



A Novel Feature Descriptor Invariant to Complex Brightness Changes

Feng Tang, Suk Hwan Lim, Nelson L. Chang

HP Laboratories
HPL-2009-71

Keyword(s):

brightness invariance, illumination invariant, feature descriptors, image features, feature matching, correspondences

Abstract:

We describe a novel and robust feature descriptor called ordinal spatial intensity distribution (OSID) which is invariant to any monotonically increasing brightness changes. Many traditional features are invariant to intensity shift or affine brightness changes but cannot handle more complex nonlinear brightness changes, which often occur due to the nonlinear camera response, variations in capture device parameters, temporal changes in the illumination, and viewpoint-dependent illumination and shadowing. A configuration of spatial patch sub-divisions is defined, and the descriptor is obtained by computing a 2-D histogram in the intensity ordering and spatial sub-division spaces. Extensive experiments show that the proposed descriptor significantly outperforms many state-of-the-art descriptors such as SIFT, GLOH, and PCA-SIFT under complex brightness changes. Moreover, the experiments demonstrate the proposed descriptor's superior performance even in the presence of image blur, viewpoint changes, and JPEG compression. The proposed descriptor has far reaching implications for many applications in computer vision.

External Posting Date: April 6, 2009 [Fulltext]

Approved for External Publication

Internal Posting Date: April 6, 2009 [Fulltext]



To be published and presented at IEEE Conference on Computer Vision and Pattern Recognition, June 2009.

© Copyright IEEE Conference on Computer Vision and Pattern Recognition.

A Novel Feature Descriptor Invariant to Complex Brightness Changes

Feng Tang, Suk Hwan Lim, Nelson L. Chang
Hewlett-Packard Labs
Palo Alto, California, USA
{feng.tang; suk-hwan.lim; nelson.chang}@hp.com

Abstract

We describe a novel and robust feature descriptor called ordinal spatial intensity distribution (OSID) which is invariant to any monotonically increasing brightness changes. Many traditional features are invariant to intensity shift or affine brightness changes but cannot handle more complex nonlinear brightness changes, which often occur due to the nonlinear camera response, variations in capture device parameters, temporal changes in the illumination, and viewpoint-dependent illumination and shadowing. A configuration of spatial patch sub-divisions is defined, and the descriptor is obtained by computing a 2-D histogram in the intensity ordering and spatial sub-division spaces. Extensive experiments show that the proposed descriptor significantly outperforms many state-of-the-art descriptors such as SIFT, GLOH, and PCA-SIFT under complex brightness changes. Moreover, the experiments demonstrate the proposed descriptor's superior performance even in the presence of image blur, viewpoint changes, and JPEG compression. The proposed descriptor has far reaching implications for many applications in computer vision including motion estimation, object tracking/recognition, image classification/retrieval, 3D reconstruction, and stereo.

1. Introduction

For many applications, such as object tracking, object recognition, image retrieval, image classification, and stereo, it is very important for the feature detectors and descriptors to be robust to changes in brightness or viewpoint and to image distortions such as noise, blur, or compression. Many prior work on local features such as SIFT (Scale-Invariant Feature Transform) [8], GLOH (Gradient Location Orientation Histogram) [14] and SURF (Speeded up Robust Features) [17] have shown to be fully or partially robust to many of the variations and distortions. While these methods are invariant to intensity shift or affine brightness changes, they cannot handle more complex nonlinear brightness changes.

Complex brightness changes are a common occurrence in real world applications or scenarios. The same physical

point may have different pixel intensities due to changes in the properties and locations (with respect to the object) of the light sources, the scene geometry and the viewing angle. When an object moves in and out of a shadow, the brightness may change abruptly between frames. In addition, changes in the capture parameters such as gain and exposure time, coupled with nonlinear processing of the pixel intensities in the imaging pipeline can cause the brightness to go through complex nonlinear changes. For example, in a simple scenario, auto exposure in the video camera may abruptly reduce the exposure time and the pixel intensities along the motion trajectories would be altered in nonlinear and spatially-varying fashion after gamma correction (i.e., $I_{\text{out}} = kI_{\text{in}}^{0.45}$ where I_{in} and I_{out} are the pixel intensities before and after gamma correction) is applied during the imaging pipeline of the camera. Such complex brightness changes may pose significant challenges for feature matching. In another scenario of stereo, differences in the photon response or capture parameters between multiple cameras may cause the pixel intensities to differ significantly. Due to modules such as auto-exposure (AE) or auto-white-balance (AWB) in the imaging pipeline, the camera responses are adaptive to the input scene and cannot be calibrated entirely. In yet another application of image retrieval/classification, the large database of images available to the system has typically been captured with some capture parameters and processed by unknown and nonlinear imaging pipelines of various digital cameras. When finding corresponding parts in the images for such database, the brightness may vary significantly between corresponding parts and it is challenging to account for them prior to applying matching.

In order to quantify how the brightness changes in real world scenarios, we define the brightness change function, $I_2 = f(I_1)$, as the function that models how the pixel intensities (I_1) in one patch (or image) map to the intensities (I_2) in the corresponding pixels in another image. In the scenario of motion tracking from a video, the two images may be consecutive frames in a video. In the scenario of stereo, the images may be two viewpoints from the two cameras. Note that many well known features such as SIFT are invariant to brightness changes only when the brightness change function has the form of $I_2 = f(I_1) = k_1 * I_1 + k_2$, where k_1 and k_2 are constants.

Complex changes in brightness cause problems in many vision applications. For instance, in optical flow estimation, it is assumed that pixel intensities along the motion trajectories remain unchanged across frames. The performance of algorithms that do not model or account for the complex brightness changes deteriorates significantly when they occur. Such problems also occur in the application of stereo matching when the illumination inconsistencies or variations in capture parameters are present between two or more cameras.

In this paper, we present a robust local feature descriptor that is invariant to complex brightness changes. We shall show later that our local feature descriptor is invariant to any brightness change if the brightness change functions for all the local patches are monotonically increasing. Suppose I_i and I_j are two intensity values, the brightness change function f is monotonically increasing if and only if $f(I_i) \geq f(I_j)$ for all $I_i \geq I_j$. Note that many brightness changes such as affine, square root, square and other nonlinear brightness changes all belong to this category. We argue that the requirement of monotonically increasing brightness change functions accurately reflect the real world scenarios and that the intensity shift or affine brightness change functions (which many prior features assume) are not adequate. Brightness inversion (where darker pixels map to brighter pixels) are rare for consecutive frames in video or different viewpoints in stereo especially when considering pixel intensities in local patches. Note that our requirement of monotonically increasing brightness change function needs to be held only for the pixels in local patches rather than for all the pixels in the image.

This paper is organized as follows: Section 2 gives an overview of the existing methods aiming to handle the brightness changes. We discuss their limitations and the key ideas for our descriptor. The OSID descriptor is presented in Section 3. The dataset and evaluation metrics are described in Section 4. The effectiveness of the descriptor is demonstrated in Section 5.

2. Related Work

Many approaches have been proposed to handle the brightness changes. Normalized cross correlation or adaptive normalized cross correlation [5] attempts to remove brightness incoherencies through intensity normalization but it is very challenging to normalize the intensities for complex brightness changes. Another approach to combat the illumination inconsistency problem is through illumination estimation and color constancy, which decomposes an image into illumination and reflectance components and works on the reflectance image. Several pieces of work belonging to this category include Retinex [7], gamut-mapping [3] and color-by-

correlation algorithm [2]. However, these approaches are not robust since accurate estimation of the illumination and the reflectance from an image or even a sequence of images is an ill-posed problem [18]. Alternatively, there are also algorithms that attempt to estimate the brightness variations. For example, Jin et al. [6] assume an affine illumination change model and simultaneously optimize the illumination and motion parameters for feature tracking. However, simple models like the affine model may not be adequate since the illumination changes may be more complex.

Another class of brightness "invariant" algorithms is through gradient or edge-based image features. For example, the scale invariant feature transform (SIFT) [8] uses magnitude weighted gradient orientation as the feature descriptor, and shape context [10] uses edges as the descriptor. Although image gradients are generally believed to be invariant to brightness changes, they are only invariant when the brightness change function is just an intensity shift. Also, since the edge detection process is dependent on illumination conditions (due to the thresholding of gradients to obtain the edge maps), edge features are not truly invariant to brightness changes.

Histogram equalization (HE) may be used as a pre-processing step to alleviate brightness change problems. However, it is very sensitive to the scope of pixels used to estimate the histogram hence often fails when objects in the scene change.

