# Rethinking the sGLOH Descriptor

Fabio Bellavia and Carlo Colombo

**Abstract**—sGLOH (shifting GLOH) is a histogram-based keypoint descriptor that can be associated to multiple quantized rotations of the keypoint patch without any recomputation. This property can be exploited to define the best distance between two descriptor vectors, thus avoiding computing the dominant orientation. In addition, sGLOH can reject incongruous correspondences by adding a global constraint on the rotations either as an a priori knowledge or based on the data. This paper thoroughly reconsiders sGLOH and improves it in terms of robustness, speed and descriptor dimension. The revised sGLOH embeds more quantized rotations, thus yielding more correct matches. A novel fast matching scheme is also designed, which significantly reduces both computation time and memory usage. In addition, a new binarization technique based on comparisons inside each descriptor histogram is defined, yielding a more compact, faster, yet robust alternative. Results on an exhaustive comparative experimental evaluation show that the revised sGLOH descriptor incorporating the above ideas and combining them according to task requirements, improves in most cases the state of the art in both image matching and object recognition.

**Index Terms**—Keypoint matching, SIFT, sGLOH, RFDs, LIOP, MIOP, MROGH, CNN descriptors, rotation invariant descriptors, histogram binarization, cascade matching.

## 1 Introduction

Finding correspondences between image keypoints is a crucial step in many 3D computer vision applications such as Structure from Motion (SfM) [1] and Simultaneous Localization And Mapping (SLAM) [2], as well as in object detection [3], recognition [4], tracking [5] and classification [6].

Keypoint descriptors are numerical vectors that encode the local properties of the keypoint image neighborhood. Good keypoints and descriptors [7] must be robust against several image transformations, such as geometric affine warping, blur and luminosity changes, while keeping high discriminative power. Furthermore, they must be fast, efficient to compute and relatively compact, especially in the case of real-time applications and devices with limited hardware capabilities.

Before computing the descriptor, the image patch representing the keypoint is usually normalized [8]. For instance, in the case of modern affine keypoint detectors [9], the elliptic region representing the keypoint is transformed into a unit circle and rotated according to the dominant orientation of the patch in order to achieve geometric affine invariance. Finally, the pixel intensities are normalized according to their mean and variance to make the patch invariant to affine luminosity changes.

The estimation of the dominant orientation can be often unreliable [10]–[12]. In this respect, recent rotation invariant descriptors such as LIOP (Local Intensity Order Pattern) [13], MIOP (Mixed Intensity Order Pattern) [14] and MROGH (Multi-Support Region Order-Based Gradient Histogram) [10] are more robust than the popular SIFT (Scale Invariant Feature Transform) descriptor [3].

The sGLOH (shifting GLOH) descriptor [11] is a histogram-based keypoint descriptor that can be associated to multiple quantized rotations of the keypoint patch by a cyclic shift of the descriptor vector without any recomputation. This property can be exploited to define the best distance between two descriptor vectors among all those induced by the quantized rotations, thus avoiding altogether computing the dominant orientation. Furthermore, using sGLOH, incongruous correspondences can be rejected by adding a global constraint on the rotations either as an a priori knowledge or according to the data. This process is similar to applying the generalized Hough transform [15] for image rotations in the context of keypoint matching.

This paper thoroughly reconsiders the basic sGLOH matching strategies and improves them in terms of correct matches, speed and descriptor dimension. In particular, as first anticipated in [16], the improved sGLOH is able to embed more quantized rotations, while avoiding decreasing the area used for each local gradient histogram, which would lead to a loss of its discriminative power. Since such refined rotation quantization increases the number of matched correspondences but also the computing time, a faster and more efficient approximated matching scheme is presented in this paper. The proposed method uses statistical clues accumulated at run-time in order to drop matching pairs which are unlikely to be correct. Without computing the full descriptor distance, this yields a very close yet faster approximation of the original matching strategy.

Additionally, a novel binary version of sGLOH, named BisGLOH, is defined based on comparisons inside the descriptor histograms, which still incorporates several patch rotations into shifts of the descriptor vector. Although its discriminative power is somewhat reduced, the BisGLOH interestingly gives a compact and fast, yet valid, descriptor based on the Hamming distance.

Results on an extensive evaluation on both image matching and object recognition show the validity of the proposed approaches as compared to the state-of-the-art descriptors. These include the popular and well investigated SIFT descriptor, the rotational invariant LIOP, MIOP and MROGH descriptors, the recent learning-based binary RFDs (Recep-

*F. Bellavia and C. Colombo are with the Department of Information Engineering, Università degli Studi di Firenze, Via di S. Marta, 3, 50139 Firenze, Italy, e-mail: name.surname@unifi.it .*

tive Fields Descriptors) [17] and two emerging approaches based on convolutional neural networks (CNNs) [18], [19]. Beside a standard evaluation in the case of planar scenes on the Oxford dataset [8] and the recent Viewpoint dataset [19], descriptors behavior is analyzed on four non-planar scene datasets using respectively the approximated overlap error [11], [20], structured light 3D data [21], epipolar constraints between triplets of calibrated views [22] and patches extracted using SfM [23]. Furthemore, descriptor properties in the case of object retrieval tests [10], [17] are also investigated through the ZuBuD [24] and Kentucky [25] datasets. Running time and implementation issues are also discussed in detail.

The rest of the paper is organized as follows: Related work is presented in Sec. 2, while the original sGLOH descriptor is introduced in Sec. 3. The proposed extensions are discussed in Sec. 4 and evaluation results are presented in Sec. 5. Finally, conclusions and future work are outlined in Sec. 6.

## 2 Related Work

Research on keypoint descriptors has experienced a strong and constant interest due to the ever increasing proliferation of computer vision applications, continuously demanding better and more efficient solutions. Keypoint descriptors are related to keypoint detectors, evolving concurrently upon the concepts of corners, blobs, saliency, scale-space and affine covariant transformations. The reader may refer to [7] for a general overview.

Most of today's descriptors are distribution-based [8], i.e. they compute a statistic for given regions of the keypoint patch, such as the gradient histogram or binary comparisons between pixel intensities. The rank and census transforms [26] can be considered the precursors of these descriptors.

Recently, accordingly to some authors [17], descriptors can be further divided into handcrafted and data driven. Data driven descriptors use machine learning techniques on training sets to extract the descriptor configuration or its structural design. Reducing the descriptor vector length by PCA (Principal Component Analysis) [27] may be considered as an early example of this kind of descriptors.

One of the most popular and yet still valid histogram-based descriptors is SIFT [3]. This descriptor considers the concatenation of the Gaussian-weighted gradient histograms associated to square regions into which the keypoint patch is divided. Rotation invariance is usually obtained by preprocessing the patch by rotating it towards the dominant gradient orientation, even though other methods exist [12], [28]–[31].

Several descriptors have been built upon SIFT. PCA-SIFT [27] applies PCA to the descriptor vector in order to reduce the dimension and increase its robustness. RIFT (Rotation Invariant Feature Transform) [32] uses rings instead of square grid regions in order to achieve rotational invariance. A log-polar grid and PCA are employed with GLOH (Gradient Local Orientation Histogram) [8] and overlapping regions in [9]. The Manhattan norm instead of the Euclidean norm in conjunction with the Bhattacharyya distance is reported to improve RootSIFT [33]. Multiple support regions [34] are successfully employed by MROGH [10] with intensity order pooling to achieve rotational invariance. Recently, DSP-SIFT (Domain Size Pooling SIFT) [35] reports better results compared to SIFT by pooling gradient orientations across different domain sizes, i.e. properly weighted SIFT histograms for distinct scales are merged together. Furthermore, ASIFT [36] compensates high perspective image distortions by using SIFT on multiple virtually generated views. This last idea is further developed in [37], with the descriptor subspace representation of multiple SIFT vectors computed at different scales.

Other histogram-based descriptors worth mentioning are the rotational invariant LIOP [13] with intensity order pooling and histograms computed on the relative order of neighbor pixels, and the CS-LBP (Center Symmetric Local Binary Pattern) [38], where histograms arise from the distribution of the intensity comparisons among center symmetric pixels. More recently, promising results have been reported with MIOP (Mixed Intensity Order Pattern) [14], obtained by applying PCA to the concatenation of the LIOP descriptor with the recent OIOP (Overall Intensity Order Pattern) [14], whose histograms encode the distribution of the intensity values of the ordered neighborhood pixels for each pixel of the patch. The fast SURF (Speeding-Up Robust Features) [28] and DAISY [39] descriptors are based respectively on Haar wavelets and Gaussian convolution.

Binary descriptors represent the state of the art of the current research towards efficient, fast, compact and yet sufficiently robust descriptors, demanded by the diffusion of real-time computer vision applications and devices with limited hardware capabilities. The robustness of binary descriptors is still noticeably inferior to that of histogram-based descriptors, although this gap is being filled nowadays [40]. Nevertheless, binary descriptors are faster and computationally more efficient [41].

BRIEF (Binary Robust Independent Elementary Features) [42], is based on binary comparisons between the intensities of random pixel pairs. With respect to BRIEF, ORB (Oriented-FAST and Rotated BRIEF) [30] compensates for patch rotations and chooses pixel comparison pairs by minimizing their correlation on training data. BRISK (Binary Robust Invariant Scalable Keypoints) [29] uses a handcrafted polar sampling pattern where short distance pixel pairs are used for the comparisons, and long pairs to determine patch orientation. FREAK (Fast Retina Keypoint) [31] further adds a matching pair selection like ORB and a cascade fast comparison to accelerate the matching process. Binary comparisons can also be built upon existent descriptors. For instance, BIG-OH (BInarization of Gradient Orientation Histograms) [43] gets a binary descriptor from the comparison of successive SIFT gradient orientation histogram bins. In [44], concatenations of successive thresholding results of the sampling pattern are used to mimic the quantization mechanism.

Data driven descriptors have also been investigated for designing efficient and compact descriptors. In the case of non-binary descriptors, beside the PCA-SIFT, linear discriminant embedding has been applied to reduce the descriptor dimension [45], while the recent ASR (Affine Subspace Representation) [46] has been used without affine normalization of the keypoint patch. Binary data driven descriptors exist as

well. LDAHash (Linear Discriminant Analysis Hashing) [47] defines thresholds on SIFT linear projections, while BGM (Boosted Gradient Map) [48] and RDFs [17] threshold on the patch gradient map, parameters are learned from training data. Recently, LATCH (Learned Arrangements of Three Patch Codes) [49] compares learned sub-patch triplets, while in [18], [19] CNNs are trained respectively to assign the reference orientation and to define a full descriptor. BOLD (Binary Online Learned Descriptor) [40] defines a binary mask so that only the descriptor vector elements minimizing the intra-class variance on affine warps of the original patch are used in the matching. ASV (Accumulated Stability Voting) [50] is obtained by thresholding the differences between descriptor vectors (e.g. SIFTs) for the same patch at different scales and summing up the results.

