# Robust Binary Feature using the Intensity Order

Yukyung Choi*, Chaehoon Park*, Joon-Young Lee, and In So Kweon

Robotics and Computer Vision Lab., KAIST, Korea

**Abstract.** Binary features have received much attention with regard to memory and computational efficiency with the emerging demands in the mobile and embedded vision systems fields. In this context, we present a robust binary feature using the intensity order. By analyzing feature regions, we devise a simple but effective strategy to detect keypoints. We adopt an ordinal description and encode the intensity order into a binary descriptor with proper binarization. As a result, our method obtains high repeatability and shows better performance with regard to feature matching with much less storage usage than other conventional features. We evaluate the performance of the proposed binary feature with various experiments, demonstrate its efficiency in terms of storage and computation time, and show its robustness under various geometric and photometric transformations.

## 1 Introduction

Finding correspondences between images is a fundamental step in many computer vision methods, such as object recognition, image retrieval, and wide-baseline stereo. The key component of a correspondence search is to extract invariant image features, and many computer vision researchers have focused on extracting invariant image features based on their importance.

The main concerns with regard to invariant features are localization accuracy, invariance to geometric and photometric deformations, and distinctiveness to be correctly matched against a large number of features. SIFT [1] and SURF [2] are known as the best known and most widely used methods among all various image features. They find scale-invariant distinctive image regions and represent local regions using feature vectors which are invariant to rotation and illumination changes. The discriminative power of SIFT and SURF has been validated in many computer vision techniques, and variants of these methods are widely used for robust image representation.

Two other important factors pertaining to invariant features are time and space efficiency levels when detecting, matching, and storing features. Recently, demand has increased for such efficient image features, as mobile and embedded vision systems are emerging for visual searches [3] and for direct 2D to 3D matching [4, 5]. Also, for mobile visual search applications, the amount of data sent over the network needs to be as small as possible so as to reduce latency and lower costs. Several binary features,

*The first and the second authors provided equal contributions to this work.

such as BRIEF [6], ORB [7], BRISK [8] have been developed to describe local image regions with small binary strings which can be matched much faster with the Hamming distance compared to SIFT. However, despite the effort and advances in this area, SIFT has remained the best option for various deformation tasks apart from non-geometric transforms [9].

In this paper, we aim to extract binary features with a method that can achieve matching performance levels comparable to those of SIFT and SURF with even less storage than that required for existing binary features. We apply FAST-like binary tests [10] to reject non-feature regions quickly and present an efficient approximation of the Determinant of the Hessian for robust feature detection with high repeatability. Motivated by earlier work [11], we employ ordinal descriptions of local image measurements for robust representations of feature regions. The ordinal description encodes the rank order of each measurement and is therefore invariant to monotonic deformations of the measurements. Also, an ordinal description is insensitive to moderate rank-order errors, thus enabling the quantization of descriptions into small-sized binary descriptors without a noticeable degradation in the performance. Experimental results show that our feature outperforms other state-of-the-art binary features with fewer dimensional descriptors in terms of repeatability and matching performance.

## 2 Related Work

### 2.1 Feature Detection

The first stage of image feature extraction is to detect interest points, which are known as keypoints. Many feature extractors detect blobs or corners as keypoints because they can be repeatedly detected despite the presence of various geometric and photometric deformations. SIFT [1] convolves images with Gaussian filters at different scales and approximates the Laplacian of the Gaussian using the Difference of Gaussians (DoG) method. SIFT then detects blob-like areas as keypoints by taking the maxima and minima of the Difference of Gaussians (DoG). Scale invariance is obtained from the scale of DoG. Instead of using DoG, SURF [2] uses an approximated Determinant of Hessian measure via box filters which are implemented efficiently using an integral image. Harris corner [12] is the best known corner detector; it uses the second moment matrix, also known as the auto-correlation matrix. Harris-Affine [13] introduces a multi-scale version of Harris corner. FAST [14, 10] is one of the fastest keypoint detectors. FAST is considered as a modification of SUSAN [15], demonstrating that a simple segment test is enough to detect corner-like areas. AGAST [16] improves FAST with an adaptive and generic accelerated segment test. CensurE [17] introduces a scale-invariant center-surround detector. A simplified center-surround filter with an integral image is used for efficiency, and non-features are removed by a Harris measure. FRIF [18] is a fast approximated LoG (FALoG) detector based on the Harris matrix. FALoG can be quickly computed by means of factorization with an integral image while preserving the properties of LoG. BRISK [8] and ORB [7] use multi-scale FAST for scale-invariance and efficiency. Inspired by SURF and FAST, we use a simple sampling pattern to measure and classify features and non-features.

