SIAM J. IMAGING SCIENCES
Vol. 8, No. 4, pp. 2558–2580

Is Repeatability an Unbiased Criterion for Ranking Feature Detectors?*

Ives Rey-Otero† and Mauricio Delbracio‡

Abstract. Most computer vision applications rely on algorithms finding local correspondences between different images. These algorithms detect and compare stable local invariant descriptors centered at scale-invariant keypoints. Because of the importance of the problem, new keypoint detectors and descriptors are constantly being proposed, each one claiming to perform better than the preceding ones. This raises the question of a fair comparison between very diverse methods. This evaluation has been based mainly on a repeatability criterion of the keypoints under a series of image perturbations (blur, illumination, noise, rotations, homotheties, homographies, etc). In this paper, we argue that the classic repeatability criterion is biased favoring algorithms producing redundant overlapped detections. To overcome this bias, we propose a variant of the repeatability rate taking into account the descriptors overlap. We apply this variant to revisit the popular benchmark by [Mikolajczyk et al. *Int. J. Comput. Vis.*, 65 (2005), pp. 43–72], comparing several classic and recently introduced feature detectors. Experimental evidence shows that the hierarchy of these feature detectors is severely disrupted by the amended comparator.

Key words. feature detectors, performance evaluation, repeatability criteria, image matching, descriptors, scale invariant feature transform (SIFT)

AMS subject classifications. 68T10, 68T40, 68T45, 93C85

DOI. 10.1137/15M1007732

1. Introduction.

Local stable features are the cornerstone of many image processing and computer vision applications such as image registration [18, 38], camera calibration [15], image stitching [16], three-dimensional (3D) reconstruction [1], object recognition [14, 11, 2, 42], or visual tracking [31, 43]. The seminal SIFT method (scale invariant feature transform) introduced by Lowe in 1999 and 2004 [23, 24] sparked an explosion of local keypoints detectors/descriptors seeking discrimination and invariance to a specific group of image transformations [39].

Ideally, one would like to detect keypoints that are stable to image noise, illumination changes, and geometric transforms such as scale changes, affinities, homographies, perspective changes, or nonrigid deformations. Complementarily, the detected features should provide information as diverse as possible. Detections should, for example, be well distributed throughout the entire image, extracting information from all image regions and from boundary features of all kinds (e.g., textures, corners, blobs). Hence, there is a variety of detectors/descriptors

*Received by the editors February 9, 2015; accepted for publication (in revised form) July 22, 2015; published electronically November 10, 2015. A preliminary short version of this work was submitted to the 2015 IEEE International Conference on Image Processing. The research of the authors was supported by the Centre National d'Etudes Spatiales (CNES, MISS Project), European Research Council (advanced grant Twelve Labours), Office of Naval research (ONR grant N00014-14-1-0023), DGA Stéréo project, ANR-DGA (project ANR-12-ASTR-0035), FUI (project Plein Phare), and Institut Universitaire de France.
http://www.siam.org/journals/siims/8-4/M100773.html

†CMLA, ENS-Cachan, 94230 Cedex Cachan, France (ives.rey-otero@cmla.ens-cachan.fr).
‡ECE, Duke University, Durham, NC 27708-0271 (mauricio.delbracio@duke.edu).

built on different principles and having different requirements. While the SIFT method and its similar competitors [3, 30, 25] detect blob-like structure in a multiscale image decomposition, other approaches [5, 30, 35, 13, 36, 20] explicitly detect corners or junctions at different scales. As opposed to interest point detectors, interest region detectors [40, 41, 19, 6] extract the invariant salient regions of an image based on its topographic map. To fairly compare the very different feature detectors it is fundamental to have a rigorous evaluation protocol.

The repeatability rate measures the detector's ability to identify the same features (i.e., *repeated* detections) despite variations in the viewing conditions. Defined as the ratio between the number of keypoints simultaneously present in all the images of the series (repeated keypoints) over the total number of detections, it can be seen as a measure of the detector's efficiency. Indeed, the repeatability rate incorporates two struggling quality criteria: the number of repeated detections (i.e., potential correspondences) should be maximized while the total number of detections should be minimized since the complexity of the matching grows with the square of the number of detections.

Interest point detectors can also be indirectly evaluated through a particular application. In [29], the authors propose to evaluate detector-descriptor combinations in an image matching/recognition scenario. Although this approach can lead to very practical observations, the conclusions about the keypoints stability is intertwined with the descriptor's discrimination ability.

In this work, we show that the repeatability criterion suffers from a systematic bias: it favors redundant and overlapped detections. This has serious consequences, as evenly distributed and independent detections are crucial in image matching applications. The concentration of many keypoints in a few image regions is generally not helpful, no matter how robust and repeatable they may be. To better measure the detectors redundancy, we introduce a modified repeatability criterion. We consider the area actually covered by the descriptor and we evaluate the *descriptor overlap* as a measure of redundancy.

A preliminary short version of this work was submitted to a conference [33]. The present version incorporates the entire experimental analysis. The remainder of the article is organized as follows. Section 2 describes the repeatability criterion, discusses its variants, and illustrates how algorithms producing redundant detections may have a good performance according to this traditional quality measure. In section 3 we introduce a correction to the repeatability criterion that overcomes this bias, by accounting for the descriptor overlap. Section 4 reviews twelve state-of-the-art feature detectors, and details the region involved in the feature extraction for each of the analyzed methods. This extracted region will be the key ingredient for the proposed overlap measure. Comparative performance tables and maps gathered in section 5 show that the hierarchy of detectors is drastically altered by the new repeatability criterion. This result is further confirmed by analyzing the detection/matching performance using the same normalized descriptor for all the detectors. Conclusions are finally summarized in section 6.

2. The repeatability criterion and its bias.

2.1. Definition of the repeatability criterion. Consider a pair of images $u_a(\mathbf{x})$, $u_b(\mathbf{x})$ defined for $\mathbf{x} \in \Omega \subset \mathbb{R}^2$ and related by a planar homography H, that is, $u_b = u_a \circ H$. The detector repeatability rate for the pair (u_a, u_b) is defined as the ratio between the number

of detections simultaneously present in both images, i.e., repeated detections, and the total number of detections in the region covered by both images.

In the repeatability framework, a detection generally consists of an elliptical region, denoted $R(\mathbf{x}, \Sigma)$, parametrized by its center $\mathbf{x}$ and a 2×2 positive-definite matrix Σ,

$$R(\mathbf{x}, \Sigma) = \left\{ \mathbf{x}' \in \Omega \mid (\mathbf{x}' - \mathbf{x})^T \Sigma^{-1} (\mathbf{x}' - \mathbf{x}) \leq 1 \right\}.$$

A pair of detections (elliptical regions $R(\mathbf{x}_a, \Sigma_a)$ and $R(\mathbf{x}_b, \Sigma_b)$) from images $u_a(\mathbf{x})$ and $u_b(\mathbf{x})$ will be considered repeated if

$$e_{\text{overlap}} = 1 - \frac{|R(\mathbf{x}_a, \Sigma_a) \cap R(\mathbf{x}_{ba}, \Sigma_{ba})|}{|R(\mathbf{x}_a, \Sigma_a) \cup R(\mathbf{x}_{ba}, \Sigma_{ba})|} \leq \epsilon_{\text{overlap}}^{\max}, \tag{2.1}$$

where e_{overlap} is the overlap error, $\mathbf{x}_{ba} = H\mathbf{x}_b$, $\Sigma_{ba} = A\Sigma_b A^T$ represents the reprojection of the ellipse $R(\mathbf{x}_b, \Sigma_b)$ from image u_b into the image u_a, and A is the local affine approximation of the homography H.

The union and intersection of the detected regions are examined on the reference image $u_a(\mathbf{x})$ by projecting the detection on the image u_b into the image u_a. The union covers an area denoted by $|R(\mathbf{x}_a, \Sigma_a) \cup R(\mathbf{x}_{ba}, \Sigma_{ba})|$ while $|R(\mathbf{x}_a, \Sigma_a) \cap R(\mathbf{x}_{ba}, \Sigma_{ba})|$ denotes the area of their intersection. The parameter $\epsilon_{\text{overlap}}^{\max}$ is the maximum overlap error tolerated. In most published benchmarks it is set to $\epsilon_{\text{overlap}}^{\max} = 0.40$ [28, 30, 25].

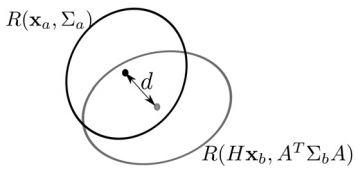


Figure 1. *Illustration of the repeatability criterion. Detection $R(\mathbf{x}_b, \Sigma_b)$ on image u_b is reprojected on the reference image u_a. If the overlap error is lower than $\epsilon_{overlap}^{max}$ (see (2.1)), the detections are considered repeated.*

Let Ω be the region covered by both images u_a and u_b. Since the number of repeated detections is upper bounded by the minimal number of detections in Ω (under the assumption that there are no multiple matches), the repeatability rate is defined as

$$\text{rep} = \frac{\text{number of repeated detections}}{\min\left(|\mathcal{K}_a|_\Omega, |\mathcal{K}_b|_\Omega\right)}, \tag{2.2}$$

where $|\mathcal{K}_a|_\Omega$ and $|\mathcal{K}_b|_\Omega$ denote the respective numbers of detections inside Ω.

2.2. Illustration and alternative definitions. To discuss and illustrate the repeatability criterion, let us consider the particular case of a pair of detections $R(\mathbf{x}_a, \Sigma_a)$ and $R(\mathbf{x}_b, \Sigma_b)$ whose reprojections on the reference image are two disks, both of radius r and with centers separated by a distance d (Figure 1). Such a pair will be considered repeated if $d/r \leq f(\epsilon_{\text{overlap}}^{\max})$, where f is a monotone function easily derived from (2.1). Figure 2(a) shows the maximum distance d under which both detections will be considered repeated as a function of the radius r.

As pointed out in [30], detectors providing larger regions have a better chance of yielding good overlap scores, boosting as a result their repeatability scores. This also means that one can artificially increase the repeatability score of any detector by increasing the scale associated with its detections. The authors of [30] proposed to avoid this objection by normalizing the detected region size before computing the overlapped error. The two detected elliptical regions $R(\mathbf{x}_a, \Sigma_a)$ and $R(\mathbf{x}_b, \Sigma_b)$ in (2.1) are replaced, respectively, by the elliptical regions $R(\mathbf{x}_a, \kappa^2/r_a R_a \Sigma_a)$ and $R(\mathbf{x}_b, \kappa^2/r_b R_b \Sigma_b)$, where r_a and R_a (respectively, r_b and R_b) are the radii of the elliptical region $R(\mathbf{x}_a, \Sigma_a)$ (respectively, $R(\mathbf{x}_b, \Sigma_b)$) and $\kappa = 30$ is the geometric mean of its radii after normalization.

The idea of such a normalization was to prevent boosting a detector's performance by enlarging its associated ellipse. Yet, such a criterion is not scale invariant, meaning that it may be overpermissive or underpermissive depending on the detection size. For example, the maximal distance separating repeated detections of equal size does not take into account the scale (e.g., the radius of the circle in our special case illustration; see Figure 2(b). In consequence, with $\epsilon^{\max}_{\text{overlap}}$ set to its standard value ($\epsilon^{\max}_{\text{overlap}} = 40\%$), two circular detections of radius 1 pixel and centers separated by 12 pixels can still be regarded as repeated, although their respective descriptors may not even overlap!