Using of a good distance metric is also crucial for matching descriptors, as can be noted by the strong relation between subspace reprojection of data-driven approaches [27], [45] and cross-bin histogram dissimilarity measure [51]. Euclidean and Manhattan distances are the most common choices for non-binary descriptors, while the fast Hamming distance is frequently used instead for binary vectors, although other choices are possible [51], [52].

The huge growth of keypoint descriptor database demanded by current applications has required the design of fast and efficient matching strategies. Beside the kd-tree search [53], cascade matching filtering [31], [54] rejects a putative match by a partial, incremental, fast analysis of the descriptor vector pairs, under the observation that some vector elements are more informative than others. A similar approach is explored in SIFT-HHM (SIFT Handed Hierarchical Matching) [55], where the most informative SIFT vector elements and the corresponding distance thresholds are learned off-line. Furthermore, MRES (Multi-Resolution Exhaustive Search) [56] employs a hierarchical matching on increasing resolution levels, with a single threshold estimated from a run-time sample of matching pairs.

## 3 The sGLOH Descriptor

The sGLOH descriptor grid is made up of $n \times m$ regions $\mathcal{R}_{r,d}$ with $r = \{0, 1, \ldots, n-1\}$ and $d = \{0, 1, \ldots, m-1\}$, defined by $n$ rings centred on the keypoint, each containing $m$ sectors, equally distributed along $m$ directions (see Fig. 1). For each region $\mathcal{R}_{r,d}$, the histogram of $m$ quantized orientations weighted by the gradient magnitude is computed, where the bin value $h_i$, $i = 0, 1, \ldots, m-1$, is obtained by the Gaussian kernel density estimation for that region

$$h^i_{r,d} = \frac{1}{\sqrt{2\pi}\sigma} \sum_{\mathbf{x} \in \mathcal{R}_{r,d}} \parallel \nabla I(\mathbf{x}) \parallel \; e^{-\frac{\left(M_{2\pi}\left(\theta_{\nabla I(\mathbf{x})} - m_i\right)\right)^2}{2\sigma^2}} \tag{1}$$

where $\parallel \nabla I(\mathbf{x}) \parallel$ and $\theta_{\nabla I(\mathbf{x})}$ are respectively the image gradient magnitude and orientation at pixel $\mathbf{x} \in \mathcal{R}_{r,d}$; $m_i = \frac{2\pi}{m} i$ is the $i$-th orientation bin center and $\sigma = \frac{2\pi}{m} c$, with $c \in \mathbb{R}^+$ the standard deviation in quantized orientation bin units. The function $M_{2\pi}(x)$ is used to take into account a periodicity of length $2\pi$

$$M_{2\pi}(x) = \begin{cases} x & \text{if} \quad x < \pi \\ 2\pi - x & \text{otherwise} \end{cases}$$

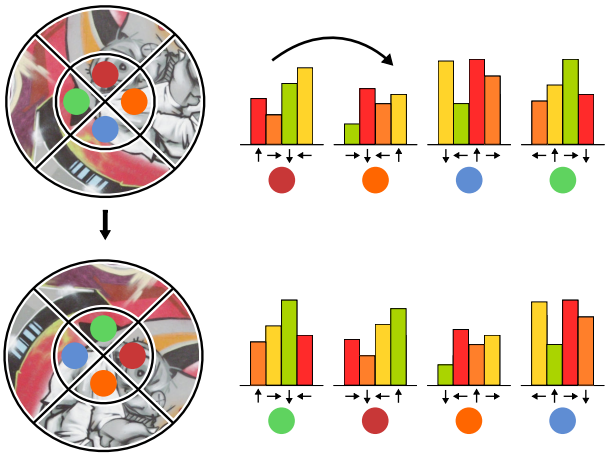


Fig. 1. Rotation of an image patch by a factor $\frac{2\pi}{m}$ with the superimposed sGLOH grid (left), which corresponds to a cyclic shift of the block histogram of each ring (right). In the example $n = 2$ and $m = 4$, color labels on the patch grid identify the corresponding block histograms on the descriptor (best viewed in color).

In modular arithmetic, the relation $a \equiv b \pmod m$ defines the congruence class $[a]_m$, so that $[d + i]_m$ shifts cyclically by $d$ positions the $i$-th element of a $m$ dimensional vector. We define a block histogram

$$H^t_{r,d} = \bigoplus_{i=0}^{m-1} h^{[t+i]_m}_{r,d} \tag{2}$$

where $\bigoplus$ is the concatenation operator, so that the first bin of each block has direction $\frac{2\pi}{m} t$. By concatenating the block histograms $H^t_{r,d}$ of each region $\mathcal{R}_{r,d}$ so that $t = d$, the vector $\dot{H}$ is obtained

$$\dot{H} = \bigoplus_{i=0}^{n-1} \bigoplus_{j=0}^{m-1} H^j_{i,j} \tag{3}$$

which becomes in a more simple notation $\dot{H} = [\dot{h}_1, \dot{h}_2, \ldots, \dot{h}_l]$, with $l = m^2 n$. The final descriptor vector $H = [h_1, h_2, \ldots, h_l]$ is obtained after unit length normalization on $L_1$ and quantization to $q$ levels

$$h_i = \left\lfloor \frac{\dot{h}_i}{\sum_{j=1}^{l} \dot{h}_j} \, q \right\rfloor \tag{4}$$

A patch rotation by a factor $\alpha k$, where $\alpha = \frac{2\pi}{m}$, corresponds to a cyclic shift of the block histograms for each ring, without any vector element recomputation (see again Fig. 1)

$$H_{\alpha k} = \bigoplus_{i=0}^{n-1} \bigoplus_{j=0}^{m-1} H^j_{i,[k+j]_m} \tag{5}$$

In this sense, the sGLOH descriptor packs $m$ different descriptors of the same patch at different orientations so that two descriptor vectors $H$ and $\overline{H}$ are compared using the distance

$$\widehat{\mathcal{D}}(H, \overline{H}) = \min_{k=0,\ldots,m-1} \mathcal{D}(H, \overline{H}_{\alpha k}) \tag{6}$$

induced by a generic distance $\mathcal{D}$, such as the Euclidean or Manhattan distance. Note that it also holds that

$$\widehat{\mathcal{D}}(H, \overline{H}) = \min_{k,k'} \mathcal{D}(H_{\alpha k'}, \overline{H}_{\alpha k}) \tag{7}$$

where both $k$ and $k'$ range in $0, \ldots, m-1$, since relative rotations are involved so that

$$\mathcal{D}(H, \overline{H}_{\alpha k}) = \mathcal{D}(\overline{H}, H_{\alpha[-k]_m}) \tag{8}$$

For the best setup [11], $n = 2$ and $m = 8$, so that $l = 128$ and $\alpha = 45°$. Furthermore, $c = 0.7$, $q = 512$ and the patch radii of the circular grid are set to 12 and 20 so that the normalized patch size is $41 \times 41$ pixels, following the indications of [8] already adopted for other histogram-based descriptors such as PCA-SIFT, LIOP and MROGH. With respect to the original implementation described in [11], we found that using a look-up table for storing the exponential weight values required by Eq. (1), dramatically reduces the computation of about one half with no loss in the final descriptor robustness.

Specific matching strategies can be arranged by exploiting the additional orientation information provided by limiting the rotations to check [11]. This can reduce the number of wrong matches, since some of these are dropped and cannot be selected by chance. Doing this either a priori or according to image context, gives rise respectively to the sCOr (shifting Constrained Orientation) and sGOr (shifting Global Orientation) matching strategies. sCOr constrains the range of orientations to be checked up to the first clockwise and counterclockwise discrete rotations, i.e. $k = m-1, 0, 1$ in Eq. (6), handling an increase in patch rotation up to $\pm 45°$, which is enough for most practical applications, such as automotive SLAM, SfM and mosaicing (nowadays cameras or phones automatically adjust the image orientation, limiting rotation errors to no more than $\pm 45°$). Similarly, sGOr uses the information provided by scene context to provide a global reference orientation, under the reasonable assumption that all keypoints of the scene undergo roughly the same rotation $\alpha g$, not known a priori. The range of discrete orientations in Eq. (6) is modified to $k = [g-1]_m, g, [g+1]_m$, where $g \in \{0, 1, \ldots, m-1\}$ can be estimated according to the most probable relative orientation among all matches, as follows. Given two images, the relative orientation $k^\star(H, S)$ of the best match pair containing the descriptor vector $H$ and any other $\overline{H}$ in the other image is

$$k^\star(H, S) = \underset{\substack{k=0,1,\ldots,m-1 \\ \overline{H} \in S}}{\arg\min} \mathcal{D}(H, \overline{H}_{\alpha k}) \tag{9}$$

where $S$ is the set of descriptor vectors of the other image. The histogram of the relative orientations is defined so that the bin $z_k$ counts the number of the best matches with relative discrete orientation $\alpha k$

$$\begin{aligned} z_k = &\sum_{H_1 \in S_1} f(k = [k^\star(H_1, S_2)]_m) + \\ &\sum_{H_2 \in S_2} f(k = [-k^\star(H_2, S_1)]_m) \end{aligned} \tag{10}$$

where $f(W)$ is the indicator function (i.e. $f = 0/1$ if $W$ is false/true respectively). $S_1$ and $S_2$ are the sets of descriptor vectors for the images $I_1$ and $I_2$ respectively. The value of $g$ is finally given by

$$g = \underset{k=0,1,\ldots,m-1}{\arg\max} z_k \tag{11}$$

Consistently with the definition of $g$, wrong matches are distributed uniformly across the bins $z_k$, while correct matches follow a Gaussian distribution centered in $z_g$.

## 4 The Revised sGLOH Descriptor

### 4.1 Doubled sGLOH

The sGLOH descriptor, especially if coupled with the sCOr and sGOr matching strategies, obtains results comparable with state-of-the-art descriptors [11], but can suffer of performance degradations when the relative rotation between the patches approaches the one between two discrete rotations, i.e. it is of the form $k\frac{2\pi}{m} + \frac{\pi}{m}$ for $k = 0, \ldots, m-1$.