### 2.2 Feature Description

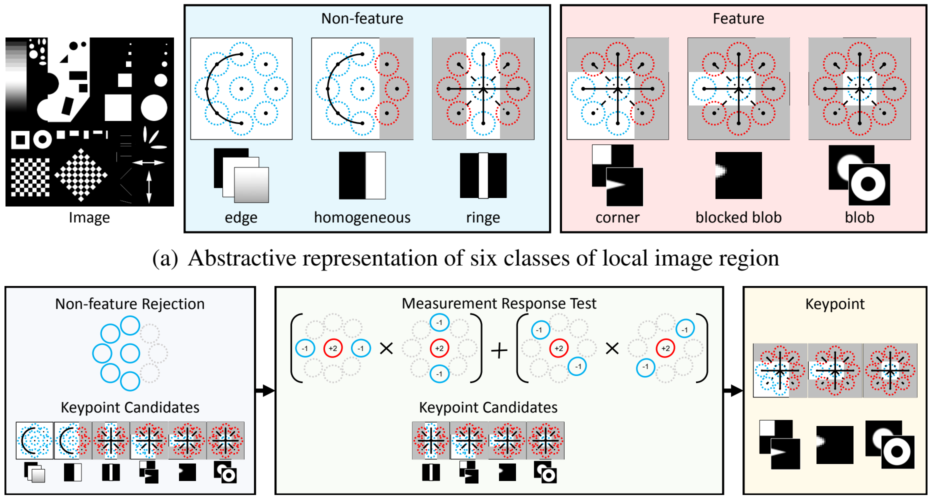
A descriptor encodes the local image information around a keypoint. It is used to distinguish keypoints under various transformations. SIFT descriptor describes a local image patch using a gradient histogram. It quantizes gradient orientations and accumulates gradient magnitudes into an orientation histogram with eight bins over 4x4 sub-regions. The form of the SURF descriptor is similar to that of the SIFT descriptor. Instead of quantizing gradient orientations, the SURF descriptor computes a histogram with four bins, $dx$, $dy$, $|dx|$, and $|dy|$. There are many SIFT variants, such as GLOH and DAISY. GLOH [19] uses a polar arrangement of sub-regions and DAISY [20] uses a flower-like spatial division for dense descriptions.

Because gradient-based descriptors can only deal with linear deformations, rank-order-based methods have been proposed to handle more general non-linear deformations. Rank-order-based methods such as SIFT-Rank [11] and LUCID [21] encode *relative* order information rather than raw values such as the gradient and intensity. Wang et al. [22] propose a Local Intensity Order Pattern (LIOP) to encode the local ordinal information and create a histogram of the LIOP for each ordinal sub-region. Motivated by these methods, we introduce a new binary descriptor using ordinal information.

In the computer-vision community, recent progress has shown that a simple brightness comparison test is a good choice when attempting to generate a robust binary descriptor. BRIEF [6] presents a binary feature using an intensity difference test and demonstrates a high recognition rate with low computational complexity during the feature construction and matching processes, though it is not designed to be rotationally invariant. BRISK [8] is a combination of the scale-normalized FAST keypoint detector and the BRISK descriptor. BRISK divides point pairs into two groups: long-distance pairs and short-distance pairs. It calculates the characteristic pattern direction using long-distance pairs and computes the descriptor using intensity comparisons of short-distance pairs after rotation- and scale-normalization. ORB [7] demonstrates that the steered BRIEF loses discriminancy from rotation-normalization and introduces rBRIEF, which uses a learning strategy to recover from the loss of variance in steered BRIEF. The rBRIEF method demonstrates variance and correlation improvements over the steered BRIEF. Inspired by the human visual system, FREAK [23] uses the learning strategy of ORB with a DAISY-like sampling pattern [20]. For our feature description, we apply a similar sampling pattern with BRISK, but we generate a binary string using a different strategy based on the rank order.

## 3 Keypoint detector

For fast keypoint detection, previous binary features [6–8] adopt the FAST detector [10] with modifications such as the scale-space feature and/or a Harris filter to obtain additional scale and rotation invariances. While these methods obtain very fast detection results, they have relatively low repeatability levels compared to SIFT and SURF, as will be shown in Sec. 5.1. Repeatability is one of the most important properties of a keypoint detector for localization accuracy.




(a) Abstractive representation of six classes of local image region

(b) Pipeline for keypoint detection

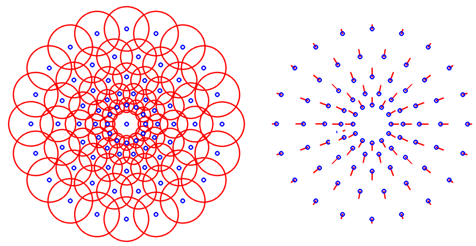
**Fig. 1.** The proposed feature detection algorithm.

We introduce a new keypoint detection algorithm in an effort to improve the repeatability. We apply a FAST-like binary comparison test and propose a keypoint measure to balance computational efficiency and repeatability performance.

We initially analyze the tendencies of image patches and categorize these into six types, *homogeneous*, *edge*, *ridge*, *corner*, *blob*, and *blocked blob*. Abstractive representations of the six types of local image regions are given in Fig. 1 (a). For repeatable keypoint detection robust to image deformations, we detect corner and blob regions as keypoints and reject homogeneous, edge, and ridge regions as non-features. Therefore, we consider the *homogeneous*, *edge*, and *ridge* types as non-feature regions and the *corner*, *blob*, and *blocked blob* types as feature regions.

The overall pipeline for our keypoint detection is depicted in Fig. 1 (b). For a given local image patch, we compute the differences between the intensities of a center and surrounding points and assign the *similar* label when the difference is under a certain threshold, assigning the *dissimilar* label otherwise. If more than five surrounding points have *similar* labels, the region is classified as a non-feature region because the *homogeneous* and *edge* types apply in this case. Classification between other categories also may be possible with additional rule-based comparison tests. However, we use another measure for classification in this case instead of adding more rules for robust and repeatable keypoint detections.

