



HAL
open science

Comparative Study of Descriptors with Dense Key points

Hermine Chatoux, François Lecellier, Christine Fernandez-Maloigne

► **To cite this version:**

Hermine Chatoux, François Lecellier, Christine Fernandez-Maloigne. Comparative Study of Descriptors with Dense Key points. 23rd International Conference on Pattern Recognition, Dec 2016, Cancun, Mexico. hal-01461562

HAL Id: hal-01461562

<https://hal.science/hal-01461562>

Submitted on 6 Mar 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Comparative Study of Descriptors with Dense Key points

Hermine Chatoux, François Lecellier and Christine Fernandez-Maloigne

Xlim, University of Poitiers, CNRS, France
Email: firstname.lastname@univ-poitiers.fr

Abstract—A great deal of features detectors and descriptors are proposed every years for several computer vision applications. In this paper, we concentrate on dense detector applied to different descriptors. Eight descriptors are compared, three from gradient based family (SIFT, SURF, DAISY), others from binary category (BRIEF, ORB, BRISK, FREAK and LATCH). These descriptors are created and defined with certain invariance properties. We want to verify their invariances with various geometric and photometric transformations, varying one at a time. Deformations are computed from an original image. Descriptors are tested on five transformations: scale, rotation, viewpoint, illumination plus reflection. Overall, descriptors display the right invariances. This paper’s objective is to establish a reproducible protocol to test descriptors invariances.

I. INTRODUCTION

Over the last two decades, several descriptors were created with different purposes: wide base line matching [1], [2], object recognition [3], [4], image retrieval [5], [6], [7], texture recognition [8], [9] or shape matching [10], [11].

To sort all these new techniques, Mikolajczyk *et al.* proposed a comparison method for detectors and descriptors with [12], [13]. In these papers, authors introduce a new dataset with a human created ground truth. To compare descriptors [13], they count the number of correct matches over the total number of matches. One of the tested descriptors, SIFT developed by Lowe in [14], [15], is based on gradient location and orientation histograms. In spite of its efficiency, SIFT remains relatively slow to compute, so, attempts were made to improve it, such as Ke and Sukthankar with PCA-SIFT [5] which select the descriptor’s strongest component. Another improvement, GLOH, introduced in [13], uses a log polar location instead of a linear one before PCA. The other descriptors presented in this paper were: invariant moments [16], shape context [10], spin images [8], steerable filters [17], differential invariant [18], and complex filters [1].

Misksik and Mikolajczyk updated these articles in [19], with others detectors and descriptors tested together to determine detector/descriptor dependences. They compare mostly binary descriptors that will be detailed in the next section [6], [7], [20]. For matching results, the best descriptors are LIOP, MRRID and MROGH proposed by Fan *et al.* in [21], [22]. These descriptors are rotation invariant through a watershed on the key point region.

Other comparison articles dedicated to a specified task were published as [23] specialised in visual tracking, a real time

task. Descriptors compared are SIFT, SURF [24] (introduced by Bay *et al.* relying upon sums of Haar wavelet), random trees [25] and ferns [26]. The last two descriptors belong to the binary descriptors family inspired from LBP, but binary comparisons are done sparsely. Difference between trees and ferns is a classification improving ferns’ time efficiency. These descriptors are tested with various detectors and algorithms on several databases.

Mandal in [27] compares six descriptors applied to face recognition with different distances. When applied to face identification, features are more important than distance, contrary to face recognition. Mainly, the best descriptor for facial application is BSIF created by Kannala and Rahtu in [28].

For class matching, Hietanen in [29] compares several binary descriptors and SIFT with different detectors, including a dense grid. SIFT performed better than other descriptors and dense grid responds very well.

This result leads us to test dense detector for different descriptors. Compared to others methods, our approach is not based on manual key points pairing. Thus descriptors will be applied on computationally transformed images. An image from database leads to several transformed images. Hence every pixel’s correspondence to original image is calculable.