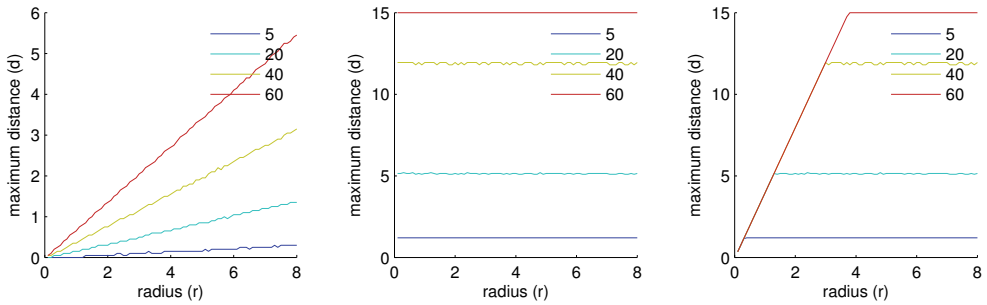


Figure 2. *Illustration of three different definitions of the repeatability criteria. Consider a pair of detections whose reprojections on the reference image are two disks of radius r with their centers separated a distance d. The maximal tolerated separation distance d_{max} between repeated detections is plotted as a function of the radius r for four values of the parameter $\epsilon^{max}_{overlap}$ (5%, 20%, 40%, and 60%).* (a) *original definition given by* (2.1), (b) *with ellipsis normalization $\kappa = 30$,* (c) *definition implemented in the code provided by the authors of* [30]. *Only the first definition is scale invariant.*

Surprisingly, the code[1] provided by the authors of [30] does not implement any of the criteria defined in their article. The code introduces a third definition by incorporating an additional criterion on the maximum distance separating two repeated keypoints that depends on the scale by

$$|\mathbf{x}_a - H\mathbf{x}_b| \leq 4\sqrt{r_a R_a}.$$

This criterion is illustrated in Figure 2(c) for the same study case of two circular detections of equal size. This third criterion is not scale invariant either. Thus in this paper we shall stick to the first definition, which is scale invariant. With the nonredundant repeatability criterion

[1] http://www.robots.ox.ac.uk./~vgg/research/affine/(retrieved on August 5th, 2014).

to be introduced in the next section, it will become pointless to try "boosting" a detector's scale. Indeed such attempts will result in decreased matching performance. The detection's characterizing scale will be the spatial extent of the descriptor ultimately computed, which is the real practical scale associated with each detector.

2.3. Repeatability favors redundant detectors. The following mental experiment sheds light on how the repeatability favors redundancy. Let DET be a generic keypoint detector, and let DET2 be a variant in which each detection is simply counted twice. The number of repeatable keypoints and the total number of detections are both artificially doubled, leaving the repeatability rate unchanged. However, although the number of costly descriptor computations has doubled, no extra benefit can be extracted from the enlarged set of repeated keypoints. The classic repeatability rate fails to report that the benefit over cost ratio of DET2 is half the one of DET. This explains why methods producing correlated detections may misleadingly get better repeatability ratios.

A popular attempt for mitigating this drawback is to compare detectors at a fixed number of detections [21, 25]. This would not, however, solve the problem for two reasons. First, by a similar reasoning as before, one can imagine a detector that repeats its best detection N times (N being the "fixed" number of detections) while discarding the rest. Such a detector would achieve optimal repeatability, despite being useless. But most importantly, given a detector, selecting the N best detections via a parameter (e.g., a threshold) is not generally an easy task. For example, in SIFT, the most popular way of adjusting the number of detected keypoints is by thresholding the analysis operator (difference of Gaussians) to retain only the most salient features. However, it is well known that this does not necessarily lead to a good selection in terms of stability [34]. To improve the selection, Li et al. [21] proposed a supervised regression process to learn how to rank SIFT keypoints. Although this scheme produces good results it requires supervised learning.

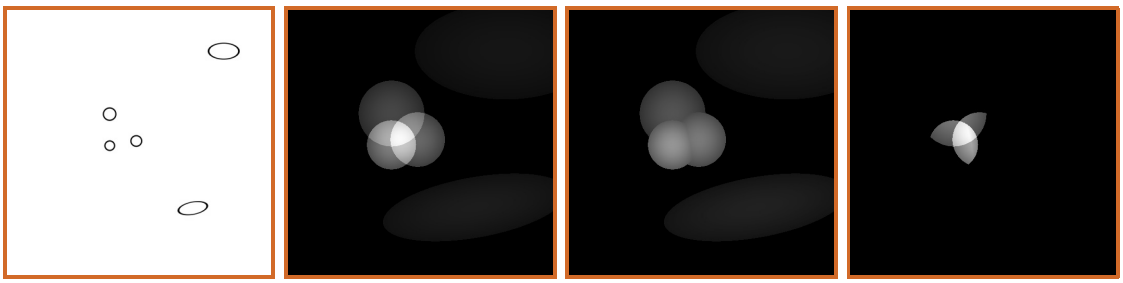
For these reasons, we believe that a fair comparison should prefer the genuinely independent detections. The metric introduced in the following section is a first attempt in this direction.

3. Nonredundant repeatability. Besides the repeatability measure, which ignores the keypoints spatial distribution, other specific metrics have been proposed. Some examine the spatial distribution of the descriptors and others evaluate how well they describe the image. The ratio between the convex hull of the detected features and the total image surface is used in [7] as a coverage measure. The harmonic mean of the detections positions is used in [10, 9] as a measure of concentration. In [8], the authors propose to measure the completeness of the detected features, namely, the ability to preserve the information contained in an image by the detected features. The *information content* metric proposed in [37] quantifies the distinctiveness of a detected feature with respect to the whole set of detections. Nondistinctive features are indeed harmful, as they can match to many others and therefore confuse the matching. Being complementary to it, these metrics are generally used in combination with the repeatability rate. Nevertheless, since the purpose of the repeatability is to report on the benefit/cost ratio of a given detector, it should also, by itself, report on the description redundancy. We shall see that the descriptors redundancy can be naturally incorporated into the repeatability criterion.

3.1. Nonredundant detections. To evaluate the redundancy of a set of detections $k \in \mathcal{K}$, each detection $(\mathbf{x}_k, \Sigma_k)$ can be assigned, in accordance with the descriptor associated canonically with the keypoint for each method, a mask function $f_k(\mathbf{x})$ consisting of a truncated elliptical Gaussian

$$f_k(\mathbf{x}) = Ke^{-\frac{1}{2\zeta^2}(\mathbf{x}-\mathbf{x}_k)^T\Sigma_k^{-1}(\mathbf{x}-\mathbf{x}_k)}$$

if $(\mathbf{x}-\mathbf{x}_k)^T\Sigma_k^{-1}(\mathbf{x}-\mathbf{x}_k) \leq \rho^2$ and 0 elsewhere. Each mask is normalized so that its integral over the image domain is equal to 1. The values ρ and ζ control the extent of the detected feature, as it can be derived from the descriptor's design. They will be fixed here for each detector by referring to the original paper where it was introduced (section 4). Indeed most detectors proposals come up with a descriptor or at least with a characterization of the region where this descriptor should be computed.



(a) (b) (c) (d)

Figure 3. *The mask functions formalizing the keypoint description on a toy example consisting of several Gaussian blobs* (a). *The sum over all detections* $\sum_{k\in\mathcal{K}} f_k(\mathbf{x})$ *maps the contribution of each image pixel to different descriptors* (b). *The max over all detections masks* $\max_{k\in\mathcal{K}} f_k(\mathbf{x})$ *maps the pixel contributions to the best available descriptor* (c). *Their difference maps the detection redundancy* (d).

The sum of all descriptor masks $\sum_{k\in\mathcal{K}} f_k(\mathbf{x})$ yields a final map showing how much each image pixel contributes to the set of all computed descriptors. Note that one pixel may contribute to several descriptors (as in the example shown in Fig 3). Similarly, the maximum taken over all detections $\max_{k\in\mathcal{K}} f_k(\mathbf{x})$ measures the contribution of pixel $\mathbf{x}$ to the best descriptor. Thanks to the mask normalization, the number of keypoints $K := card\,(\mathcal{K})$ is given by

$$K = \int_\Omega \left(\sum_{k\in\mathcal{K}} f_k(\mathbf{x})\right) d\mathbf{x}, \tag{3.1}$$

where Ω denotes the image domain. On the other hand,

$$K_{\text{nr}} := \int_\Omega \left(\max_{k\in\mathcal{K}} f_k(\mathbf{x})\right) d\mathbf{x} \tag{3.2}$$

measures the number of *nonredundant* keypoints. This value can be interpreted as a count of the independent detections.

To gain some intuition and see why this measurement is quite natural, let us examine four illustrative cases. Assume that there are only two detected keypoints so that $K = 2$. If the two detections

a. completely overlap, then $K_{\text{nr}} = 1$;
b. share the same center but have different sizes, then $1 < K_{\text{nr}} < K = 2$. But if their sizes are significantly different, then $K_{\text{nr}} \approx 2$, which makes sense. Indeed, one of them describes a fine detail and the other one a detail at a larger scale. Their information contents are roughly independent;
c. have both keypoints very close to each other, then, again, $1 < K_{\text{nr}} < K = 2$ and the above remark on scales still applies;
d. have descriptors that do not overlap at all, then $K_{\text{nr}} = K = 2$.

The propensity of a given algorithm to extract overlapped and redundant detections can therefore be measured by computing the *nonredundant detection ratio*

$$\text{nr-ratio} := K_{\text{nr}}/K. \tag{3.3}$$

3.2. Nonredundant repeatability. The above definitions entail a straightforward modification of the repeatability criterion (2.2). Let $\mathcal{K}_r$ be the set of repeatable keypoints (satisfying (2.1)) between two snapshots, and Ω the area simultaneously covered by both images. We define the *nonredundant repeatability rate* by

$$\text{nr-rep} := \frac{\int_\Omega \max_{k\in\mathcal{K}_r} f_k(\mathbf{x})d\mathbf{x}}{\min\left(|\mathcal{K}_a|_\Omega, |\mathcal{K}_b|_\Omega\right)}, \tag{3.4}$$

where $|\mathcal{K}_a|_\Omega$ and $|\mathcal{K}_b|_\Omega$ denote the respective numbers of detections inside Ω. The number of repeated detections in (2.2) is replaced in (3.4) by the number of nonredundant detections.

4. Spatial coverage of state-of-the-art feature detectors. In this section we review the twelve state-of-the-art feature detectors that will be compared using the nonredundant repeatability criteria. Our goal is to specify the region of the descriptor associated with each detector. A classical objection is that the descriptors associated with a detector may influence its matching performance. Hence the detector performance should be evaluated independently of its associated descriptor, and conversely. Fortunately, most papers introducing a detector also specify the area of interest around each detector as a circular or elliptical region. This is the region on which the final descriptor will be computed, regardless of its description technique. This information about the descriptor's region can be taken from the original papers. It is independent of the ultimate choice of a description technique, which may indeed vary strongly. In our discussion of each detector, we shall nevertheless also associate a fixed type of descriptor to each method, so as to be able to compare matching performance on an equal footing. This comparison is performed at the end of section 5.

Some of the detectors considered here were also compared in the original benchmark by Mikolajczyk et al. [30], namely, the Harris–Laplace and Hessian–Laplace [30], Harris-Affine and Hessian-Affine [30], EBR [40], IBR [41], and MSER [27]. We also included here, for completeness, methods published since: SIFT [23, 24], SURF [3], SFOP [13], BRISK [20], and SIFER [25]. Table 1 summarizes the algorithms' invariance properties. For details, we refer the reader to the original methods' publications and to the survey by Tuytelaars and Mikolajczyk [39].

Furthermore, we shall show detection maps on pattern images as well as on natural photographs to illustrate the behavior of each algorithm.

