2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing

# BE-SIFT: a more brief and efficient SIFT image matching algorithm for computer vision

*Jian Zhao, Student Member, CCF, Hengzhu Liu, Member, IEEE, Yiliu Feng, Shandong Yuan, Wanzeng Cai*
*School of Computer*
*National University of Defense Technology*
*No.109 Road Deya, District Kaifu, Changsha 410073, Hunan, P.R. China*
*E-mail: zhaojian9014@gmail.com*

***Abstract*—In the area of computer vision, pattern recognition and image processing, image match is a research hotspot with important theoretical significance and practical value. Recently, the image matching algorithm based on SIFT has drown wide attention for its outstanding local feature matching performance. In view of high computation complexity, poor anti-noise ability, and difficulty for practical use of the original SIFT method, a more brief and efficient SIFT image matching algorithm: BE-SIFT is proposed in this paper for computer vision. Through collaborative improvement and optimization on detection of SIFT scale space feature points, allocation of key point principal direction, and calculation of feature descriptors, the proposed method has achieved more brief dimension expression and more efficient local feature match than other existing methods. Experiments based on the Affine Covariant Features dataset provided by Oxford VGG Group demonstrate that BE-SIFT has stronger robustness for image match under the condition of viewpoint changes, rotation + scale changes, blur changes and noise changes. What's more, the computational overhead is sharply reduced and the real-time performance is greatly improved while ensuring the uniqueness of local feature and satisfactory accuracy.***

***Keywords*—SIFT; BE-SIFT; image match; local feature; real-time**

## I. Introduction

In recent years, with continuously intensive research on the theory of computer graphics, artificial intelligence, computer vision, and the rapid development based on multimedia application technologies, implementation of accurate and real-time image match for binocular stereo vision has become an urgent research topic [1].

Image match adopts some effective matching algorithm to match and to fuse the image with the same scene acquired by different sensors under the conditions of different time, different viewpoint and different shooting [2]. The matching methods are generally divided into two types: ones based on gray value and others based on feature, with both advantages and disadvantages. The gray-level-based methods are simple and fast while the matching precision is not satisfactory. In contrast, the feature-based methods have stronger robustness for complex environment, and easier for real-time matching implementation [3]. Therefore, the research of real-time image matching method with high precision, strong robustness, stable performance and fine applicability still faces huge challenges, with important theoretical significance and practical value.

The remainder of this paper is organized as follows. Section 2 introduces the mathematical modeling of image match and briefly trade off several conventional matching methods. Section 3 detailedly elaborate five stages of BE-SIFT. Section 4 provides experimental results from a comparison between BE-SIFT and original SIFT on feature detection and matching experiments. Finally, Section 5 concludes this paper and outlines the future work.

## II. RELATED WORK ON IMAGE MATCH

### *A. Mathematical Modeling*

Match between different images can be boiled down to a correlation measurement problem: stronger correlation means more common information between the images, and ensures higher matching degree [4].

If two matrixes $I_1(x,y)$ and $I_2(x,y)$ with given size are taken as the respective definition of the gray value of two different kinds of images at the pixel $(x,y)$, then the matching process can be expressed as:

$$I_2(x,y) = g\left(I_1\left(f(x,y)\right)\right) \tag{1}$$

Where, function $f(x,y)=(x',y')$ denotes two-dimensional transformation of space coordinate, $g(\bullet)$ denotes one-dimensional transformation of gray degree or radiancy degree.

It can be seen from equation (1) that the essence of image match is the alignment of geometrical space (solve the coordinate transformation relationship $f(\bullet)$ between pixels of $I_1(x,y)$ and $I_2(x,y)$) and the match of gray value between corresponding pixels (solve the gray degree transformation relationship $g(\bullet)$ between pixels of $I_1(x,y)$ and $I_2(x,y)$). Namely, it aims to achieve the best match between images through searching the optimal relationships of space transformation and gray degree transformation.