The next section is dedicated to a detailed presentation of compared descriptors. The third one lays out the experimental protocol, images transformations and measuring results formulae. Fourth section introduces results. The last part is reserved for conclusion, perspective and acknowledgement.

II. DESCRIPTORS

Dense grid is chosen as detector, therefore we are only interested in the descriptor part of all detector/descriptors combination. We will present in this section all descriptors we intend to compare. Table I summarizes invariance assumed by each descriptors.

A. Gradient based descriptors

1) *SIFT*: As stated in introduction, SIFT was introduced by Lowe *et al.* in [14] and enhanced for scale invariance few years later [15]. During detection, SIFT estimates key points’ location, scale and orientation and uses these data to create the descriptor. SIFT, founded on a human vision behaviour, favours gradients and orientations to slight shift location in order to recognize objects. This method calculates gradients

TABLE I
DESCRIPTORS' INVARIANCE

	Scale	Rotation	Viewpoint	Lightness
SIFT	x	x	x	x
SURF	x	x	x	x
DAISY			x	x
BRIEF				x
ORB		x		x
BRISK	x	x		x
FREAK	x	x	x	x
LATCH		x		x

on key point's region which is divided into 4×4 sub-regions. From each one, orientation histograms are estimated. Descriptor is normalized using $L_2 - norm$ inducing illuminations invariance. Rotation and scale invariances are computed from detector's information.

2) *SURF*: SURF, proposed by Bay *et al.* in [24], is based on Haar wavelet transform to measure a gradient's approximation on a smoothed image. Before descriptor's calculation, SURF requires key points' location and scale. First, descriptor estimates interest point's orientation, then gradients' approximation (dx and dy) are estimated. Similarly to SIFT, the key point area is divided into sub-regions. For each one, $\sum dx$, $\sum dy$, $\sum |dx|$ and $\sum |dy|$ are calculated. Descriptor is finally normalized using $L_2 - norm$.

3) *DAISY*: To the best of our knowledge, DAISY, created by Tola in [2], is the only descriptor originally designed for dense utilisation. Consisting in histogram of oriented gradients, measures are made around key point with different orientations and radii. Then, different-orientation gradients are estimated and positives directions are kept. The descriptor is designed as a flower with overlapping areas. Each area's centre is convolved with a Gaussian kernel proportional to key point's distance. Hence, series of smoothed-gradient histograms are estimated. Each histogram is normalized independently to ensure lightness invariance.

The three previous descriptors use Euclidean distance to measure the difference between two key points. At the same time, another faster method appeared: binary descriptors.

B. Binary descriptors

The principle is to compare two pixels leading to a boolean response. These comparison are made locally around a key point. The five following descriptors rely on this method.

1) *BRIEF*: Binary Robust Independent Elementary Features presented by Calonder *et al.* in [6], has no associated detector. Based on random difference of pixels selected around a key point, description is made in two steps. First, image is blurred. Next, pixel's pairs (x, y) are randomly picked around a key point. Selection follows Gaussian distribution centred on the key point. Each pair of pixels are compared with:

$$b = \begin{cases} 1 & \text{if } p(x) < p(y) \\ 0 & \text{else} \end{cases} \quad (1)$$

The pixels difference has no influence on the descriptor calculation which ensures the invariance to contrast and brightness.

However, this descriptor is sensitive to rotation.

2) *ORB*: Oriented fast and Rotated Brief was introduced by Rublee in [7]. As the name indicates, the descriptor principle is the same as BRIEF but is oriented with the angle calculated in the FAST detection process. In our case, orientation is given by the transformation. To take orientation into account, a rotation matrix is applied to all pixel's pairs coordinates before computing binary comparison. To improve discrimination, an algorithm trained pairs to compare.