In order to tackle more general brightness changes, methods have been proposed to use relative ordering of the pixel intensities rather than the original intensities. The observation is that although the pixel intensities in the corresponding locations may change due to the changes in illumination or capture parameters, the relative ordering of the pixel intensities in the local patch remains unchanged if the brightness change function is monotonically increasing.¹ The idea of using relative ordering of the pixel intensities has been explored in the task of feature matching [14, 11, 1, 12, 9, 4]. Given two patches of the same size, instead of working in the original pixel intensity space, these algorithms compare the relative orderings of the pixels. The census transform and its variant local binary pattern [11] create a vector from a given pixel by comparing it with its neighborhood pixels. Image matching is then performed in this transformed space. However, these methods depend heavily on the center pixel and any slight variation in the center pixel intensity may cause the descriptor to vary significantly. Bhat and Nayar [1] propose a distance measure by penalizing when the relative pixel order flips in the two images. Scherer et al. [12] improves on this by designing a

¹ When the brightness change function is monotonically decreasing the order is reversed.

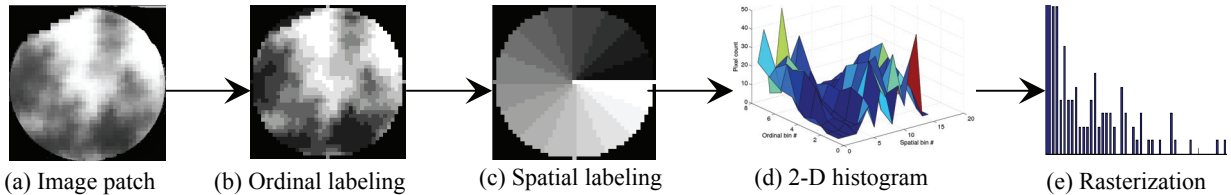


Figure 1. Workflow of our descriptor

more discriminative matching function. Mittal and Ramesh [9] augment the ordinal matching with the original intensity values by considering the change in the pixel intensities in order to achieve greater robustness. In [20], Gupta and Mittal choose point pairs in extremal regions, and design a distance function based on the number of order flips. In [21] the authors use intensity comparison for keypoint detection. These methods [1, 12, 9, 20] generally have several major limitations. They work on the raw ordinal space and have many restrictions (e.g. the number of pixels in the two image windows must be the same to make the ordering comparable). Also, illumination invariance highly depends on the distance measure that needs to be carefully designed to account for the ordinal information. Finally all the spatial information among the pixels, which is very important for image matching, is not considered in the distance function.

In order to overcome the limitations described above, we create a 2-D histogram where the pixel intensities are grouped (or binned) both ordinally and spatially. The key properties of our descriptor and hence our contributions are:

- The feature descriptor is compact and robust (invariant to monotonically increasing local brightness changes.)
- Spatial information is naturally encoded into the new descriptor by using the sub-patch configurations to enhance the discriminative power of the descriptor. Note that simple 1-D histogram of pixel intensity orderings loses spatial information which is important for the discriminative power of the features.
- The 2-D histogram is more robust to image deformations than previous approaches of using raw pixel orderings.

3. The Descriptor: Ordinal-Spatial Intensity Distribution

One of the key ideas of the proposed feature descriptor is that the relative ordering of the pixel intensities in a local patch remains unchanged or stable under monotonically increasing brightness changes. However, simply extracting the feature vector based on the raw ordinal information of pixel intensities in the local patch may not be appropriate since the dimension of the feature vector is too high (i.e., equal to the number of pixels in the

patch). Furthermore, such an approach would make the features sensitive to perturbations such as image distortions or noise.

We propose a novel feature descriptor called *ordinal-spatial intensity distribution* (OSID). The feature is constructed by rasterizing a 2-D histogram where the pixel intensities are grouped (or binned) in the ordinal space as well as spatial space. Binning the pixels in the ordinal space ensures that the feature is invariant to complex brightness changes while binning the pixels spatially captures the structural information of the patch that would have been lost if the feature was obtained from a naïve histogram of the pixels.

To extract features and their descriptors, the image is passed through a pre-processing step and the feature detection step to localize the feature points. Local patches around the feature points are extracted and fed into the feature descriptor. An overview of how OSID is computed from each patch is illustrated in Figure 1.

3.1. Pre-processing and Feature Detection

Before computing the features, the image is smoothed with a Gaussian filter to remove noise since relative ordering of the pixel intensities can be sensitive to noise (later detailed in Section 5). Then, feature points or key points need to be localized. Note that OSID does not require a particular feature detector and any existing feature detector can be used for this step. In order to take full advantage of the brightness-invariant property of OSID, detectors such as intensity-extrema [16] may be a good choice. For each detected feature point, a local patch of size $d \times d$ is extracted where the typical choice of d is 41 but it may vary with the image resolution and scale.

3.2. Ordinal and Spatial Labeling

For ordinal distribution, the pixels in the patch are grouped into $nbins$ bins where each bin has pixels with similar ordinal pixel intensities. Unlike the conventional histograms with $nbins$ bins, each bin represents the ordinal range instead of the range in raw pixel intensities. For example, if there are 400 intensity levels and 5 bins, each bin has 80 intensity levels (i.e., orderings of 1~80, 81~160, 161~240, 241~320 and 321~400). Note that the raw values of the pixel intensities are not used since they may have changed from frame to frame or camera to camera. The pixels are labeled with the bin number that they belong to. A naïve implementation of this process is

to sort all the pixel intensities and group the pixels based on the indices which results in the complexity of $O(n \log_2(n))$ (quicksort) [13], where n is the number of pixels in the patch. We rather use a selection algorithm [13] to find the $nbins$ boundary elements (e.g., orderings of 80, 160, 240, 320 and 400). This reduces the complexity to $O(n \log_2(nbins))$.

For spatial distribution, the pixels in the $d \times d$ patch are labeled based on $npies$ spatial subdivisions. There are many patch/sub-patch configurations possible but we chose to implement our feature with the configuration shown in Fig 1(c), where the circle is divided into pies. Note the assignment of a pixel to a sub-patch is pre-computed to save the computation time.

3.3. 2-D Histogram and Rasterization

We create a 2-D histogram for each local patch where the x -axis in the histogram encodes the pixel intensity distributions with the relative ordering of the pixel intensities and the y -axis encodes the spatial distributions. The value for the bin at location (x,y) in the 2-D histogram denotes the number of pixels in y sub-division that have pixel orderings of x . For example, if x represents the ordinal bin location for the brightest pixels, then the value for the bin (x,y) conveys how many pixels in sub-patch y falls into the category of the brightest pixels.

Each row in the 2-D histogram is a 1-D histogram which represents how the pixels are distributed in the ordinal space given a sub-patch (e.g., pie). It is a $npies$ -dimensional vector where each dimension represents how many of the pixels have the ordinal-spatial label. Each column in the 2-D histogram represents how pixels of similar intensities are distributed across the sub-patches.

After the ordinal-spatial 2-D histogram is constructed for each patch, we perform rasterization on the histogram to form an $nbins \times npies$ dimensional vector as the descriptor of the $d \times d$ patch. The starting bin and the direction (or order) of rasterization may be arbitrary but needs to be pre-defined. For example, we choose to start from the horizontal sub-patch and traverse in a counter-clockwise order (and from inside to outside if radial division configuration is used). Optionally, the starting bin may be dynamically chosen by the content in the patch. For example, the starting row may always be from the darkest/brightest sub-patch or from the sub-patch with the highest gradient energy. Note that such constraints can also be combined for higher accuracy. The adaptive rasterization has been useful for enabling the feature descriptor to be robust to viewpoint changes. Finally, the feature vector is normalized by the number of pixels in the patch to eliminate its effect on the descriptor. Thus, the size of the patch may be varied from one feature point to another.

4. Dataset and Evaluation

4.1. Dataset

We evaluate our descriptor and compare the performance against other well-known descriptors on the standard dataset and method commonly used for feature descriptors evaluation by most vision researchers. The code and dataset is downloaded from the website linked to [14] (<http://www.robots.ox.ac.uk/~vgg/research/affine/>). This dataset consists of real images with different geometric and photometric transformations (e.g. viewpoint changes, image blur, illumination changes, JPEG compression) and has the ground-truth matches through estimated homography. In order to perform more thorough experiments involving complex brightness changes, we augment the ‘‘Leuven’’ dataset by synthesizing more challenging illumination changes. We perform a square root and square operation (after normalizing the pixel intensities such that the maximum is 1) on the 6th image which has the largest illumination change with respect to the reference image (the 1st image). This nonlinear transformation makes the images (see Figure 2) very challenging due to complex brightness changes. Note no additional geometric transformation is applied, so the ground-truth matches remain the same.