Most keypoint detection methods share the use of the *Gaussian scale-space* $u(\mathbf{x}, \sigma)$ defined by

$$u(\mathbf{x}, \sigma) := (G_\sigma * u)(\mathbf{x}), \text{ with } G_\sigma(\mathbf{x}) = \frac{1}{2\pi\sigma} e^{-\frac{\|\mathbf{x}\|^2}{2\sigma^2}},$$

where σ and $\mathbf{x}$ are, respectively, called the scale and space variables.

Table 1

Summary of algorithms' invariance properties. A zoom is the combination of a homothety and a Gaussian smoothing modeling the camera's point spread function. The considered detectors detect elliptical regions $(\mathbf{x}, \Sigma)$*, circular regions* $(\mathbf{x}, \sigma)$*, regions, or parallelograms.*

	detects	feature	rotation	zoom	homothety	affine
SIFT	$(\mathbf{x}, \sigma)$	blob	yes	yes	no	no
EBR	parallelograms	corners	yes	no	yes	limited
IBR	$(\mathbf{x}, \Sigma)$	blob	yes	no	yes	yes
Hessian-Laplace	$(\mathbf{x}, \sigma)$	blob	yes	yes	no	no
Hessian-Affine	$(\mathbf{x}, \Sigma)$	blob	yes	yes	no	limited
Harris-Laplace	$(\mathbf{x}, \sigma)$	corner	yes	yes	no	no
Harris-Affine	$(\mathbf{x}, \Sigma)$	corner	yes	yes	no	limited
MSER	regions	contrasted level lines	yes	no	no	yes
SURF	$(\mathbf{x}, \sigma)$	blob	limited	yes	no	no
SFOP	$(\mathbf{x}, \sigma)$	junction, circles	yes	no	yes	no
BRISK	$(\mathbf{x}, \sigma)$	corners	yes	yes	no	no
SIFER	$(\mathbf{x}, \sigma)$	blob	no	no	yes	limited

SIFT [23, 24] is probably the most popular local image comparison method. SIFT computes a multiscale image representation, detects keypoints from this scale-space, and extracts patch descriptors for each of the detections. For detecting keypoints, SIFT takes extrema of the convolution of the image with the normalized Laplacian of Gaussians (LoG). More precisely, SIFT approximates the LoG kernel by a difference of Gaussians (DoG),

$$w_{\text{SIFT}}(\sigma, \mathbf{x}) = \sigma^2 \Delta G_\sigma * u(\mathbf{x}) \approx (G_{k\sigma} - G_\sigma) * u(\mathbf{x}),$$

where $k = 2^{\frac{1}{3}}$ is a constant factor. The stable interpolated 3D extrema of the multiscale representation are the SIFT keypoints. The description of a keypoint consists of a feature vector assembled from the gradient distribution over an oriented patch surrounding the detected keypoint. For a detection at scale σ, the described patch covers a circular area of radius $\rho\sigma = 6\sqrt{2}\sigma$ weighted by a Gaussian mask of standard deviation[2] $\zeta\sigma = 6\sigma$. The described patch is oriented along a dominant orientation of the gradient distribution. SIFT considers multiple dominant orientations. This means that one keypoint may be described by various feature vectors, each corresponding to one of the dominant orientations (see [32] for a detailed description of the SIFT algorithm). We shall also consider a variant of SIFT that only takes one feature vector per detection, the one corresponding to the dominant orientation. We shall call it SIFT-single (SIFT-S).

[2]In the original SIFT algorithm the area covered by the descriptor is a square patch of size $12\sigma \times 12\sigma$. However, to make uniform all the algorithms since some of them do not give a reference keypoint orientation, we opted to replace the patch by the smallest disk containing it, which therefore covers a slightly larger area.

EBR (edge based region) [40] is an affine-invariant region detector. This method is not based on a scale-space image representation but on explicitly searching the image for structures of various sizes. Starting from a Harris corner point, EBR localizes the two nearby edges and analyzes their curvature to assign to each segment a characteristic direction and length. EBR returns the parallelogram bounded by the two edge segments. The parallelogram regions can be mapped into elliptical shapes having the same first and second moments. The EBR descriptor consists of a set of invariant moments computed over the elliptical region. For the sake of comparison, we will rely on the matching experiments on an affine normalized SIFT feature vector computed over the same elliptical region. In contrast with the SIFT method, the normalized patch is not weighted by a Gaussian mask.

IBR (intensity based region) [41] is an affine-invariant method which detects elliptical shapes of various sizes centered on specific gray level extrema. This method is not based on the Gaussian scale-space. By detecting abrupt changes in the intensity profiles along a set of rays originating from a gray value extremum, IBR extracts contrasted regions of various sizes and associates with them elliptical shapes. Similarly to EBR, invariant moments are computed over the detected region to build the feature vector. For the sake of homogeneity in our matching comparisons we shall instead use a SIFT descriptor computed on the affine normalized patch, without applying a Gaussian weighting mask.

Harris–Laplace and Hessian–Laplace detectors [30]. Unlike SIFT, these methods use two multiscale representations instead of one. The first one is used to determine the keypoint location and the second one is used to select its characteristic scale. In the case of the Hessian–Laplace method, the first multiscale representation is the two-dimensional (2D) Hessian determinant while the second one is the normalized Laplacian, both computed on the Gaussian pyramid [22]. The 2D Hessian determinant extremum gives the keypoint location $\mathbf{x}$. Then, the extremum of the scale-space Laplacian $\Delta u(\mathbf{x}, \sigma)$ with respect to σ gives the keypoint scale. The detector goes back and forth between both multiscale representations to iteratively refine $\mathbf{x}$ and σ. The Harris–Laplace method proceeds almost identically. Only the Harris operator [17] is used in place of the 2D Hessian to extract the keypoint location $\mathbf{x}$. The Harris–Laplace features are predominantly corners while the Hessian–Laplace mostly detects blobs. Unlike in the SIFT method, the extrema are not interpolated to subpixel precision. Once extracted, each keypoint is locally described, using the SIFT or the gradient location and orientation histogram (GLOH) descriptor [30, 29]. Consequently, for a detection at scale σ, the described patch covers a circular area of radius $\rho\sigma = 6\sqrt{2}\sigma$ weighted by a Gaussian mask of standard deviation $\zeta\sigma = 6\sigma$.

Harris-Affine and Hessian-Affine detectors [30] are affine extensions of the Harris–Laplace and Hessian–Laplace detectors. Instead of detecting keypoints, both methods detect elliptical regions. Compared to the Harris–Laplace and Hessian–Laplace methods, the affine variants contain an additional step in which the second-moment matrix is used to estimate an elliptical shape around each keypoint.[3] These elliptical shapes are used to normalize the local neighborhood by an affine transformation before its description (using the SIFT or

[3]The elliptical shape is estimated via an iterative procedure. Unreliable detections with degenerated second-moment matrices are also discarded in the process.

the GLOH descriptor). The SIFT descriptor is adopted in the present study. If σ denotes the geometric mean of the ellipse radii, then the described patch covers a circular area in the affine-normalized neighborhood of radius $\rho\sigma = 6\sqrt{2}\sigma$ weighted by a Gaussian mask of standard deviation $\zeta\sigma = 6\sigma$.

MSER (maximally stable extremal region) [27] is an affine-invariant method which extracts regions that are connected components of image upper level sets. By examining how the area of the image upper level sets evolves with respect to an image intensity threshold, MSER measures the region stability. The MSERs are the regions that achieve a local maximum of the (nonpositive) derivative of the region area with respect to its level. MSER proposes to compute feature descriptors at different scales of the detected region size (1.5, 2, and 3 times the convex hull of the detected region). In addition, MSER regions can be easily mapped into elliptical shapes and then used to compute an affine descriptor of the detected region. In the present framework, for each of the detected regions a SIFT feature vector on an affine normalized patch of twice the size of the detected region was computed.

SURF (speeded-up robust feature) [3] can be regarded as a fast alternative to SIFT. SURF keypoints are the 3D extrema of a multiscale image representation that approximates the 2D Hessian determinant computed on each scale of the Gaussian scale-space. The Gaussian convolution is approximated using box filters computed via integral images. SURF descriptors are computed over a Gaussian window centered at the keypoint, and encode the gradient distribution around the keypoint using 2D Haar wavelets. The described patch for a detection at scale σ covers a circular area of radius $\rho\sigma = 10\sqrt{2}\sigma$ weighted by a Gaussian mask of standard deviation $\zeta\sigma = 3.3\sigma$. Note that the described areas used in SIFT and SURF are slightly different. A SURF descriptor patch is larger but uses a more concentrated Gaussian mask.

SFOP (scale-invariant feature operator) [13] is a versatile multiscale keypoint detector that explicitly models and detects corners, junctions, and circular features. SFOP is built on the Förstner feature operator [12] for detecting junctions and on the spiral model [4] for unifying different feature types into a common mathematical formulation. For detecting keypoints at different scales, the input image is decomposed into a series of images using a Gaussian pyramid. Each image is then scanned for various feature types, namely, circular structures of various sizes and junctions of different orientations. At each pixel, the algorithm takes a surrounding patch and evaluates its consistency to the feature model. Although SFOP only concerns keypoint detection, the authors recommend combining the SFOP detector with SIFT's descriptor. Consequently, the described patch for a detection at scale σ also covers a circular area of radius $\rho\sigma = 6\sqrt{2}\sigma$ weighted by a Gaussian mask of standard deviation $\zeta\sigma = 6\sigma$.

BRISK (binary robust invariant scalable keypoint) [20] focuses on speed and efficiency. The BRISK detector is a multiscale adaptation of FAST and its optimized version AGAST [35, 26] corner detectors. The AGAST corner detector is first applied separately to each scale of a Gaussian pyramid decomposition to rapidly identify potential regions of interests. For each pixel in such regions, a corner score quantifying the detection confidence is computed (see [26] for details). Based on the AGAST corner score, BRISK performs a

3D nonmaxima suppression and a series of quadratic interpolations to extract the BRISK keypoints $(\mathbf{x}, s)$, $(\mathbf{x})$ being the 2D position and s the feature size. The BRISK descriptor is a binary string resulting from brightness differences computed around the keypoint.

In the current analysis, we calibrated the size of the detections s provided by the BRISK binary to make it comparable to the other methods. We empirically found that the image of a 2D Gaussian function of standard deviation σ produces a SIFT detection of scale σ while it produces a BRISK feature of size $s = 4\sigma$. In consequence, for a BRISK detection of size s, the described patch in the present study covers a circular area of radius $\rho s = \frac{3}{2}\sqrt{2}s$ weighted by a Gaussian mask of standard deviation $\zeta s = \frac{3}{2}s$.

SIFER (scale-invariant feature detector with error resilience) [25]. The recently introduced SIFER algorithm tightly follows SIFT, but computes a different multiscale image representation. Instead of smoothing the image with a set of Gaussian filters and computing its Laplacian, SIFER convolves the image with a bank of cosine modulated Gaussian kernels (see Figure 4).

$$\mathrm{cmg}_\sigma(x, y) = \left(2\pi\sigma^2 \left(\cos\left(\frac{cx}{\sigma}\right) + \cos\left(\frac{cy}{\sigma}\right)\right) G_\sigma\right). \tag{4.1}$$

The 3D extrema of the resulting multiscale representation are the SIFER keypoints. The method is homothety invariant. Unlike SIFT, however, SIFER is not zoom-out invariant. Indeed, its kernel does not commute with a Gaussian camera blur. The authors claim that, despite losing rotation invariance, the approach increases the detection precision in both scale and space thanks to the better localization of the modulated cosine filters. The descriptor computed at each extracted keypoint is identical to the SIFT descriptor. Therefore, the described patch considered in the present study covers a circular area of radius $\rho\sigma = 6\sqrt{2}\sigma$ weighted by a Gaussian mask of standard deviation $\zeta\sigma = 6\sigma$.