3) *BRISK*: proposed by Leutenegger in [20], Binary Robust Invariant Scalable Key points need an information on scale prior to description. Contrary to BRIEF, all pairs are fixed and uniformly extracted around key points. Pixel's smoothing size varies with the distance to the key point. The set of pairs is divided in two sets, \mathcal{L} and \mathcal{S} , corresponding to large and small distances between two elements of a pair, respectively. \mathcal{L} is used to determine orientation of key point:

$$\begin{pmatrix} g_x \\ g_y \end{pmatrix} = \frac{1}{L} \sum_{(p_i, p_j) \in \mathcal{L}} \frac{I(p_i, \sigma_i) - I(p_j, \sigma_j)}{\|p_i - p_j\|^2} \begin{pmatrix} x_i - x_j \\ y_i - y_j \end{pmatrix}, \quad (2)$$

where L is the cardinal of \mathcal{L} , p a pixel, (x, y) its coordinates and g the resulting vector of equation. From g , orientation is extracted. \mathcal{S} set elements location is modified by the angle estimated: \mathcal{S}_θ . The elements of \mathcal{S}_θ are compared to one another to obtain the descriptor.

4) *FREAK*: presented by Alahi in [3], Fast Retina Key point relies upon human retina and analysis perception. Similar to retina, the surrounding of the key point is separated in three areas, fovea, para foveal and peri foveal. Ganglion cells' number diminish with the distance to the centre, so applied Gaussian depends on area observed (Standard deviation grows with the distance). Pixels are uniformly selected on these areas. Orientation is computed with the same formula as BRISK but fewer pairs are picked and are chosen symmetrically. Results concurs with human processing vision, coarse to fine analysis.

5) *LATCH*: Learned Arrangements of Three Patch Codes is one of the latest descriptor published, proposed by Levi and Hassner in [4]. Gaussian kernels causes a loss of information for most binary descriptors. To solve the problem, LATCH uses a " 3×3 patches" for its binary comparison. Three patches are selected, one is classified as anchor, and Frobenius distances between anchor and the other two pixels are calculated. By comparing both distances, we obtain a binary element. The best patches triplets are learned, then selected to avoid high correlation.

To be able to compare descriptors, once the distances are calculated, we normalized the results using infinite norm for all binary descriptors. As they are constituted of binary elements, the distance to use is Hamming's counting different bits' numbers between two key points. All descriptors have been introduced, now we will detailed the transformations applied to initial image.

III. EXPERIMENTAL PROTOCOL

To estimate invariance of the previously detailed descriptors, all images from the photo CV KODAK base (768 × 512 sized images) are used. Descriptors used are only lightness, so photos are immediately transformed in grey level images. All images are positioned at the center of a squared image with zero-padding to keep it whole when it rotates. Descriptors are computed with *OpenCV*: SIFT, SURF, DAISY, BRIEF, ORB, BRISK, FREAK and LATCH using recommended parameters values except for key point size when necessary.

TABLE II
SUMMARY OF TRANSFORMATIONS CARRIED OUT

Transformation	Total number	From	To	Step
Scale	8	/1.5	/5	0.5
Rotation	35	10°	350°	10°
Viewpoint	16	10°	80°	10°
Lightness	15	0.2	3	0.2
Reflection	2			

Five types of transformations are applied: scale, rotation, shearing, lighting and reflection. The proposed transformations are recapitulated in Table II.

A. Scale

Image's size is divided by 1.5 to 5 with a step of 0.5. We do not perform image extension since this may lead to interpolation of pixels value. In this case, key points size is adapted accordingly on the transformed image.

B. Rotation

For the rotation transformation, we apply the following matrix for every angle between 10° and 350° with a step of 10°:

$$\begin{pmatrix} x_\theta \\ y_\theta \end{pmatrix} = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}, \quad (3)$$

where (x_θ, y_θ) and (x, y) are pixels coordinates and θ is the rotation angle. Origin coordinate is in the image's centre. To assess angle estimation impact on descriptor invariance, the measure is performed twice, one without any information on angle, the second with the known angle.