When we compute the difference between intensity levels on a central and a surrounding point, large differences arise along a single axis in *ridge*-type regions. For *corner*- and *blob*-type regions, large differences arise along two perpendicular directions ($+$, $\times$). We use this characteristic as a keypoint measure and formulate an equation to have high response values for features and low response values for non-features.



(a) (b)

**Fig. 2.** Sample pattern of our descriptor with N = 80 points. (a) The blue dots indicate sampling points, the red circles show the radii corresponding to the standard deviation of a box kernel that is used to smooth intensity values around sampling points. (b) The red lines denote the predefined local directional vector at each sampling point and those are used for estimating dominant orientation.

The proposed keypoint measure $\mu$ is defined as

$$\mu = (2I_C - I_L - I_R)(2I_C - I_T - I_B) + (2I_C - I_{TL} - I_{BR})(2I_C - I_{TR} - I_{BL}), \quad (1)$$

where $I_{C,L,R,T,B}$ represents the intensities of the center, left, right, top, and bottom locations. This measure can be seen as an approximation of the Determinant of Hessian (DoH). Here, $(2I_C - I_L - I_R)$ denotes the second-order partial derivatives in the x direction and $(2I_C - I_T - I_B)$ denotes second-order partial derivatives in the y direction. Similarly, $(2I_C - I_{TL} - I_{BR})$ and $(2I_C - I_{TR} - I_{BL})$ represent the second-order partial derivative in the xy direction. The keypoint measure $\mu$ has a large response on corner or blob regions; therefore, we finally classify locations as keypoints when there are large keypoint responses.

In practice, we construct a scale-space for scale invariance and sample intensities from one center and eight surrounding points. The surrounding points are centrally symmetric and equidistant from the center point, as depicted in Fig. 1.

## 4 Descriptor

In this section, we present a new binary descriptor using the intensity order. Given a keypoint location with a scale, we extract pattern intensities from a sampling pattern and determine the dominant orientation for rotation invariance. Then, we establish the rank order of the pattern intensities and binarize it to produce a binary descriptor.

### 4.1 Sampling pattern

To generate the descriptor, we use a sampling pattern, as shown in Fig. 2 (a). As in previous methods, we sample intensities from a spatial division of the polar form. We assign more sampling points to the center than to the surrounding areas according to the retina model of the human visual system, which has a higher cell density at the center than in the surrounding areas [23]. The central part is less affected than the surrounding areas when there are geometric transformations; therefore, it is beneficial to assign more points to the center. The sampling pattern consists of one center point and 79 surrounding points around the center.

We sample the pattern intensity from each sampling point with spatial smoothing. Spatial smoothing is applied to prevent aliasing, and the scale of the smoothing region is determined according to the distances between adjacent sampling points. In Fig. 2 (a), the blue dots indicate the sampling points and the red circles represent the radius corresponding to the standard deviation of the smoothing kernels.

### 4.2 Dominant orientation

For rotation invariance, we estimate the dominant orientation of a keypoint and construct a descriptor vector along the dominant orientation. The dominant orientation is determined by the weighted average of the local directional vectors. Each local directional vector is oriented toward the center point from a sampling point, and its magnitude set such that it is inversely proportional to the distance from the sampling points and to the center point. Fig. 2 (b) shows the local directional vectors. The red lines show the directions of the local vectors, and the lengths of these lines indicate the magnitudes of the vectors.

The weight of each local directional vector is assigned as the difference between its pattern intensity and the median of the pattern intensities. With the pattern intensities and predefined local directional vectors, the dominant orientation $\theta$ of a keypoint is computed as

$$\theta = \arctan \sum_{i=1}^{N} |I_i - M| \frac{dy_i}{dx_i} \quad \text{s.t.} \quad M = \underset{i \in \{1,\ldots,N\}}{\text{median}} (I_i), \tag{2}$$

where $N$ is the total number of sampling points, $I_i$ represents an $i^{\text{th}}$ pattern intensity, $M$ is the median of pattern intensities, and $(dx, dy)$ indicates a local directional vector.

### 4.3 Binary descriptor

We employ an ordinal description of pattern intensities. The ordinal description for an invariant feature was introduced in earlier work [11]. It describes each measurement using its rank order with sorted measurement values. Employing the ordinal description technique is shown to have strong discriminative power and to be invariant to monotonic deformations of embedding measurements. While the ordinal description has good invariance characteristics for many deformations, it is not designed for use with binary descriptors. The binary descriptor is of greater importance due to its compact storage size and fast matching performance. To take advantage of both approaches, we present an ordinal binary descriptor.

