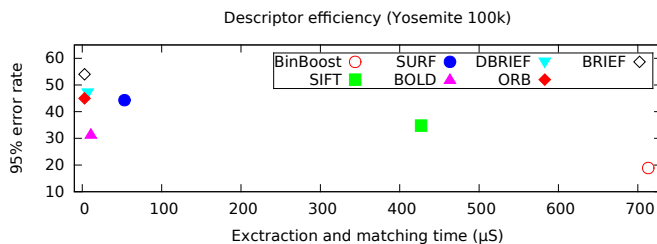


Fig. 15: BOLD descriptor has low error rates and high computational efficiency. In particular, it exhibits speed that is similar to the fastest available binary descriptors that are based on intensity tests, while at the same time reaches error rates comparable with the much less efficient descriptors that are based on pooling local gradients.

Descriptor	YOSEMITE	NOTREDAME	LIBERTY
BRIEF [3]	54.73	50.87	56.95
DBRIEF [16]	47.12	40.41	50.10
ORB [13]	45.10	42.80	50.90
SURF [1]	44.30	31.85	49.85
SIFT [8]	34.79	23.34	39.28
BOLD	31.55	24.65	40.68
BINBOOST [15]	18.87	13.03	21.44
DEEPLIT [24]	57.61	26.61	32.06
DEEPCOMPARE [26]	5.00	2.76	4.85

TABLE 5: FPR95 results for several state of the art binary descriptors and deep learning CNN based descriptors i.e. DEEPLIT and DEEPCOMPARE. For comparison we also report the results for two commonly used real-valued descriptors, SIFT and SURF.

optimization of binary descriptors that is performed online for each image patch.

The results from several experiments on different datasets and different tasks show that using a local optimization leads to significant improvements over a global one. Furthermore, the efficiency of the proposed implementation is comparable to other binary descriptors and significantly better than real-valued descriptors. Our approach is the first attempt to use per-patch descriptor with successful results in terms of matching performance and speed in

typical computer vision applications.

The proposed method can be applied to other techniques such as decision trees or ferns. An interesting extension would be to apply the proposed selection approach to other quantized gradient based features such as SIFT or deep CNN descriptors.

ACKNOWLEDGMENTS

This work was supported by EPSRC EP/K01904X/2 Visen and EP/N007743/1 FACER2VM projects.

REFERENCES

- [1] H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. In *ECCV*, 2006.
- [2] H. Cai, K. Mikolajczyk, and J. Matas. Learning linear discriminant projections for dimensionality reduction of image descriptors. *IEEE TPAMI*, 33(2):338–352, 2010.
- [3] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: binary robust independent elementary features. In *ECCV*, 2010.
- [4] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, R. Grzeszczuk, B. Girod. CHoG: Compressed histogram of gradients a low bit-rate feature descriptor. In *CVPR* 2009.
- [5] S. Hare, A. Saffari, and P. Torr. Struck: Structured output tracking with kernels. In *ICCV*, 2011.
- [6] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE TPAMI*, 34(7):1409–1422, 2012.
- [7] S. Leutenegger, M. Chli, and R. Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *ICCV*, 2011.
- [8] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004.
- [9] M. Brown, G. Hua and S. Winder. Discriminative learning of local image descriptors. *IEEE TPAMI*, 33(1):43–57, 2010.
- [10] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE TPAMI*, 27(10):1615–1630, 2005.
- [11] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *IJCV*, 60(1): 63–86, 2004.
- [12] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua. Fast keypoint recognition using random ferns. *IEEE TPAMI*, 32(3):448–461, March 2010.
- [13] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, 2011.
- [14] K. Simonyan, A. Vedaldi, and A. Zisserman. Learning local feature descriptors using convex optimisation. *IEEE TPAMI*, 36(8):1573–1585, 2014.
- [15] T. Trzcinski, M. Christoudias, P. Fua and V. Lepetit. Boosting Binary Keypoint Descriptors. In *CVPR*, 2013.
- [16] T. Trzcinski and V. Lepetit. Efficient Discriminative Projections for Compact Binary Descriptors. In *ECCV*, 2012.

