Description of Interest Regions with Center-Symmetric Local Binary Patterns

Marko Heikkilä[1], Matti Pietikäinen[1], and Cordelia Schmid[2]

[1] Machine Vision Group, Infotech Oulu and Department of Electrical and
Information Engineering, PO Box 4500, FI-90014, University of Oulu, Finland
[2] INRIA Rhône-Alpes, 655 Avenue de l'Europe, 38334 Montbonnot, France
`markot@ee.oulu.fi, mkp@ee.oulu.fi, cordelia.schmid@inrialpes.fr`

Abstract. Local feature detection and description have gained a lot of
interest in recent years since photometric descriptors computed for in-
terest regions have proven to be very successful in many applications. In
this paper, we propose a novel interest region descriptor which combines
the strengths of the well-known SIFT descriptor and the LBP texture
operator. It is called the *center-symmetric local binary pattern (CS-LBP)
descriptor*. This new descriptor has several advantages such as tolerance
to illumination changes, robustness on flat image areas, and computa-
tional efficiency. We evaluate our descriptor using a recently presented
test protocol. Experimental results show that the CS-LBP descriptor
outperforms the SIFT descriptor for most of the test cases, especially for
images with severe illumination variations.

1   Introduction

Local features extracted from images have performed very well in many applica-
tions, such as image retrieval [1], wide baseline matching [2], object recognition
[3], texture recognition [4], and robot localization [5]. They have many advan-
tages over the other methods. They can be made very distinctive, they do not
require segmentation, and they are robust to occlusion. The idea is to first de-
tect interest regions that are covariant to a class of transformations. Then, for
each detected region, an invariant descriptor is built. In this paper, we focus on
interest region description. For more information on interest region detection the
reader is referred to [6].

A good region descriptor can tolerate illumination changes, image noise, im-
age blur, image compression, and small perspective distortions, while preserving
distinctiveness. In a recent comparative study the best results were reported for
the SIFT-based descriptors [7]. For some interesting recent work on interest re-
gion description done after this study, see [8,9,10,11]. The local binary pattern
(LBP) texture operator [12], on the other hand, has been highly successful for
various problems, but it has so far not been used for describing interest regions.
In this paper, we propose a novel interest region descriptor which combines the
strengths of the SIFT descriptor [3] and the LBP operator [12]. Our descriptor
is constructed similarly to SIFT, but the individual features are different. The

gradient features used by SIFT are replaced with features extracted by a *center-symmetric local binary pattern (CS-LBP) operator* similar to the LBP operator. The new features have many desirable properties such as tolerance to illumination changes, robustness on flat image areas, and computational simplicity. They also allow a simpler weighting scheme to be applied. For evaluating our approach, we use the same test protocol as in [7]. It is available on the Internet together with the test data [13]. The evaluation criterion is recall-precision, i.e., the number of correct and false matches between two images.

The rest of the paper is organized as follows. In Section 2, we first briefly describe the SIFT and LBP methods, and then introduce the proposed descriptor in detail. The experimental setup is described in Section 3, and Section 4 presents the experimental results. Finally, we conclude the paper in Section 5.

2   Interest Region Description

Our interest region descriptor is based on the SIFT descriptor [3] which has shown to give excellent results [7]. The basic idea is that the appearance of an interest region can be well characterized by the distribution of its local features. In order to incorporate spatial information into the representation, the region is divided into cells and for each cell a feature histogram is accumulated. The final representation is achieved by concatenating the histograms over the cells and normalizing the resulting descriptor vector. The major difference between the proposed descriptor and the SIFT descriptor is that they rely on different local features. Instead of the gradient magnitude and orientation used by the SIFT, we introduce novel center-symmetric local binary pattern (CS-LBP) features that are motivated by the well-known local binary patterns (LBP) [12]. Before presenting in detail the CS-LBP descriptor, we give a brief review of the SIFT descriptor and the LBP operator.