978-1-5090-0154-5/15 $31.00 © 2015 IEEE
DOI 10.1109/CIT/IUCC/DASC/PICOM.2015.81

### *B. Conventional Methods for Image Match*

- Similarity detection algorithm (SDA), which was proposed by professor Barnea in 1972, has fast template matching speed at the cost of matching precision, besides, it is susceptible to noise interference [5].

- Normalized gray degree correlation algorithm, Fourier-Mellin phase transformation correlation algorithm based on Fourier transformation, mutual information algorithm, and Levenberg-Marquardt optimization algorithm, which have high matching precision at the cost of computation overhead, complexity and consuming time [6, 7].

- Harris corner point detection algorithm, small univalue segment assimilating nucleus (SUSAN) corner point detection algorithm, and iterative closest point (ICP) algorithm, which have strong robustness and universality with mere consideration of feature information [8, 9].

- Algorithms based on regional primary color features, edge detection, neural network, projection features, wavelet transformation and image segmentation, which have satisfactory precision upon certain types of image match, however, the matching process is relatively complex and the universality is usually poor [10].

- SIFT algorithm, which was proposed by Professor David Lowe, University of British Columbia, Canada, in 1999, has its main application in detection and description of interest points. SIFT was published in 2004 on the foundation of Professor David Lowe's summary of the existing feature detection methods based on invariant, and it is a kind of local feature extraction and description algorithm which is able to keep invariance against image transformations, such as scale, translation, rotation, illumination, and affine [11, 12]. The main idea of SIFT is to transform the match between images to the match between feature vectors. SIFT has been widely used in the field of image match for its strong robustness and fast computing speed, and it has already become a hotspot in the research areas such as terminal guidance, data science and data analytics. Currently, some improved algorithms based on SIFT have been proposed in the literature, such as principle component analysis (PCA)-SIFT, Affine-SIFT, and gradient location-orientation histogram (GLOH), etc [13]. These algorithms have perfected the original method to some extent, but the implementation process is still relatively complex, with large computation overhead. Consequently, the key of research in this paper is to further lower computation complexity, and to improve robustness and real-time performance while ensuring high efficiency of the original SIFT method.

## III. A MORE BRIEF AND EFFICIENT SIFT IMAGE MATCHING ALGORITHM FOR COMPUTER VISION

Massive features may sharply increase the computation complexity of image matching algorithms, which is not in conformity with the real-time demand in practical applications, while the original SIFT method could complete local feature matching task very well only with a few necessary key points, which is quite suitable for fast and accurate match among the database with huge quantity of features [13]. Although the original SIFT algorithm has superior performance, there are still some problems, such as high computation complexity, poor anti-noise ability, and difficulty for practical use. In view of the above problems, in this paper, a more brief and efficient SIFT image matching algorithm is proposed for computer vision. The original SIFT algorithm consists of five major stages: (1) construction of SIFT scale space; (2) detection of SIFT feature points; (3) allocation of key point principal direction; (4) calculation of feature descriptors; (5) match of feature points. As is shown in Fig. 1, all five stages are also included in our proposed BE-SIFT. The difference between BE-SIFT and SIFT is that we carry out collaborative improvement and optimization on stage (2), (3), and (4).

### *A. Construction of BE-SIFT scale space*

Construction of BE-SIFT scale space mainly includes construction of Gaussian pyramid and construction of difference of Gaussian (DoG) scale space:

#### *1) Construction of Gaussian pyramid*

Theories of scale space demonstrate that Gaussian convolution kernel is the only linear transformation kernel for scale conversion. An image in scale space is expressed as the convolution of grey value function with Gaussian convolution kernel whose scale factor is variable, to obtain the scale space