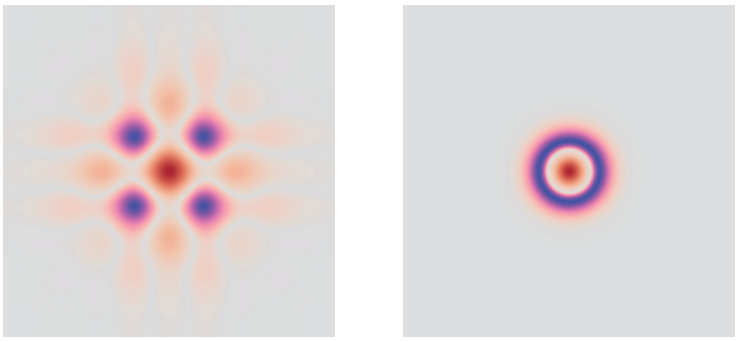


Figure 4. *SIFER (left) and SIFT (right) filter kernels. The SIFER kernel, a Gaussian modulated along the two axes by cosine functions is not rotation invariant, while the difference of Gaussians used in SIFT is.*

Table 2 summarizes the values of ρ and ζ for each method.

4.1. Detection maps. Different detectors extract different kinds of features, in different amounts, and with different spatial distributions. To visually inspect the algorithms general

Table 2

Summary of the parameters controlling the spatial coverage of a detection for each evaluated method. The parameter ρ controls the size of the patch encoded in the descriptor. For methods that apply a Gaussian weighting window to the described patch, the parameter ζ controls the standard deviation of the Gaussian function.

	ρ	ζ
SIFT	$6\sqrt{2}$	6
Hessian-Laplace	$6\sqrt{2}$	6
Hessian-Affine	$6\sqrt{2}$	6
Harris-Laplace	$6\sqrt{2}$	6
Harris-Affine	$6\sqrt{2}$	6
MSER	2	-

	ρ	ζ
EBR	1	-
IBR	1	-
SURF	$10\sqrt{2}$	3.3
SFOP	$6\sqrt{2}$	6
BRISK	$3/2\sqrt{2}$	$3/2$
SIFER	$6\sqrt{2}$	6

behavior, Figures 5 and 6 show the detection maps for the twelve compared methods on the `siemens` pattern and on the `bike` image from the Oxford dataset [30].

The detection number varies from one method to the other, and also from one sequence to the next. MSER generally detects fewer features than the rest while SIFT and the Harris and Hessian based methods detect many more.

The rotation invariance of the methods is easily tested by examining the detections on the `siemens star` test image shown in Figure 5. Unsurprisingly, SIFT and SFOP are rotation invariant while SIFER is not. More surprisingly, the Hessian and Harris based methods are not rotation invariant. Although the Hessian determinant and the Laplacian of the Gaussian smoothing are isotropic, the methods fail to maintain the theoretical invariance properties due to the discretization of the differential operators.

Several feature detectors generate multiple detections from a single local feature. This is clearly the case for Harris-Affine, Hessian-Affine, and, to a lesser extent, for BRISK. In general, with the exception of SIFT, SFOP, and MSER, all the detectors appear to be visually highly redundant.

In some cases, while detections are numerous, they cluster on a reduced part of the scene. This is observed for instance with SIFER, (see, e.g., Figure 6). This seems to imply that the information contained in the descriptors computed from SIFER keypoints is both redundant and incomplete.

5. Experiments.

Using the proposed nonredundant repeatability criterion, we examined the performance of the described feature detectors on the Oxford dataset [30].[4] The Oxford dataset contains eight sequences of six images each designed to help assess the stability of the detections with respect to habitual image perturbations, namely, rotation and scale changes, viewpoint changes, camera blur, illumination changes, and JPEG compression artefacts. The eight sequences are shown in Figure 7. The original and publicly available binaries of all but one method were used.[5] No reference implementation of SIFER was available; we therefore relied on our own implementation rigorously following the published description [25]. The

[4]Dataset available at http://www.robots.ox.ac.uk/~vgg/research/affine/

[5]Methods binaries are found at http://www.robots.ox.ac.uk/~vgg/research/affine/, http://docs.opencv.org/doc/tutorials/features2d/feature_detection/feature_detection.html, http://www.vision.ee.ethz.ch/~surf/ and http://www.cs.ubc.ca/~lowe/keypoints/ http://www.ipb.uni-bonn.de/sfop/

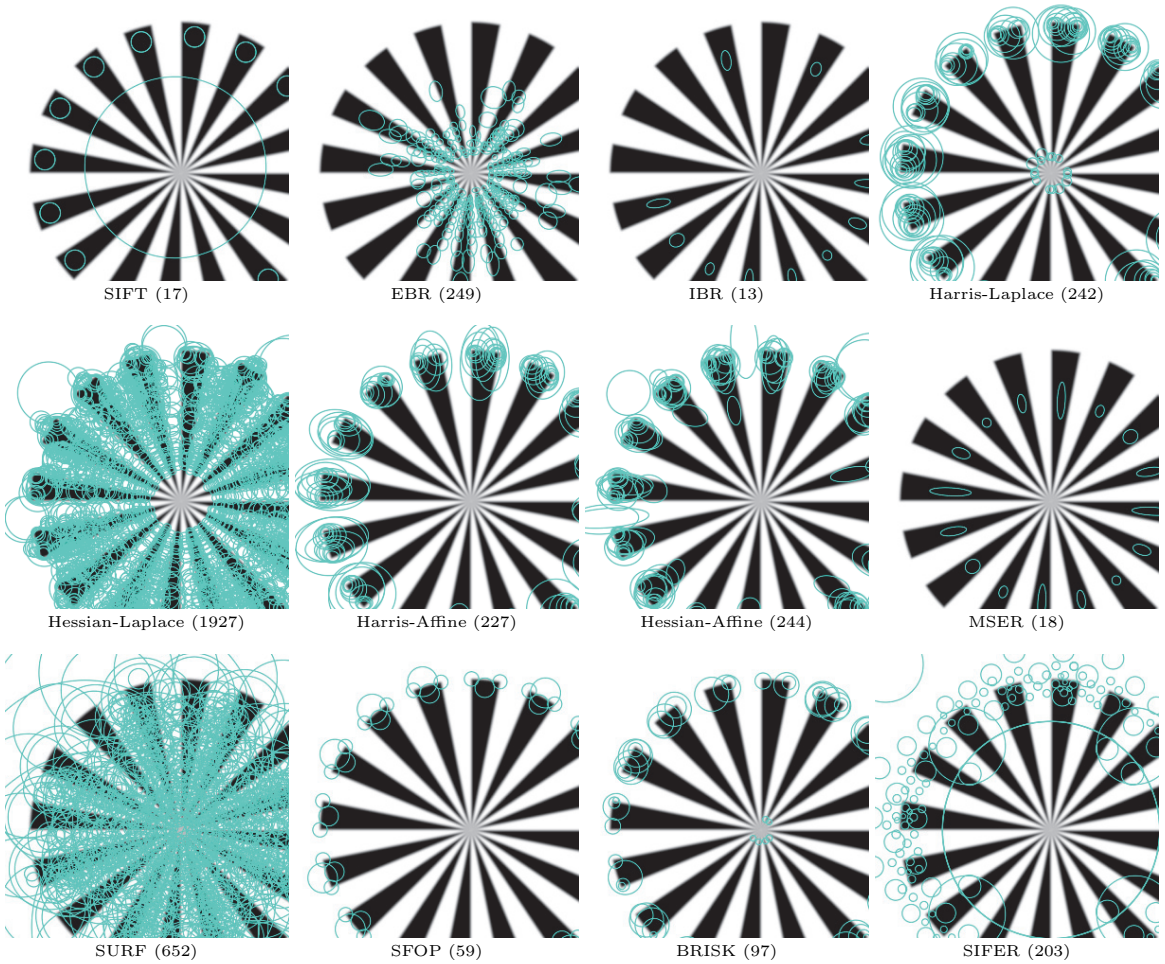


Figure 5. *Keypoints map on* `siemens star` *test image. For better readability of the figure, the descriptor ellipses are reduced to one sixth of their real size. Thus, when two ellipses overlap, their associated descriptors are in strong overlap. This is particularly conspicuous for the Hessian and Harris detectors. The total number of detected keypoints by each method is shown in brackets. SIFT and SFOP seem to be the only (experimentally) rotationally invariant methods. The elliptical shapes deduced from the MSER regions have different sizes in each rotated triangle. By design, SIFT detects blob-like structures and SFOP additional features, such as corners and edges.*

parameters of each method were set to their default values. All scripts and codes are available for download.[6]

The performance evaluation of a detector is two-dimensional. On the one hand, a detector should produce as many detections as possible, while on the other, it should keep to a minimum the number of nonrepeatable detections. In other words, the best detector is the one that has simultaneously the largest repeatability ratio and the largest number of detections.

[6]In particular a documented and optimized version of the repeatability criteria [30] along with the two variants discussed in section 2 are available for download at http://dev.ipol.im/~reyotero/comparing_20140906.tar.gz.

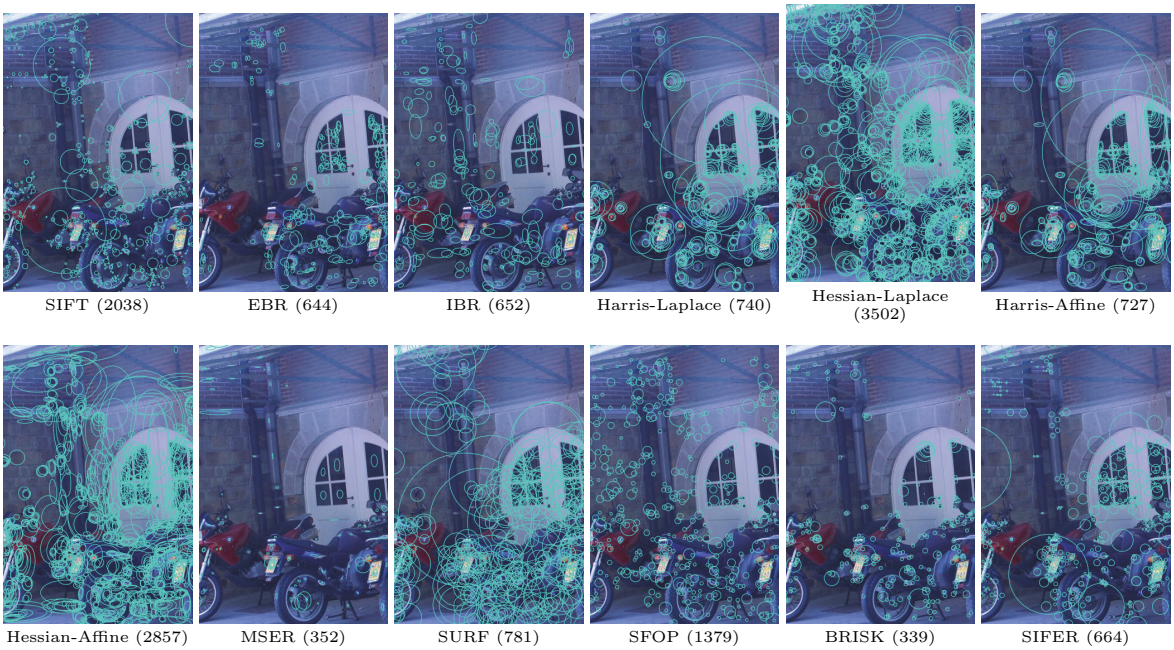


Figure 6. *Keypoints map on an image from the* `bikes` *sequence. For a better readability of the figure, we reduced six times the descriptors ellipses with respect to their real size. This also means that when two ellipses overlap, their associated descriptors are in strong overlap. The total number of detected keypoints by each method is shown in brackets. The number of detections significantly varies with the algorithm. Hessian based methods and SIFT produce many more detections than the rest. All methods, with the exception of IBR and EBR, detect features at very different scales. In particular, SIFT and SFOP detect very small structures. Most algorithms detect the same structure several times, producing significantly overlapped detections. The SIFER detections are disturbingly concentrated on clusters not necessarily overlapped. Yet the proposed nonredundant repeatability metric will not penalize such behavior. For the Harris and Hessian based methods, note how corners generate trails of detections of increasing size.*

As we showed in the previous section, a quick visual examination of the detection maps already reveals that some methods are more redundant than others. For example, it is clear from Figure 5 (`siemens star`) that SIFER, SURF, and the Hessian based methods produce highly redundant detections. The nonredundancy ratio shown in Table 3(a) for the eight Oxford sequences helps rank the methods in terms of redundancy. With nonredundant ratios lower than 7% on all eight sequences, the Hessian based detectors are the most redundant methods. On the other end of the spectrum, the least redundant method is MSER having an average nonredundant ratio of 51%. SIFT and its SIFT-S variant come second, with nonredundant ratios ranging from 20% to 36%. Since the number of detections of SIFT and Hessian-Laplace are comparable (Table 3(b)), the cost of extracting and matching descriptors is similar for both methods. Notwithstanding this fact, SIFT produces well-spread detections while the Hessian-Laplace are redundant and overlapped. Under such circumstances, we expect that taking into account the descriptors overlap will change significantly the hierarchy given by the repeatability rates.