To fix this issue, a novel doubled sGLOH descriptor $H^\star = H^1 \bigoplus H^2$ is defined, concatenating the standard sGLOH descriptor of the patch $H^1$ with the sGLOH descriptor $H^2$ obtained after applying a rotation of $\frac{\pi}{m}$ to the patch. The proposed descriptor, referred to as sGLOH2, can handle up to $2m$ discrete rotations of $\frac{\pi}{m}$ degrees. Note that this design is more advantageous than imposing $2m$ directions in the sGLOH setup (see Sec. 3), as in the latter case smaller discriminative regions and more noisy histograms would be obtained [3], together with a longer descriptor length of $4m^2n$ instead of the $2m^2n$ sGLOH2 length. In the additional material, it has been shown experimentally that more than doubling sGLOH, i.e. concatenating three or more descriptors, does not bring any concrete advantages in terms of performances, but merely increases the computational effort.

Considering the sequence $\{0, \frac{\pi}{m}, \frac{2\pi}{m}, \frac{3\pi}{m}, \ldots\}$ of the $2m$ successive discrete rotations by a step of $\frac{\pi}{m}$, the corresponding ordered set of cyclic shifted descriptors is given by

$$Q(H^\star) = \{H^1_0, H^2_0, H^1_1, H^2_1, \ldots, H^1_{m-1}, H^2_{m-1}\} \tag{12}$$

where $H^1_k$ is the cyclic block shift applied to $H^1$ to get a patch rotation of $\alpha k$ as defined in Eq. (5), and similarly for $H^2_k$. Analogously to Eq. (6), the distance between sGLOH2 features $H^\star$ and $\overline{H}^\star$ is given by

$$\widehat{\mathcal{D}}_2(H^\star, \overline{H}^\star) = \min_{K \in Q(\overline{H}^\star)} \mathcal{D}(H^1_0, K) \tag{13}$$

Notice that although the descriptor length is now doubled, the computation of the distance $\mathcal{D}$ for a single rotation remains the same as for sGLOH.

Different matching strategies can be obtained in analogy with sCOr and sGOr. By limiting the rotations up to $\pm\frac{\pi}{m}$, i.e. using the subset $\{\overline{H}^1_0, \overline{H}^2_0, \overline{H}^2_{m-1}\}$ instead of $Q(\overline{H}^\star)$ in Eq. (13) we get the sCOr2.1 strategy. Using a wider rotation range up to $\pm 2\frac{\pi}{m}$ results instead in sCOr2.2, with the subset $\{\overline{H}^1_0, \overline{H}^0_2, \overline{H}^1_1, \overline{H}^1_{m-1}, \overline{H}^2_{m-1}\}$ replacing $Q(\overline{H}^\star)$ in Eq. (13).

Analogously to sGOr, the estimation of the global reference orientation $g$ can be achieved either using all the $2m$ rotations in $Q$ (sGOr2a), or only the $m$ rotations belonging to the first concatenated sGLOH descriptor $H^1$ (sGOr2h), constraining the relative rotation window after finding $g$ to $\pm\frac{\pi}{m}$ as for sCOr2.1.

### 4.2 Fast Matching

Given a descriptor $H$, the general matching process can be regarded as looking for the corresponding descriptor $\overline{H} = \arg\min_{\widetilde{H} \in S} \widehat{\mathcal{D}}(H, \widetilde{H})$, where the set $S$ defines the chosen matching strategy. The time required for matching with a standard descriptor of length $l = 128$ as SIFT on a set $S$ of $|S| = s$ descriptors is proportional to $T_s = sl$. Instead, for the corresponding sGLOH-based matching, $|S| = \rho s$, where $\rho$ is the number of rotations to check, so that the computational time is proportional to $T_l = \rho s l$, with $\frac{T_l}{T_s} = \rho$. The same considerations hold for the memory required to store data.

In order to speed-up the matching process, candidate matches can be filtered according to partially computed distances at run-time. In particular, we consider the $z$ partial descriptor vector blocks $P_i$, $1 \leq i \leq z$ of size $\lceil l/z \rceil$ on which the descriptor vector $H = [h_1, h_2, \ldots, h_l] = \bigoplus_{i=1}^{z} P_i$ is split. Notice that in this partition, $P_i$ does not have to be equal to a block histogram. The distances $\mathcal{D}$ on $H$ used in Eq. (6) and (13) can be defined in terms of sums on the partial blocks $P_i$, i.e.

$$\mathcal{D}(H, \widetilde{H}) = \mathcal{P}_z(H, \widetilde{H}) = \sum_{i=1}^{z} \mathcal{D}(P_i, \widetilde{P}_i) \tag{14}$$

therefore, assuming $S_0 = S$, $\mathcal{P}_0(H, \widetilde{H}) = 0$ and $\mu_0 = \infty$, we can define the filtered set $S_i$ recursively as

$$S_i = \begin{cases} \{\widetilde{H} \in S_{i-1} \,|\, \mathcal{P}_{i-1}(H, \widetilde{H}) < \mu_{i-1}\} & \text{if } |S_{i-1}| > t_s \\ S_{i-1} & \text{otherwise} \end{cases} \tag{15}$$

In Eq. 15, $t_s$ is a threshold to limit the set shrinking and $\mu_i$ is the average precedent partial distance on the previous filtered set $S_{i-1}$

$$\mu_i = \sum_{\widetilde{H} \in S_{i-1}} \frac{\mathcal{P}_{i-1}(H, \widetilde{H})}{|S_{i-1}|} \tag{16}$$

The cardinality of the set $S_i$ is about halved at each iteration $i = 0, \ldots, z$ up to the fixed limit $t_s$, and incremental distances are only needed to be computed on the partial block $P_i$. Under the assumption that the descriptor vector elements are in decreasing order according to their discriminative ability, the final set $S_z$ would contain with high probability the descriptor $\overline{H}$ that best matches $H$, or anyway a close approximation to it, and its matching distance is given by $\mathcal{P}_z$. The parameters were set experimentally to $z = 10$ and $t_s = 32$.

This approximated fast matching scheme is quite robust and efficient (see Sec. 5). sGLOH histograms are concatenated in a spiral-like manner from the inner ring (which is more discriminative according to the analysis in [55]) to the outer rings (see Eq. (3)). In this way, fast matching achieves a good approximation of the original matching strategy. Inside a ring no particular starting sector is preferred, due to the inherent symmetry of the descriptor structure. In the case of sGOr strategies, the fast approximated matching efficiently avoids having to store all the computed distance values for each direction, since a very sparse distance matrix is obtained due to the shrinking constraint induced by $t_s$. In particular, instead of a $\rho \times n_1 \times n_2$ distance table for $\rho$ orientations and $n_1$, $n_2$ keypoints from images $I_1$ and $I_2$ respectively, only a $t_s \times \min(n_1, n_2)$ distance table is required.

Using the approximated fast matching scheme the running time is

$$T_f \approx \rho s \frac{l}{z} \sum_{i=0}^{z-1} 2^{-i} \leq 2\rho s \frac{l}{z} \tag{17}$$

This implies that $\frac{T_f}{T_l} \approx \frac{2}{z} = 0.2$, i.e. a speedup of about 5 is achieved by the approximated fast matching with respect to the exhaustive matching. Detailed time ratios that would be achieved for each sGLOH-based strategy are reported in Table 1. Notice that, theoretically, the slowest exhaustive matching strategies requiring respectively 16 and 10 times more than a standard descriptor, are reduced by fast matching to only 3 and 2 times. Actually, to further speedup the matching, $\mu_1$ and consequentially $S_1$ are updated online and not after scanning all the elements of $S_0$, thus progressively decreasing the value of $\mu_1$.

The proposed fast matching approach is similar to those described in [55], [56]. Yet, it does not require the additional structure and descriptor manipulations done in [56] and, differently from [55], it is adaptive.

TABLE 1
Time ratios with respect to SIFT for the different sGLOH-based exhaustive and fast matching strategies, $z = 10$ and $m = 8$

| | $\rho$ | $T_l/T_s$ (exhaustive) | $T_f/T_s$ (fast) |
|---|---|---|---|
| sGLOH | $m$ | 8 | 1.5 |
| sCOr | 3 | 3 | 0.6 |
| sGOr | $m$ | 8 | 1.5 |
| sGLOH2 | $2m$ | 16 | 3 |
| sCOr2.1 | 3 | 3 | 0.6 |
| sCOr2.2 | 5 | 5 | 1 |
| sGOr2h | $m+2$ | 10 | 2 |
| sGOr2a | $2m$ | 16 | 3 |

### 4.3 Binary sGLOH

In this section, a novel approach to binarize sGLOH-based descriptors is given, named BisGLOH (Binary sGLOH). Differently from most of the existing binary descriptors, BisGLOH does not operate comparisons directly on the patch intensities, but on histograms. With respect to the approach proposed in [43], that uses only consecutive bin comparisons, BisGLOH exploits more bin relations, obtaining a richer and more robust descriptor. Moreover, BisGLOH still maintains the same cyclic shift rotation property of the original sGLOH, but can be compressed in roughly half the space and uses the faster Hamming distance. Although less robust than the original sGLOH, as it is expected from a binary descriptor, it still provides valid results at a lower computational cost (see Sec. 5). Last, but not least, the approach behind BisGLOH is sufficiency general to be applied to other histogram-based descriptors.