2.1   SIFT and LBP

SIFT Descriptor. The SIFT descriptor is a 3D histogram of gradient locations and orientations. Location is quantized into a 4×4 location grid and the gradient angle is quantized into 8 orientations, resulting in a 128-dimensional descriptor. First, the gradient magnitudes and orientations are computed within the interest region. The gradient magnitudes are then weighted with a Gaussian window overlaid over the region. To avoid boundary effects in the presence of small shifts of the interest region, a trilinear interpolation is used to distribute the value of each gradient sample into adjacent histogram bins. The final descriptor is obtained by concatenating the orientation histograms over all locations. To reduce the effects of illumination change the descriptor is first normalized to unit length. Then, the influence of large gradient magnitudes is reduced by thresholding the descriptor entries, such that each one is no larger than 0.2, and renormalizing to unit length.

Neighborhood   **Binary Pattern**
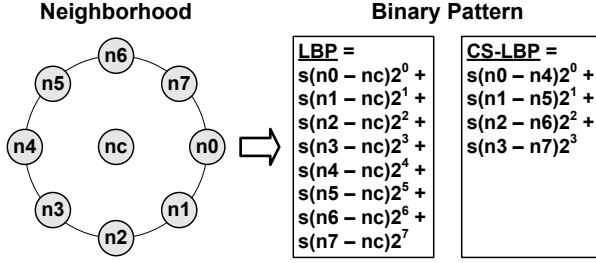
Fig. 1. LBP and CS-LBP features for a neighborhood of 8 pixels

LBP Operator. The local binary pattern is a powerful graylevel invariant texture primitive. The histogram of the binary patterns computed over a region is used for texture description [12]. The operator describes each pixel by the relative graylevels of its neighboring pixels, see Fig. 1 for an illustration with 8 neighbors. If the graylevel of the neighboring pixel is higher or equal, the value is set to one, otherwise to zero. The descriptor describes the result over the neighborhood as a binary number (binary pattern):

$$LBP_{R,N}(x,y) = \sum_{i=0}^{N-1} s(n_i - n_c)2^i, \quad s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & otherwise \end{cases}, \qquad (1)$$

where n_c corresponds to the graylevel of the center pixel of a local neighborhood and n_i to the graylevels of N equally spaced pixels on a circle of radius R. The values of neighbors that do not fall exactly on pixels are estimated by bilinear interpolation. Since correlation between pixels decreases with distance, a lot of the texture information can be obtained from local neighborhoods. Thus, the radius R is usually kept small. In practice, (1) means that the signs of the differences in a neighborhood are interpreted as an N-bit binary number, resulting in 2^N distinct values for the binary pattern. The LBP has several properties that favor its usage in interest region description. The features are robust against illumination changes, they are very fast to compute, do not require many parameters to be set, and have high discriminative power.

2.2 CS-LBP Descriptor

In the following, we provide details on our interest region descriptor which combines the strengths of the SIFT descriptor and the LBP texture operator.

Region Preprocessing. We first filter the region with an edge-preserving adaptive noise-removal filter (we used `wiener2` in Matlab). The edge-preserving nature of the filter is essential for good performance, since much of the information comes from edges and other high-frequency parts of a region. Our experiments have shown that this filtering improves the performance on average around 5 percent (depending on the test images), and therefore all the experiments presented in

this paper are carried out with this kind of filtering. Furthermore, the region data is scaled between 0 and 1 such that 1% of the data is saturated at the low and high intensities of the region. This increases the contrast of the region.