The classic repeatability and the nonredundant repeatability rates as well as the number of detections for the eight Oxford sequences are provided in Table 3. Also, in Figure 8 the

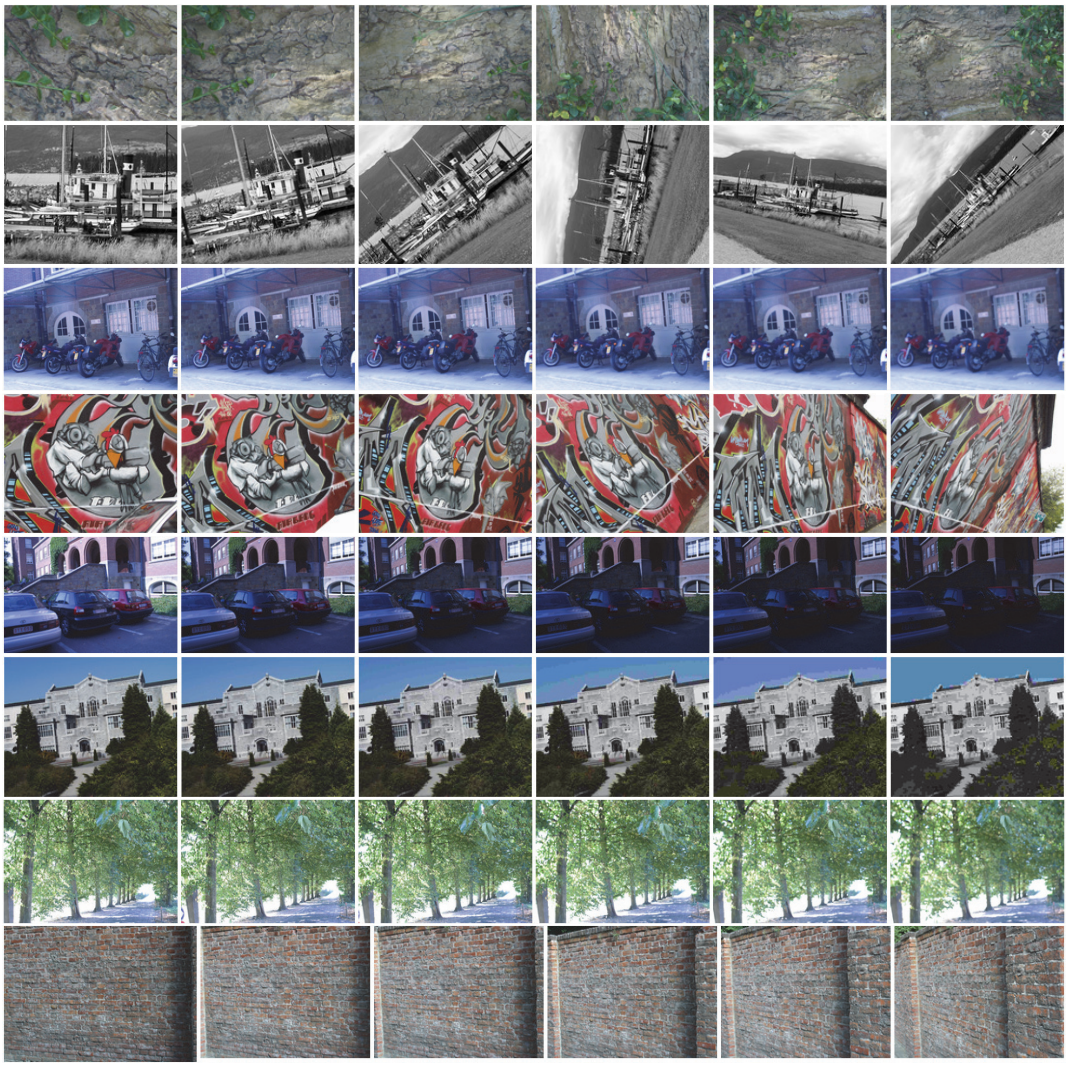


Figure 7. *The Oxford dataset. Different series of images simulating different image transformations. From top to bottom:* `bark` *and* `boat` *(scale changes and rotations),* `bikes` *(camera blur),* `graf` *(viewpoint changes),* `leuven` *(illumination changes),* `trees` *(camera blur),* `ubc` *(JPEG compression),* `wall` *(viewpoint changes) image series.*

average repeatability rates for all the compared detectors are plotted as a function of the number of detections. Note that in general, the number of repeated points oscillates around 40% of the total number of detections. This is a much lower rate than usually achieved with the more permissive definition of the repeatability criterion; see section 2.

As previously said, the repeatability score must be compared alongside the number of detections to have a complete performance evaluation of detectors. The methods that provide, in general, the largest number of detections are SIFT, SIFER, and the Hessian based methods. MSER, EBR, and IBR produce significantly fewer detections. The methods that are the most redundant also happen to be the methods that perform well according to the classic

Table 3

Detectors comparison regarding repeatability and nonredundant repeatability rates on the eight sequences of the Oxford dataset. The algorithm with the best number is colored in ***red*** *and the next three in* ***bordeaux****. Each table focuses on a single metric: the (nonredundant) repeatability of the number of detection in the common area. A fair comparison should consider both metrics simultaneously (see Figure 8).*

	bark scale	bikes blur	boat scale	graf viewp	leuven illum	trees blur	ubc jpeg	wall viewp	*mean*
SIFT	**0.26**	**0.27**	**0.26**	**0.29**	**0.35**	**0.20**	**0.24**	**0.23**	*0.24*
SIFT-S	**0.31**	**0.32**	**0.31**	**0.34**	**0.41**	**0.24**	**0.29**	**0.27**	*0.29*
EBR	0.23	0.18	0.10	0.11	0.15	**0.23**	0.1	0.08	*0.12*
IBR	0.20	0.24	0.21	0.21	**0.30**	0.20	**0.21**	**0.28**	*0.22*
HARLAP	0.11	0.10	0.06	0.06	0.12	0.04	0.08	0.07	*0.07*
HESLAP	0.04	0.04	0.03	0.03	0.05	0.03	0.04	0.03	*0.03*
HARAFF	0.12	0.10	0.07	0.07	0.13	0.05	0.08	0.08	*0.07*
HESAFF	0.04	0.05	0.05	0.04	0.07	0.03	0.04	0.04	*0.04*
MSER	**0.61**	**0.55**	**0.51**	**0.55**	**0.58**	**0.48**	**0.55**	**0.48**	*0.51*
SURF	0.16	0.16	0.11	0.11	0.16	0.10	0.13	0.13	*0.12*
SFOP	0.17	0.24	**0.21**	0.26	0.25	0.17	0.19	0.18	*0.20*
BRISK	0.26	**0.28**	0.14	**0.27**	0.26	0.10	0.17	0.13	*0.15*
SIFER	**0.31**	0.23	0.18	0.21	0.22	0.16	0.18	0.19	*0.19*

(a) Average *nonredundant ratio* $\mathrm{nr} := K_{\mathrm{nr}}/K$.

	bark scale	bikes blur	boat scale	graf viewp	leuven illum	trees blur	ubc jpeg	wall viewp	*mean*
SIFT	**1021.6**	**1034.8**	**3802.8**	**1906.6**	**1736.6**	**9143.0**	**5296.0**	**8677.6**	*4077.4*
SIFT-S	**848.0**	871.0	**3225.6**	**1641.8**	1473.6	7506.8	**4272.6**	**7255.2**	*3386.8*
EBR	75.2	366.4	665.2	577.4	458.0	535.2	756.0	2012.4	*680.7*
IBR	131.8	573.2	280.6	293.6	238.2	1141.0	563.2	453.4	*459.4*
HARLAP	118.0	541.0	1438.8	1120.8	568.4	4419.6	1549.0	1963.4	*1464.9*
HESLAP	**814.6**	**2936.4**	**2794.8**	**3164.8**	**2233.2**	**8201.6**	**3594.0**	**4913.8**	*3581.7*
HARAFF	120.2	533.2	1392.2	1103.0	555.8	4397.6	1501.0	1931.6	*1441.8*
HESAFF	**807.2**	**2470.0**	2217.2	**2180.2**	**1538.6**	**7875.8**	3146.0	4798.4	*3129.2*
MSER	85.4	195.2	592.4	280.4	276.4	1839.4	716.0	1372.8	*669.8*
SURF	183.0	546.6	948.2	913.4	607.8	3000.0	1194.0	1564.2	*1119.7*
SFOP	476.0	**1040.8**	825.8	530.2	1014.0	3293.0	1859.4	2243.2	*1410.3*
BRISK	119.2	194.2	1149.6	374.0	521.4	3016.6	1408.8	2413.2	*1149.6*
SIFER	159.4	729.8	**4321.4**	1570.6	**2591.4**	**8818.2**	**6609.8**	**8535.2**	*4167.0*

(b) Average number of detections.

	bark scale	bikes blur	boat scale	graf viewp	leuven illum	trees blur	ubc jpeg	wall viewp	*mean*
SIFT	23.4	44.3	17.6	11.8	42.5	6.7	29.1	8.0	*22.9*
SIFT-S	23.3	44.6	18.1	11.9	43.5	7.1	30.0	8.3	*23.4*
EBR	7.5	**66.6**	**53.5**	**38.6**	55.1	16.0	51.4	**38.8**	*40.9*
IBR	37.2	51.9	46.4	**50.6**	58.1	**33.4**	45.6	36.1	*44.9*
HARLAP	**52.5**	52.4	40.2	21.3	50.2	**23.2**	**73.6**	29.9	*42.9*
HESLAP	**57.9**	**69.5**	**50.0**	22.4	**70.1**	**33.1**	**73.8**	**36.4**	*51.7*
HARAFF	48.6	50.0	36.7	26.9	47.5	20.2	**71.8**	27.9	*41.2*
HESAFF	**54.7**	**66.8**	46.8	**30.7**	**65.9**	**28.4**	**72.7**	35.8	*50.2*
MSER	32.9	52.2	42.4	**55.6**	**72.8**	18.0	44.8	**40.4**	*44.9*
SURF	**63.6**	**72.6**	**48.2**	19.4	**64.6**	29.5	70.9	**36.7**	*50.7*
SFOP	29.7	31.8	25.9	13.7	42.6	8.4	36.2	18.8	*25.9*
BRISK	2.4	9.9	4.0	4.3	18.2	5.4	16.6	5.8	*8.3*
SIFER	1.4	49.9	7.4	1.5	37.5	9.1	50.9	10.0	*20.9*