For each sGLOH histogram of the patch region $\mathcal{R}_{r,d}$ defined by Eq. 2, $n \times m$ linearized tables $T^t_{r,d}$ of all binary comparisons are obtained (see Fig. 2)

$$T^t_{r,d} = \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} 2^{im+j} f^{t,i,j}_{r,d} \tag{18}$$

where

$$f_{r,d}^{t,i,j} = f\left(h_{r,d}^{[t+i]_m} \leq h_{r,d}^{[t+j]_m}\right) \tag{19}$$

is a binary comparison between histogram bins of the region $\mathcal{R}_{r,d}$ and $f(W)$ the indicator function of Eq. 10. Analogously, the strings $D_{r,d}$ comparing the gradient sum for each region histogram

$$C_{r,d} = \sum_{w=0}^{m-1} h_{r,d}^{w} \tag{20}$$

are built up for each ring so as to improve the descriptor robustness

$$D_{r,d} = \sum_{i=0}^{m-1} 2^i f_{r,d}^{i} \tag{21}$$

with

$$f_{r,d}^{i} = f\left(C_{r,d} \leq C_{r,[d+i]_m}\right) \tag{22}$$

Strings $D_{r,d}^{t}$ can be stacked into tables

$$D_r = \bigoplus_{i=0}^{m-1} D_{r,i} \tag{23}$$

As for Eq. (3) we concatenate the strings $T_{r,d}^{d}$ and $D_r$ to get the final descriptor $B$

$$\begin{aligned} B &= \bigoplus_{i=0}^{n-1}\bigoplus_{j=0}^{m-1} T_{i,j}^{j} \bigoplus_{i=0}^{n-1} D_i \\ &= \bigoplus_{i=0}^{n-1}\bigoplus_{j=0}^{m-1} T_{i,j}^{j} \bigoplus_{i=0}^{n-1}\bigoplus_{j=0}^{m-1} D_{i,j} \end{aligned} \tag{24}$$

Experiments have shown that using $q = 2048$ instead of 512 quantization levels in Eq. (4) for defining the sGLOH normalized vector $H$ used to generate $B$ works better, as finer comparisons are obtained. Note that patch rotations by a factor $\alpha k$ still correspond to cyclic shifts $B_{\alpha k}$ of the vector elements of $B$, as illustrated in Fig. 2

$$B_{\alpha k} = \bigoplus_{i=0}^{n-1}\bigoplus_{j=0}^{m-1} T_{i,[j+k]_m}^{j} \bigoplus_{i=0}^{n-1}\bigoplus_{j=0}^{m-1} D_{i,[j+k]_m} \tag{25}$$

The length of $B$ is $b_u = m^3 n + m^2 n$ bits, i.e. $b_u = 144$ bytes for the usual parameters $m = 8, n = 2$. Considering the byte alignment into memory for efficient descriptor rotations, each $T_{r,d}^{d}$ table, occupying $m^2$ bits can be easily decimated into $\left\lceil \frac{1}{8}\frac{m(m-1)}{2} \right\rceil$ bytes due to the inherent skew-symmetry of the tables. This would also be possible for each $D_r$ table up to a permutation of its elements, but that does not permit efficient implementations of the descriptor rotations. Hence, each $D_{r,d}$ string actually requires 1 byte. Under these observations, $B$ is easily decimated into $b_c = nm\left\lceil \frac{1}{8}\frac{m(m-1)}{2} \right\rceil + n\left\lceil \frac{m^2}{8} \right\rceil$ byte strings, i.e. 80 bytes, and the faster Hamming distance can be used as base distance $\mathcal{D}$ in Eq. (6). Operating on the decimated string, the Hamming distance weights the $D_{r,d}$ strings twice with respect to the $T_{r,d}$ tables. Notice that this design can also benefit of the fast matching described in Sec. 4.2 and the BisGLOH descriptor length is only $\left\lceil \frac{1}{8}\left(nm\frac{m(m-1)}{2} + n\frac{m(m-1)}{2}\right) \right\rceil = 63$ bytes for storage purposes, or in the case the right patch rotation $k$ in Eq. (25) is known a priori.

The BisGLOH descriptor can be easily doubled as done in Sec. 4.1 and the same matching strategies can be used, leading to the effective BisGLOH2 descriptor, that can be stored into 126 bytes and expanded into a 160 byte string for matching.

Note that no theoretical comparisons concerning the speedup with respect to the SIFT descriptor can be done in this case, due to the different descriptor length and distance used.

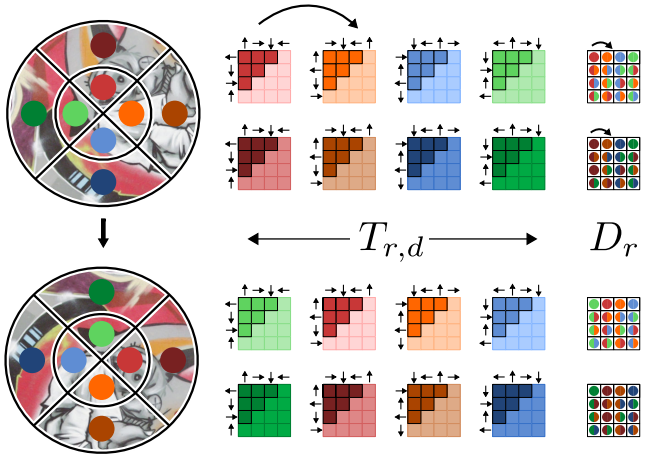



Fig. 2. Rotation of an image patch, and the corresponding cyclic shift of the BisGLOH histogram comparison tables. Color labels for each region $\mathcal{R}_{r,d}$ in the image patch identify the corresponding table $T_{r,d}^{t}$, whose darker entries only are concatenated in the descriptor, due to the skew-symmetry table decimation. Similarly, the two color cell entries in the $D_r$ tables indicate the two regions whose gradient sums $D_{r,d}$ are compared. In the example $n = 2$ and $m = 4$ (best viewed in color and zoomed in).

## 5 Experimental Evaluation

In order to evaluate the proposed sGLOH2 and BisGLOH2 descriptors together with the fast matching distance computation, several experiments on both image matching and object recognition were carried out. The code used for this evaluation is freely available[1]. The proposed matching strategies were compared against usual matching with several remarkable descriptors. These include the well known SIFT, that is considered as reference, LIOP, MIOP and MROGH, that represent the state of the art for rotational invariant descriptors, and RFDs, that are among the best binary descriptors. Additionally, the descriptor proposed in [18], here referred to as DeepDesc, and the SIFT coupled with the orientation estimation described in [19], both based on CNNs, were also included in the evaluation as interesting emerging approaches.

### 5.1 Setup

#### 5.1.1 Image Matching

The evaluation consists of seven different experiments, one of which dealing with keypoint matching in the case of synthetic rotations, two with general image transformations for real planar scenes, and four dealing with non-planar real scenes. The average image resolution is $800 \times 600$ pixels.

1. http://cvg.dsi.unifi.it/

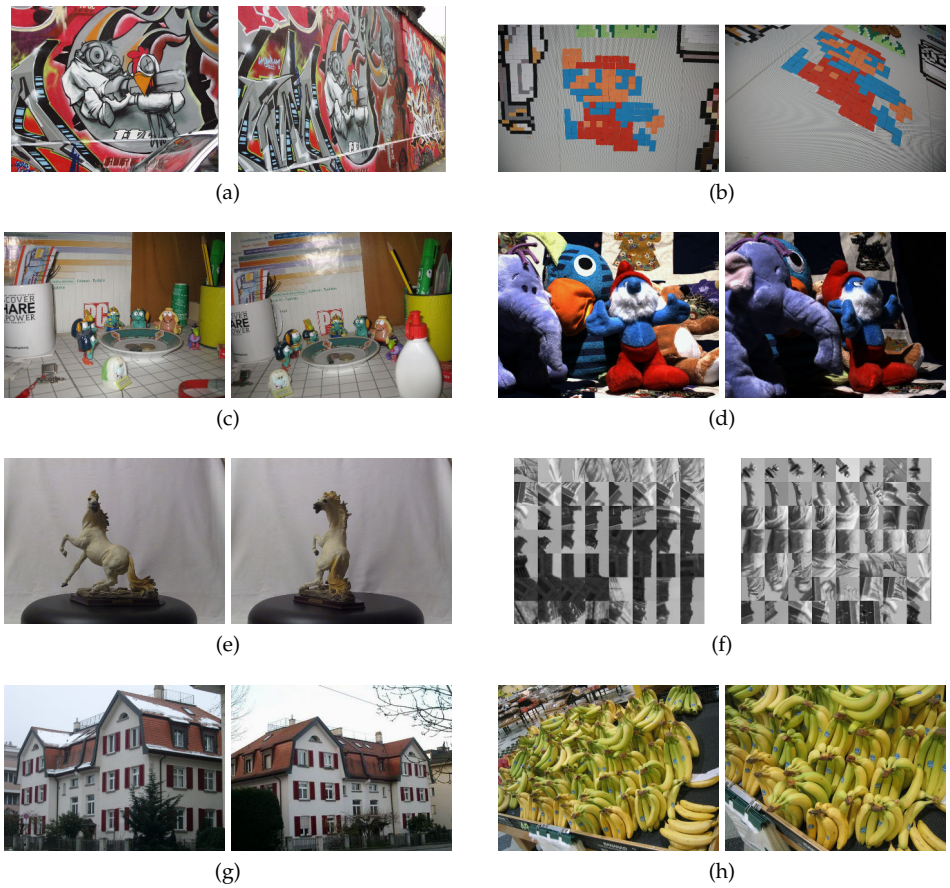


Fig. 3. Corresponding image pairs from the Oxford (a), Viewpoint (b), Approximated overlap (c), DTU (d), Turntable (e), Patch (f), ZuBuD (g) and Kentucky (h) datasets (best viewed in color).

In order to test descriptors under image rotations, 17 different images were artificially rotated up to $90°$ with a step of $3°$, and correct matches are evaluated as described in [11].

Matching in the case of planar scenes was carried out on the Oxford dataset [8], containing image pairs of planar scenes undergoing several image transformations including scale, rotation, image blur, illumination, JPEG compression and viewpoint changes. An image pair from this dataset is shown in Fig. 3a. Ground-truth homographies for each image pair are provided in order to compute precision-recall curves according to the overlap error [8] between matched keypoint patches. Note that the actual support region for MROGH is 2.5 times bigger than the elliptic region employed by the other descriptors [10]. This implies that better results are expected for MROGH in the case of non-planar image transformations, since more discriminative data is available, as pointed out in [11], [14]. MROGH results can thus be used as upper bounds for the planar scene evaluation.

In order to gain further insight into the descriptor robustness, another evaluation in the case of planar scenes was done using the recent Viewpoint dataset [19] (see Fig. 3b), containing 5 planar scenes where images are subject to different incremental viewpoint and scale changes.

Results with non-planar scenes were evaluated using the Approximated overlap [11], DTU [21], Turntable [22] and the Patch [23] datasets. In the first case, ground-truth data were computed according to the approximated overlap error [20] on 42 different images pairs, extending the original dataset (see Fig. 3c). The approximated overlap error is computed on ground-truth fundamental matrices and it has a low false positive rate (less than 5%), not affecting descriptor ranking in unsupervised evaluations [20].