C. Viewpoint

Shearing projects the image as if rotated toward depth axis, with either left or bottom side fixed. The projection observer is set at the image's centre. As for the rotation the step is 10° from 10° to 80°. Transformations matrices are for vertical shearing:

$$\begin{pmatrix} x_\theta \\ y_\theta \\ p_\theta \end{pmatrix} = \begin{pmatrix} D \cos(\theta) & 0 & \frac{DCol(\cos(\theta)-1)}{2} \\ 0 & D & 0 \\ \sin(\theta) & 0 & D + \frac{Col \sin(\theta)}{2} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad (4)$$

and for horizontal:

$$\begin{pmatrix} x_\theta \\ y_\theta \\ p_\theta \end{pmatrix} \begin{pmatrix} D & 0 & 0 \\ 0 & D \cos(\theta) & \frac{DRow(\cos(\theta)-1)}{2} \\ \sin(\theta) & 0 & D + \frac{Row \sin(\theta)}{2} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad (5)$$

where p_θ is the depth of the resulting pixel. 3D coordinates are then projected to obtain a 2D image. θ stands for the angle toward the z axis, *Col* and *Row* the column and row of the original image size respectively, and D is the observer viewing distance. Here D is chosen for c to be the usual 80° of human vision. Note that matrices (4) and (5), as for rotation, is calculated for origin coordinate at the image's centre. As for scaling, key points are adapted to forbid overlapping. One may consider a rotation on key points but as it is not uniformed, angle is left unchanged.

D. Illumination

We want to measure invariance to illumination changes, so modifications are applied to pixels intensity and not on their position. We chose to make a linear change with saturation when intensity exceeds maximum. This transformation represents well what would happen if camera exposition is incorrectly set while taking pictures. Lightness coefficient varies from 0.2 to 3 with a 0.2 step. New grey level follows:

$$G = \begin{cases} \alpha G & \text{if } \alpha G < 1 \\ 1 & \text{else,} \end{cases} \quad (6)$$

where G is a grey level and α a linear coefficient to apply to the lightness transformation.

E. Reflection

This case is a bit different since we do not want this transformation to be invariant for our descriptors. One can easily see that horizontal or vertical flip may lead to some modification in the image interpretation (e.g. 2 and 5).

Cumulating these five alterations, 76 transformations are obtained. We choose not to measure recognition rate for blur and *jpeg* compression. In our field of experimentation and with digital-archiving improvement, we can assume our pictures to be necessarily sharp and uncompressed.

The next operation consists in estimating key points position for the transformed images. To perform this task, we densely create an x pixels sized grid on image. At each transformation, the new position of each key point is measured with the corresponding transformation matrix. Therefore, theoretically we should obtain the same descriptors for all transformations. Once key points are estimated for all transformations, descriptors are computed for each transformed image with the appropriated set of key points.

IV. RESULTS AND DISCUSSION

Each transformation is matched to the original image using nearest neighbour algorithm. Since no key points detection is done, the number of correspondences is identical to the number of matches. We then compute matching rate:

$$r = \frac{|\mathcal{T}|}{|\mathcal{C}|}, \quad (7)$$

where \mathcal{T} corresponds to the set of correct matches and \mathcal{C} is the set of possible correspondences.

In this experiment we also test different key points size to assess the more efficient one for invariance. Test results are only presented on scale invariance, but they lead to the same conclusion for others invariances.