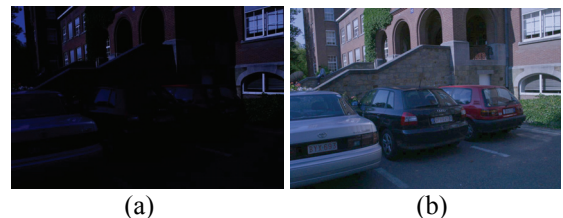


Figure 2. Synthesized images from image 6 of ‘‘Leuven’’: (a) squared brightness change; (b) square root brightness change

4.2. Detector

To make fair comparisons of the descriptors, we follow the well-adopted methodology described in [14]. For the purpose of comparison with other methods, we use Hessian affine region detector [16] because it is widely used and it already adapts to the affine shape changes. The output of the detector is a set of affine normalized image patches and they are used to compute descriptors. Note the patches are the same for all the descriptors.

4.3. Evaluation Metric

We adopt the same evaluation metric as in [14], which is based on the number of generated matches correct matches and false matches obtained for an image pair. The number of correspondences is defined as the number of pairs of features whose distance is below a threshold. This threshold is varied to generate different points in curves shown in the Figures 3~8. A correct match is when the pair of features with the smallest distance coincides with

the pair of points/regions matched by the ground-truth. The results are presented with recall versus 1-precision where recall is the number of correctly matched regions over the number of corresponding regions between two images of the same scene. The number of false matches relative to the total number of matches is represented by 1-precision.

$$recall = \frac{\#correct\ matches}{\#correspondences}$$

$$1 - precision = \frac{\#false\ matches}{\#correct\ matches + \#false\ matches}$$

5. Experiments

5.1. Simulations for Parameter Selection

Our descriptor has three free parameters: the smoothing sigma in the pre-process step along with the number of ordinal and spatial bins for the 2-D histogram. We performed various simulations to study the effect of the parameters on the performance of our descriptor.

Smoothing sigma: Before the descriptors are extracted, the image is smoothed using a Gaussian filter. Smoothing reduces the noise in the image making the descriptor more robust. We performed simulations with and without smoothing and verified that the performance is significantly improved with smoothing prior to the descriptor computation. We also performed simulations to select the appropriate the smoothing parameters (e.g. sigma of the Gaussian kernel). After such simulations, we chose a Gaussian kernel with the size of 5x5 and sigma=1. Although it is beneficial to apply more aggressive smoothing with noisier images, we did not observe large performance variations with smoothing parameters.

Number of ordinal and spatial bins: In general, with more bins, more ordinal and spatial information can be captured by the descriptor. However, the performance will degrade if there are too many bins relative to the number of pixels in the patches due to sensitivity to noise. Also, a high number of bins significantly increases the descriptor's dimension and makes matching more computationally intensive. We simulated various parameters to study the effect of these parameters on the performance. The performance evaluations varying *n*bins (8, 12 and 16) and *n*pies (8, 12, 16 and 20) for 41x41 patches are shown in Figure 3. As can be observed, setting the number of ordinal bins to 8, and the number of spatial bins to 16 gives the best result with a reasonable descriptor dimension (16x8=128).

5.2. Comparison of Performance for OSID and Other Features

The performance of the proposed OSID descriptor is compared against 10 existing features including SIFT,

gradient location and orientation histogram (GLOH), shape context, PCA-SIFT, spin images, steerable filters, differential invariants, complex filters, moment invariants, and cross-correlation of sampled pixel values. We use the binaries and MATLAB code downloaded from the website [14] to localize feature points by extracting the Hessian affine regions. All the descriptors are extracted using the default parameters. In the implementation of our descriptor, we fix the smoothing sigma to be 1 and the number of bins to be 8x16.

Brightness changes: We first tested our descriptor under brightness changes using “Leuven” dataset. We compute the descriptor, perform matching and compare the results to the ground-truth while varying the matching threshold to generate the performance curves. Experiments are run between the first image in this dataset and the remaining five images with increasing amount of brightness changes. The 2nd image has the least brightness change compared with the first and the 6th has the largest brightness change. The results are shown in Figure 4. Due to the space limit, only the performance of the two most difficult image pairs is shown. As can be observed, our descriptor performs consistently and substantially better than SIFT and all other feature descriptors in these tests. We also evaluated the performance using the test images augmented by severe or complex (synthesized) brightness changes as described in Section 4.1. The performance is shown in Figure 5. As can be observed, our descriptors outperform all the other features by a large margin.

Image blur: We also test our feature under image blur using the “Bike” dataset. The results are shown in Figure 6. As can be observed, although image blur was not the highest priority for our descriptor, it outperforms all other descriptors by a wide margin.

Viewpoint changes: Our descriptor is also tested under viewpoint changes using the “Wall” dataset. The results are shown in Figure 7. As can be observed, our descriptor performs best with reasonable amount of viewpoint changes (see Figure 7 (a)) and performs comparable to SIFT under severe viewpoint changes (see Figure 7 (b)).

JPEG compression: The descriptor is also tested under different levels of JPEG compression using “UBC” dataset. Our descriptor performs best under reasonable amount of JPEG compression shown in Figure 8 (a), and good performance in the presence of significant JPEG compression in Figure 8 (b).

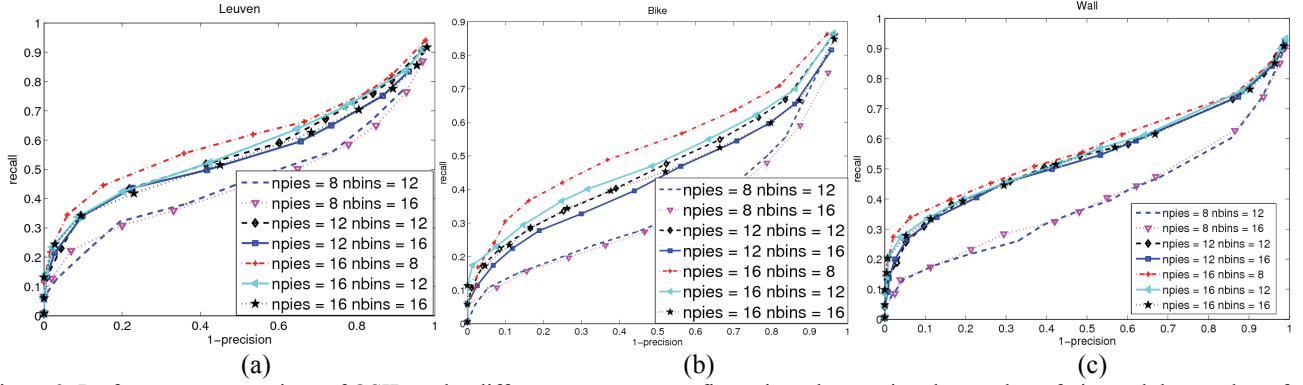


Figure 3. Performance comparison of OSID under different parameter configurations, by varying the number of pies and the number of ordinal space bins. (a) testing on the “Leuven” dataset, matching between the 1st and the 6th image. (b) testing on the “Bike” dataset, matching between the 1st and the 6th image. (c) testing on the “Wall” dataset, matching between the 1st and the 6th image.

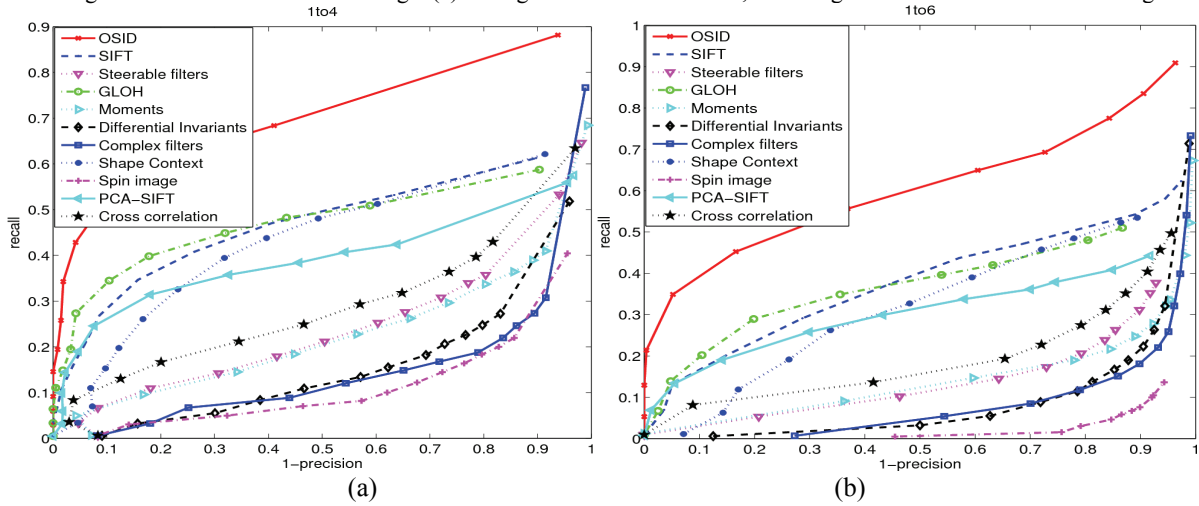


Figure 4. Comparison of OSID against other descriptors under illumination changes using “Leuven” dataset. (a) matching performance between the 1st and the 4th image. (b) matching performance between the 1st and the 6th.