Feature Extraction with Center-Symmetric Local Binary Patterns. After pre-processing, we extract a feature for each pixel of the region using the center-symmetric local binary pattern (CS-LBP) operator which was inspired by the local binary patterns (LBP). The LBP operator produces rather long histograms and is therefore difficult to use in the context of a region descriptor. To produce more compact binary patterns, we compare only center-symmetric pairs of pixels, see Fig. 1. We can see that for 8 neighbors, LBP produces 256 different binary patterns, whereas for CS-LBP this number is only 16. Furthermore, robustness on flat image regions is obtained by thresholding the graylevel differences with a small value T:

$$CS-LBP_{R,N,T}(x,y){=} \sum_{i=0}^{(N/2)-1} s(n_i-n_{i+(N/2)})2^i, \ s(x) = \begin{cases} 1 & x > T \\ 0 & otherwise \end{cases} , \ (2)$$

where n_i and $n_{i+(N/2)}$ correspond to the grayvalues of center-symmetric pairs of pixels of N equally spaced pixels on a circle of radius R. The value of the threshold T is 1% of the pixel value range in our experiments. Since the region data lies between 0 and 1, T is set to 0.01. The radius is set to 2 and the size of the neighborhood is 8. All the experiments presented in this paper, except the parameter evaluation, are carried out for these parameters ($CS-LBP_{2,8,0.01}$) which gave the best overall performance for the given test data. It should be noted that the gain of CS-LBP over LBP is not only due to the dimensionality reduction, but also to the fact that the CS-LBP captures better the gradient information than the basic LBP. Experiments with LBP and CS-LBP have shown the benefits of the CS-LBP over the LBP, in particular, significant reduction in dimensionality while preserving distinctiveness.

Feature Weighting. Different ways of weighting the features are possible. For example, in the case of SIFT, the bins of the gradient orientation histograms are incremented with Gaussian-weighted gradient magnitudes. A comparison of different weighting strategies, including the SIFT-like weighting, showed that simple uniform weighting is the most suitable choice for the CS-LBP features. This is, of course, good news, as it makes our descriptor computationally very simple.

Descriptor Construction. In order to incorporate spatial information into our descriptor, the region is divided into cells with a location grid. Our experiments showed that a Cartesian grid seems to be the most suitable choice. For the experiments presented in this paper, we selected a 4×4 Cartesian grid. For each cell a CS-LBP histogram is built. In order to avoid boundary effects in which the descriptor abruptly changes as a feature shifts from one histogram bin to another, a bilinear interpolation is used to distribute the weight of each feature
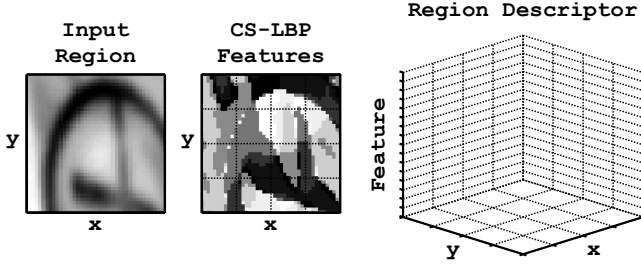
Fig. 2. The CS-LBP descriptor

into adjacent histogram bins. The resulting descriptor is a 3D histogram of CS-LBP feature locations and values, as illustrated in Fig. 2. As explained earlier, the number of different feature values depends on the neighborhood size of the chosen CS-LBP operator.

Descriptor Normalization. The final descriptor is built by concatenating the feature histograms computed for the cells to form a $(4 \times 4 \times 16)$ 256-dimensional vector. The descriptor is then normalized to unit length. The influence of very large descriptor elements is reduced by thresholding each element to be no larger than 0.2. This means that the distribution of CS-LBP features has greater emphasis than individual large values. Finally, the descriptor is renormalized to unit length.

3   Experimental Setup

For evaluating the proposed descriptor, we use the same test protocol as in [7]. The protocol is available on the Internet together with the test data [13]. The test data contains images with different geometric and photometric transformations and for different scene types. Six different transformations are evaluated: *viewpoint change*, *scale change*, *image rotation*, *image blur*, *illumination change*, and *JPEG compression*. The two different scene types are *structured* and *textured* scenes. These test images are shown on the left of Fig. 3. The images are either of planar scenes or the camera position was fixed during acquisition. The images are, therefore, always related by a homography (included in the test data). In order to study in more detail the tolerance of our descriptor to illumination changes, we captured two additional image pairs shown on the right of Fig. 3.