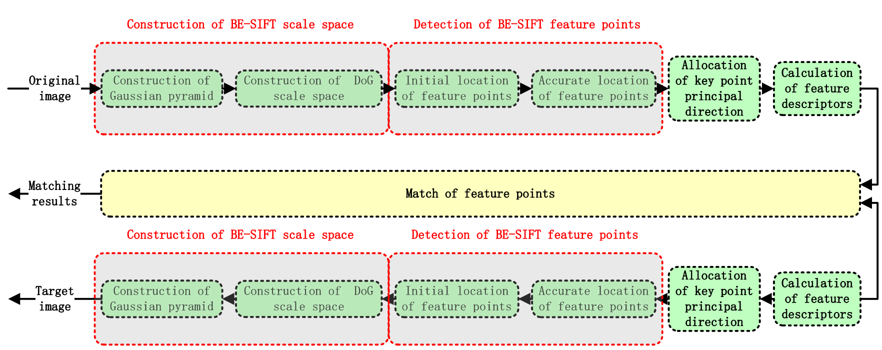


Fig. 1. BE-SIFT block diagram.

expression sequences under multi-scales, thus to extract scale space features from the sequences, which can be expressed by Laplace of Gaussian (LoG) operator as follows:

$$L(x,y,\sigma)=G(x,y,\sigma)\otimes I(x,y) \tag{2}$$

Where, $G(x,y,\sigma)=\frac{1}{2\pi\sigma^2}e^{\frac{-(x^2+y^2)}{2\sigma^2}}$, $\sigma$ denotes the scale factor, $\otimes$ denotes the convolution operation between functions. Equation (2) can be expressed by Gaussian pyramid as Fig. 2.

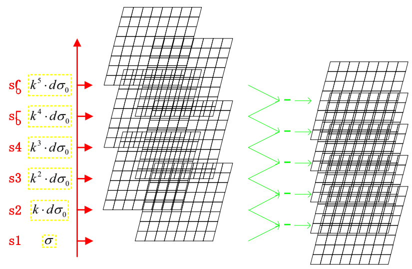

Fig. 2. Construction of DoG scale space.

Fig. 2 shows that Gaussian pyramid is a representational model of the image sequence structure. The group number ‘Octave’ usually values 4. The sub-layer number of images contained in each group is $S+3$, where, $S$ usually values 3. The first stratum image (s1) of Octave $i$ ($2\le i\le 4$) is obtained by reduced sampling between interlaced columns from the second bottom stratum image (s5) of Octave $i-1$. The first stratum image (s1) of each Octave is preprocessed by filtering the input image $I_{input}(x,y)$ with Gaussian convolution kernel, whose scale factor is $\sigma$. The $j$th ($2\le j\le 6$) stratum image of each Octave is processed by filtering the $j-1$th stratum image of the same Octave with Gaussian convolution kernel, whose scale factor is $k^{j-1}\cdot d\sigma_0$, where, $k$ denotes the constant factor of scale space, $d\sigma_0$ denotes the scale factor of fundamental stratum.

*2) Construction of DoG scale space*

In order to smooth quantitative expressions of BE-SIFT scale space, thus to obtain more stable image features and to further reduce computation complexity, DoG operator is adopted to carry out differential treatment upon Gaussian pyramid:

$$\begin{aligned} D(x,y,\sigma)&=(G(x,y,k\sigma)-G(x,y,\sigma))\otimes I(x,y)\\ &=L(x,y,k\sigma)-L(x,y,\sigma)\end{aligned} \tag{3}$$

Where, $k=2^{\frac{1}{S}}$. The specific process of DoG scale space construction is shown in Fig. 3. The group number ‘Octave’ remains the same, but the sub-layer number of images contained in each group changes to $S+2$.

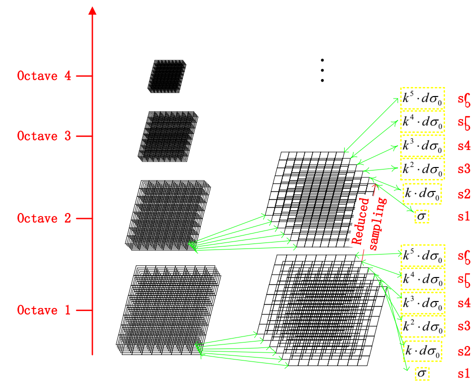


Fig. 3. Construction of Gaussian pyramid.