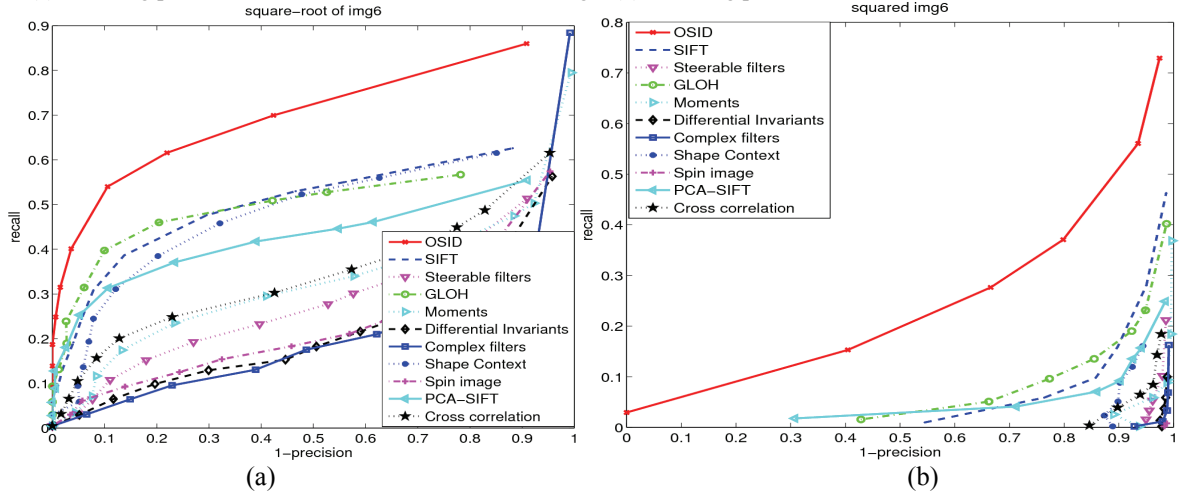


Figure 5. Comparison of OSID against other descriptors under illumination changes using synthesized image (square-root -(a), squared -(b)) in Figure 2

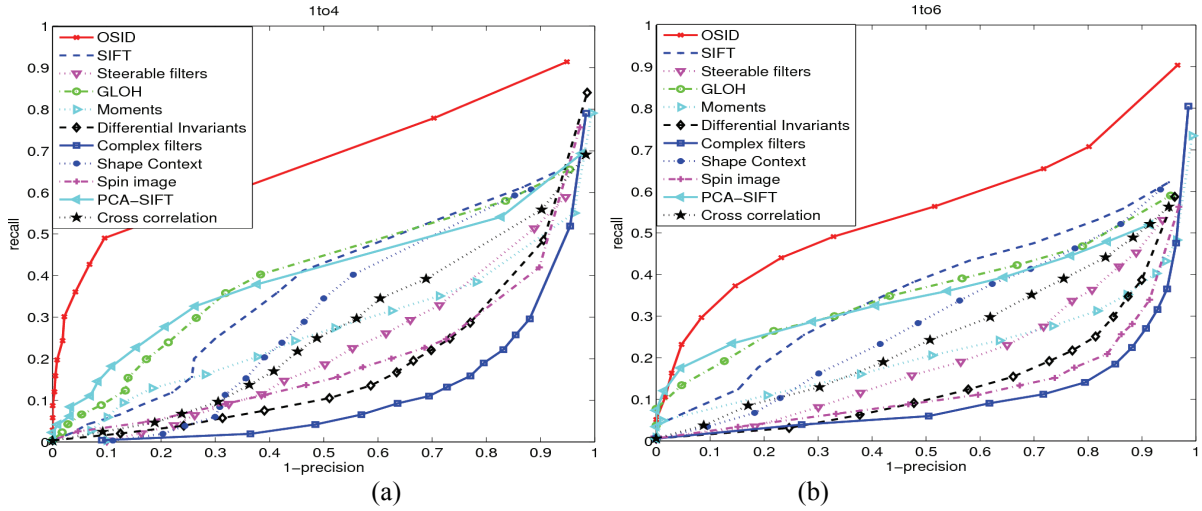


Figure 6. Comparison of OSID against other descriptors under image blur using “Bike” dataset. (a) matching performance between the 1st and the 6th image. (b) matching performance between the 1st and the 6th image.

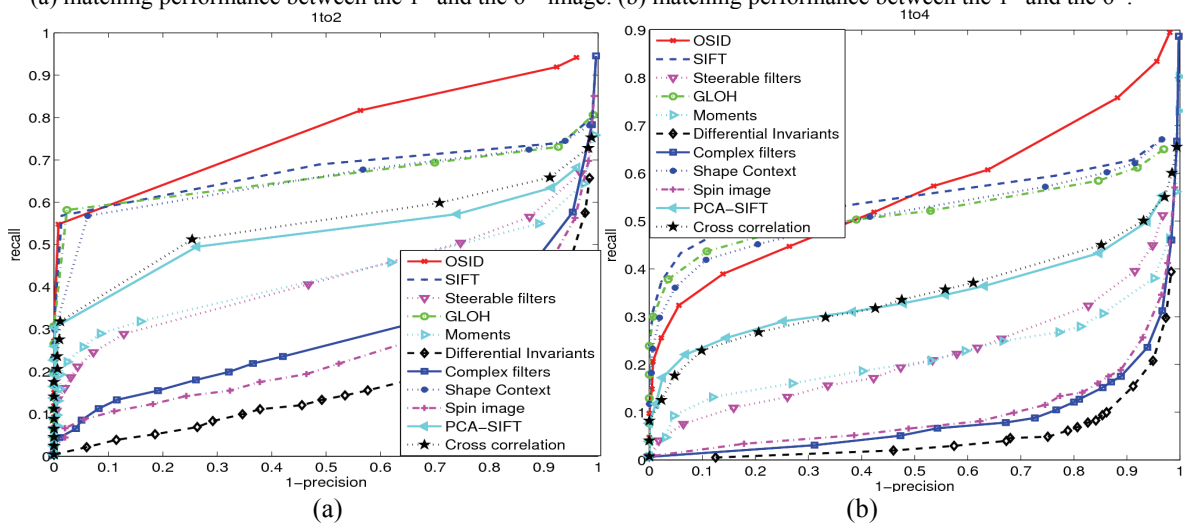


Figure 7. Comparison of OSID against other descriptors under view point changes using “Wall” dataset. (a) matching performance between the 1st and the 2nd image. (b) matching performance between the 1st and the 4th image.

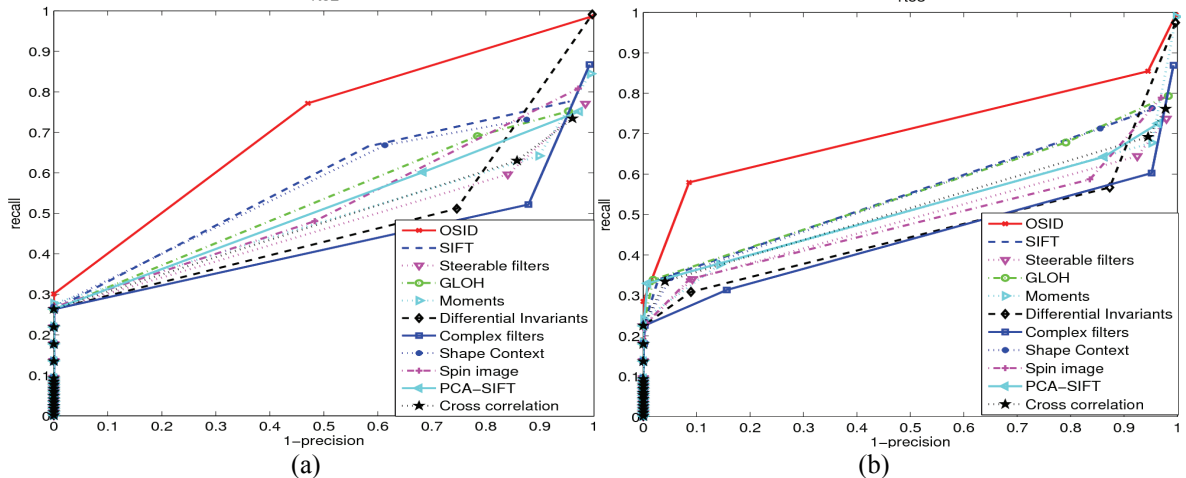


Figure 8. Comparison of OSID against other descriptors under JPEG compression using “UBC” dataset. (a) matching performance between the 1st and the 2nd image. (b) matching performance between the 1st and the 3rd image.

