Performance Evaluation of Feature Detection Methods for Visual Measurements

Ya Zhang, Fei Yu, Yanyan Wang, and Kai Wang

Abstract—**The visual measurement restricts the navigation accuracy of the vision-aided integrated navigation system. Thus, how to obtain the visual measurement quickly and accurately which involves the feature extraction becomes a key focus. Among the various feature extraction methods, the most commonly used feature extraction methods are the scale invariant feature transform (SIFT), the speeded up robust features (SURF) and the features from accelerated segment test (FAST). The performance evaluation is beneficial to choosing appropriate feature extraction methods for visual measurements. Although a great many of studies on their performance evaluation exist, there is lack of performance comparison among the abovementioned three feature extraction methods. Therefore, researching on the evaluation of SIFT, SURF and FAST is of great importance, which is the main objective of this manuscript. In this paper, the theoretical principles of these three methods were firstly overviewed. And then their performance was compared and analyzed from three aspects: the computing time, the capability of extracting features and their invariances. In order to make the comparative analysis systematically, the sequences of the image transformations used in this paper were carried on rotation, scale, blur, compression and illumination, respectively. The experimental results showed that among the three methods, the FAST method was the fastest one and the SIFT method possessed the strongest extraction capability. The rotation, scale and compression invariances with the SIFT method were all superior to the ones with the other two methods. For the blur invariance, the SIFT and SURF methods had similar performance which was better than the one of the FAST method. Besides, the illumination invariance with the FAST was not as good as with the other methods.**

Index Terms—**scale invariant feature transform (SIFT), speeded up robust features (SURF), features from accelerated segment test (FAST), feature extraction method, invariance, performance evaluation**

I. Introduction

IN modern kinematic positioning and navigation, the MEMS (Micro-electro-mechanical System) IMUs have been more and more popularized due to their advantages for being low cost, very compact and low power consumption [1], [2], [3]. However, low positioning accuracy is a fatal flaw of the MEMS IMUs which seriously discourages their utilization in practice. At the same time, how to well aid the inertial navigation in poor GNSS and/or GNSS denied environment also becomes essential. Therefore, more and more researchers devote to studying alternate aiding techniques in order to improve the system's accuracy [4], [5], [6].

With the rapid development of the digital imaging sensors and the computer techniques, the vision-aided integrated navigation system has drawn extensive attention of plenty research and development activities due to its low price, small size and rich information and other numerous merits [7], [8], [9]. In the vision-aided integrated navigation system, the visual measurement obtained by utilizing its onboard visual sensor. Therefore, how to effectively extract useful visual information through visual sensors is one of the key components in vision-aided integrated navigation systems.

So far, there are different varieties of feature extraction methods. It is worth to mention a few of the most well-known feature extractors, the Harris feature extractor proposed by Harris and Stephens [10] based on the second moment matrix, the Smallest Univalue Segment Assimilating Nucleus (SUSAN) detector proposed by Smith et.al [11] based on the morphology, and the Hessian extractor based on the second order partial derivative matrix [12], [13], [14]. Besides, the scale invariant feature transform (SIFT), the speeded up robust features (SURF) and the features from accelerated segment test (FAST) methods are most widely used for visual measurements in vision-aided integrated navigation systems now [15], [16], [17]. With the emergence of these new feature extraction methods, the studies on their performance become paramount and badly needed in practice. Accordingly, [12], [13], [18] made outstanding contributions. Although some of extraction methods were evaluated and compared, there is still a lack of systematic analysis of the FAST method. Under the consideration that FAST is one of the most increasingly widespread used feature extraction methods, a systematical evaluation of its performance is necessary and meaningful so that it becomes the focus of this manuscript.

The fundamentals of the SIFT, SURF, and FAST methods were firstly introduced in detail in this paper. Thereafter, their performance was systematically analyzed and compared. Since a good feature extraction method should have many advantages including running faster, detecting more features, and having good invariance for different transforms etc. Therefore, the above indicators of these three feature extraction methods were taken as the evaluation metrics to perform the comparison and evaluation here.