## B. Detection of BE-SIFT feature points

Detection of BE-SIFT feature points mainly includes initial location of feature points and accurate location of feature points:

*1) Initial location of feature points*

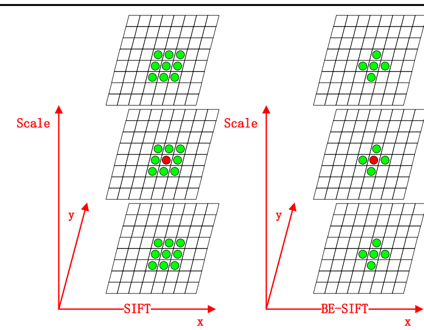


Fig. 4. Initial location of feature points.

SIFT feature points are the local extreme points with scale invariance obtained within DoG scale space. As is shown in Fig. 4, for the original SIFT method, the middle detection point is compared with the other $3\times3-1+3\times3\times2=26$ points within neighborhood with the same scale and corresponding neighborhoods with adjacent scales to successfully obtain extreme points within two-dimensional image space and scale space, thus the feature points detected via initial location have both position coordinates and scale coordinates [14]. This method can undoubtedly enable sufficient extreme points, however, it has high computation complexity and many redundant information, which directly affect the subsequent matching precision and speed.

The BE-SIFT method proposed in this paper has fully considered that pixels which are closer to feature point will make greater contribution to key point principal direction. Thus, the detection point is compared with the other $3\times3-5+(3\times3-4)\times2=14$ points within neighborhood with the same scale and corresponding neighborhoods with adjacent scales, as a result, the computation overhead has reduced by 46.15% on average, and the processing speed has been nearly doubled.

*2) Accurate location of feature points*

In order to generate the stable feature descriptors, it is still necessary to do further refinement upon feature points, which mainly includes inhibition of low contrast points and removal of edge response points:

*a) Inhibition of low contrast points*

In order to accurately locate feature points in the sub pixel level, the method based on three-dimensional quadratic Taylor expansion is adopted at the candidate extreme point to fit local sampling points within SIFT scale space:

$$D(X) = D(X_0) + \frac{\partial D^T}{\partial X}(X_0)\cdot X + \frac{1}{2}\cdot X^T \cdot \frac{\partial^2 D^T}{\partial X^2}(X_0)\cdot X \tag{4}$$

Where, $X = (x, y, \sigma)^T$ denotes the position vector and the scale vector of feature point, $X_0$ denotes the original information vector of feature point. Demand equation (4) on the $x$ derivative and then set the derivative to 0 to achieve the information vector of revised extreme point:

$$\hat{X} = -\frac{\partial^2 D^{-1}}{\partial X^2}(X_0)\cdot\frac{\partial D^T}{\partial X}(X_0) \tag{5}$$

If any dimension of $|\hat{X}|$ is larger than 0.5, then the real extreme point is definitely closer to the other one, and new data point should be interpolated to replace the current extreme point. Substitute equation (5) into equation (4) to obtain the contrast of revised feature point:

$$D(\hat{X}) = D(X_0) + \frac{1}{2}\cdot\frac{\partial D^T}{\partial X}(X_0)\cdot\hat{X} \tag{6}$$

If $|D(\hat{X})|$ is less than the threshold $\frac{0.1}{S}$, then the current extreme point is determined as the unstable extreme point with low contrast, which should be inhibited.

*b) Removal of edge response points*

The DoG operator can produce strong edge response and it can be easily affected by noise. In order to ensure the stability of feature points, it is still necessary to remove edge response points, whose principal curvatures along the edge direction are larger, and the principal curvatures perpendicular to the edge direction are smaller, thus the ratio of the two principal curvatures is generally greater than those of none edge response points. The principal curvature can be calculated through a $2\times 2$ Hessian matrix:

$$H = \begin{bmatrix} D_{XX} & D_{XY} \\ D_{XY} & D_{YY} \end{bmatrix} \tag{7}$$

Based on equation (7), equation (8) and (9) can be obtained:

$$Trace(H) = D_{XX} + D_{YY} = \alpha + \beta \tag{8}$$

$$Det(H) = D_{XX}\cdot D_{YY} - (D_{XY})^2 = \alpha\cdot\beta \tag{9}$$

Where, $\alpha$ denotes the maximum eigenvalue of $H$, corresponding to the principal curvature along the edge direction. $\beta$ denotes the minimum eigenvalue of $H$, corresponding to the principal curvature perpendicular to the edge direction. Set $\alpha = r\cdot\beta$, then:

$$\frac{Trace(H)^2}{Det(H)} = \frac{(\alpha+\beta)^2}{\alpha\cdot\beta} = \frac{(r\cdot\beta+\beta)^2}{r\cdot\beta^2} = \frac{(r+1)^2}{r} \tag{10}$$