5.3. Example Application

The proposed OSID feature descriptor may be used as a core element of various computer vision based applications. As an example, we have demonstrated using the descriptor to facilitate automatic weak and color calibration between different cameras. Figures 9 (a) and (b) show the raw (uncalibrated) camera views. Despite illumination and viewpoint differences, the features are automatically and accurately extracted with the intensity extrema detector [15] and OSID descriptor directly from the scene without any calibration pattern. The color transformation matrix, transforming the color space between the two cameras, is solved by linear least squares using the pixel intensity values at the matched features points. The matched features are also used for the rectification algorithm from [19] and the resulting image pair is shown in Figure 9 (c) and (d). Note that the reddish color hue of the left image (a) has been corrected in (c) and the book shelf has been automatically aligned to be parallel.

6. Conclusion

We presented a new brightness invariant feature descriptor that is invariant to any monotonically increasing brightness changes. The descriptor captures texture information as well as structure information using an ordinal and spatial intensity histogram. Experiments show the proposed descriptor is very effective in terms of feature matching under complex brightness changes, image blur, viewpoint changes and JPEG compression. We believe the proposed descriptor has far reaching implications for many applications in computer vision including motion estimation, object tracking/recognition,

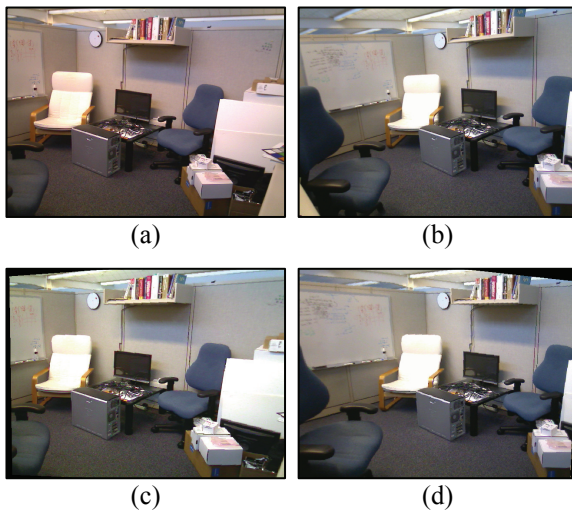


Figure 9. Example of using OSID to extract features for image rectification and color correction. (a) and (b) show the original camera images, while (c) and (d) show the result after rectification and color correction.

image classification/retrieval, 3D reconstruction, and stereo.

7. References

- [1] D. Bhat and S. Nayar. Ordinal measures for image correspondence. *PAMI*, 20(4):415–423, April 1998.
- [2] G. Finlayson, S. Hordley, and P. Hubel. Color by correlation: A simple, unifying framework for color constancy. *PAMI*, 23(11):1209–1221, November 2001.
- [3] D. Forsyth. A novel algorithm for color constancy. *IJCV*, 5(1):5–36, August 1990.
- [4] R. Gupta and A. Mittal. Illumination and affine- invariant point matching using an ordinal approach. In *ICCV*, pp 1–8, 2007.
- [5] Y. Heo, K. Lee, and S. Lee. Illumination and camera invariant stereo matching. In *CVPR*, pages 1–8, 2008.
- [6] H. Jin, P. Favaro, and S. Soatto. Real-time feature tracking and outlier rejection with changes in illumination. In *ICCV*, pages 684–689, 2001.
- [7] E. Land. An alternative technique for the computation of the designator in the retinex theory of color vision. In *Proc. Nat. Acad. Sci.*, pages 3078–3080, 1986.
- [8] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, November 2004.
- [9] A. Mittal and V. Ramesh. An intensity-augmented ordinal measure for visual correspondence. In *CVPR*, pages I: 849–856, 2006.
- [10] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. In *PAMI*, 27(11):1832–1837, 2005.
- [11] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29(1):51–59, 1996.
- [12] S. Scherer, P. Werth, and A. Pinz. The discriminatory power of ordinal measures: Towards a new coefficient. In *CVPR*, pages I: 76–81, 1999.
- [13] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery *Numerical Recipes in C*, Cambridge Press, 1992
- [14] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors. In *PAMI* 27(10):1615-1630, 2004
- [15] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Van Gool, A comparison of affine region detectors. In *IJCV* 65(1/2):43-72, 2005.
- [16] K. Mikolajczyk and C. Schmid, Scale and Affine invariant interest point detectors. In *IJCV* 60(1):63-86, 2004
- [17] H. Bay, T. Tuytelaars, and L. Van Gool, SURF: Speeded Up Robust Features. *ECCV*, pp. 404-417, 2008
- [18] Y. Weiss. Deriving intrinsic images from image sequences. In *ICCV*, pages 68–75, 2001.
- [19] A. Fusiello and L. Irsara, Quasi-Euclidean Uncalibrated Epipolar Rectification, *ICPR*, pp 1-4, 2008.
- [20] R. Gupta and A. Mittal, SMD: A Locally Stable Monotonic Change Invariant Feature Descriptor. *ECCV*, pp. 265-277 2008.
- [21] Vincent Lepetit, Pascal Laguerre, Pascal Fua: Randomized Trees for Real-Time Keypoint Recognition. *CVPR*, pp. 775- 781, 2005

Supplementary material

In this supplementary material, we include a more complete set of results for the performance comparison of our descriptor (OSID) using the standard dataset [14]. We show the results for 5 pairs of images (i.e., 1-2, 1-3, 1-4, 1-5 and 1-6). Note that a subset of these figures was included in the main paper submission. We mark those in **red**.

Fig. 1 shows the performance comparison of OSID against other descriptors under illumination changes using all the images in “Leuven” dataset.

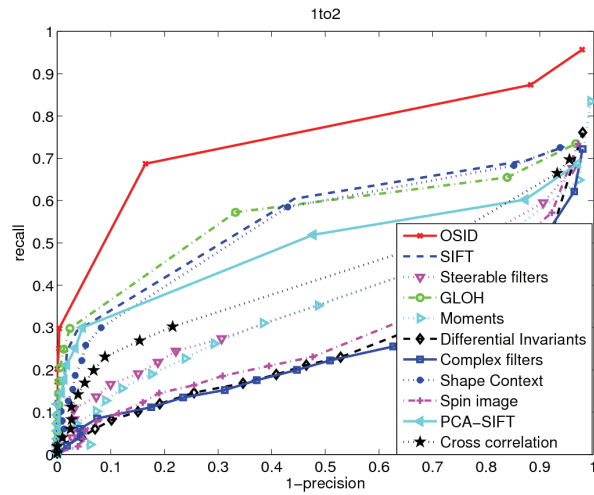
We also show the performance comparison using Harris-affine detector in Fig. 2 using “Leuven” dataset. Note all the rest experiments are using Hessian-affine detector. As can be observed, OSID descriptor performs the best using Harris-affine detector with “Leuven” dataset.

Fig. 3 shows the performance comparison of OSID against other descriptors under image blur using all the images in “Bike” dataset.

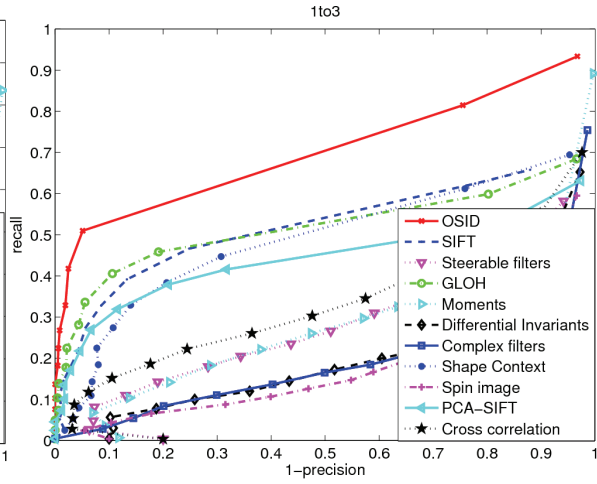
Fig. 4 shows the performance comparison of OSID against other descriptors under viewpoint changes using all the images in “Wall” dataset.