The evaluation criterion is based on the number of correct and false matches between a pair of images. The definition of a match depends on the matching strategy. As in [7], we declare two interest regions to be matched if the Euclidean distance between their descriptors is below a threshold. The number of correct matches is determined with the *overlap error* [14]. It measures how well the regions A and B correspond under a known homography H, and is defined by the ratio of the intersection and union of the regions: $\epsilon_S = 1 - (A \cap H^T BH)/(A \cup$

Graf	Wall	Boat	Bark	Mvlab1

Bikes	Trees	Leuven	Ubc	Mvlab2
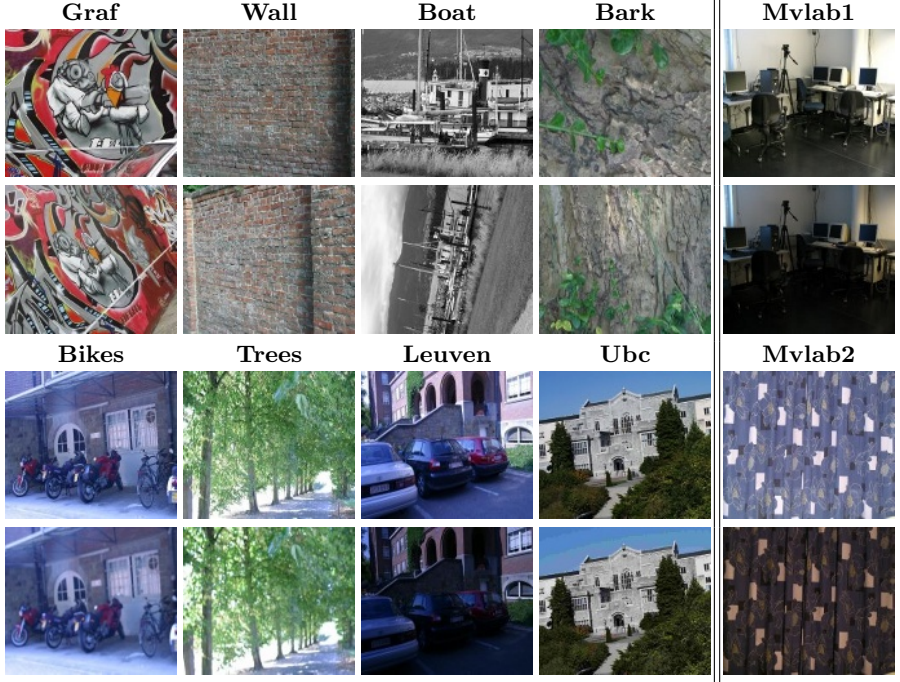
Fig. 3. Test images (left): **Graf** (viewpoint change, structured scene), **Wall** (viewpoint change, textured scene), **Boat** (scale change + image rotation, structured scene), **Bark** (scale change + image rotation, textured scene), **Bikes** (image blur, structured scene), **Trees** (image blur, textured scene), **Leuven** (illumination change, structured scene), and **Ubc** (JPEG compression, structured scene). Additional test images (right): **Mvlab1** (illumination change, structured scene) and **Mvlab2** (illumination change, textured scene).

$H^T BH$). A match is assumed to be correct if $\epsilon_S < 0.5$. A descriptor can have several matches and several of them may be correct. The results are presented with *recall* versus *1-precision*:

$$recall = \frac{\#correct\ matches}{\#correspondences}, \quad 1 - precision = \frac{\#false\ matches}{\#all\ matches}, \quad (3)$$

where the *#correspondences* stands for the ground truth number of matching regions between the images. The curves are obtained by varying the distance threshold and a perfect descriptor would give a recall equal to 1 for any precision.