(c) Average repeatability.

	bark scale	bikes blur	boat scale	graf viewp	leuven illum	trees blur	ubc jpeg	wall viewp	*mean*
SIFT	7.3	**15.2**	6.6	4.5	**19.2**	3.2	10.6	3.7	*8.8*
SIFT-S	8.8	**18.1**	**7.8**	5.3	**22.6**	3.9	**13.1**	4.4	*10.5*
EBR	5.3	**15.4**	6.6	**9.2**	10.3	**6.8**	7.4	5.4	*8.3*
IBR	**19.8**	15.2	**15.4**	**17.5**	**26.2**	**11.9**	**13.7**	**17.7**	*17.2*
HARLAP	**11.1**	9.1	4.1	2.6	10.1	3.1	6.8	4.9	*6.5*
HESLAP	3.8	3.7	2.4	1.2	4.6	2.1	3.5	2.6	*3.0*
HARAFF	11.1	9.4	4.0	4.2	10.4	3.0	7.2	5.3	*6.8*
HESAFF	4.1	4.6	2.8	2.8	6.4	2.2	4.1	3.0	*3.7*
MSER	**27.2**	**35.9**	**24.0**	**32.8**	**49.8**	**13.1**	**29.9**	**25.4**	*29.8*
SURF	**14.7**	13.3	7.1	3.7	14.1	**5.6**	10.2	**8.0**	*9.6*
SFOP	10.0	11.5	**10.3**	**6.2**	16.7	4.2	10.7	**6.1**	*9.5*
BRISK	2.3	7.2	2.7	3.4	11.8	3.5	7.7	3.9	*5.3*
SIFER	1.2	14.7	3.4	1.2	12.3	3.7	**11.2**	3.8	*6.4*

(d) Average nonredundant repeatability.

repeatability criteria (see Table 3(d)). Indeed, the Hessian based methods are among the methods with the largest repeatability while providing numerous detections. Note that SFOP is outperformed by the Harris based methods in all eight sequences, while providing a similar number of detections.

These conclusions are drastically altered when the redundancy of detections is taken into account. According to the nonredundant repeatability shown in Table 3(d), the hardly redundant SIFT method achieves one of the top three best scores while providing, in general, one of the largest number of detections. The Hessian based methods and SIFER, while achieving detection numbers comparable to those of SIFT, perform poorly according to the nonredundant repeatability. Despite having fewer detections, the nonredundant repeatability of SURF is lower than the one of SIFT in five sequences out of eight. Unlike what was concluded with the classic criterion, SFOP outperforms the Harris based methods in seven out of eight sequences. In fact, SFOP performs generally well. In all sequences, SFOP is one of the three best algorithms according to the nonredundant repeatability while it performed poorly for the traditional repeatability. On average, MSER and IBR produce the best nonredundant repeatability scores. Nevertheless, with up to ten times more detections, SIFT should be preferred to MSER except for severe changes of viewpoint (see Figure 8). In principle, MSER is not blur invariant. Yet, it performs surprisingly well on the sequence `bikes`, containing well-contrasted large geometric features. MSER may benefit here from its low number of detections.

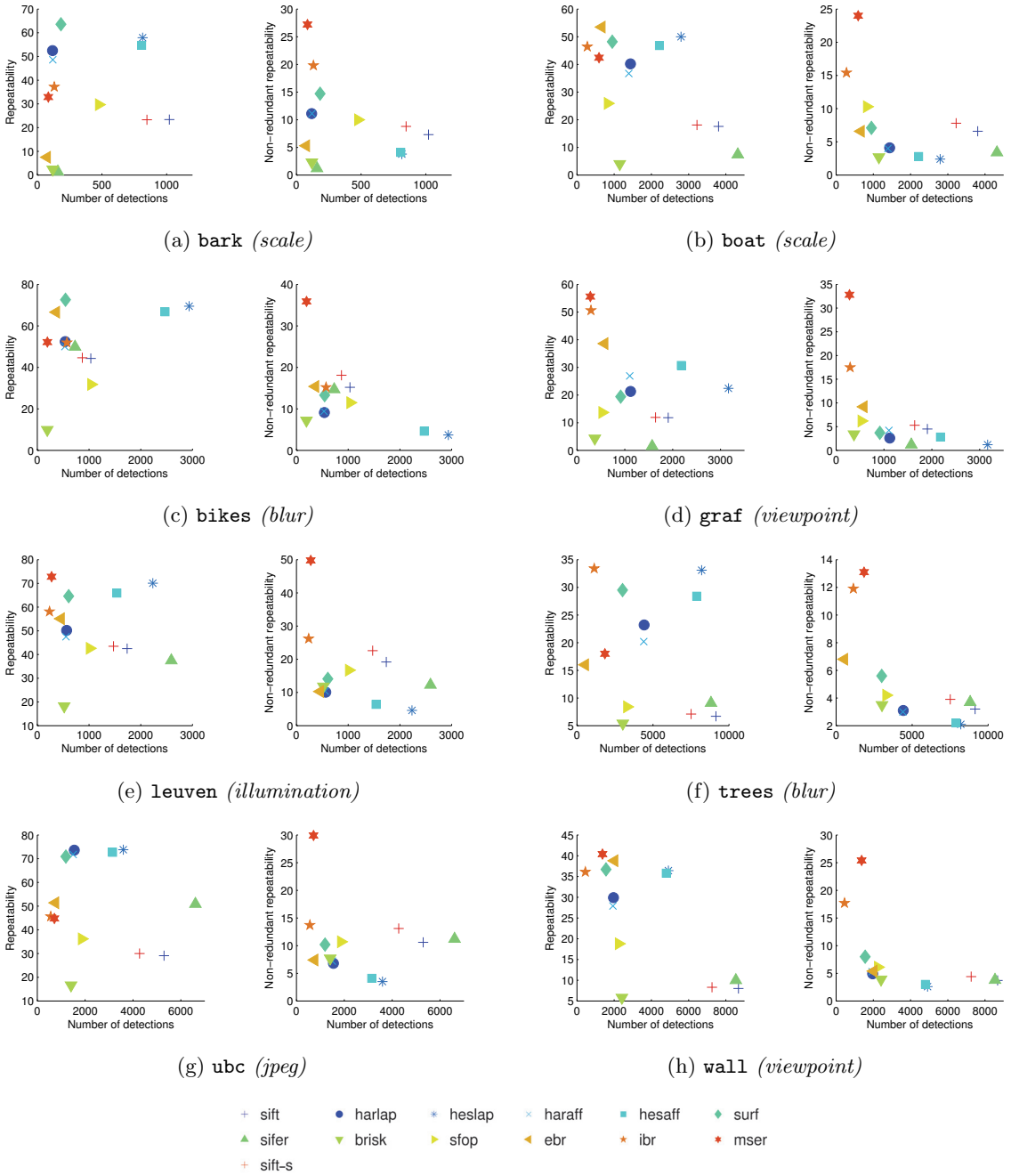


Figure 8. *The average of the repeatability and nonredundant repeatability on each Oxford sequence is plotted as a function of the average number of keypoints detected. The performance evaluation of a detector is two-dimensional. On the one hand, a detector should detect as many keypoints as possible (abscissa). On the other, the detections should be as repeatable as possible (ordinate). Good detectors are on the top-right region of this plot. To compare a single detector performance the reader might follow the relative ordinate position of a particular detector in a particular scene in the traditional repeatability (left) and the nonredundant repeatability plots (right). For instance, MSER and SIFT algorithms always go up from the traditional to the nonredundant repeatability plots. This means that MSER and SIFT detections are less redundant than the average.*

To summarize the relative performance of each method on the entire Oxford data set we proceeded as follows. First, the number of detections, the repeatability, and nonredundant repeatability rates on each sequence were rescaled to cover the interval $[0, 1]$. Then, we computed the mean of the rescaled detectors performance over the eight sequences. Figure 9 shows the relative repeatability and nonredundant repeatability scores as a function of the number of the normalized number detections. In this map a method performs optimally if it is simultaneously extremal in ordinate and in abscissa, and performs well if it is extremal in at least one of the coordinates. Thus, the normalized benchmark reveals that the ranking of detectors is severely disrupted when considering the detectors redundancy. While, for example, Harris and Hessian based methods, SURF and EBR, significantly reduce their performance (going down in the plot), MSER and BRISK improve their position relative to the others. When the redundancy is not taken into account the method producing the most detections and with the highest repeatability is Hessian Laplace, while when considering the nonredundant variant it is SIFT.

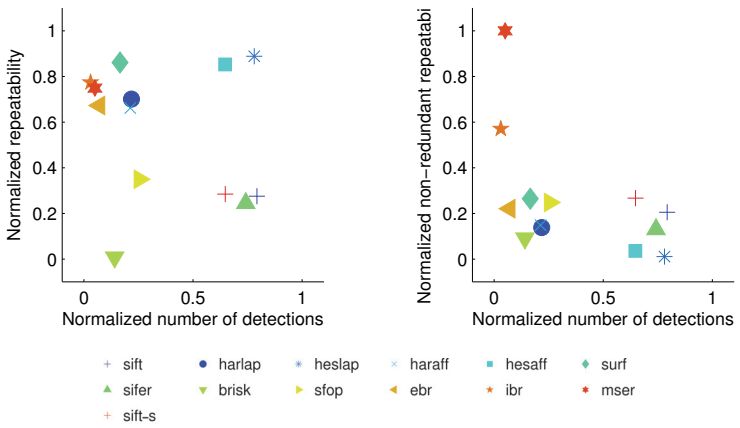


Figure 9. *Qualitative visualization of the methods repeatability performance. This is an average over eight sequences. For each sequence, the number of detections, the matching rates, and the nonredundant matching rates are scaled to the full range* $[0, 1]$ *and averaged into a single map. Once normalized, the mean values of each method over the eight sequences are computed. On the left, the normalized repeatability is plotted as a function of the normalized number of detections. On the right, the normalized nonredundant repeatability is plotted as a function of the normalized number of detections. The same conclusions observed in each of the eight Oxford sequences apply in this qualitative contest.*

Matching scenario. We also explored the algorithms performance on a matching scenario. For that purpose, we adopted the same protocol as in [30]. Each detector is combined with a SIFT descriptor. Around each detection, a patch is extracted to compute the dominant orientation and a SIFT feature vector. The width of the extracted patch is computed as the mean of the detected ellipse radii multiplied by the method's parameter ρ, as described for each method in section 4. For all the SIFT feature vectors in one image we found the most similar feature vector on the other image (in terms of the Euclidean distance). If the distance to the most similar one is less than 60% of the distance to the second nearest feature, then the pair of detections is considered as a match (as proposed in [24]) Table 4(a) gives the number of detections in the common area. Table 4(b) shows the average total number of matches