Fig. 5 shows the performance comparison of OSID against other descriptors under JPEG compression using all the images in “UBC” dataset.

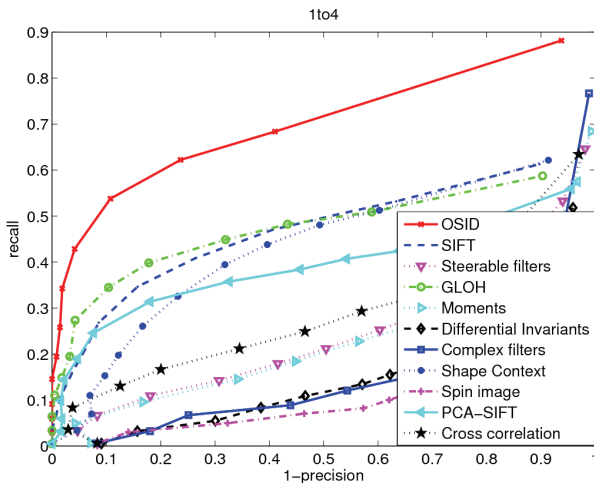
As can be observed from the figures our descriptor significantly outperforms all the other features for most of the cases. In Figure 4 (d) and (e) (i.e., significant viewpoint changes), OSID does not have the best performance but is among the top performers. We believe this is mainly due to OSID’s inability to adapt to the changes in scale and our future work is on addressing this. In Figure 5 (d) and (e) (i.e., significant JPEG compression), the performance of OSID deteriorates rapidly with higher compression. As shown in Fig. 6, the 6th image contains significant compression artifacts such as heavy quantization and blockiness. Figure 7 illustrates how severe the quantization is using the histogram of the pixel intensities for both images. As can be seen, quantization in the 6th image is so severe that the image effectively has 5~7 intensity levels. Such heavy quantization maps many pixel intensities in the original image into only a handful (roughly 5~7 levels of) pixel intensities after compression. Since OSID makes use of relative ordering of pixel intensities, many-to-one mapping of pixel intensities creates significant challenges for OSID. Fortunately, we believe that such heavy quantization rarely occur in the real world.



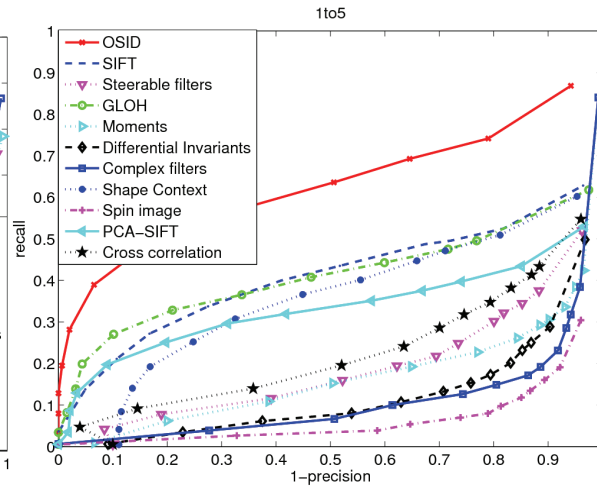
(a)



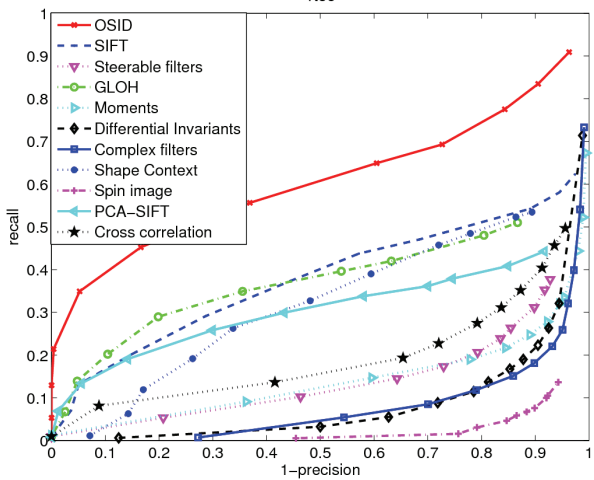
(b)



(c)



(d)



(e)

Fig. 1. Comparison of our descriptor against other descriptors under illumination changes using “Leuven” dataset (with Hessian-affine detector). (a) matching performance between the 1st and the 2nd image; (b) matching performance between the 1st and the 3rd image; (c) matching performance between the 1st and the 4th image; (d) matching performance between the 1st and the 5th image; (e) matching performance between the 1st and the 6th image

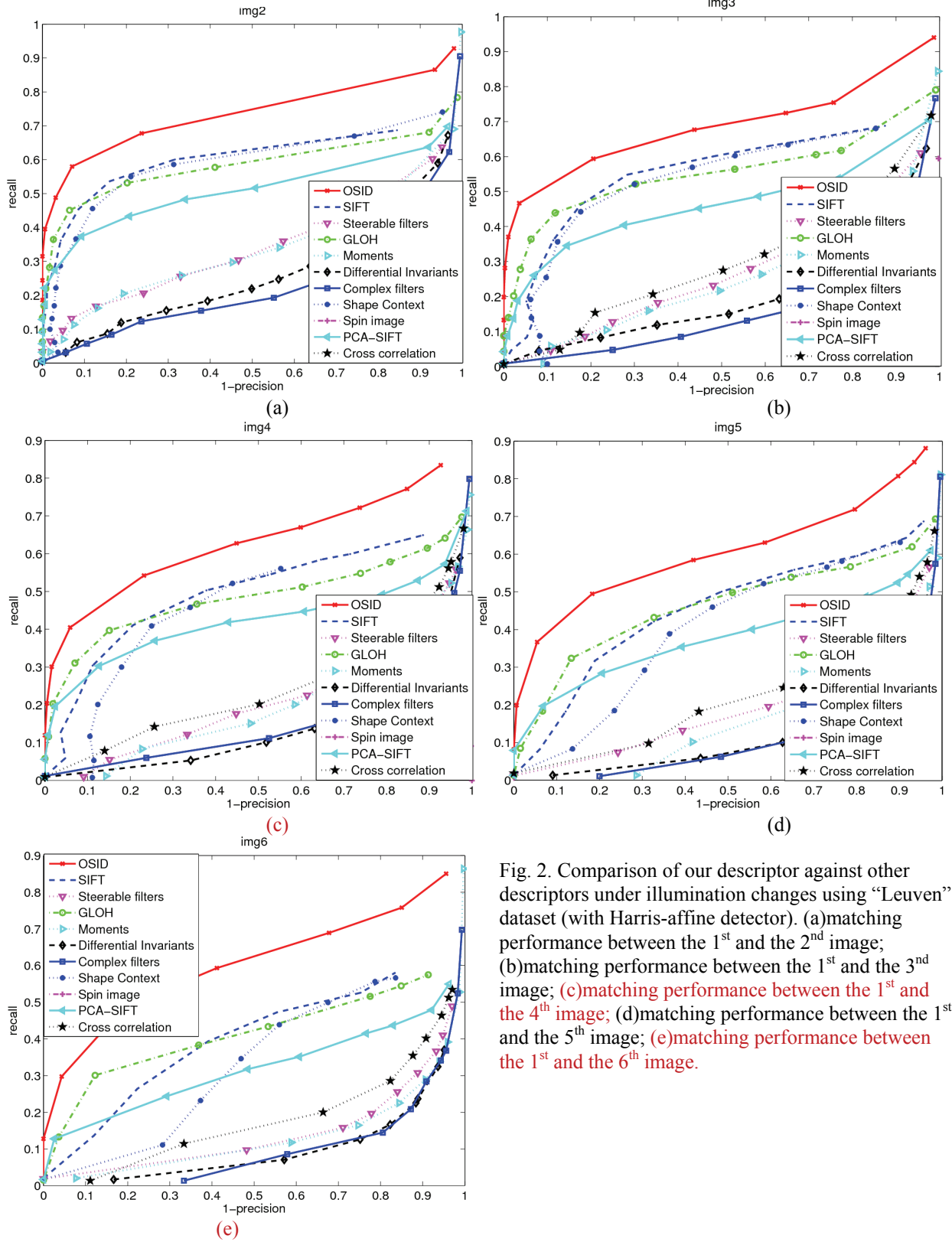


Fig. 2. Comparison of our descriptor against other descriptors under illumination changes using “Leuven” dataset (with Harris-affine detector). (a) matching performance between the 1st and the 2nd image; (b) matching performance between the 1st and the 3rd image; (c) matching performance between the 1st and the 4th image; (d) matching performance between the 1st and the 5th image; (e) matching performance between the 1st and the 6th image.