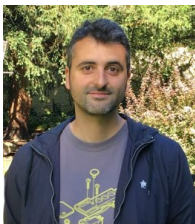
- [17] T. Tuytelaars and C. Schmid. Vector quantizing feature space with a regular lattice. In *ICCV*, 2007.
- [18] S. Winder and M. Brown. Learning local image descriptors. In *CVPR*, 2007.
- [19] C. Strecha, A. Bronstein, M. Bronstein, and P. Fua. LDAHash: Improved matching with smaller descriptors. *IEEE TPAMI*, 34(1):964–973, 2012.
- [20] S. Winder, G. Hua, and M. Brown. Picking the best daisy. In *CVPR*, 2009.
- [21] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *CVPR*, 2013.
- [22] K. P. Hollingsworth, K. W. Bowyer and P. J. Flynn. The Best Bits in an Iris Code. *IEEE TPAMI*, 31(6):964–973, 2009.
- [23] R. M. Bolle, S. Pankanti, J. H. Connell and N. K. Ratha. Iris individuality: a partial iris model. In *ICPR*, 2004.
- [24] K. Lin, J. Lu, C.S. Chen and J. Zhou. Learning Compact Binary Descriptors with Unsupervised Deep Neural Networks. In *CVPR*, 2016.
- [25] G. Zhang, M.J. Lilly, P.A. Vela. Learning Binary Features Online from Motion Dynamics for Incremental Loop-Closure Detection and Place Recognition. In *ICRA*, 2016.
- [26] S. Zagoruyko and N. Komodakis. Learning to Compare Image Patches via Convolutional Neural Networks. In *CVPR*, 2015.
- [27] Y. Tsun-Yi and L. Yen-Yu and C. Yung-Yu. Accumulated Stability Voting: A Robust Descriptor From Descriptors of Multiple Scales. In *CVPR*, 2016.
- [28] L. Zhang, Y. Zhang, J. Tang, K. Lu, Q. Tian. Binary Code Ranking with Weighted Hamming Distance. In *CVPR*, 2013.
- [29] P. Fischer, A. Dosovitskiy, T. Brox. Descriptor Matching with Convolutional Neural Networks: a Comparison to SIFT. In *arXiv*, 2014.
- [30] J. Bromley, J.W. Bentz, L. Bottou, , I. Guyon , Y. LeCun, R. Shah. Signature verification using a Siamese time delay neural network. *IJPRAI*, 7(04), 669-688, 1993.
- [31] S. Chopra, R. Hadsell and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005.
- [32] R. Hadsell and S. Chopra and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.
- [33] S. Zagoruyko and N. Komodakis. Learning to Compare Image Patches via Convolutional Neural Networks. In *CVPR*, 2015.
- [34] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua and F. Moreno-Noguer. Discriminative Learning of Deep Convolutional Feature Point Descriptors. In *ICCV*, 2015.
- [35] X. Han T. Leung, Y. Jia , R. Sukthankar and A. Berg. MatchNet: Unifying Feature and Metric Learning for Patch-Based Matching. In *CVPR*, 2015.
- [36] V. Balntas, E. Riba, D. Ponsa and K. Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks In *BMVC*, 2016.
- [37] S. Urban and S. Hinz. A Fast Online Adaptable, Distorted Binary Descriptor for Real-Time Applications Using Calibrated Wide-Angle Or Fisheye Cameras. In *arXiv*, 2016.



Lilian Tang Lilian Tang received her PhD degree in Medical Informatics in 2001 from the University of Cambridge, United Kingdom. She then joined the Department of Computer Science at the University of Surrey where she is now a Senior Lecturer. Her research interests are medical image analysis, machine learning, and object recognition.



Krystian Mikolajczyk Krystian Mikolajczyk is an Associate Professor at Imperial College London. He completed his PhD degree at the Institute National Polytechnique de Grenoble and held a number of research positions at INRIA, University of Oxford and Technical University of Darmstadt, as well as faculty positions at the University of Surrey, and Imperial College London. His main area of expertise is in image and video recognition, in particular methods for image representation and learning. He has served in various roles at major international conferences co-chairing British Machine Vision Conference 2012, 2017 and IEEE International Conference on Advanced Video and Signal-Based Surveillance 2013. In 2014 he received Longuet-Higgins Prize awarded by the Technical Committee on Pattern Analysis and Machine Intelligence of the IEEE Computer Society.



Vassileios Balntas Vassileios Balntas is a research associate at the Imperial Computer Vision and Learning Lab, Imperial College London, UK. Previously he was a PhD student at the University of Surrey, UK and a research assistant at National Technical University of Athens, Greece. He holds a MSc in Computer Science from University of Surrey, UK, and a MEng in Electrical and Computer Engineering from Democritus University of Thrace, Greece.