Because the ratio of the two principal curvatures is an increasing function of $r$, a threshold can be set for $r$ (average value: 10), if the ratio is larger than the threshold, then the point is determined as the edge response point, which should be removed.

## C. Allocation of key point principal direction

After accurate location, the feature points are called key points. In order to achieve rotation invariance via local invariant features, it is necessary to allocate each key point a principal direction, which only depends on the image local information.

On this stage, the original SIFT method samples and weighs within the neighborhood window which takes the key point as the center, and $3\times 1.5\times\sigma$ as the radius. Then every three adjacent pixels are continuously weighed twice with the template $[0.25, 0.5, 0.25]$, and the gradient directions of the neighborhood pixels are counted up with the gradient histogram, thus the key point principal direction can be determined as the direction corresponding to the peak value. If there are gradient values of other directions 80% larger than that of the principal direction, then these directions are determined as the auxiliary directions of the key point, to improve the robustness of feature match and to realize the rotation invariance of feature descriptors [15]. This method can undoubtedly enable accurate allocation of key point principal direction and excellent distinguishability of feature descriptors. However, it requires a lot of storage space and has poor real-time performance and anti-noise ability.

The BE-SIFT method proposed in this paper takes advantage of first order central moment to determine key point principal direction, thus it has excellent translation invariance and strong anti-noise ability. As a result, the algorithm computation complexity has been sharply slashed. The $(p+q)$th order original moment of the input image is defined as:

$$m_{pq} = \sum x^p y^q I(x,y) \tag{11}$$

The barycenter of the input image is defined as:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right) \tag{12}$$

The original moment contains a lot of useful information, and the original image can be reconstructed by finite and enough original moments. The first order central moment of the input image can be obtained from equation (12):

$$M_{pq} = \sum (x - C_x)^p (y - C_y)^q I(x,y) \tag{13}$$

The corresponding first order moment is:

$$C' = \left( \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right) \tag{14}$$

Then, the principal direction of key point can be obtained:

$$\theta = \tan^{-1}\left( \frac{M_{01}}{M_{10}} \right) \tag{15}$$

In consideration of the fact that farther distance between pixels and the center will produce higher risk of information loss, thus $M_{00}$ is weighed in this paper:

$$M_{00} = \sum \sqrt{(x^2 + y^2)} I(x, y) \tag{16}$$

*D. Calculation of feature descriptors*

The feature descriptor of each detected key point possesses high distinctiveness, thus the mutation of eigenvector can be avoid. Besides, the feature description is compact and detailed, and it has strong robustness against the transformations such as scale, rotation, and illumination. What's more, through the combination of the neighborhood directional information, the anti-noise performance of the algorithm is enhanced, thus satisfactory fault tolerance can be achieved for the feature match containing location error.

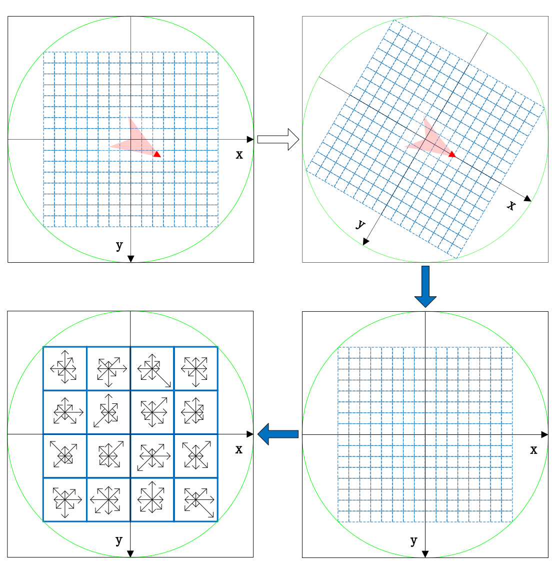


Fig. 5. The generative process of feature descriptors.