After determining the dominant orientation, we obtain a measurement vector by tracing the pattern intensities aligned with the dominant orientation. Then, we transform the measurement vector into a rank-order vector by computing the rank of each element value in the measurement vector. Our descriptor is formed by binarizing the rank-order

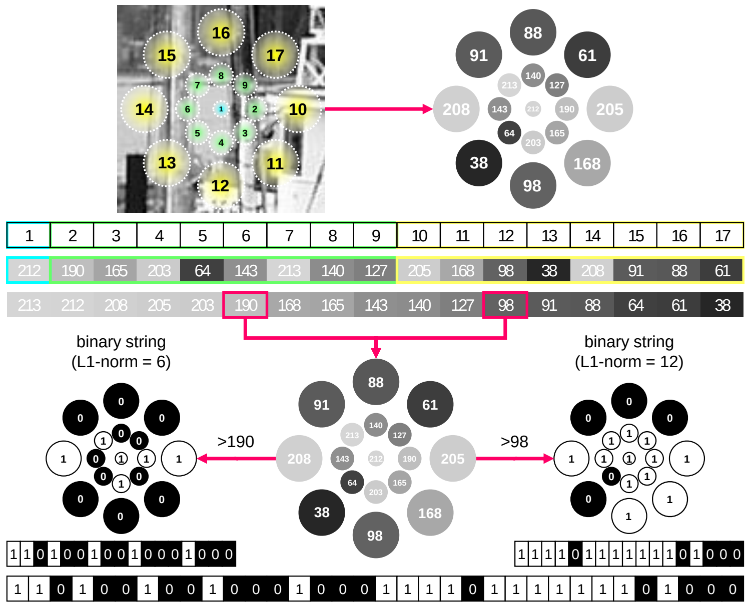



**Fig. 3.** The overall process of our binary description, which is explained in Sec. 4.3.

vector. The binarization process is performed by means of a binary comparison test of a certain threshold rank. Our binary descriptor $D$ with the binary comparison test is denoted as

$$D = \sum_{j=1}^{k} \sum_{i=1}^{N} 2^{N(j-1)} 2^{i-1} b_i, \quad \text{s.t.} \quad b_i = \begin{cases} 1, & r_i \geq T_j \\ 0, & otherwise, \end{cases} \tag{3}$$

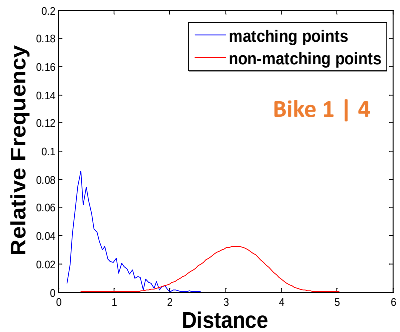
where $N$ is the total number of sampling points, $k$ is the number of threshold values, $T_j$ represents a $j^{\text{th}}$ threshold rank, and $r_i$ denotes the $i^{\text{th}}$ element of the rank-order vector. Given $N$ and $k$, we determine the threshold rank $T$ as

$$T_j = \frac{j}{k+1} N, \quad \text{s.t.} \quad j \in \{1, ..., k\}. \tag{4}$$

We set the number of threshold values $k$ to 2 throughout the paper; therefore, our descriptor becomes a 160-dimensional binary descriptor with 80 sampling points. The overall process of our binary description is illustrated in Fig. 3.

Our simple binarization process experiences reduced coding efficiency compared to the direct quantization of the rank order into $2^k$ ranks. However, the important property of our binarization process is that we can utilize the Hamming distance to compare encoded rank orders, which is one order of magnitude faster than the Euclidean distance.

Ordinal descriptions are insensitive to moderate rank-order errors, which enables us to quantize rank-order descriptions into binary descriptors without a noticeable degradation in the performance. Fig. 4 shows a comparison of an intensity-based ordinal



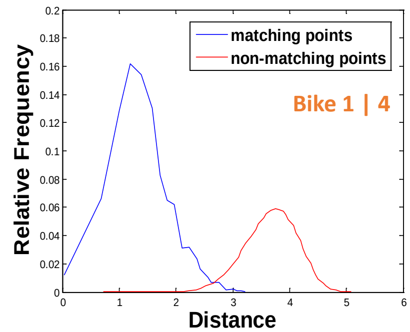




**Fig. 4.** Distance distribution of matching pairs (blue lines) and non-matching pairs (red lines). The Bikes dataset in Fig. 5 (c) is used in this experiment. (a) Intensity-based ordinal descriptor (b) Our descriptor ($k = 2$)

descriptor and our quantized binary descriptor. For this experiment, the intensity-based ordinal descriptor directly employs the rank orders as a descriptor, while our descriptor uses the binarized version of the intensity-based ordinal descriptor according to Eq. (3) with $k = 2$. The distributions of the two descriptors are very similar, indicating that our descriptor retains the discriminative power of the intensity-based ordinal descriptor even after binarization with $k = 2$.

Also, as illustrated in Sec. 5, we achieve good performance only with two threshold values, which shows that the reduction in the coding efficiency is very small while the gain in the computational efficiency is considerable.

### 4.4 Comparison to other descriptors

Our descriptor is closely related to ordinal descriptors [11, 21, 22] and binary descriptors [6, 8, 7, 23] based on a brightness comparison test. For clarity, we present a description of the similarities and the differences between the proposed method and two category descriptors.