The interest region detectors provide the regions which are used to compute the descriptors. In the experiments, we use two different detectors: *Hessian-Affine* [6] and *Harris-Affine* [15]. The two detectors output different types of image structures. Hessian-Affine detects blob-like structures while Harris-Affine looks for corner-like structures. Both detectors output elliptic regions of varying

size which depends on the detection scale. Before computing the descriptors, the detected regions are mapped to a circular region of constant radius to obtain scale and affine invariance. Rotation invariance is obtained by rotating the normalized regions in the direction of the dominant gradient orientation, as suggested in [3]. For region detection and normalization, we use the software routines provided by the evaluation protocol. In the experiments, the normalized region size is fixed to 41×41 pixels.

4   Experimental Results

In this section we first evaluate the performance of our CS-LBP descriptor for different parameter settings and then compare the resulting version to the SIFT descriptor.

Descriptor Parameter Evaluation. The evaluation of different parameter settings is carried out for a pair of images with a viewpoint change of more than 50 degrees. The images are shown in Fig. 4. We use the Hessian-Affine detector which extracts 2454 and 2296 interest regions in the left and right images, respectively. The performance is measured with nearest neighbor matching, i.e., a descriptor has only one match. We keep the 400 best matches and report the percentage of correct matches. Note that there are 503 possible nearest neighbor correspondences identified between the images.

We compare the matching performance (percentage of correct matches) for differently spaced location grids, different parameters of the CS-LBP operator, and two weighting schemes. Fig. 5 shows that a 4×4 Cartesian grid outperforms all the other grid spacings. The left graph clearly shows that a uniform weighting outperforms a SIFT-like one and that a neighborhood size 8 is better than 6 or 10. The graph on the right compares different values for the radius and the threshold and shows that a radius of 1 and a threshold of 0.01 give best results. In conclusion, the 4×4 Cartesian grid and the $CS - LBP_{1,8,0.01}$ with uniform weighting give the best performance. For the given image pair, the best results are obtained with a radius of 1. However, experiments with many other image pairs have shown that a radius of 2 actually gives better overall performance. Thus, in the comparison with SIFT, we set the radius to 2 instead of 1. The results also show that our descriptor is not very sensitive to small changes in its parameter values. Note that due to space constraints, Fig. 5 does not cover all

Image 1	Image 2

	CS-LBP	SIFT
Recall	0.386	0.316
1 - Precision	0.515	0.603
Correct Matches	194 / 400	159 / 400

Fig. 4. Left: Image pair with a viewpoint change of more than 50 degrees. Right: The matching results for the 400 nearest neighbor matches between the images.
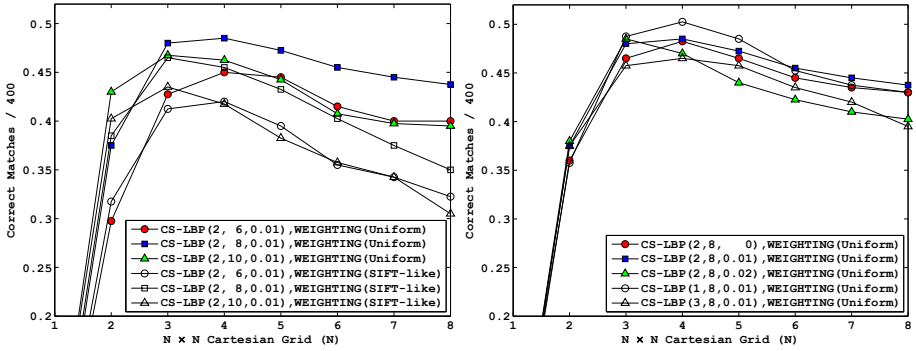
Fig. 5. Evaluation of different parameter settings. See text for details.

the tested parameter settings and that the omitted results are consistent with our conclusions.