For the DTU dataset (see Fig. 3d) we used 18 sequences with 9 camera positions for each scene, corresponding to the leftmost, middle and rightmost positions for each of the three camera arc paths present, and 4 different lighting conditions. The reference frame is fixed, as suggested by authors, to the middle inner arc camera view, so that a total of $8 \times 4 \times 4 = 128$ image pairs were evaluated for each sequence. Ground-truth is established according to the 3D mapping derived by the structured light 3D data accompanying the dataset. Such 3D map is used to define the overlap error as done in [57].

The Turntable dataset is composed by 88 sequences, each showing 3D objects with different shapes rotating on a turntable. For each rotation, both clockwise and counterclockwise rotations up to $\pm 50°$ with a step of $5°$ were taken into account (see Fig. 3e). The system is calibrated so that ground-truth matches can be established by using epipolar constraints between triplets of calibrated views. When a match between the two input views does not have a correspondence in the auxiliary view, it is excluded and

does not contribute to any statistic [22]. Views at $0^\circ$ and $90^\circ$ were employed as reference, so that for each object a total of $2 \times 4 \times 10 = 80$ image pairs were tested.

The Patch dataset, consists of corresponding patches sampled from the 3D reconstruction of 3 sequences (Liberty, Notre Dame and Yosemite), obtained with SfM followed by a dense stereo map estimation. For each sequence, more than 400k patches were obtained both with the DoG (Difference of Gaussian) and Harris interest point detectors. Patches are normalized to a size of $64 \times 64$ pixels, so that the orientations between corresponding patches differ by no more than $\frac{\pi}{8} = 22.5^\circ$ (see Fig. 3f). We randomly selected about 65k distinct patch pairs for each available detector and sequence, of which only half of them are correct.

#### 5.1.2 Object Recognition

Tests were carried out using the ZuBuD and Kentucky datasets. The ZuBuD dataset [24] contains 1005 images of 201 buildings, each taken from 5 random arbitrary viewpoints and under different conditions (see Fig. 3g). The Kentucky dataset [25] contains images of 2550 objects, each seen from 4 different viewpoints (see Fig. 3h). All images in both datasets have a resolution of $640 \times 480$ pixels. In the case of the Kentucky dataset, we used only the images of the first 750 objects, for a total of $4 \times 750 = 3000$ images. Query images were matched with all the others in dataset, and the first 4 (ZuBuD), 3 (Kentucky) most similar images with the highest number of keypoint matches were returned. Two keypoints are said to correspond if their matching distance is below a threshold. For each descriptor, we selected the threshold value that gives the best results on the considered dataset. Differently from [10], the number of keypoint matches per image pair was not normalized by the product of the keypoints of the two images.

A further object retrieval test was taken into account, using the Kentucky and Turntable datasets. Specifically, for each of the 88 Turntable objects, the view taken at $\pm 30^\circ$, $\pm 40^\circ$ and $\pm 50^\circ$ with respect to the reference view were used as query images against a database composed of 750 images from the Kentucky dataset (i.e. one for each distinct objects) plus the reference Turntable object view. As described above, only the first most similar image was returned.

#### 5.1.3 Setup Protocol

We used the descriptor implementations provided by the authors, except for SIFT for which both the Mikolajczyk's implementation [8] and the one included in the VLFeat library [58] were used. They are respectively denoted as SIFT and VL SIFT. VLFeat was also used to get patches and orientation estimation for DeepDesc, since the DeepDesc implementation works only on already normalized patches. The CNN-based orientation estimation proposed in [19] was coupled with VL SIFT and denoted by the superscript "$*$" (more in detail, we used the EdgeFoci/SIFT with random rotation learned CNN).

Except for the Patch dataset, the HarrisZ corner detector [59] was used to extract keypoints from images, whose results are similar to those of state-of-the-art detectors. The HarrisZ detector outputs a lower number of similar keypoints (i.e. with close scale, rotation and location) with respect to the Hessian-affine detector [9], but both obtain similar relative ranks among descriptors. We also validated this choice on the Viewpoint dataset in terms of mean Average Precision (mAP), computed similarly to [60], by comparing the HarrisZ detector against the EdgeFoci detector [61] that provided the best results on this dataset in a recent evaluation [19] (see additional material). Note that, unlike [19], we did not retain only the first 1000 keypoints for each detector, since some detectors, including HarrisZ, output keypoints by increasing scale, so that more robust and discriminant keypoints would be excluded with this setup.

In the case of image matching for planar and non-planar scenes, results in terms of absolute values could change dramatically according to the image transformations. To better appreciate the relative differences between descriptors given the generic quality metric $e(a, I)$ for the descriptor $a$ and the image pair $I$, so that higher values of $e(a, I)$ implies better results, we define the *soft rank* $r(a, I)$ as

$$r(a, I) = \frac{|e(a, I) - b(I) + \varepsilon|}{\sum_{a' \in \{a\}} |e(a', I) - b(I) + \varepsilon|} \tag{26}$$

where $\varepsilon$ is a small constant to avoid a zero-denominator and $b(I)$ is the best value among the descriptors for the image pair $I$

$$b(I) = \max_{a' \in \{a\}} e(a, I) \tag{27}$$

The soft rank $r(a, I)$ ranges between $[0, 1]$, achieves lower values for better descriptors and is equal to $\frac{1}{|\{a\}|}$ when $e(a, I)$ is the same for all descriptors.

Table 2 shows the main characteristics of the descriptors involved in this evaluation. Binary descriptors use the Hamming distance $H$. For histogram-based descriptors, the $L_1$ Manhattan distance is used instead of the $L_2$ Euclidean distance in all cases except for MIOP and DeepDesc. This choice is motivated by better mAP results on the Oxford dataset (reported in the additional material) and by previous evaluations [11]. Table 2 also reports the descriptor length, indicating the char, integer and floating-point type vectors respectively with subscripts "$C$", "$I$" and "$F$". In our experiments no integer vector entry was greater than 255, hence, for storage purposes, integers can be considered as chars. In this sense, floating-point vectors require at least four times the memory of the other vectors. Notice also that, in the case of sGLOH-based descriptors, sGLOH2 and BisGLOH2 use only half of the data available when computing the distance, while for storage purposes, both binary BisGLOH and BisGLOH2 can be packed more compactly.

In the case of image matching tests, Nearest Neighbor (NN) matching is preferred to Nearest Neighbor Ratio (NNR) matching [3] in ranking descriptor distances for strategies relying on sGLOH, since Eq. (6) minimizes the score across matches, that the NNR matching would instead maximize [11]. Nevertheless, in the case of object recognition tests, an improvement of about 5-10% on the successfully retrieved queries was observed in evaluations using the NNR matching. This is likely due to the impossibility to set an absolute fixed distance NN threshold valid for all the database queries, so that a relative measure such as NNR yields better results.

TABLE 2
Descriptor evaluation setup details

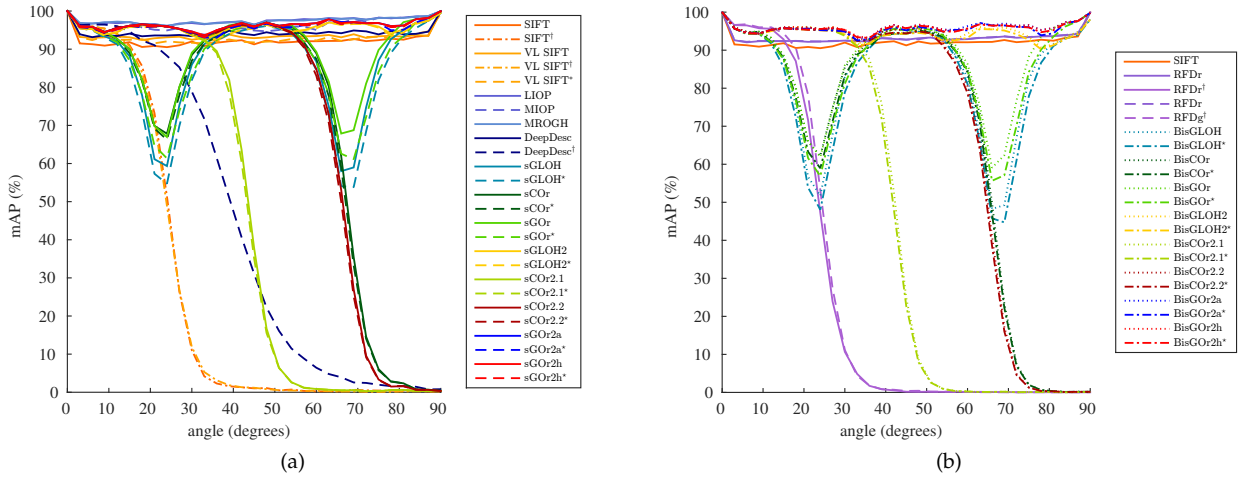
| | $\mathcal{D}$ | Descriptor length | | | Matching strategy | |
|---|---|---|---|---|---|---|
| | | Packed | Expanded | Used | Image matching | Object recognition |
| SIFT | $L_1$ | – | – | $128_I$ | NNR | NNR |
| VL SIFT | $L_1$ | – | – | $128_F$ | NNR | NNR |
| VL SIFT* | $L_1$ | – | – | $128_F$ | NNR | NNR |
| LIOP | $L_1$ | – | – | $144_I$ | NNR | NNR |
| MIOP | $L_2$ | – | – | $128_F$ | NNR | NNR |
| MROGH | $L_1$ | – | – | $192_I$ | NNR | NNR |
| DeepDesc | $L_2$ | – | – | $128_F$ | NNR | NNR |
| RFDr | $H$ | – | – | $40_C$ | NNR | NNR |
| RFDg | $H$ | – | – | $56_C$ | NNR | NNR |
| sGLOH | $L_1$ | – | $128_I$ | $128_I$ | NN | NNR |
| sGLOH2 | $L_1$ | – | $256_I$ | $128_I$ | NN | NNR |
| BisGLOH | $H$ | $63_C$ | $80_C$ | $80_C$ | NN | NNR |
| BisGLOH2 | $H$ | $126_C$ | $160_C$ | $80_C$ | NN | NNR |

## 5.2 Results

### 5.2.1 Image Matching

Figure 4 shows mAP results respectively for histogram-based and binary descriptors, in the case of the synthetic rotation test. A match is defined as correct if the overlap error is less than 50%, the "$\star$" superscript indicates that the fast matching is used, while the "†" superscript that the up-right version of the descriptor is used (i.e. no orientation estimation is done on the normalized patch). Detailed results for each image sequence, also in terms of correct match ratios that give similar ranking results, are reported as additional material. Among histogram-based descriptors, sGLOH2-based matching clearly improves on the original sGLOH-based matching, since the issue due to patch relative rotations between two descriptor discrete rotations is solved. The sCOr2.1 strategy can correctly handle rotations up to $\pm\frac{2\pi}{m} = 45°$, while sCOr2.2 up to $\pm\frac{3\pi}{m} = 67.5°$, similarly to sCOr but without in-between rotation issues. For both the sGLOH-based an BisGLOH-based strategies, using fast matching only slightly degrades the performances achieved with full matching. Fully rotational invariant LIOP and MROGH achieve the best results, followed by sGLOH2 and BisGLOH2 strategies, and MIOP with a difference of about 2%. DeepDesc, RFDs and the various SIFTs, come next. By inspecting the plots for the up-right descriptors, it is clear that no descriptor except DeepDesc can handle rotations of more than about $\pm\frac{\pi}{8} = 22.5°$ without a rotation handling mechanism.