**Ordinal Descriptor** The SIFT-Rank and the proposed descriptor are similar in terms of their use of ordinal information to take advantage of ordinal descriptions which are invariant to any monotonic transformations of the raw measurements. SIFT-Rank encodes ordinal information from raw SIFT descriptor values and the proposed descriptor utilizes the rank of the pattern intensities to describe a keypoint. The difference between the two methods is as follows. SIFT-Rank utilizes post-processing of the gradient-based SIFT descriptor and is therefore not designed to improve the time and storage efficiency. Also, SIFT-Rank uses a rank-order vector as a descriptor directly; thus, it may require specialized matching metrics. On the other hand, the proposed descriptor is an independent descriptor that uses the rank of the pattern intensities. It is also a binary descriptor with an elaborate binarization method. This allows the proposed descriptor to have very efficient time performance with low memory usage.

**Binary descriptor** The similarity between existing binary descriptors and the proposed descriptor is the use of a brightness comparison test to binarize a descriptor. The difference between them lies in the method used to select point pairs for the binary test. In existing binary descriptors, those point pairs are fixed to all keypoints. BRIEF chooses point pairs randomly or it depends on a certain distribution. BRISK uses the distance between point pairs as a condition and selects some from all possible point pairs. ORB selects point pairs according to decreases in correlations and increases in the degree of variation. Instead of predetermining point pairs for the binary test, our method implicitly selects point pairs based on an analysis of the pattern intensity distribution.

In the next section, it will be shown through experiments that our method shows better or comparable results relative to those of existing methods with fewer bits.

## 5 Experiments

To validate the performance of the proposed feature detector and descriptor, we conduct experiments using publicly available evaluation toolkits and datasets. First, we give a brief description of the experimental configuration, after which we present the experimental results pertaining to the detector and descriptor evaluations. In this evaluation, we mainly compare our method to methods that proposed both a detector and a descriptor simultaneously.

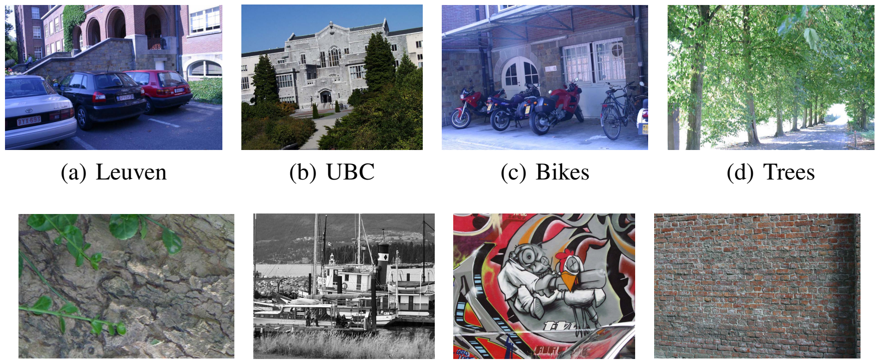
*Evaluation toolkits and Dataset* Our method was mainly tested using the evaluation toolkits proposed by Mikolajczyk *et al.* [24], Mikolajczyk and Schmid [19], and the OpenCV-Features-Comparison toolkits (CVT)[1]. The evaluation codes are available on the authors' webpage. With the evaluation toolkits, we perform the evaluations using the dataset used in earlier studies [24, 19]. There are eight datasets with different geometric and photometric transformations. These transformations include brightness changes (Leuven), JPEG compression (UBC), blur (bikes and tress), zoom and rotation (bark and boat), and view-point changes (graffiti and wall). Each set consists of six images with gradually increasing levels of transformation. The dataset provides homographies between the first and the other images, and the homographies are used to estimate ground-truth matches. Fig. 5 shows sample images of the dataset.

*Experimental Settings* We compare our method to the state-of-the-art keypoint detectors and descriptors of SIFT, SURF, STAR, ORB, and BRISK. STAR uses only a keypoint detector while the other methods use both detectors and descriptors. All of the compared methods are implemented in OpenCV 2.4.5, and we use the library with default parameters, except for ORB. For ORB, we set the number of features adaptively to have it extract the same number of features used by our method. We present more detailed information about each method in Table 1.

### 5.1 Repeatability performance

Mikolajczyk *et al.* [24] proposed the concept of *repeatability* to evaluate the performance levels of keypoint detectors. Repeatability measures how much the keypoints

[1] http://computer-vision-talks.com/2011/08/feature-descriptor-comparison-report/



(a) Leuven (b) UBC (c) Bikes (d) Trees

(e) Bark (f) Boat (g) Graffiti (h) Wall

**Fig. 5.** Example images of Mikolajczyk and Schmid's dataset used for evaluation: brightness (Leuven), JPEG compression (UBC), blur (Bikes, Trees), zoom and rotation (Bark, Boat), viewpoint change (Graffiti, Wall).