The dimensionality of the CS-LBP descriptor can be reduced without loss in performance. When reducing the dimension from 256 to 128 with PCA, the results seemed to remain unchanged. The performance of the 64-dimensional descriptor is still very close to that of the original one. This property makes our descriptor applicable in systems where the matching speed is important. Note that a data set different from the test data was used to estimate the covariance matrices for PCA. The comparison experiments presented next are carried out without using dimension reduction.

Comparison with the SIFT Descriptor. Figures 6 and 7 show the comparison results for Hessian-Affine and Harris-Affine regions, respectively. For Hessian-Affine regions, our descriptor is better than SIFT for most of the test cases and performs about equally well for the remaining ones. A significant improvement of CS-LBP is obtained in the case of illumination changes. For example, for the *Leuven* images, our descriptor gives approximately 20% higher recall for 1-precision of 0.4. The difference is even larger for the additional two test pairs (*Mvlab1* and *Mvlab2*). Clearly better results are also obtained for the *Graf*, *Bikes*, and *Ubc* images which measure the tolerance to viewpoint change, image blur, and JPEG compression, respectively. As we can see, the CS-LBP descriptor performs significantly better than SIFT for structured scenes, while the difference for textured scenes is smaller. Similar results are achieved for Harris-Affine regions. Both descriptors give better overall results for Hessian-Affine regions than for Harris-Affine ones. This is consistent with the findings in [6] and can be explained by the fact that Laplacian scale selection used by the region detectors works better on blob-like structures than on corners [7]. In other words, the accuracy of interest region detection affects the descriptor performance.