A. Scale invariance

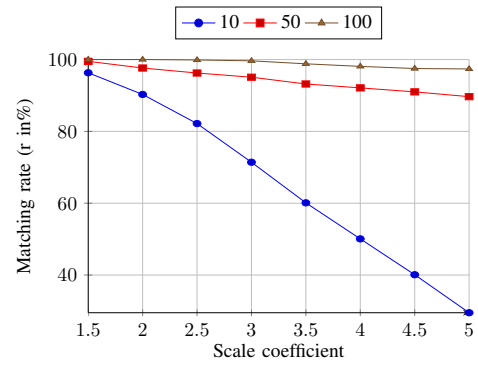
First experiment realised on scale invariance was to estimate key points size influence. Descriptors tendencies are summarized in the three graphics of Figure 1. One can easily see that for all descriptors, key points size influences results. The larger it is, the more information is stacked in descriptors, hence a better matching rate. Only three descriptors are represented on Figure 1 since all tendencies are summarized with these. In Figure 1a, SIFT and SURF results are similar and decrease regularly with key point size. BRIEF, ORB, DAISY and LATCH (Fig. 1b) decrease very quickly regardless the size. It is consistent with their non invariance to scale presented in Table I. Figure 1c displays BRISK's results, FREAK's are very close. For 10 pixels key points, results collapse and no correct match is made for a scaling superior to 2. This size is smaller than regions used to estimate key points inducing information overlap in the descriptor. However, for 50 and 100 pixels, the matching is almost perfect.

For the rest of the experiment we have used a 50 pixels diameter key point, to be consistent with the previous results. On Figure 3a, we plot matching rates of all descriptors. We confirm scale invariance for SIFT, SURF, BRISK and FREAK with advantage for the latter ones. Others descriptors are strongly influenced by scale.

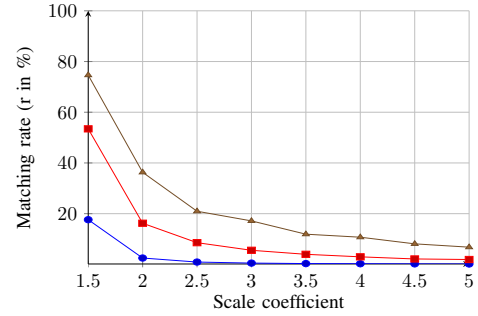
B. Rotation invariance

Results on rotation invariance are summarised in Figure 3b. The matching rates are computed with an *a priori* on rotation's angle. BRIEF and DAISY are the worst invariant descriptors, allowing no correct match for angles between 50° and 300° . SURF and FREAK measure the angle in the descriptor chain and not in the detector one, so pre-giving an angle has no effect on their results. On Figure 2, we display results with and without *a priori* for SIFT, ORB and LATCH since on these case, orientation is computed on the detection phase. We can clearly see that for SIFT, ORB and LATCH the descriptor needs information about the angle on rotation to keep their rotation invariance. Thus, when computing these descriptors, one must always provide an orientation information to ensure a correct description. This brought two issues: how to estimate a good orientation angle and does orientation need to be measured within the detector or descriptor part. In [21], [22], Fan *et al.* could provide an answer as their descriptors are free from angle by computing watershed on key point region.

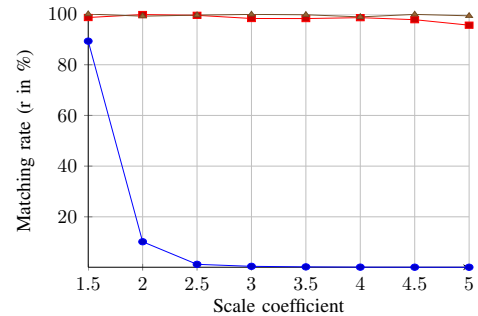
For almost all descriptors, some angle rotation has an effect on distance. Gradients directions are computed in histogram and Haar wavelets are summed, so SIFT and SURF are invariant to 90° rotation. For ORB, approximation error is caused by precomputed rotated pairs. Results are repetitive every 90° . We can see a symmetric period. The largest distance is reached



(a) SIFT



(b) BRIEF



(c) BRISK

Fig. 1. Comparison results on scale invariance with different key points size. All descriptors have been tested for different sizes of key points. All descriptors are sensitive to key points size. A larger patch, contains more information. Scale invariance diminish with the key point size in 1a. Despite large key point giving better results, 1b shows this descriptor is very sensitive to scale. In 1c, the descriptor loses its scale invariance when key point is too small.

every 90° plus 45° (for rotation invariant descriptors). All descriptors are consistent with results expected by Table I.