As is shown in Fig. 5, firstly, rotate the coordinate axis to the key point principal direction to ensure the rotation invariance of feature descriptors. Then select the corresponding DoG image according to the scale factor of the key point, and divide the circular neighborhood (center: key point, radius: $8 \times \sqrt{2}$) into $4 \cdot 4$ subdomains. Whereafter, adopt the gradient histogram to count up gradient modulus values and gradient directions of the 8 directions (0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°) within each subdomain. Thus each key point can be expressed by a $4 \times 4 \times 8 = 128$ dimensional feature descriptor with scale invariance. Finally, normalize feature descriptors to ensure strong robustness against the transformation of illumination.

*E. Match of feature points*

The proposed BE-SIFT method adopts the ratio of the nearest neighborhood Euclidean distance and the next nearest neighborhood Euclidean distance for feature point match between two images. Suppose the collection of feature points for reference image $A$ is $P_A = \{P_A(1), P_A(2), \cdots, P_A(p)\}$, where, $p$ denotes the number of feature points for image $A$. The collection of feature points for image $B$ (image to match) is $P_B = \{P_B(1), P_B(2), \cdots, P_B(q)\}$, where, $q$ denotes the number of feature points for image $B$. For each point within the feature point collection $P_A$, calculate its Euclidean distance with each point within the feature point collection $P_B$ to obtain the Euclidean distance collection $E$ between feature points. Rank the elements of the Euclidean distance collection $E$ to obtain the nearest neighborhood Euclidean distance and the next nearest neighborhood Euclidean distance. Therefore, the criterion can be expressed as: if $\frac{d_{\min}}{d_{n-\min}} \leq 0.8$, accept the matching pair as a candidate, otherwise, refuse the matching pair.

## IV. SIMULATION EXPERIMENT AND RESULT ANALYSIS

The Affine Covariant Features dataset (Mikolajczyk) provided by Oxford VGG Group is adopted in this paper to systematically assess the proposed BE-SIFT image matching algorithm. There are 6 images in each sub dataset and all of them are derived from one original image. Besides, all of the experiments are realized on Intel (R) Core (TM) i5-3470 CPU @ 3.20 GHz, 4G RAM computer based on Matlab (R2014a) environment, and the resolution of the selected reference image and image to match is set to $240 \times 320$.

*A. Viewpoint changes*

The proposed BE-SIFT method and the original SIFT method are respectively adopted to conduct matching experiment, which takes the first image of viewpoint change sub dataset in Mikolajczyk as the reference image, and matches with the third image of the same group. The results are shown in TABLE I and Fig. 6.

TABLE I. THE PERFORMANCE COMPARISON BETWEEN BE-SIFT AND SIFT UNDER VIEWPOINT CHANGES

| | *Key point number (image 1 / image 3)* | *Matching pair number* | *Matching time (ms)* |
|---|---|---|---|
| **BE-SIFT** | 1106/1304 | 28 | 20896 |
| **SIFT** | 3142/4371 | 96 | 50664 |

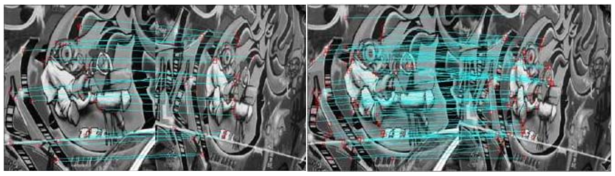

Fig. 6. The matching results comparison between BE-SIFT and SIFT under viewpoint changes (left for BE-SIFT and right for SIFT).