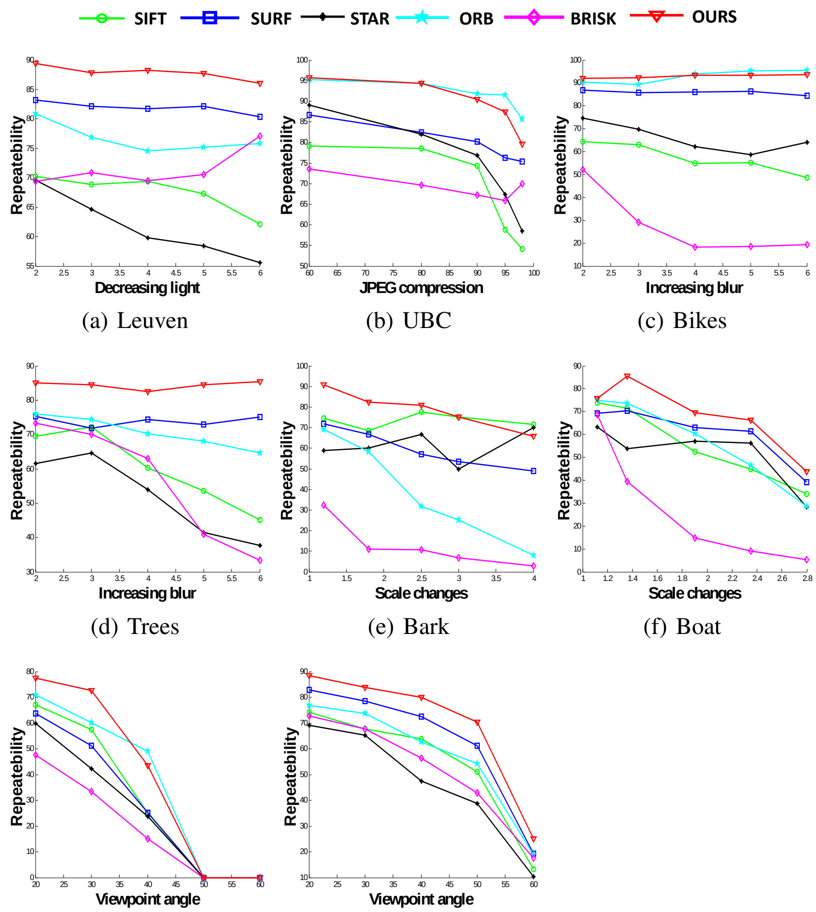
detected from two images overlap the same regions. It is a desirable property for invariant local features, as a high degree of repeatability refers to the robustness of a keypoint detector under various transformations.

We evaluate the performance of the detectors according to a method used in earlier work [24]. Fig. 6 shows the evaluation results pertaining to the repeatability of each detector under various transformations. SURF shows high repeatability over all images. For binary features, BRISK has relatively low repeatability and ORB has different appearances depending on the dataset. Both binary features commonly have low repeatability for scale changes, as they are based on multi-scale FAST, which is oriented for fast keypoint detections. The proposed method demonstrates repeatability performance similar to that of SURF because both methods detect keypoints using approximated DoH measures.

### 5.2 Matching performance

We evaluate the matching performance using Mikolajczyk and Schmids evaluation toolkit [19] and the CVT toolkit. Their evaluation toolkit [19] is publicly available and evaluates descriptors by means of a precision-recall approach. The CVT toolkit evaluates the performances of descriptors using synthetically transformed images. This evaluation tool provides brightness, blur, rotation, and scale changes as the transformations.

**Precision-Recall** Fig. 7 shows the evaluation results using the aforementioned descriptor evaluation toolkit [19]. Both the SIFT and the proposed methods outperform the other methods in general. Specifically, SIFT outperforms other methods in terms of scale and rotation transformations, and our method shows good performance with transformations of brightness changes, blurring, and JPEG images.