C. Viewpoint invariance

Vertical and Horizontal shear results are summarised in Figure 3c and 3d respectively and display similar results. The best results are obtained by FREAK which keeps its rate above 80 % until 60° . SIFT, BRISK and SURF are slightly worse but remain above 60 % on almost all angles. Since BRISK was not set to viewpoint invariance, it yields surprisingly good results almost as good as SIFT. The four other ones are decreasing rapidly and do not present shear invariance.

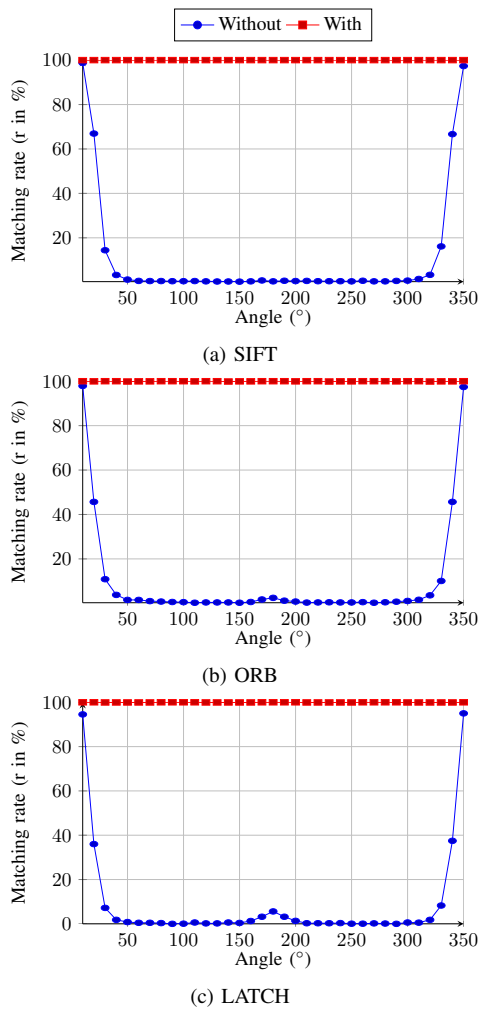


Fig. 2. **Comparison results on rotation invariance with and without *a priori*.** When the correct orientation is provided, matching rate is almost perfect for every angle. If this is not set, angles between 40° and 310° do not match any point.

D. Illumination invariance

Results on lightness are recapitulated in Figure 3e. Apart from DAISY which will be discussed later, all other descriptors display a decreasing matching rate as lightness coefficient pulls away from 1. Our method to change illumination leads to saturate areas when coefficient is larger than 1 which explain why curves decrease rapidly over 2. However, when change remains relatively small, all descriptors maintain matching rate higher than 80 %. DAISY gives poor results less than 60 % correct matching if factor is under 0.5 or over 1.5. Tola, in [2], mentioned that normalisation of each histogram separately could induce strange effects for other application than baseline matching.

E. Reflection invariance

DAISY and LATCH's matching rate differ a lot between horizontal and vertical flip. 20 % and 29 % for DAISY's horizontal and vertical respectively, 2 % and 6% for LATCH. Apart from DAISY and LATCH, all descriptors responds