Experiment results demonstrate that the matching pair number for successfully matching image 1 and image 3 required by BE-SIFT method is 29.2% of the original SIFT method, and the matching time required by BE-SIFT method is 41.2% of the original SIFT method.

### B. *Rotation + scale changes*

The proposed BE-SIFT method and the original SIFT method are respectively adopted to conduct matching experiment, which takes the first image of rotation + scale change sub dataset in Mikolajczyk as the reference image, and matches with the second image. The results are shown in TABLE II and Fig. 7.

TABLE II. THE PERFORMANCE COMPARISON BETWEEN BE-SIFT AND SIFT UNDER ROTATION + SCALE CHANGES

| | *Key point number (image 1 / image 2)* | *Matching pair number* | *Matching time (ms)* |
|---|---|---|---|
| **BE-SIFT** | 1080/983 | 63 | 15624 |
| **SIFT** | 1184/1133 | 82 | 21432 |

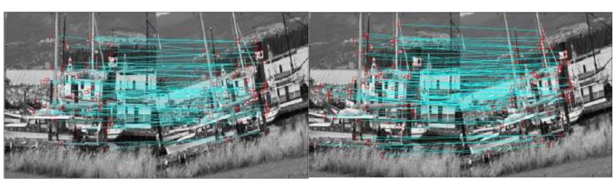

Fig. 7. The matching results comparison between BE-SIFT and SIFT under rotation + scale changes (left for BE-SIFT and right for SIFT).

Experiment results demonstrate that the matching pair number for successfully matching image 1 and image 2 required by BE-SIFT method is 76.8% of the original SIFT method, and the matching time required by BE-SIFT method is 72.9% of the original SIFT method.

### C. *Blur changes*

The proposed BE-SIFT method and the original SIFT method are respectively adopted to conduct matching experiment, which takes the first image of blur change sub dataset in Mikolajczyk as the reference image, and matches with the sixth image. The results are shown in TABLE III and Fig. 8.

TABLE III. THE PERFORMANCE COMPARISON BETWEEN BE-SIFT AND SIFT UNDER BLUR CHANGES

| | *Key point number (image 1 / image 6)* | *Matching pair number* | *Matching time (ms)* |
|---|---|---|---|
| **BE-SIFT** | 304/166 | 30 | 2937 |
| **SIFT** | 318/241 | 108 | 5460 |

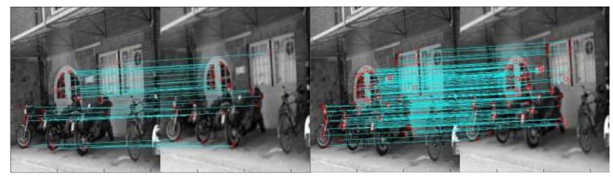

Fig. 8. The matching results comparison between BE-SIFT and SIFT under blur changes (left for BE-SIFT and right for SIFT).

Experiment results demonstrate that the matching pair number for successfully matching image 1 and image 6 required by BE-SIFT method is 27.8% of the original SIFT method, and the matching time required by BE-SIFT method is 53.8% of the original SIFT method.

### D. *Noise changes*

The proposed BE-SIFT method and the original SIFT method are respectively adopted to conduct matching experiment, which takes the first image of noise change sub dataset in Mikolajczyk as the reference image, and matches with the sixth image. The results are shown in TABLE IV and Fig. 9.

TABLE IV. THE PERFORMANCE COMPARISON BETWEEN BE-SIFT AND SIFT UNDER NOISE CHANGES

| | *Key point number (image 1 / image 6)* | *Matching pair number* | *Matching time (ms)* |
|---|---|---|---|
| **BE-SIFT** | 351/255 | 73 | 6461 |
| **SIFT** | 2152/1466 | 490 | 24632 |

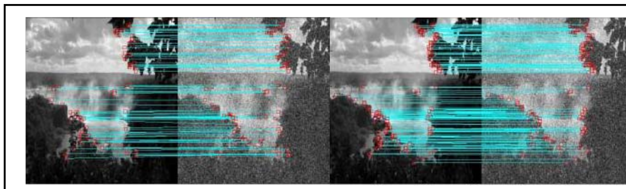

Fig. 9. The matching results comparison between BE-SIFT and SIFT under noise changes (left for BE-SIFT and right for SIFT).