Table 4

*The matching performance of the compared detectors on the eight sequences of the Oxford dataset. In **red** is the algorithm with the largest number in the column. The other top three are in **bordeaux**. The best algorithm is the one that produces the largest number of correct (nonredundant) matches, provided it does not make too many detections. This is a bidimensional criterion that is not fully represented in a single table. Another comparison will consider both components simultaneously (Figure 10).*

	bark scale	bikes blur	boat scale	graf viewp	leuven illum	trees blur	ubc jpeg	wall viewp	*mean*
SIFT	**1021.6**	**1034.8**	**3802.8**	**1906.6**	**1736.6**	**9143.0**	**5296.0**	**8677.6**	*4077.4*
SIFT S	**848.0**	871.0	**3225.6**	**1641.8**	1473.6	7506.8	**4272.6**	**7255.2**	*3386.8*
EBR	75.2	366.4	665.2	577.4	458.0	535.2	756.0	2012.4	*680.7*
IBR	131.8	573.2	280.6	293.6	238.2	1141.0	563.2	453.4	*459.4*
HARLAP	118.0	541.0	1438.8	1120.8	568.4	4419.6	1549.0	1963.4	*1464.9*
HESLAP	**814.6**	**2936.4**	**2794.8**	**3164.8**	**2233.2**	**8201.6**	**3594.0**	**4913.8**	*3581.7*
HARAFF	120.2	533.2	1392.2	1103.0	555.8	4397.6	1501.0	1931.6	*1441.8*
HESAFF	**807.2**	**2470.0**	2217.2	**2180.2**	**1538.6**	**7875.8**	3146.0	4798.4	*3129.2*
MSER	85.4	195.2	592.4	280.4	276.4	1839.4	716.0	1372.8	*669.8*
SURF	183.0	546.6	948.2	913.4	607.8	3000.0	1194.0	1564.2	*1119.7*
SFOP	476.0	**1040.8**	825.8	530.2	1014.0	3293.0	1859.4	2243.2	*1410.3*
BRISK	119.2	194.2	1149.6	374.0	521.4	3016.6	1408.8	2413.2	*1149.6*
SIFER	159.4	729.8	**4321.4**	1570.6	**2591.4**	**8818.2**	**6609.8**	**8535.2**	*4167.0*

(a) Average number of detections.

	bark scale	bikes blur	boat scale	graf viewp	leuven illum	trees blur	ubc jpeg	wall viewp	*mean*
SIFT	**328.0**	271.6	**567.0**	**170.6**	**518.2**	335.0	889.6	**1189.6**	*533.7*
SIFT-S	**388.4**	**322.4**	**637.8**	**198.6**	**585.4**	370.4	**1040.0**	**732.4**	*534.4*
EBR	5.0	60.0	19.8	14.2	51.8	15.0	186.0	0.0	*44.0*
IBR	9.2	69.8	13.2	13.4	30.0	48.8	112.2	19.8	*39.6*
HARLAP	21.4	203.0	244.4	53.2	154.0	338.6	943.2	210.4	*271.0*
HESLAP	**168.2**	**1125.2**	**378.4**	**124.8**	**653.6**	**705.0**	**2022.8**	**572.6**	*718.8*
HARAFF	9.4	155.8	125.0	48.8	122.8	225.6	840.2	201.6	*216.2*
HESAFF	50.2	**857.4**	148.4	**67.6**	400.2	**507.6**	**1636.0**	567.2	*529.3*
MSER	7.2	67.0	37.8	12.0	108.6	61.2	194.6	154.6	*80.4*
SURF	47.2	311.0	177.8	54.4	233.4	**410.2**	741.2	261.4	*279.6*
SFOP	**132.8**	310.2	217.8	64.0	357.0	186.2	587.8	384.2	*280.0*
BRISK	5.2	29.0	67.6	20.2	114.8	125.8	344.8	160.2	*108.5*
SIFER	8.6	**313.0**	**384.0**	55.4	**872.6**	**553.0**	**2329.6**	**1694.4**	*776.3*

(b) Total number of matches.

	bark scale	bikes blur	boat scale	graf viewp	leuven illum	trees blur	ubc jpeg	wall viewp	*mean*
SIFT	**106.8**	240.8	**365.2**	**104.8**	**420.2**	161.8	758.6	**303.0**	*307.7*
SIFT-S	**133.0**	286.0	**413.0**	**124.2**	**474.6**	180.6	894.4	128.4	*329.3*
EBR	0.4	56.6	13.0	6.8	43.8	7.0	174.4	0.0	*37.8*
IBR	2.2	64.6	8.4	5.8	27.2	30.2	102.4	14.2	*31.9*
HARLAP	18.8	189.2	**219.6**	48.0	141.2	258.0	**927.6**	173.6	*247.0*
HESLAP	**138.2**	**1047.8**	**326.8**	**101.8**	**609.8**	**536.8**	**1942.0**	**456.4**	*645.0*
HARAFF	7.8	142.6	102.8	42.2	109.2	172.2	819.6	158.6	*194.4*
HESAFF	41.2	**782.2**	123.0	**49.6**	366.4	**372.2**	**1585.4**	**443.4**	*470.4*
MSER	4.6	65.6	32.0	8.0	106.2	50.6	189.8	133.8	*73.8*
SURF	41.0	**294.2**	157.2	45.0	211.2	**312.4**	694.6	227.6	*247.9*
SFOP	**76.4**	248.8	179.8	47.8	295.6	116.0	532.2	241.8	*217.3*
BRISK	2.0	13.6	28.2	8.2	57.0	47.8	176.8	51.2	*48.1*
SIFER	0.8	**286.2**	136.4	9.8	**703.6**	**263.0**	**2195.8**	**504.2**	*512.5*

(c) Number of correct matches.

	bark scale	bikes blur	boat scale	graf viewp	leuven illum	trees blur	ubc jpeg	wall viewp	*mean*
SIFT	**47.1**	**106.7**	**173.3**	**52.8**	**234.3**	**91.0**	**344.1**	**181.3**	*153.8*
SIFT-S	**52.5**	**118.8**	**190.4**	**59.5**	**264.6**	**101.2**	**387.4**	70.4	*155.6*
EBR	0.0	6.9	2.3	1.5	5.4	2.3	7.9	0.0	*3.3*
IBR	0.5	9.6	2.3	1.1	6.8	7.5	9.3	12.5	*6.2*
HARLAP	7.1	38.0	36.7	11.1	39.1	59.2	89.0	47.8	*41.0*
HESLAP	**19.2**	80.5	39.5	13.9	74.9	**78.3**	93.0	64.4	*58.0*
HARAFF	3.1	36.4	29.4	12.5	36.8	51.2	89.6	51.1	*38.8*
HESAFF	12.3	83.7	29.5	**14.2**	70.3	69.7	98.8	72.2	*56.4*
MSER	2.2	43.2	26.1	6.3	81.3	39.5	129.5	**106.2**	*54.3*
SURF	11.0	47.9	28.8	11.2	49.0	63.9	71.8	68.9	*44.0*
SFOP	**31.2**	**96.9**	**70.2**	**21.6**	**130.4**	62.6	**160.8**	**96.3**	*83.8*
BRISK	0.8	8.5	19.8	6.2	39.1	32.5	81.2	39.2	*28.4*
SIFER	0.2	**91.0**	**73.7**	7.5	**253.1**	**109.7**	**536.8**	**210.8**	*160.4*

(d) Number of nonredundant correct matches.

while Table 4(c) presents the number of correct matches, namely, those that are consistent with the ground truth. Like in the repeatability criterion, one match is considered correct if the overlap error between the two matched keypoints (elliptical regions) is less than 40%. Table 4(d) gives the number of nonredundant correct matches.

Due in part to their large number of detections, the Hessian based methods achieve, in general, the largest number of correct matches. In particular, in the `ubc` sequence, the Hessian-Laplace and Hessian-Affine provide almost twice as many correct matches than SIFT on average. However, this apparent advantage of the Hessian based methods fades away once the detection redundancy is taken into account, as revealed by the number of nonredundant correct matches.

SIFT and its single orientation variant achieve the largest number of nonredundant correct matches in most sequences. Although SIFER produces, on average, the maximum number of nonredundant correct matches on the whole data set, it performs poorly on two sequences (`graf` and `bark`). In Figure 10, the average ratios of correct matches for the 13 compared detectors are plotted as a function of the number of detections. Figure 11 summarizes the methods matching performance relative to each other. For that purpose, the number of detections, the ratio of correct matches, and the ratio of nonredundant correct matches were rescaled, and the mean values over the eight sequences of the rescaled ratios are plotted as a function of the normalized number of detected keypoints.

Similarly to what we have observed on the repeatability ratio, the normalized matching benchmark reveals that the ranking of detectors is significantly disrupted when considering

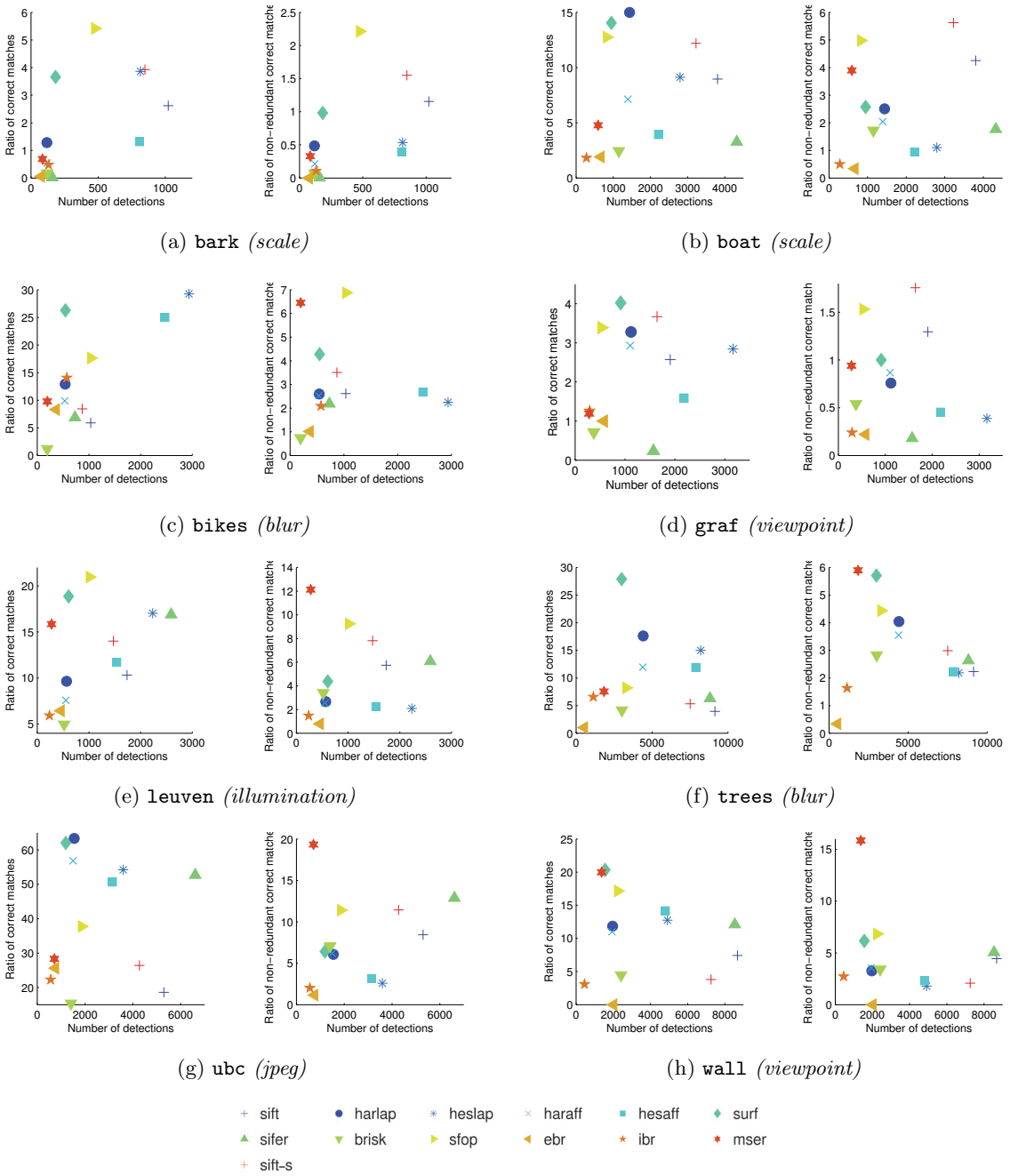


Figure 10. *Ratio of correct matches (left) and nonredundant correct matches (right), i.e., the number of matches over number of detections in the area covered by both images. Again, to compare a single detector matching performance the reader might follow the relative ordinate position of a particular detector in a particular scene. Generally, MSER, SIFT, and SFOP algorithms go up once the redundancy of matches is taken into account. On the other side, Hessian based methods and EBR/IBR always go down once the matches redundancy is taken into account.*