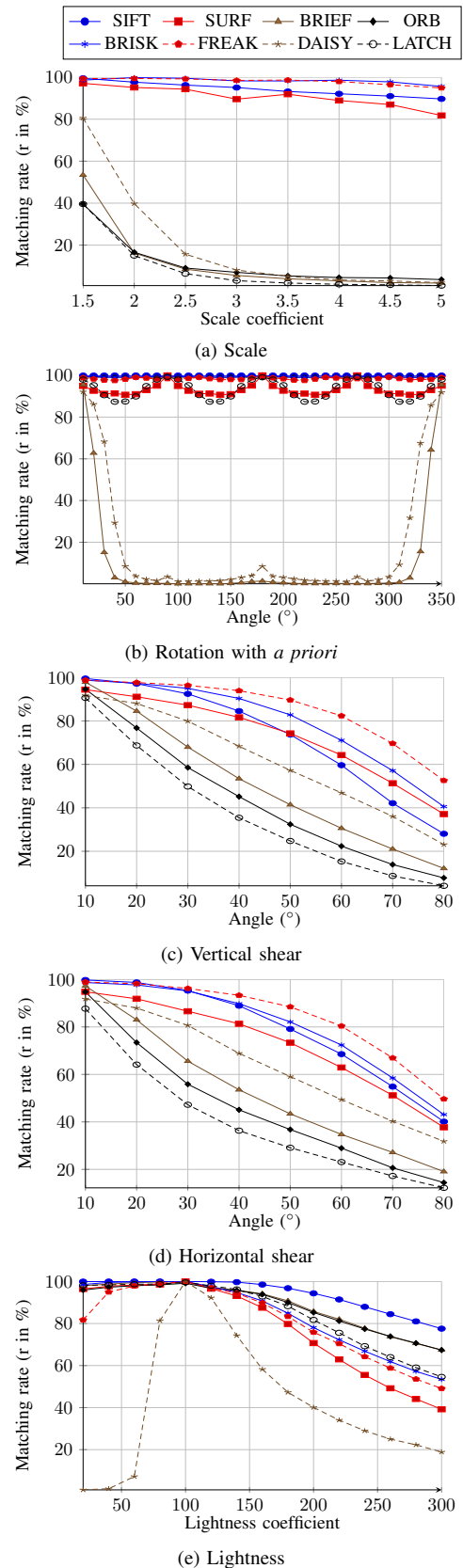


Fig. 3. **Comparison results.** All descriptors results have been regrouped per transformation. There is scale changes in 3a, then orientation shift in 3b. The following (3d and 3c) displays viewpoint change. Last, 3e shows lightness variation.

similarly to both reflections. SURF rate is slightly above 20 %. Both BRISK and FREAK are around 9 % and the last three are below. Globally all descriptors match poorly to reflection which was our expectation.

V. CONCLUSION

To summarize, descriptors respond as expected to transformations. Their invariance capacities match Table I information. Nevertheless, each descriptor has a range of efficiency. The matching rate decreases outside the interval. Computation time responds as expected: gradient based descriptors are slower than binary ones. There is an inversion, SURF is faster than LATCH which measures several Frobenius norms, hence its slowness. Overall, flip transformations give the expected results and images do not match.

In this paper we present only key points with 50 pixels size but tests have been conducted in different sizes and may be enhanced to a complete study on key point size effect on matching rate. We also intend to conduct an experiment on the grid size to assess influence of overlapping in key points description. Our initial data set is composed of coloured images. Therefore, an experiment on colour descriptor invariances is currently in progress with all these transformations and coloured ones. Tests were made as if manipulating the cover of a book. To complete the analysis, a new set of data will be realized. A real 3D scene is created and will be shot in the same conditions as the previously described transformations. It will allow to obtain true occlusions and deformations from 3D observation.

ACKNOWLEDGEMENT

We would like to thank the Aquitaine-Limousin-Poitou-Charentes Region and European Union for supporting the project ESALHI which is financially supported by FEDER.