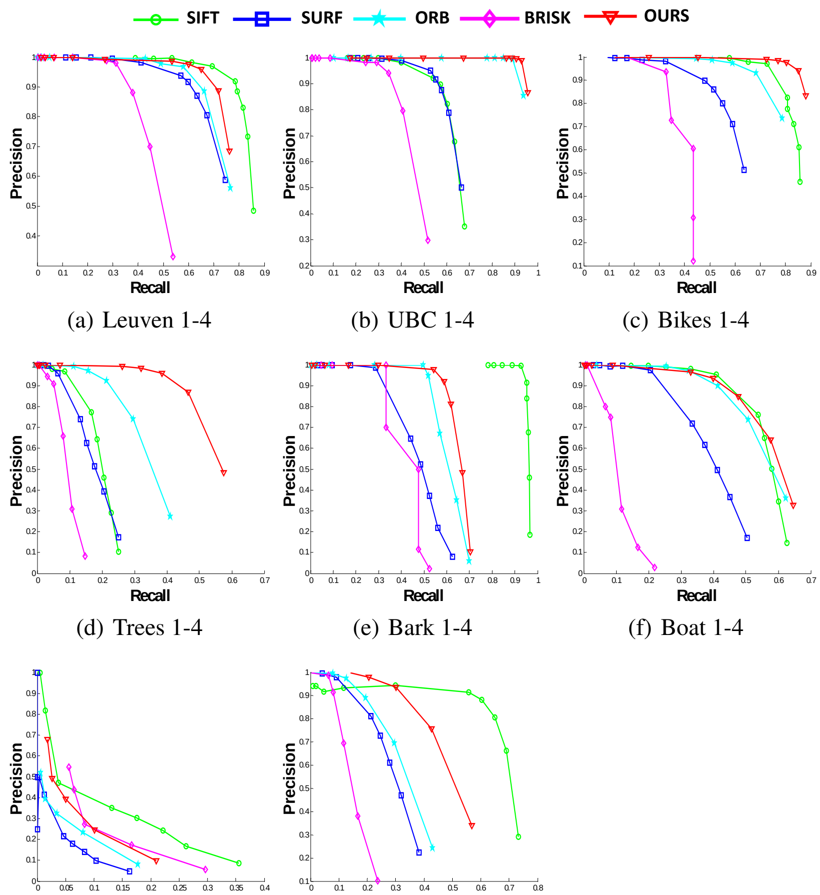

(a) Leuven (b) UBC (c) Bikes

(d) Trees (e) Bark (f) Boat

(g) Graffiti (h) Wall

**Fig. 6.** Repeatability of keypoint detection.

**CVT toolkit** As the second matching experiment, the CVT evaluation toolkit is used and the degree of transformation is finely adjusted for the evaluation. Fig. 8 shows the performance with each level of transformation. SIFT generally shows good performance, as shown in the previous precision-recall test. SURF shows matching performance that follows SIFT. ORB shows relatively strong performance compared to BRISK; however, both methods do not perform as well as SURF or SIFT. Our method specifically shows robustness to brightness changes and Gaussian blurring. Our method approaches state-of-the-art performance levels with relatively few bits.




(a) Leuven 1-4 (b) UBC 1-4 (c) Bikes 1-4

(d) Trees 1-4 (e) Bark 1-4 (f) Boat 1-4

(g) Graffiti 1-3 (h) Wall 1-4

**Fig. 7.** Precision-recall performance.

### 5.3 Storage and time

All experiments are performed using an Intel i7 3.4GHz processor without multi-core parallelism. Table 2 summarizes the computation time and storage amounts for feature extractions, descriptions, and matching using the Boat dataset. To measure the computation time, we run 100 trials for each task and average them after dropping the best 10% and worst 10% of the results. For matching, we use a brute-force method.

The storage of the proposed descriptor is the lowest among the compared methods. SIFT and SURF are 128- and 64-dimensional floating descriptors which require 512 and 256 bytes, respectively. ORB and BRISK require 64 bytes (512 bits), while our method requires only 20 bytes (160 bits).

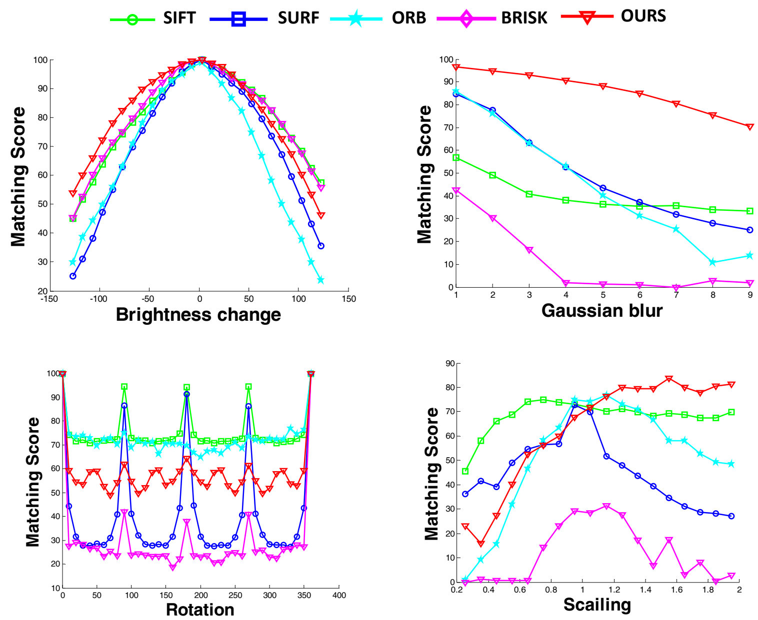



**Fig. 8.** Matching performance under four kinds of transformations (Brightness change, Gaussian blur, Rotation and Scaling)

Though our method is not fully optimized, it is nonetheless one order of magnitude faster than SIFT and SURF in terms of feature extraction and is comparable to ORB and BRISK in a trade-off between detection time efficiency and the repeatability gain. Distance computation times between the descriptors are directly influenced by the descriptor length; therefore, the use of a short descriptor has advantages in terms of the matching time and the amount of required storage. Also, because our descriptor is a binary descriptor consisting of 0s and 1s, we can use a bitwise XOR operator to compare two descriptors with the Hamming distance, which is much more efficient than the Euclidean distance.

## 6 Conclusions

In this paper, we presented a robust binary feature using the intensity order. We achieved better detection results than other binary features in terms of repeatability. For robust feature descriptions, we employ an ordinal description which is invariant to monotonic transformations. We also presented a binarization method which encodes the intensity order into a binary descriptor, which enables us to take advantage of better storage and computational efficiency. We evaluated the proposed binary feature with various experiments and demonstrated that our feature shows performance analogous to that of SIFT and that it outperforms other binary features under various transformations with much less storage use for feature descriptions.

| | **SIFT** | **SURF** | **ORB** | **BRISK** | **Ours** |
|---|---|---|---|---|---|
| non-feature removal | edge | - | FAST | FAST | FAST-variant |
| detector measure | DoG | DoH | Harris | FAST | DoH-variant |
| sampling pattern | grid | grid | polar | polar | polar |
| dimensions | 128 | 64 | 256 | 256 | 160 |
| storage/dimension | 4bytes | 4bytes | 1bit | 1bit | 1bit |
| feature information | gradient | gradient | intensity | intensity | order of intensity |