Experiment results demonstrate that the matching pair number for successfully matching image 1 and image 6 required by BE-SIFT method is 14.9% of the original SIFT method, and the matching time required by BE-SIFT method is 26.2% of the original SIFT method.

The above simulation experiment results show that the speedup of BE-SIFT relative to SIFT is acceptable for the real-time application. What's more, BE-SIFT is also valid upon a larger image dataset.

## V. CONCLUSION AND EXPECTATION

In view of high computation complexity, poor anti-noise ability, and difficulty for practical use of the original SIFT method, a more brief and efficient SIFT image matching algorithm: BE-SIFT is proposed in this paper for computer vision. Through collaborative improvement and optimization on detection of SIFT scale space feature points, allocation of key point principal direction, and calculation of feature descriptors, the proposed method has achieved more brief dimension expression and more efficient local feature match than other existing methods. Experiments based on the Affine Covariant Features dataset provided by Oxford VGG Group demonstrate that BE-SIFT has stronger robustness for image match under the condition of viewpoint changes, rotation + scale changes, blur changes and noise changes. What's more, the computational overhead has sharply reduced and the real-time performance has greatly improved while ensuring the uniqueness of local feature and satisfactory accuracy.

However, the work of this paper is still a lot of deficiencies and defects, for instance, other standard image datasets haven't been adopted to compare experiment results with those of Mikolajczyk, and to evaluate the algorithm universality. With the development of computer vision, pattern recognition and image processing, SIFT and its optimized algorithms will definitely attract more attention.

## ACKNOWLEDGMENT

This work is supported by the Aerospace Science and Technology Innovation Foundation of China Aerospace Science and Technology Corporation (CASC), under grant No. 1602066115001.

## REFERENCES

[1] Witkin A. Scale-space filtering: A new approach to multiscale description[C]. IEEE International Conference on ICASSP'84, 1984, 9: 150-153.

[2] Lindeberg T. Scale-space theory in computer vision[M]. Springer, 1993.

[3] Mikolajczyk K, and Schmid C. A performance evaluation of local descriptors[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(10): 1615-1630.

[4] Smith S M, and Brady J M. SUSAN. A new approach to low level image pro-cessing[J]. International Journal of Computer Vision, 1997, 23(1): 45-78.

[5] Harris C, and Stephens M. A combined corner and edge detector[C]. Alvey Vision Conference, 1988, 15: 50.

[6] Armstrong M, and Zisserman A. Robust object tracking[C]. Proceedings of Second Asian Conferenceon Computer Vision, Singapore, 1995: 58−62.

[7] Rosten E, and Drummond T. Machine learing for high-speed detection[C]. Proceedings of the European Conference on Computer Vision, 2006, 1(1): 430-443.

[8] Marr D, and Hildreth E. Theory of edge detection[J]. Proceedings of the Royal Society of London, 1980, 207(1167): 187-217.

[9] Lowe D G. Object recognition from local scale-invariant features[C]. Proceedings of The Seventh IEEE International Conference on Computer Vision, 1999, 2: 1150-1157.

[10] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110.

[11] Bay H, Tuytelaars T, and Van Gool L. Surf: Speeded up robust features[M]. Computer Vision–ECCV, 2006: 404-417.

[12] Bay H, Ess A, Tuytelaars T, et al. Speeded-up robust features (SURF)[J]. Computer Vision and Image Understanding, 2008, 110(3): 346-359.

[13] Chiu L C, Chang T S, Chen J Y, et al. Fast SIFT design for real-time visual feature extraction. IEEE Transactions on Image Processing. 2013.

[14] Zhengfa Liang, Hengzhu Liu, Botao Zhang, and Benzhang Wang. Real-time hardware accelerator for single image haze removal dark channel prior and guided filter[J]. IEICE Electronics Express, 2014, 11(24).

[15] Forssén P E. Maximally stable colour regions for recognition and matching[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2007: 1-8.