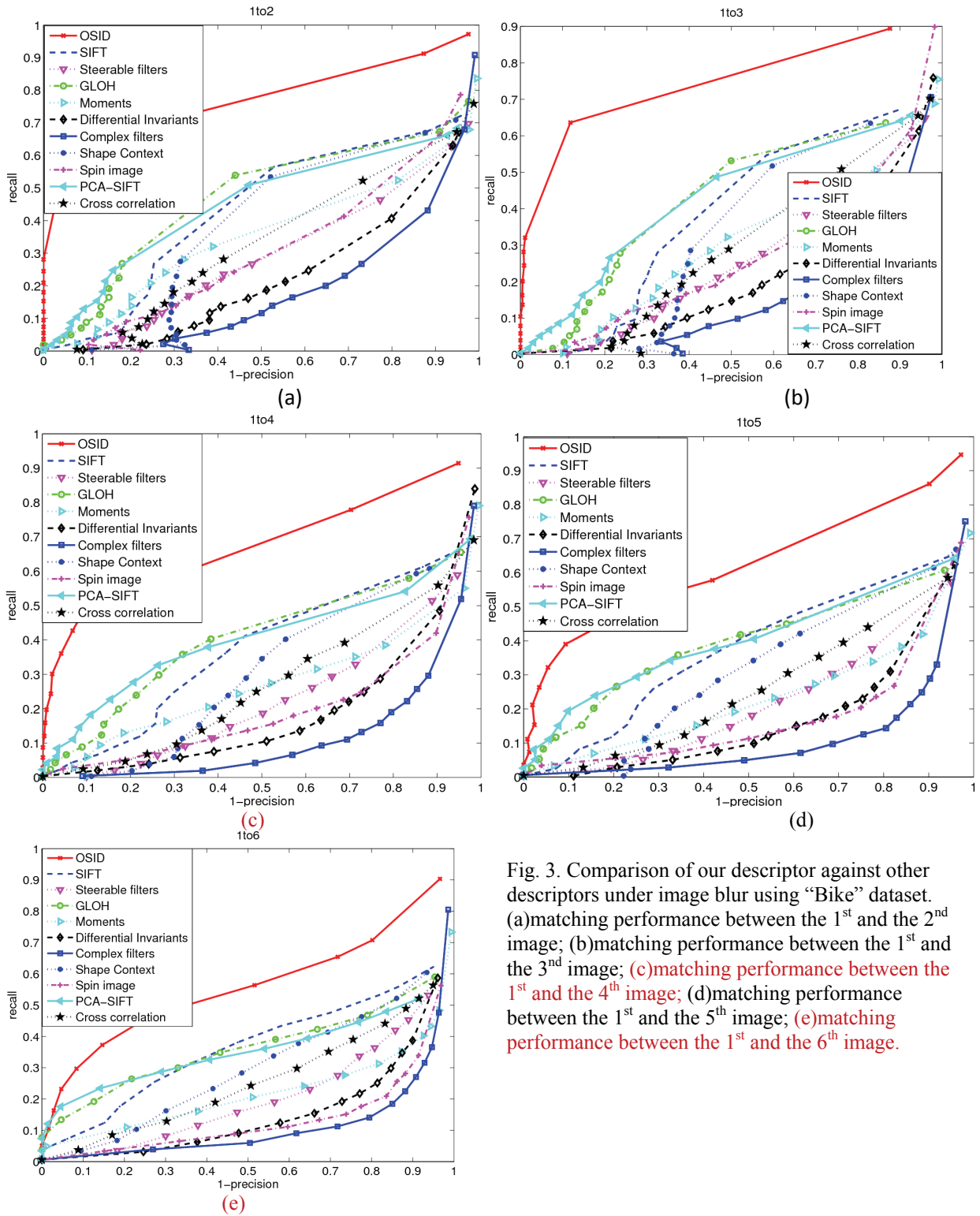


Fig. 3. Comparison of our descriptor against other descriptors under image blur using “Bike” dataset. (a) matching performance between the 1st and the 2nd image; (b) matching performance between the 1st and the 3rd image; (c) matching performance between the 1st and the 4th image; (d) matching performance between the 1st and the 5th image; (e) matching performance between the 1st and the 6th image.

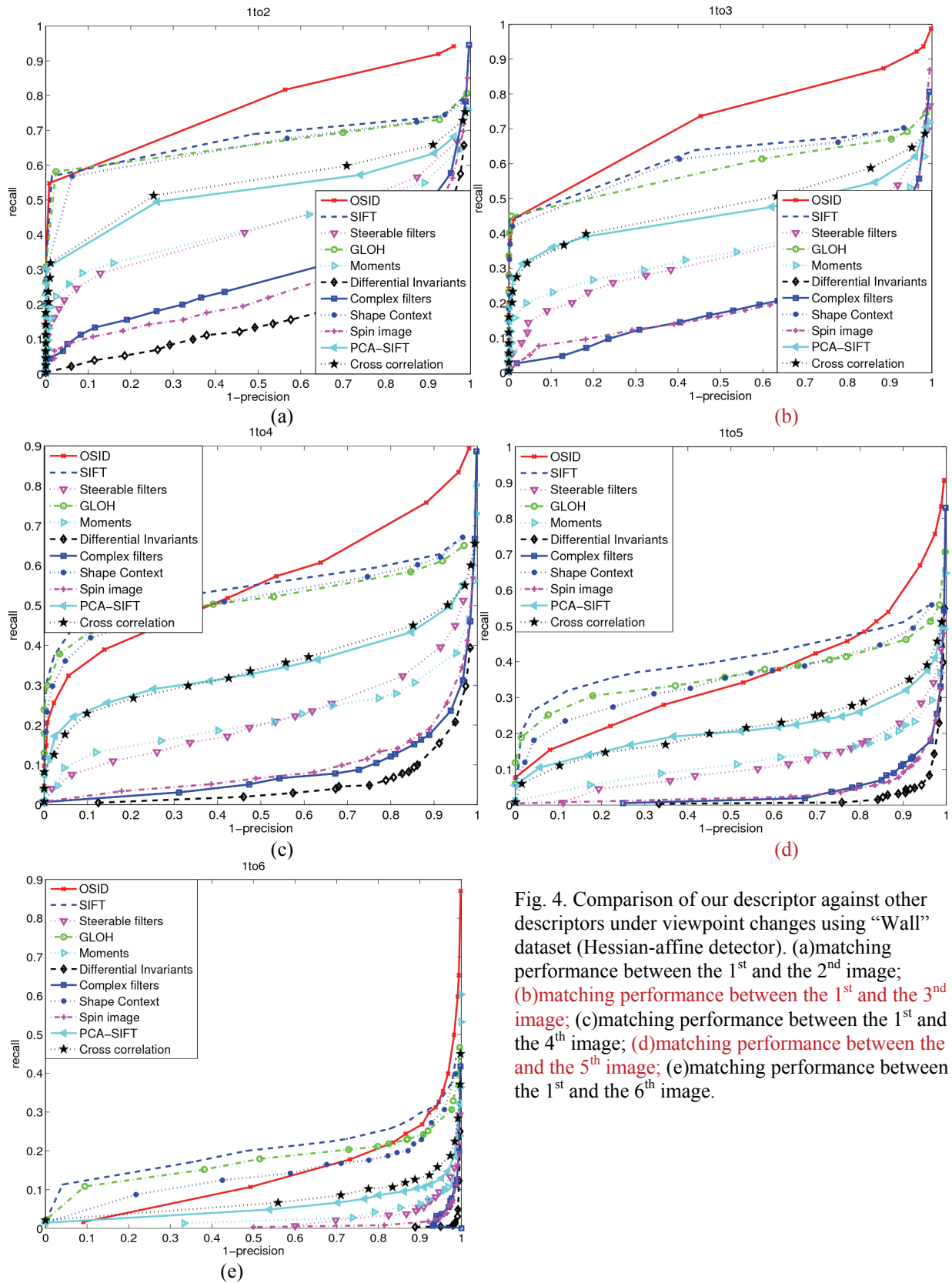


Fig. 4. Comparison of our descriptor against other descriptors under viewpoint changes using “Wall” dataset (Hessian-affine detector). (a) matching performance between the 1st and the 2nd image; (b) matching performance between the 1st and the 3rd image; (c) matching performance between the 1st and the 4th image; (d) matching performance between the 1st and the 5th image; (e) matching performance between the 1st and the 6th image.