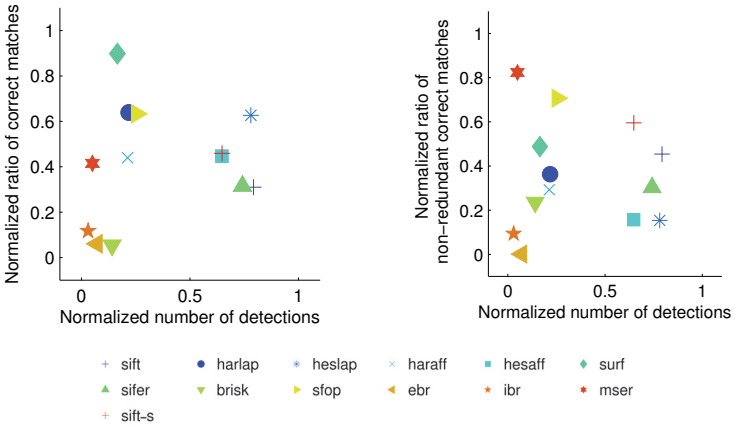


Figure 11. *Qualitative visualization of the methods relative matching performances. This is an average over eight sequences. For each sequence, the number of detections, the matching rates, and the nonredundant matching rates are scaled to the full range* $[0, 1]$ *and averaged into a single map. In a matching scenario taking into account the redundancy of matches, SIFT outperforms Hessian based methods.*

the detectors redundancy. Indeed, when the redundancy is not taken into account, the Hessian Laplace detector is the one producing more detections and a greater number of correct matches per detection. If instead we consider the redundancy, SIFT is the method producing more detections and more nonredundant correct matches per detection.

Interestingly, computing a single orientation for each keypoint improves the performance of the SIFT method. Indeed, this lowers the computational cost of descriptor computations, increases the nonredundant repeatability, and maintains the number of nonredundant correct matches.

6. Discussion. In this paper, we have shown that the classic repeatability criterion is biased towards favoring algorithms producing redundant overlapped detections. This bias motivated the introduction of a variant of the repeatability rate taking into account the descriptor overlap. To illustrate the new repeatability criterion, the performance of several state-of-the-art methods was examined. Experimental evidence showed that, once the descriptors' overlap is taken into account, the traditional hierarchy of several popular methods is severely disrupted. Thus, the detections and associated descriptions generated by some methods are highly correlated. Such redundant parasite detections are arguably caused by scale-space sampling issues (as in the case of Hessian and Harris based methods) or the method's design. For example, the SIFER's kernel generates clusters of scale-space extrema for each blob. The proposed repeatability criterion seems in agreement with the redundancies observed on patterns and on natural images. It also agrees with the detectors matching performance when combined with a common descriptor technique. Experimental evidence reveals that the SIFT and SFOP methods perform best overall as they offer the best balance between a large number of detections and a strong nonredundant repeatability, while MSER performs best for strong affine distortions with fewer detections. The amended metric aims at giving a general yet realistic assessment of keypoint detectors. The revisited benchmark along with detection maps on simple patterns seems to invalidate the performance gains reported over the last decade.

REFERENCES

[1] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, *Building Rome in a day*, Commun. ACM, 54 (2011), pp. 105–112.

[2] H. Bay, B. Fasel, and L. Van Gool, *Interactive museum guide: Fast and robust recognition of museum objects*, in Proceedings of the First International Workshop on Mobile Vision, 2006.

[3] H. Bay, T. Tuytelaars, and L. Van Gool, *SURF: Speeded Up Robust Features*, in Computer Vision–ECCV 2006, Lecture Notes in Comput. Sci. 3951, Springer, Berlin, 2006, pp. 404–417.

[4] J. Bigün, *A structure feature for some image processing applications based on spiral functions*, Comp. Vis. Grap. Image Process., 51 (1990), pp. 166–194.

[5] M. Brown, R. Szeliski, and S. Winder, *Multi-image matching using multi-scale oriented patches*, in Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, IEEE Computer Society, Los Alamitos, CA, 2005, pp. 510–517.

[6] F. Cao, P. Musé, and F. Sur, *Extracting meaningful curves from images*, J. Math. Imaging Vision, 22 (2005), pp. 159–181.

[7] T. Dickscheid and W. Förstner, *Evaluating the suitability of feature detectors for automatic image orientation systems*, in Computer Vision Systems, Springer, Berlin, 2009, pp. 305–314.

[8] T. Dickscheid, F. Schindler, and W. Förstner, *Coding images with local features*, Int. J. Comput. Vis., 94 (2011), pp. 154–174.

[9] S. Ehsan, A. F. Clark, and K. D. McDonald-Maier, *Rapid online analysis of local feature detectors and their complementarity*, Sensors, 13 (2013), pp. 10876–10907.

[10] S. Ehsan, N. Kanwal, A. F. Clark, and K. D. McDonald-Maier, *Measuring the coverage of interest point detectors*, in Image Analysis and Recognition, Springer, Berlin, 2011, pp. 253–261.

[11] R. Fergus, P. Perona, and A. Zisserman, *Object class recognition by unsupervised scale-invariant learning*, in Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, Los Alamitos, CA, 2003, pp. 264–271.

[12] W. Förstner, *A framework for low level feature extraction*, in Computer Vision–ECCV'94, Lecture Notes in Comput. Sci. 801, Springer, Berlin, 1994, pp. 383–394.

[13] W. Förstner, T. Dickscheid, and F. Schindler, *Detecting interpretable and accurate scale-invariant keypoints*, in Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, ICCV 2009, IEEE, Piscataway, NJ, 2009, pp. 2256–2263.

[14] W. Grimson and D. Huttenlocher, *Object Recognition by Computer: The Role of Geometric Constraints*, MIT Press,Cambridge, MA, 1990.

[15] R. Grompone von Gioi, P. Monasse, J-M. Morel, and Z. Tang, *Towards high-precision lens distortion correction*, in Proceedings of the 2010 IEEE International Conference on Image Processing, ICIP 2010, IEEE, Piscataway, NJ, 2010, pp. 4237–4240.

[16] G. Haro, A. Buades, and J-M. Morel, *Photographing paintings by image fusion*, SIAM J. Imaging Sci., 5 (2012), pp. 1055–1087.

[17] C. Harris and M. Stephens, *A combined corner and edge detector*, in Proceedings of the 4th Alvey Vision Conference, University of Manchester, Manchester, England, 1988, pp. 147–151.

[18] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, 2003.

[19] T. Kadir, A. Zisserman, and M. Brady, *An affine invariant salient region detector*, in Computer Vision–ECCV 2004, Lecture Notes in Comput. Sci. 3023, Springer, Berlin, 2004, pp. 228–241.

[20] S. Leutenegger, M. Chli, and R.Y. Siegwart, *BRISK: Binary Robust Invariant Scalable Keypoints*, in Proceedings of the 2011 International Conference on Computer Vision, ICCV 2011, IEEE, Piscataway, NJ, 2011, pp. 2548–2555.

[21] B. Li, R. Xiao, Z. Li, R. Cai, B.-L. Lu, and L. Zhang, *Rank-SIFT: Learning to rank repeatable local interest points*, in Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, IEEE, Piscataway, NJ, 2011, pp. 1737–1744.

[22] T. Lindeberg, *Scale-Space Theory in Computer Vision*, Springer, Boston, 1994.

[23] D. Lowe, *Object recognition from local scale-invariant features*, in Proceedings of the Seventh IEEE International Conference on Computer Vision, IEEE Computer Society, Los Alamitos, CA, 1999, pp. 1150–1157.

[24] D. Lowe, *Distinctive image features from scale-invariant keypoints*, Int. J. Comput. Vis., 60 (2004), pp. 91–110.
[25] P. Mainali, G. Lafruit, Q. Yang, B. Geelen, L. Van Gool, and R. Lauwereins, *SIFER: Scale-Invariant Feature Detector with Error Resilience*, Int. J. Comput. Vis., 104 (2013), pp. 172–197.
[26] E. Mair, G. D. Hager, D. Burschka, M. Suppa, and G. Hirzinger, *Adaptive and generic corner detection based on the accelerated segment test*, in Computer Vision–ECCV 2010, Lecture Notes in Comput. Sci. 6312, Springer, Berlin, 2010, pp. 183–196.
[27] J. Matas, O. Chum, M. Urban, and T. Pajdla, *Robust wide-baseline stereo from maximally stable extremal regions*, Image Vis. Comp., 22 (2004), pp. 761–767.
[28] K. Mikolajczyk and C. Schmid, *Scale & affine invariant interest point detectors*, Int. J. Comput. Vis., 60 (2004), pp. 63–86.
[29] K. Mikolajczyk and C. Schmid, *A performance evaluation of local descriptors*, IEEE Trans. Pattern Anal. Mach. Intell., 27 (2005), pp. 1615–1630.
[30] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, *A comparison of affine region detectors*, Int. J. Comput. Vis., 65 (2005,), pp. 43–72.
[31] D. B. Reid, *An algorithm for tracking multiple targets*, IEEE Trans. Automat. Control, 24 (1979), pp. 843–854.
[32] I. Rey-Otero and M. Delbracio, *Anatomy of the SIFT Method*, Image Process. On Line, 4 (2014), pp. 370–396.
[33] I. Rey-Otero, M. Delbracio, and J.-M. Morel, *Comparing feature detectors: A bias in the repeatability criteria*, in Proceedings of the 2015 IEEE International Conference on Image Processing, IEEE.
[34] I. Rey-Otero, J.-M. Morel, and M. Delbracio, *An Analysis of scale-space sampling in SIFT*, in 2014 IEEE International Conference on Image Processing, ICIP 2014, IEEE, Piscataway, NJ, 2014, pp. 4847–4851.
[35] E. Rosten and T. Drummond, *Machine learning for high-speed corner detection*, in Computer Vision–ECCV 2006, Lecture Notes in Comput. Sci. 3951, Springer, Berlin, 2006, pp. 430–443.
[36] E. Rosten, R. Porter, and T. Drummond, *Faster and better: A machine learning approach to corner detection*, IEEE Trans. Pattern Anal. Mach. Intell., 32 (2010), pp. 105–119.
[37] C. Schmid, R. Mohr, and C. Bauckhage, *Evaluation of interest point detectors*, Int. J. Comput. Vis., 37 (2000), pp. 151–172.
[38] N. Snavely, S. M. Seitz, and R. Szeliski, *Photo tourism: Exploring photo collections in 3D*, ACM Trans. Graphics, 25 (2006), pp. 835–846.
[39] T. Tuytelaars and K. Mikolajczyk, *Local invariant feature detectors: A survey*, Found. Trends Comp. Graph. Vis., 3 (2008), pp. 177–280.
[40] T. Tuytelaars and L. Van Gool, *Content-based image retrieval based on local affinely invariant regions*, in Visual Information and Information Systems, Lecture Notes in Comput Sci. 1614, Springer, Berlin, 1999, pp. 493–500.
[41] T. Tuytelaars and L. Van Gool, *Wide baseline stereo matching based on local, affinely invariant regions.*, in Proceedings of the 11th British Machine Vision Conference, BMVC 2000, British Machine Vision Association, Manchester, England, 2000, pp. 412–425.
[42] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, *Local features and kernels for classification of texture and object categories: A comprehensive study*, Int. J. Comput. Vis., 73 (2007), pp. 213–238.
[43] H. Zhou, Y. Yuan, and C. Shi, *Object tracking using SIFT features and mean shift*, Comp. Vis. Image Underst., 113 (2009), pp. 345–352.