**Table 1.** Feature detectors used for evaluation. Information about each feature detector is summarized.

| | **SIFT** | **SURF** | **ORB** | **BRISK** | **Ours** |
|---|---|---|---|---|---|
| storage/keypoint [bytes] | 512 | 256 | 64 | 64 | 20 |
| # of keypoints | 8802 | 6752 | 5682 | 2442 | 5682 |
| detection time [ms] | 221 | 278 | 19 | 27 | 67 |
| description time [ms] | 506 | 961 | 16 | 12 | 38 |
| total extraction time [ms] | 727 | 1239 | 35 | 43 | 105 |
| description time/keypoint [ns] | 57.5 | 142.3 | 2.8 | 4.9 | 6.7 |
| total time/keypoint [ns] | 82.6 | 183.5 | 6.16 | 17.6 | 18.4 |

**Table 2.** Computation time and storage. The table represents storage, feature detection, and description time. Time is measured using the first images in the Boat dataset. SIFT shows relatively faster than SURF since SIFT is delicately optimized in the OpenCV 2.4.5 implementation.

## Acknowledgement

We would like to thank Jungho Kim and Jiyoung Jung for their support. This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (No. 2010-0028680).

## References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal on Computer Vision (IJCV) **60** (2004) 91–110
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Computer Vision and Image Understanding (CVIU) **110** (2008) 346–359
3. Chandrasekhar, V., Takacs, G., Chen, D.M., Tsai, S.S., Reznik, Y.A., Grzeszczuk, R., Girod, B.: Compressed histogram of gradients: A low-bitrate descriptor. International Journal on Computer Vision (IJCV) **96** (2012) 384–399
4. Sattler, T., Leibe, B., Kobbelt, L.: Fast image-based localization using direct 2d-to-3d matching. In: Proceedings of International Conference on Computer Vision (ICCV). (2011) 667–674
5. Li, Y., Snavely, N., Huttenlocher, D., Fua, P.: Worldwide pose estimation using 3d point clouds. In: Proceedings of European Conference on Computer Vision (ECCV). (2012) 15–29

6. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: Brief: Binary robust independent elementary features. In: Proceedings of European Conference on Computer Vision (ECCV). Volume 6314. (2010) 778–792
7. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: Orb: An efficient alternative to sift or surf. In: Proceedings of International Conference on Computer Vision (ICCV). (2011) 2564–2571
8. Leutenegger, S., Chli, M., Siegwart, R.: Brisk: Binary robust invariant scalable keypoints. In: Proceedings of International Conference on Computer Vision (ICCV). (2011) 2548–2555
9. Heinly, J., Dunn, E., Frahm, J.M.: Comparative evaluation of binary features. In: Proceedings of European Conference on Computer Vision (ECCV). (2012) 759–773
10. Rosten, E., Porter, R., Drummond, T.: Faster and better: A machine learning approach to corner detection. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **32** (2010) 105–119
11. Toews, M., III, W.M.W.: Sift-rank: Ordinal description for invariant feature correspondence. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2009) 172–177
12. Harris, C., Stephens, M.: A Combined Corner and Edge Detection. In: Proceedings of The Fourth Alvey Vision Conference. (1988) 147–151
13. Mikolajczyk, K., Schmid, C.: Scale and affine invariant interest point detectors. International Journal on Computer Vision (IJCV) **60** (2004) 63–86
14. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: Proceedings of European Conference on Computer Vision (ECCV). (2006) 430–443
15. Smith, S.M., Brady, J.M.: SUSAN-A New Approach to Low Level Image Processing. International Journal on Computer Vision (IJCV) **23** (1997) 45–78
16. Mair, E., Hager, G.D., Burschka, D., Suppa, M., Hirzinger, G.: Adaptive and generic corner detection based on the accelerated segment test. In: Proceedings of European Conference on Computer Vision (ECCV). (2010) 183–196
17. Agrawal, M., Konolige, K., Blas, M.R.: Censure: Center surround extremas for realtime feature detection and matching. In: Proceedings of European Conference on Computer Vision (ECCV). (2008) 102–115
18. Zhenhua Wang, Bin Fan, F.W.: Frif:fast robust invariant feature. In: Proceedings of British Machine Vision Conference (BMVC). (2013)
19. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **27** (2005) 1615–1630
20. Tola, E., Lepetit, V., Fua, P.: Daisy: An efficient dense descriptor applied to wide-baseline stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **32** (2010) 815–830
21. Ziegler, A., Christiansen, E.M., Kriegman, D.J., Belongie, S.J.: Locally uniform comparison image descriptor. In: Neural Information Processing Systems (NIPS). (2012) 1–9
22. Wang, Z., Fan, B., Wu, F.: Local intensity order pattern for feature description. In: Proceedings of International Conference on Computer Vision (ICCV). (2011) 603–610
23. Alahi, A., Ortiz, R., Vandergheynst, P.: Freak: Fast retina keypoint. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2012) 510–517
24. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Gool, L.V.: A comparison of affine region detectors. International Journal on Computer Vision (IJCV) **65** (2005) 43–72