Table 3 shows the results on planar scenes obtained with the Oxford dataset. In the table, msRE denotes the mean soft rank for the maximum recall achieved for a precision greater than 70%, and msAP denotes the mean soft rank for the average precision. mAP values are also reported. Detailed results for each image pair are reported in the additional material. No sCOr-based method is included, since the dataset does not meet the rotation constraints for some image pairs. Nevertheless, as reported in [16], results similar to their sGOr-based counterparts are expected in the case the rotation constraints are met. All the adopted metrics give similar ranks: MROGH, using a wider support region than other descriptors, achieves the best results, followed by MIOP, sGOr2 strategies, LIOP and sGLOH2. The BisGLOH2-based strategies come next, while lower rank positions are obtained in order by RFDs, SIFTs and DeepDesc. Results show that sGLOH2-based strategies behave better




(a) (b)

Fig. 4. mAP (%) on the rotation dataset for histogram-based (a) and binary (b) descriptors, SIFT is included in (b) as reference (best viewed in color and zoomed in).

TABLE 3
Results on the Oxford dataset. Lower values are better for msRE and msAP, higher for mAP

| | msRE (%) | | | | msAP (%) | | | | mAP (%) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Histogram | | Binary | | Histogram | | Binary | | Histogram | | Binary | |
| | Std | Fast | Std | Fast | Std | Fast | Std | Fast | Std | Fast | Std | Fast |
| SIFT | 6.1 | – | – | – | 7.4 | – | – | – | 60.3 | – | – | – |
| VL SIFT | 5.6 | – | – | – | 5.7 | – | – | – | 58.0 | – | – | – |
| VL SIFT* | 5.6 | – | – | – | 5.5 | – | – | – | 59.9 | – | – | – |
| LIOP | 2.3 | – | – | – | 2.3 | – | – | – | 69.5 | – | – | – |
| MIOP | 1.5 | – | – | – | 1.4 | – | – | – | 72.6 | – | – | – |
| MROGH | 0.4 | – | – | – | 0.5 | – | – | – | 75.6 | – | – | – |
| DeepDesc | 8.8 | – | – | – | 9.5 | – | – | – | 48.4 | – | – | – |
| RFDr | – | – | 5.4 | – | – | – | 5.5 | – | – | – | 61.6 | – |
| RFDg | – | – | 5.2 | – | – | – | 5.2 | – | – | – | 62.5 | – |
| sGLOH | 3.4 | 3.8 | 4.5 | 4.7 | 3.3 | 3.7 | 4.2 | 4.4 | 62.4 | 61.1 | 57.9 | 57.4 |
| sGOr | 2.5 | 3.1 | 3.5 | 3.9 | 2.4 | 3.1 | 3.4 | 3.8 | 65.3 | 62.9 | 61.0 | 59.9 |
| sGLOH2 | 2.5 | 2.8 | 3.6 | 3.8 | 2.5 | 2.7 | 3.4 | 3.6 | 68.2 | 67.0 | 64.0 | 63.6 |
| sGOr2a | 1.6 | 2.0 | 2.3 | 2.7 | 1.5 | 2.0 | 2.2 | 2.5 | 71.2 | 69.9 | 68.8 | 67.7 |
| sGOr2h | 1.6 | 2.0 | 2.4 | 2.7 | 1.5 | 2.0 | 2.2 | 2.6 | 71.4 | 69.1 | 67.9 | 66.8 |

TABLE 4
Results on the Viewpoints, Approximated overlap and DTU datasets. Lower values are better for msRE and msAP, higher for mAP

| | Viewpoint | | | Approximated overlap | | | DTU | | |
|---|---|---|---|---|---|---|---|---|---|
| | msRE (%) | msAP (%) | mAP (%) | msRE (%) | msAP (%) | mAP (%) | msRE (%) | msAP (%) | mAP (%) |
| SIFT | 10.0 | 10.4 | 53.4 | 8.7 | 9.7 | 40.0 | 7.7 | 8.3 | 28.9 |
| VL SIFT | 11.8 | 12.4 | 47.4 | 9.9 | 10.9 | 38.4 | 8.3 | 8.8 | 28.0 |
| VL SIFT* | 9.9 | 10.1 | 53.0 | 7.3 | 7.9 | 42.7 | 7.5 | 7.9 | 29.3 |
| LIOP | 6.5 | 6.2 | 58.2 | 7.5 | 7.9 | 41.9 | 8.7 | 9.7 | 27.4 |
| MIOP | 7.9 | 7.6 | 56.0 | 7.4 | 7.6 | 42.4 | 8.5 | 9.3 | 27.6 |
| MROGH | 3.8 | 3.4 | 63.0 | 6.8 | 5.7 | 45.5 | 10.2 | 11.0 | 25.4 |
| DeepDesc | 12.3 | 12.7 | 47.8 | 10.4 | 10.6 | 38.9 | 10.5 | 10.8 | 25.4 |
| RFDr | 7.6 | 7.5 | 57.5 | 8.5 | 9.5 | 40.0 | 8.6 | 9.2 | 27.6 |
| RFDg | 6.3 | 6.3 | 58.9 | 7.7 | 8.4 | 41.6 | 7.5 | 8.1 | 28.9 |
| sGLOH2* | 4.9 | 4.9 | 61.4 | 3.9 | 4.3 | 48.4 | 4.3 | 4.0 | 34.3 |
| sGOr2a* | 1.8 | 1.6 | 66.6 | 1.6 | 1.4 | 52.5 | 2.3 | 1.4 | 37.2 |
| sGOr2h* | 1.2 | 1.2 | 67.3 | 1.5 | 1.2 | 52.8 | 2.3 | 1.2 | 37.5 |
| BisGLOH2* | 7.7 | 7.8 | 55.3 | 8.2 | 7.4 | 43.3 | 6.1 | 5.5 | 32.5 |
| BisGOr2a* | 3.9 | 4.0 | 62.0 | 5.4 | 3.8 | 48.5 | 3.7 | 2.4 | 36.0 |
| BisGOr2h* | 4.0 | 4.1 | 62.1 | 5.2 | 3.7 | 48.6 | 3.8 | 2.4 | 36.1 |

than the original sGLOH-based strategies, fast matching introduces a minimal loss of correct matches (more evident on the binary descriptors), and binarization reduces the original discriminative power of descriptors. Notice also that the global orientation used in sGOr-based strategies allows finding more correct matches.

Table 4 shows the results obtained with the Viewpoint, Approximated overlap and DTU datasets (detailed results are reported in the additional material). According to the mAP results, the three datasets are of increasing scene complexity, as clear from inspecting the corresponding dataset images. Notice how progressively MROGH, LIOP and MIOP lose rank positions with respect to SIFTs and RFDs as the datasets become more challenging. The fast sGOr2 strategies achieve in all cases the best results, followed by sGLOH2 and their binary counterparts. This trend is consistent whatever evaluation metric is used. In the case of the DTU dataset, a further test is reported in the additional material, where the most natural lighting conditions are set for all the image pairs. Also in this case, no relevant changes in the ranking results are observed.

Fig. 5a shows mAP results on the Turntable dataset for increasing viewpoint angles (further details can be found in the additional material). In this case the best results are obtained by sCOr2h$^\star$ and sCOr2a$^\star$, followed by their binary counterparts, sGLOH2, BisGLOH2 and then MIOP and SIFTs. Notice that in terms of correct matches, the BisGLOH2-based strategies provide more matches than their non-binary versions, possibly due to a less discriminant power, compensated by a higher tolerance to patch distortions. This is also supported by the results on the Patch dataset discussed hereafter.

Figures 5b-c plot the ROC curves on the whole 200k matching pairs for the DoG and Harris keypoints, respectively (see the additional material for more detailed results). No sGOr-based results are given, since no image context data are available for this dataset. MROGH results are also absent, due to the wider support region that would be required. As expected, up-right descriptors obtain the best results (RFDs and DeepDesc are actually learned on subsets on this dataset) together with BisGLOH2, while sGLOH2 followed by LIOP, MIOP and VL SIFT$^\dagger$ come next, suggesting a better orientation handling than the canonical SIFT approach [3]. Unlike the plots reported in [14], we did

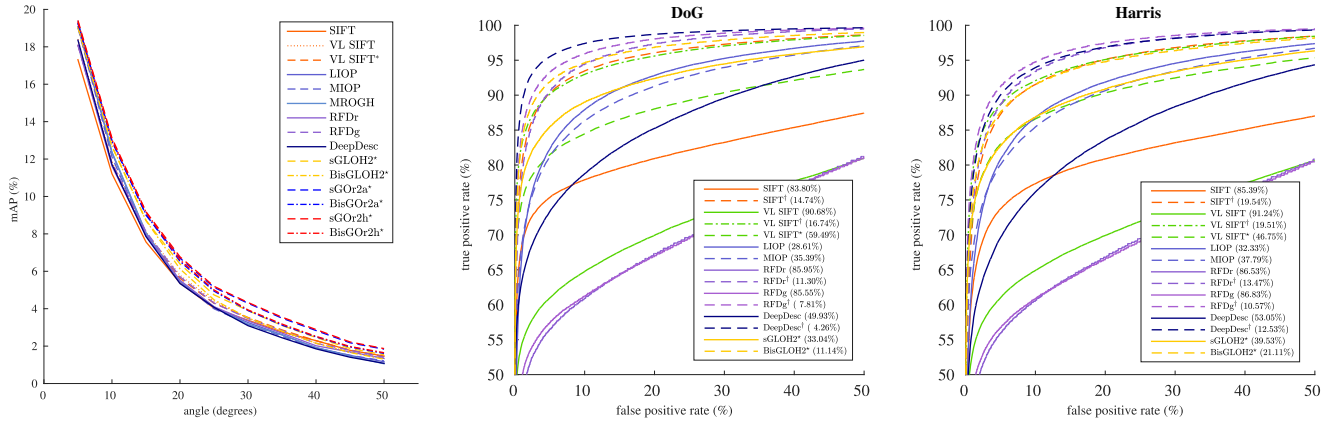



Fig. 5. mAP for increasing viewpoint angle on the Turntable dataset (a) and overall ROC curves on the Patch dataset for DoG (b) and Harris (c) keypoint patches. Error rate at 95% recall for each descriptor is also shown in the legend (best viewed in color and zoomed in).

not restrain the patch to an incircle of the original patch, as this would not represent the true detector output, since relevant data are also present in the patch boundary [8]. Moreover, sub-patches matching does not guarantee that the true corresponding patches still match. On the other hand, avoiding sub-patches can give rise to unwanted out-of-the-boundary patch rotations, that can successfully be dealt with as explained in the additional material.