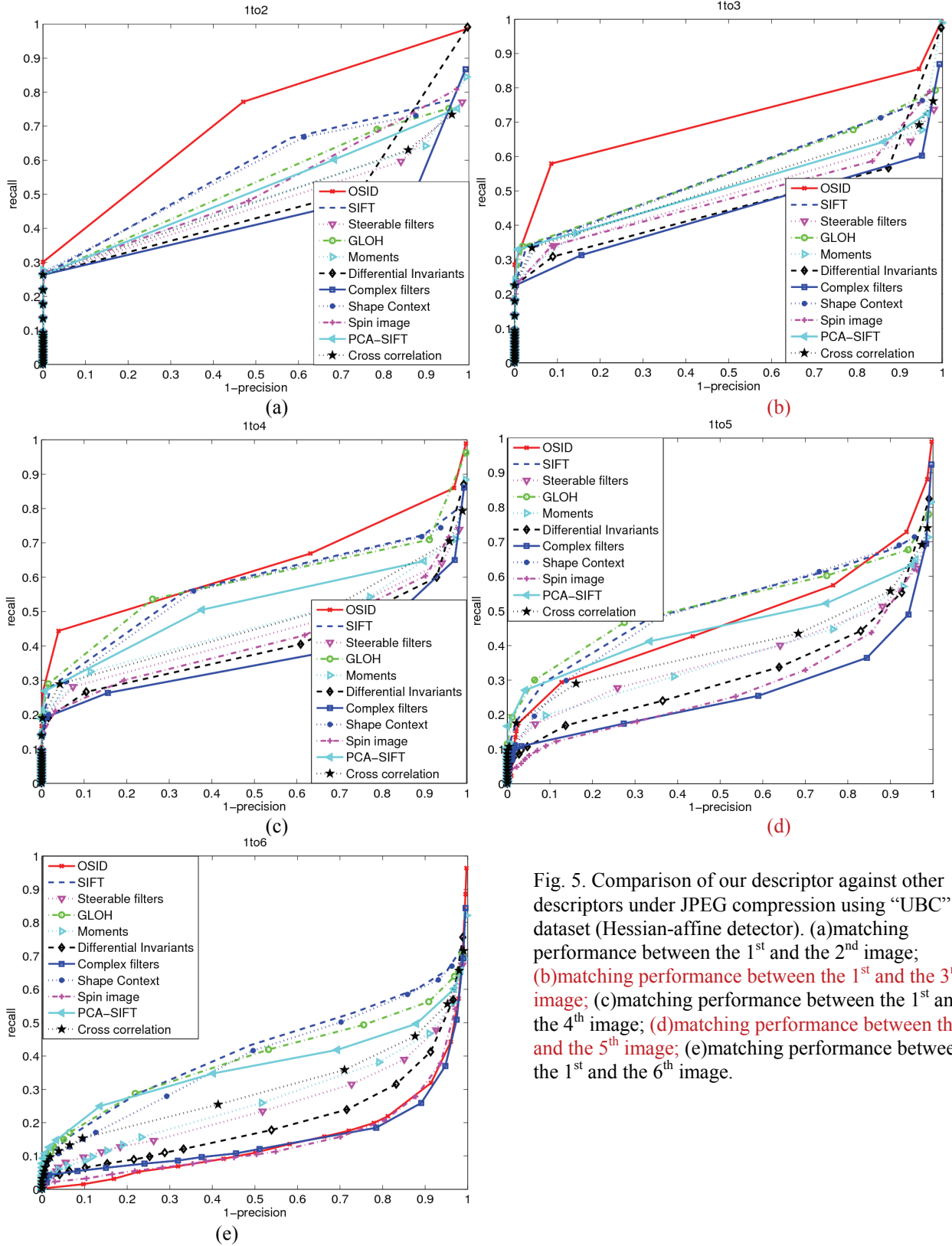


Fig. 5. Comparison of our descriptor against other descriptors under JPEG compression using “UBC” dataset (Hessian-affine detector). (a) matching performance between the 1st and the 2nd image; (b) matching performance between the 1st and the 3rd image; (c) matching performance between the 1st and the 4th image; (d) matching performance between the 1st and the 5th image; (e) matching performance between the 1st and the 6th image.



Fig. 6. 1st and 6th image in the “UBC” dataset for JPEG compression.

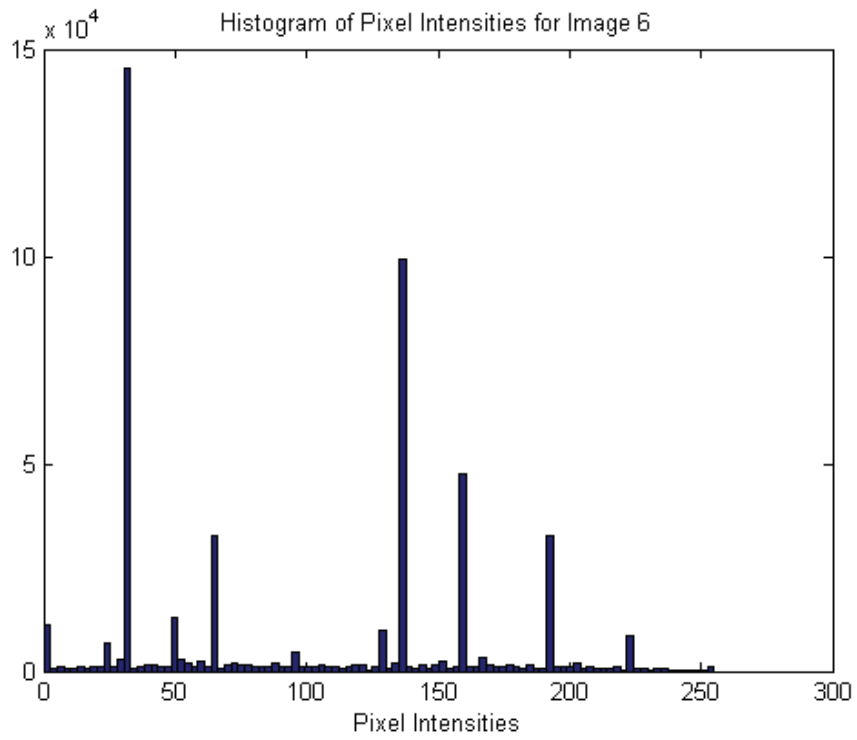
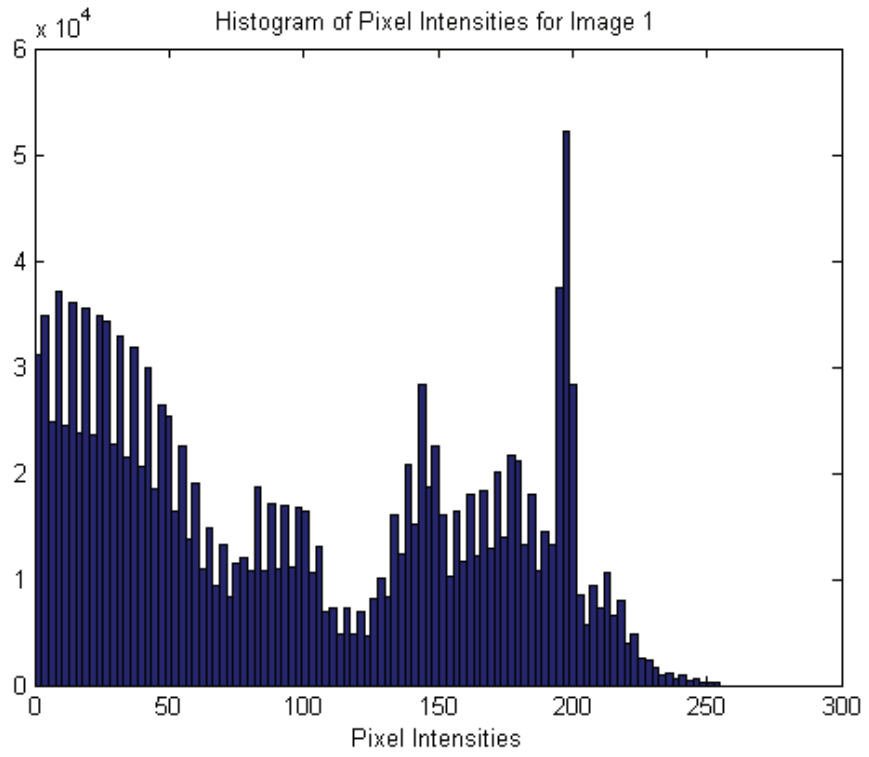


Figure 7: Histogram of intensities for 1st and 6th image in the “UBC” dataset