Additional experiments were carried out for the scale invariant versions of the detectors, i.e., *Hessian-Laplace* and *Harris-Laplace* [7]. They differ from the affine invariant detectors in that they omit the *affine adaptation* step [15]. The results are not presented due to space limitation, but the ranking of the two
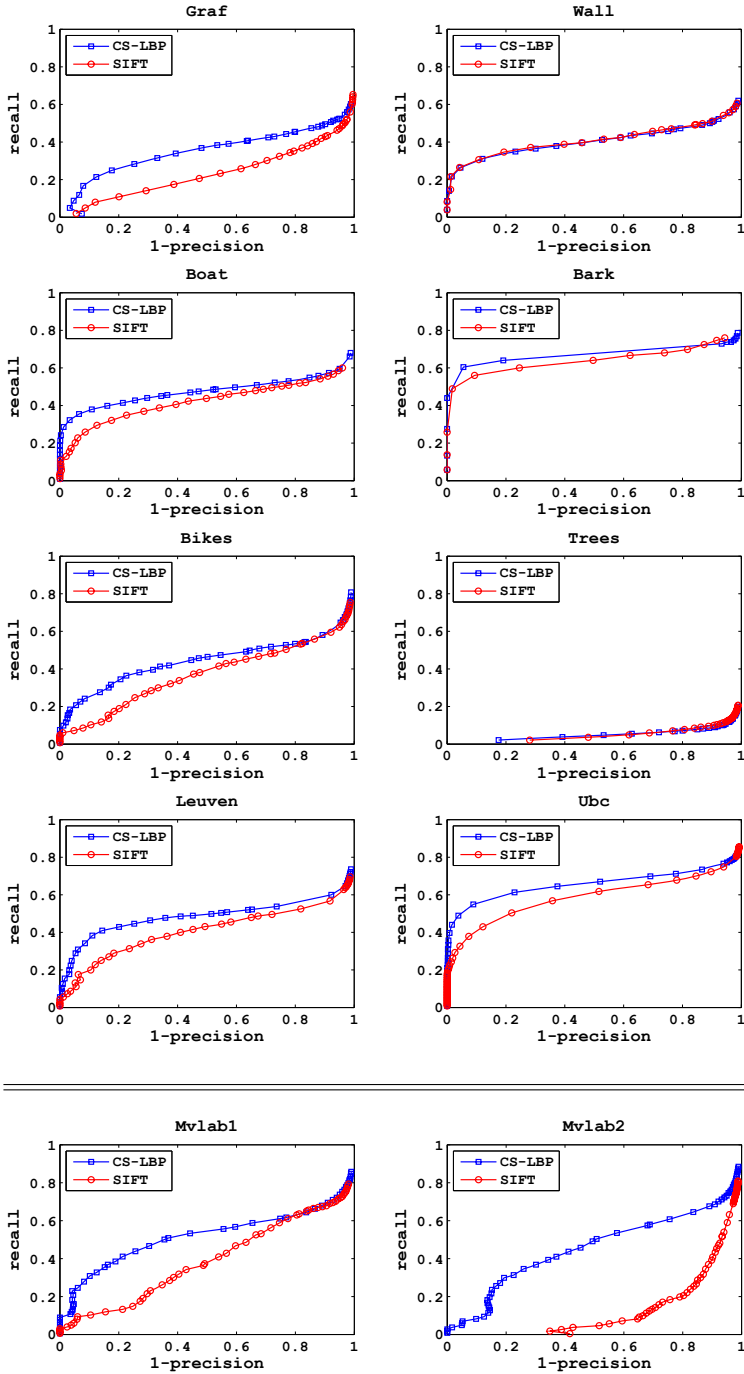
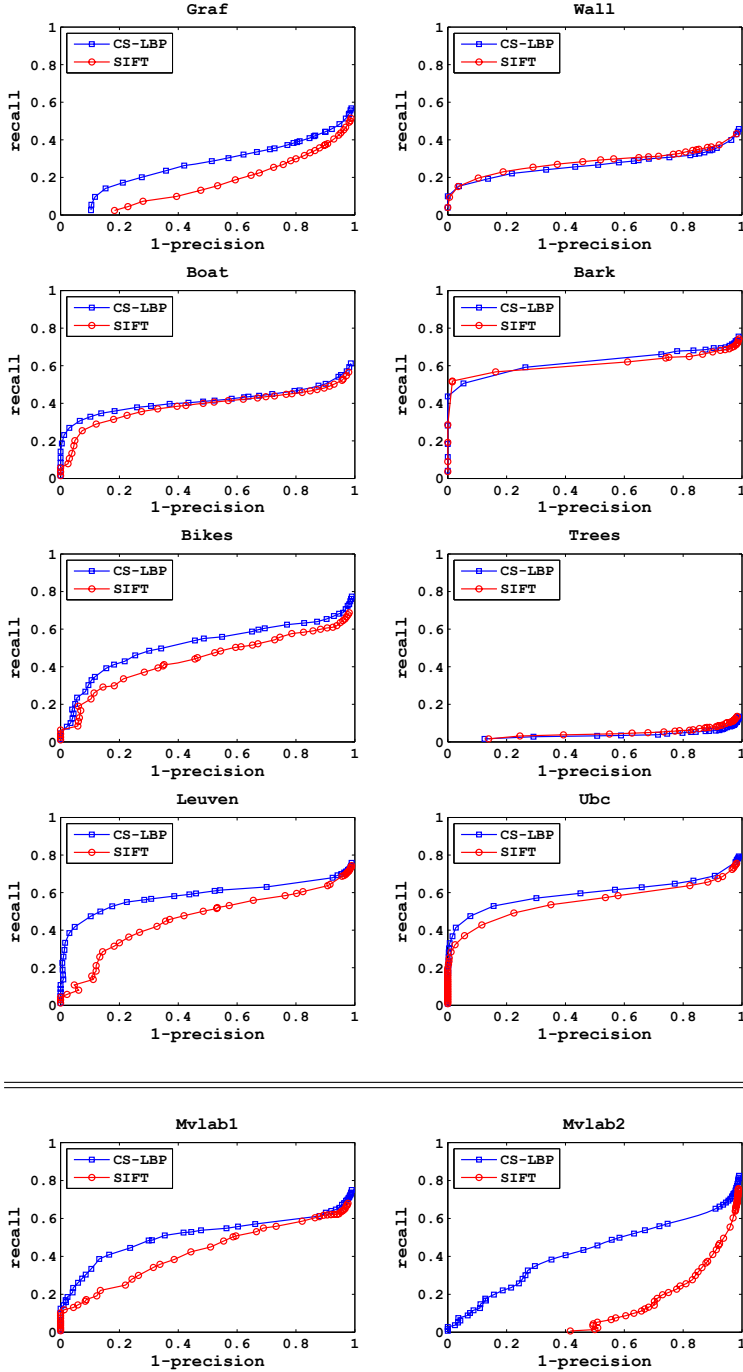**Fig. 6.** Comparison results for Hessian-Affine regions