REFERENCES

- [1] F. Schaffalitzky and A. Zisserman, "Multi-view matching for unordered image sets, or how do i organize my holiday snaps?," in *Computer Vision/ECCV 2002*. Springer, 2002, pp. 414–431, descripteur math filtre complexe.
- [2] E. Tola, V. Lepetit, and P. Fua, "DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 5, pp. 815–830, May 2010.
- [3] A. Alahi, R. Ortiz, and P. Vandergheynst, "Freak: Fast retina keypoint," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. Ieee, 2012, pp. 510–517.
- [4] G. Levi and T. Hassner, "LATCH: Learned Arrangements of Three Patch Codes," *arXiv preprint arXiv:1501.03719*, 2015.
- [5] Y. Ke and R. Sukthankar, "Pca-sift: A more distinctive representation for local image descriptors," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2. IEEE, 2004, pp. II–506.
- [6] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," *Computer Vision ECCV 2010*, pp. 778–792, 2010.
- [7] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2564–2571.
- [8] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using affine-invariant regions," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2. IEEE, 2003, pp. II–319.
- [9] J. B. Florindo, L. Assirati, and O. M. Bruno, "Locally enhancing fractal descriptors by using the non-additive entropy," *Pattern Recognition Letters*, vol. 70, pp. 32–37, Jan. 2016.
- [10] S. Belongie, J. Malik, and J. Puzicha, "Matching shapes," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 1. IEEE, 2001, pp. 454–461.
- [11] C. Ma, X. Yang, C. Zhang, X. Ruan, and M.-H. Yang, "Sketch retrieval via local dense stroke features," *Image and Vision Computing*, 2016.
- [12] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool, "A Comparison of Affine Region Detectors," *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, Nov. 2005, comparaison detecteur rptabilit, localisation (viewpoint, scale, blur, jpeg, light).
- [13] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005, comparaison descripteur.
- [14] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.
- [15] —, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [16] F. Mindru, T. Tuytelaars, L. Van Gool, and T. Moons, "Moment invariants for recognition under changing viewpoint and illumination," *Computer Vision and Image Understanding*, vol. 94, no. 1, pp. 3–27, 2004.
- [17] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE transactions on pattern analysis and machine intelligence*, no. 9, pp. 891–906, 1991, descripteur math filtre a rotation.
- [18] J. J. Koenderink and A. J. van Doorn, "Representation of local geometry in the visual system," *Biological cybernetics*, vol. 55, no. 6, pp. 367–375, 1987, descripteur math invariant différentiel.
- [19] O. Miksik and K. Mikolajczyk, "Evaluation of local detectors and descriptors for fast feature matching," in *Pattern Recognition (ICPR), 2012 21st International Conference on*. IEEE, 2012, pp. 2681–2684, comparaison detecteur descripteur.
- [20] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2548–2555.
- [21] Z. Wang, B. Fan, and F. Wu, "Local intensity order pattern for feature description," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 603–610.
- [22] B. Fan, F. Wu, and Z. Hu, "Rotationally invariant descriptors using intensity order pooling," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 10, pp. 2031–2045, 2012.
- [23] S. Gauglitz, T. Hllerer, and M. Turk, "Evaluation of Interest Point Detectors and Feature Descriptors for Visual Tracking," *International Journal of Computer Vision*, vol. 94, no. 3, pp. 335–360, Sep. 2011, comparaison detecteur descripteur suivi visuel.
- [24] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer vision/ECCV 2006*. Springer, 2006, pp. 404–417.
- [25] V. Lepetit and P. Fua, "Keypoint recognition using randomized trees," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 9, pp. 1465–1479, 2006.
- [26] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua, "Fast Keypoint Recognition Using Random Ferns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 448–461, Mar. 2010.
- [27] B. Mandal, Z. Wang, L. Li, and A. A. Kassim, "Performance evaluation of local descriptors and distance measures on benchmarks and first-person-view videos for face identification," *Neurocomputing*, Nov. 2015, comparaison detecteur descripteur reconnaissance faciale.
- [28] J. Kannala and E. Rahtu, "Bsf: Binarized statistical image features," in *Pattern Recognition (ICPR), 2012 21st International Conference on*. IEEE, 2012, pp. 1363–1366.
- [29] A. Hietanen, J. Lankinen, J.-K. Kmrinen, A. G. Buch, and N. Krger, "A comparison of feature detectors and descriptors for object class matching," *Neurocomputing*, Dec. 2015, comparaison detecteur descripteur matching.