### 5.2.2 Object Recognition

Table 5 shows the average percentage of correctly retrieved query images with the ZuBuD, Kentucky and Turntable datasets. Rank results appear to be stable and independent of the complexity of the datasets (ZuBuD and the Turntable at $\pm 30^\circ$ being the less complex ones). Notice that better quantitative results in terms of correctly retrieved queries are obtained with respect to previous similar evaluations on the same ZuBuD and Kentucky datasets [17], possibly due to the usage of a different keypoint detector and matching criteria. DeepDesc achieves the best results. Almost all other descriptors follow, with results comparable to each other. LIOP and MIOP come last. BisGLOH2-based strategies provide better results than their non-binary counterparts, as the viewpoint angle increases in the Turntable dataset, with a behavior similar to that discussed in the Patch dataset evaluation. Notice also that sGOr2h strategies and sGLOH2 perform better than sGOr2a strategies, possibly due to a higher noise introduced by the finer rotation quantization into the global rotation estimation.

It is worth noticing that all state-of-the-art descriptors included in our evaluation that are top ranked at image matching are among the last ranks in the object recognition and vice-versa. This fact underlines the subtle differences between these two tasks. In particular, image matching requires high sensitivity, i.e. to correctly identify correct matches, while object recognition requires high specificity, i.e. to correctly discard wrong matches. On the other hand, the sGLOH-based descriptors and matching strategies appear in the first rank positions both for image matching and object recognition. This means that they enjoy a good balance between sensitivity and specificity, and are equally suitable for matching and recognition.

### 5.2.3 Running Times

Figure 6 shows the average running times on an Intel Core i7-4790K processor for a sample subset of input image pairs

TABLE 5
Average correctly retrieved queries (%) for the object retrieval tests

| | ZuBuD | | Kentucky | | Turntable / Kentucky $\pm 30^\circ$ | | $\pm 40^\circ$ | | $\pm 50^\circ$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Histogram | Binary | Histogram | Binary | Histogram | Binary | Histogram | Binary | Histogram | Binary |
| SIFT | 94.9 | – | 89.6 | – | 94.8 | – | 86.7 | – | 82.1 | – |
| VL SIFT | 95.0 | – | 90.7 | – | 97.7 | – | 94.5 | – | 85.9 | – |
| VL SIFT* | 95.5 | – | 91.8 | – | 98.0 | – | 91.1 | – | 85.9 | – |
| LIOP | 92.5 | – | 81.8 | – | 94.5 | – | 83.0 | – | 78.4 | – |
| MIOP | 93.6 | – | 83.4 | – | 93.9 | – | 84.1 | – | 78.7 | – |
| MROGH | 94.4 | – | 88.4 | – | 96.8 | – | 93.4 | – | 85.9 | – |
| RFDr | – | 95.0 | – | 88.9 | – | 95.1 | – | 90.2 | – | 80.7 |
| RFDg | – | 95.5 | – | 89.6 | – | 97.1 | – | 89.0 | – | 82.1 |
| DeepDesc | 94.6 | – | 91.6 | – | 98.3 | – | 96.8 | – | 91.9 | – |
| sGLOH2* | 95.2 | 94.6 | 89.8 | 87.4 | 95.7 | 97.7 | 89.6 | 91.6 | 81.0 | 83.6 |
| sGOr2a* | 95.3 | 94.3 | 85.9 | 81.3 | 92.5 | 95.7 | 87.9 | 90.5 | 78.4 | 82.4 |
| sGOr2h* | 95.7 | 95.1 | 89.6 | 82.8 | 96.5 | 98.0 | 92.5 | 93.7 | 85.6 | 87.3 |

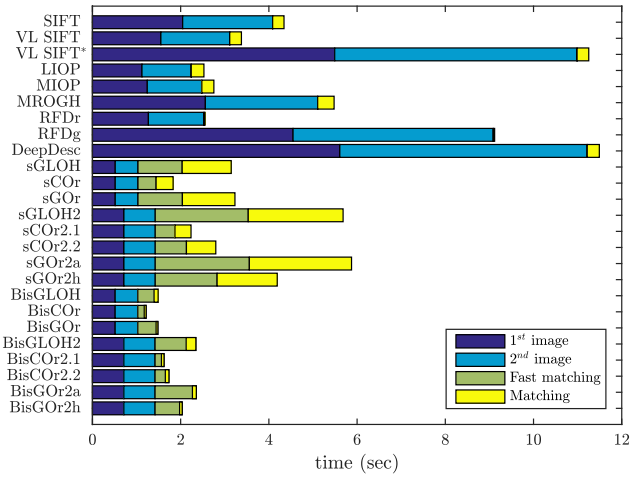



Fig. 6. Average cumulative histogram of the running times for each evaluated descriptor (best viewed in color and zoomed in).

from the employed datasets. The total cumulative time needed to compute the descriptors on both images and matching them is reported. Descriptor times (dark and light blue bars) include the computations required to normalize and rotate (if needed) the input patches. On average, about 1500 keypoints are computed for each image of the input pair. Fast matching times (green bars) are superimposed to full matching times (yellow bars). All algorithms run on a single CPU thread, except for DeepDesc than runs on a GeForce GTX 960 GPU (a batch of 1000 keypoints at a time is fed as input to the CNN, due to the 4Gb memory limitations of our system). We also developed free multi-threaded implementations for sGLOH-based methods and for general matching strategies[1]. Running times for sGLOH-based descriptors are the lowest, and using the fast matching scheme the total running times are comparable to those of the other methods. Binary BisGLOH matching strategies are faster than their sGLOH counterparts, as they use the Hamming distance instead of the Manhattan distance and have a shorter descriptor length. The fast matching speedup is about $2\times$, less than half of that expected according to the theoretical analysis in Sec. 4.2, due to the overhead in managing the additional data structures. The speedup is even lower in the case of binary descriptors, which nevertheless require less memory, especially in the case of sGOr-based matching (see again Sec. 4.2). While the differences between sGOr2a and sGOr2h are minimal in terms of outputs, this is not true for the computational times, as sGOr2h checks only about half of the directions needed by sGOr2a to guess the global orientation (see Table 1). Notice also the additional speed advantage in using sCOr-based strategies in the case of constrained rotations known a priori.

The matching time for non-binary sGLOH-based strategies is quite relevant with respect to the descriptor computation time, so that higher running times are expected for these methods in the case of many-vs-many matching applications, since descriptor computation would be linear with the number of images to match but pairwise image matching would be quadratic. Figure 7 shows, for each evaluated descriptor, the parabolic fitting of the total running time according to the average number $k$ of keypoints between the input pair images, $k = \sqrt{k_1\, k_2}$, where $k_1$ and $k_2$ are the number of keypoints in the first and second input images, respectively.

#### 5.2.4 Evaluation Summary

According to the experimental evaluation carried out, sGOr2h⋆ obtained the best overall ranking in image matching and object recognition tests. The sGLOH2⋆ descriptor follows. In addition, the sGOr2h⋆ running time is quite effective among histogram-based descriptors. Similar considerations hold for binary descriptors, where BisGOr2h⋆ is the fastest, among the approaches presented, in the case of one-to-one image matching.

## 6 Conclusions and Future Work

We proposed a revised sGLOH descriptor that solves all problems of performance loss due to rotation quantization. To cope with the increased matching time, we designed an approximated fast matching method that provides a $2\times$ speedup, also reducing the memory usage, with a negligible loss in descriptor discriminative power. In addition, we provided a new binarization technique to further reduce the running time and provide a more compact, yet valid, descriptor. All the proposed techniques can be combined together according to the task requirements in order to obtain valid, robust and efficient matching strategies with results better than the current state of the art, especially in the case of image matching. Both the approximated fast matching and the binarization techniques are general and could be applied and validated on other histogram-based descriptors as future work.

## Acknowledgment

The authors would like to thank Giosuè Lo Bosco, Domenico Tegolo and Cesare Valenti for granting access to the computational resources of the University of Palermo.

This work was supported by the SUONO project (Safe Underwater Operations iN Oceans), SCN_00306, ranked first in the challenge on “Sea Technologies” of the competitive call named “Smart Cities and Communities” issued by the Italian Ministry of Education and Research.

## References

[1] N. Snavely, S. Seitz, and R. Szeliski, “Modeling the world from internet photo collections,” *International Journal of Computer Vision*, vol. 80, no. 2, pp. 189–210, 2008.

[2] R. Mur-Artal, J. Montiel, and J. Tardos, “ORB-SLAM: a versatile and accurate monocular slam system,” *IEEE Transactions on Robotics*, 2015, to appear.

[3] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[4] K. Grauman and T. Darrell, “The pyramid match kernel: Discriminative classification with sets of image features,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2005.

[5] J. Shi and C. Tomasi, “Good features to track,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1994, pp. 593–600.

[6] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, “The devil is in the details: an evaluation of recent feature encoding methods,” in *Proceedings of the British Machine Vision Conference (BMVC)*, 2011.