Fig. 7. Comparison results for Harris-Affine regions

descriptors for the scale invariant regions is comparable to that of the affine invariant regions.

We also performed an additional matching experiment which uses the same setup that was used in the parameter evaluation. Fig. 4 presents recall, 1-precision, and the number of correct matches obtained with the two descriptors for a fixed number of 400 nearest neighbor matches. As we can see, the CS-LBP descriptor clearly outperforms the SIFT descriptor.

5   Conclusions

A novel CS-LBP interest region descriptor which combines the strengths of the well-known SIFT descriptor and the LBP texture operator was proposed. Instead of the gradient orientation and magnitude based features used by SIFT, we proposed to use center-symmetric local binary pattern (CS-LBP) features introduced in this paper. The CS-LBP descriptor was evaluated against the SIFT descriptor using a recently presented test framework. Our descriptor performed clearly better than SIFT for most of the test cases and about equally well for the remaining ones. Especially, the tolerance of our descriptor to illumination changes is clearly demonstrated. Furthermore, our features are more robust on flat image areas, since the graylevel differences are allowed to vary close to zero without affecting the thresholded results. It should be also noted that the CS-LBP descriptor is computationally simpler than the SIFT descriptor. Future work includes applying the proposed descriptor to different computer vision problems such as object recognition and tracking.

Acknowledgment

References

1. Mikolajczyk, K., Schmid, C.: Indexing based on scale invariant interest points. In: 8th IEEE International Conference on Computer Vision. Volume 1. (2001) 525–531
2. Tuytelaars, T., Gool, L.V.: Matching widely separated views based on affine invariant regions. International Journal of Computer Vision **59** (2004) 61–85
3. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60** (2004) 91–110
4. Lazebnik, S., Schmid, C., Ponce, J.: A sparse texture representation using local affine regions. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005) 1265–1278
5. Se, S., Lowe, D., Little, J.: Global localization using distinctive visual features. In: IEEE/RSJ International Conference on Intelligent Robots and System. Volume 1. (2002) 226–231

6. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffal-itzky, F., Kadir, T., Gool, L.V.: A comparison of affine region detectors. International Journal of Computer Vision **65** (2005) 43–72
7. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005) 1615–1630
8. Bay, H., Tuytelaars, T., Gool, L.V.: SURF: Speeded up robust features. In: European Conference on Computer Vision. Volume 1. (2006) 404–417
9. Abdel-Hakim, A.E., Farag, A.A.: CSIFT: A SIFT descriptor with color invariant characteristics. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Volume 2. (2006) 1978–1983
10. Brown, M., Szeliski, R., Winder, S.: Multi-image matching using multi-scale oriented patches. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Volume 1. (2005) 510–517
11. Ling, H., Jacobs, D.W.: Deformation invariant image matching. In: 10th IEEE International Conference on Computer Vision. Volume 2. (2005) 1466–1473
12. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence **24** (2002) 971–987
13. http://www.robots.ox.ac.uk/~vgg/research/affine/.
14. Mikolajczyk, K., Schmid, C.: An affine invariant interest point detector. In: European Conference on Computer Vision. Volume 1. (2002) 128–142
15. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. International Journal of Computer Vision **60** (2004) 63–86