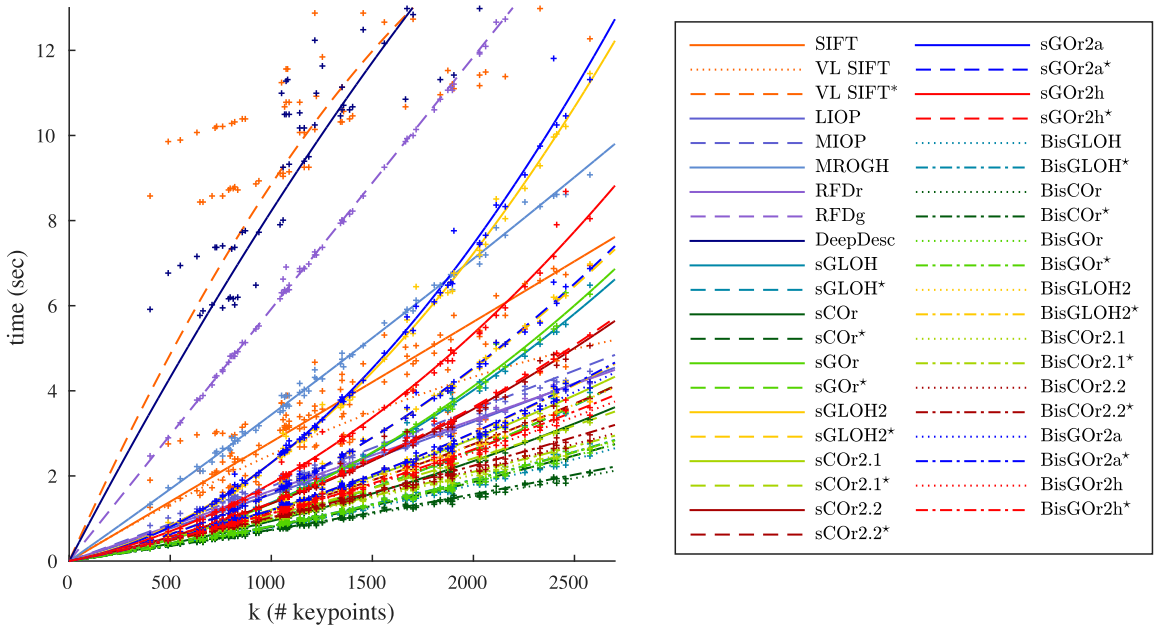



Fig. 7. Descriptor total running time vs average number of keypoints (best viewed in color and zoomed in).


[7] T. Tuytelaars and K. Mikolajczyk, "Local invariant feature detectors: a survey," *Foundations and Trends in Computer Graphics and Vision*, vol. 3, no. 3, pp. 177–280, 2008.

[8] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[9] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, 2005.

[10] B. Fan, F. Wu, and Z. Hu, "Rotationally invariant descriptors using intensity order pooling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 2031–2045, 2012.

[11] F. Bellavia, D. Tegolo, and C. Valenti, "Keypoint descriptor matching with context-based orientation estimation," *Image and Vision Computing*, vol. 32, no. 9, pp. 559–567, 2014.

[12] S. Gauglitz, M. Turk, and T. Höllerer, "Improving keypoint orientation assignment," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2011.

[13] Z. Wang, B. Fan, and F. Wu, "Local intensity order pattern for feature description," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 603–610.

[14] Z. Wang, B. Fan, G. Wang, and F. Wu, "Exploring local and overall ordinal information for robust feature description," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 99, 2016.

[15] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[16] F. Bellavia and C. Colombo, "Extending the sGLOH descriptor," in *Proceedings of the International Conference on Image Analysis and Processing (ICIAP)*, 2015, pp. 354–363.

[17] B. Fan, Q. Kong, T. Trzcinski, Z. Wang, C. Pan, and P. Fua, "Receptive fields selection for binary feature description," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2583–2595, 2014.

[18] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015.

[19] K. Yi, Y. Verdie, P. Fua, and V. Lepetit, "Learning to assign orientations to feature points," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1–8.

[20] F. Bellavia, C. Valenti, C. A. Lupascu, and D. Tegolo, "Approximated overlap error for the evaluation of feature descriptors on 3D scenes," in *Proceedings of the International Conference on Image Analysis and Processing (ICIAP)*, 2013, pp. 270–279.

[21] H. Aanæs, A. L. Dahl, and K. S. Pedersen, "Interesting interest points," *International Journal of Computer Vision*, vol. 97, no. 1, pp. 18–35, 2012.

[22] P. Moreels and P. Perona, "Evaluation of features detectors and descriptors based on 3d objects," *International Journal of Computer Vision*, vol. 73, no. 3, pp. 263–284, 2007.

[23] M. A. Brown, G. Hua, and S. A. J. Winder, "Discriminative learning of local image descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 1, pp. 43–57, 2011.

[24] H. Shao, T. Svoboda, and L. Van Gool, Computer Vision Lab, Swiss Federal Institute of Technology, Tech. Rep. No.260.

[25] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2006, pp. 2161–2168.

[26] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proceedings of the European Conference on Computer Vision (ECCV)*, vol. 2, 1994, pp. 151–158.

[27] Y. Ke and R. Sukthankar, "PCA-SIFT: a more distinctive representation for local image descriptors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2004, pp. 506–513.

[28] H. Bay, A. Ess, T. Tuytelaars, and L.Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[29] S. Leutenegger, M. Chli, and R. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011.

[30] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2564–2571.

[31] A. Alahi, R. Ortiz, and P. Vandergheynst, "Freak: Fast retina keypoint," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 510–517.

[32] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using local affine regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1265–1278, 2005.

[33] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2911–2918.

[34] H. Cheng, Z. Liu, N. Zheng, and J. Yang, "A deformable local image descriptor," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[35] J. Dong and S. Soatto, "Domain-size pooling in local descriptors: DSP-SIFT," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[36] J. Morel and G. Yu, "ASIFT: A new framework for fully affine invariant image comparison," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 438–469, 2009.
[37] T. Hassner, V. Mayzels, and L. Zelnik-Manor, "On SIFTs and their scales," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 1522–1528.
[38] M. Heikkila, M. Pietikainen, and C. Schmid, "Description of interest regions with local binary patterns," *Pattern Recognition*, vol. 42, no. 3, pp. 425–436, 2009.
[39] E. Tola, V. Lepetit, and P. Fua, "Daisy: an efficient dense descriptor applied to wide baseline stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 5, pp. 815–830, 2010.
[40] V. Balntas, L. Tang, and K. Mikolajczyk, "Bold - binary online learned descriptor for efficient image matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 2367–2375.
[41] O. Miksik and K. Mikolajczyk, "Evaluation of local detectors and descriptors for fast feature matching," in *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 2012, pp. 2681–2684.
[42] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010, pp. 778–792.
[43] J. Baber, M. Dailey, S. Satoh, N. Afzulpurkar, and M. Bakhtyar, "BIG-OH: BInarization of gradient orientation histograms," *Image and Vision Computing*, vol. 32, no. 11, pp. 940–953, 2014.
[44] Y. Choi, C. Park, J. Lee, and I. Kweon, "Robust binary feature using the intensity order," in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2015, pp. 569–584.
[45] G. Hua, M. Brown, and S. Winder, "Discriminant embedding for local image descriptors," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, vol. 0, 2007, pp. 1–8.
[46] Z. Wang, B. Fan, and F. Wu, "Affine subspace representation for feature description," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.
[47] C. Strecha, A. Bronstein, M. Bronstein, and P. Fua, "LDAHash: Improved matching with smaller descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 66–78, 2012.
[48] T. Trzcinski, M. Christoudias, V. Lepetit, and P. Fua, "Learning image descriptors with the boosting-trick," in *Advances in neural information processing systems*, 2012, pp. 269–277.
[49] G. Levi and T. Hassner, "LATCH: Learned arrangements of three patch codes," *arXiv*, 2015.
[50] T. Yang, Y. Lin, and Y. Chuang, "Accumulated stability voting: A robust descriptor from descriptors of multiple scales," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 327–335.
[51] Y. Rubner, C. Tomasi, and L. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.
[52] H. Ling and K. Okada, "Diffusion distance for histogram comparison," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2006, pp. 246–253.
[53] C. Silpa-Anan and R. Hartley, "Optimised KD-trees for fast image descriptor matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.
[54] X. Xu, L. Tian, J. Feng, and J. Zhou, "OSRI: A rotationally invariant binary descriptor," *IEEE Transactions on Image Processing*, vol. 23, no. 7, pp. 2983–2995, 2014.
[55] G. Treen and A. Whitehead, "Efficient SIFT matching from keypoint descriptor properties," in *Proceedings of the Workshop on Applications of Computer Vision (WACV)*, 2009, pp. 1–7.
[56] C. Tsai, A. Tsao, and C. Wang, "Real-time feature descriptor matching via a multi-resolution exhaustive search method," *Journal of Software*, vol. 8, no. 9, pp. 2197–2201, 2013.
[57] F. Fraundorfer and H. Bischof, "A novel performance evaluation method of local detectors on non-planar scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2005, p. 33.
[58] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.
[59] F. Bellavia, D. Tegolo, and C. Valenti, "Improving Harris corner selection strategy," *IET Computer Vision*, vol. 5, no. 2, pp. 86–96, 2011.
[60] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," vol. 88, no. 2, 2010, pp. 303–338.
[61] C. L. Zitnick and K. Ramnath, "Edge foci interest points," in *IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 359–366.

**Fabio Bellavia** received his PhD degree in Computer Science from the University of Palermo, Italy, in 2011, where he also received his BSc and MSc degrees in Computer Science in 2004 and 2007, respectively. His research interests include computer vision and image processing, in particular feature detectors and descriptors, stereo matching, 3D reconstruction, mosaicing, color correction, and related evaluation methods. He is currently a Post Doctoral researcher at the Computational Vision Group of the University of Florence, Italy, where he has been working on national and European projects aimed at developing intelligent vision systems for autonomous underwater vehicles.

**Carlo Colombo** graduated cum laude in Electronic Engineering from the University of Florence, Italy, in 1992. In 1996 he obtained a PhD in Robotics from the Sant'Anna School of University Studies and Doctoral Research, Pisa, Italy. He currently is associate professor of Computational Vision at the Department of Information Engineering, University of Florence, where he leads the Computational Vision Group. His main research interests are in computer vision with applications to autonomous robotics, biomedicine and aids to disabled people, advanced human-machine interaction, multimedia systems and image forensics. He has published over 100 papers on international journals, books, and conference proceedings. He has been associate/area editor for the journals *Robotics and Autonomous Systems* (2001-2011), *Journal of Multimedia* (2006-2009), and *Computer Vision and Image Understanding* (2009-2012). He has also been General co-Chair of the 12th European Conference on Computer Vision ECCV 2012, and guest co-Editor of the special issue of the *International Journal of Computer Vision* on Large-Scale Computer Vision (2014).