The rest of the paper was organized as follows. Section II introduced the related work. The principles of these three methods were described in Section III. Section IV presented the datasets and the evaluation criteria used in this paper. Experimental results along with specific analysis were given in Section V. Section VI concluded this manuscript.

II. Related Work

So far, there are a variety of feature extraction methods. Each of them has its own properties. The Harris detector is based on the second moment matrix describing the gradient distribution in a local neighborhood of a point [12], [13],

Manuscript received March 6, 2019. This work is also supported by National Natural Science Foundation of China (No. 61701134, 51679047 and 51709068), Postdoctoral Foundation of Heilongjiang Province (No.LBH-Z16044).

All the authors are with School of Electrical Engineering and Automation, Harbin Institute of Technology, Harbin, Heilongjiang, 150001, China e-mail: (see yufei@hit.edu.cn).

(Advance online publication: 27 May 2019)

[14]. It was first used to find the salient features in the image by Davison and Murray and it also uses the local image to describe them. Although the Harris detector has the translation invariance, the rotation invariance and the illumination invariance, it does not possess the scale invariant [19], [20]. SIFT was proposed in 1999 and was improved in 2004 by Lowe [21], [22]. This feature extraction method, by combining extraction, description and matching steps, is widely used in many computer vision applications. Furthermore, SIFT was used in 3D space by Se and Lowe through utilizing the extracted features to aid the tracking algorithm, and improving its robustness [23], [24]. However, one of the biggest deficiencies with the SIFT method is that the dimension of its descriptor is too high, which increases the computational complexity, lengthens the running time and reduces the speed of the feature matching. To solve this problem, the SURF method was proposed by Bay in 2006 [16], [25], which was modified from SIFT, and also contained extraction, description and matching steps. It was used for location in omni-directional images, and the expected results were achieved [26]. Although the SURF method runs faster than the SIFT does, it does not perform as good as the SIFT in terms of scale, rotation and illumination transformations.

Unlike the abovementioned methods, the SUSAN method was proposed based on the morphology by Smith et.al, to avoid the expensive computation burden since this method only compares the gray values among the pixels and does not involve the computation of the gradients [17], [27]. The FAST method was further proposed by Rosten et.al based on the SUSAN [17], [28] for simplicity, effectiveness and rapidity. However, this method may not be robust to the noise because only a limited number of the pixels were processed with this feature extraction method.

With the emergence of these new feature extraction methods, the research on their performance evaluation attracts plenty of attention. Mikolajczyk et.al [13], [29] compared shape context, steerable filters, PCA-SIFT, differential invariants, SIFT, complex filters, and moment invariants for different types of regions of interest. They also proposed an extension of the SIFT descriptor and showed that it outperformed the original method [13], [19], [30]. The performance was measured against the changes in viewpoint, s-cale, illumination, defocus and image compression. [12], [18] compared the behavior of different detectors and descriptors in VSLAM, such as the Harris, the SUSAN, the SIFT and the SURF methods. They evaluated the repeatability of these detectors, as well as the invariance and distinctiveness of the descriptors. The performance evaluation of the SIFT and SURF for the cross band matching of multispectral images was present in [31]. Miksik et.al [30] evaluated BRIEF, BRISK, ORB, and MRRID extractors and compared their matching precisions. In [32], the SIFT, PCA-SIFT and SURF methods were evaluated for their extracting and matching speed. In summary, there have been many of works on the performance evaluation. However, the question about which feature extraction method is more suitable for visual measurements is still open.

III. Fundamentals of SIFT, SURF and FAST

Among varieties of feature extraction methods, three are commonly used for visual measurements, which are SIFT, SURF, FAST.So here these three feature extraction methods will be introduced in this section.

A. SIFT

SIFT which is utilized for detecting and describing local features in images and was first published in [22]. The significant advantage of SIFT is its scale invariance. To achieve this efficiently, the scale space is first established on the basis of the scale space theory, i.e., convolution operations to the preprocessing images are carried out using the Gaussian kernel function. Different scale images can be obtained by changing the scale factor σ so that the Difference-of-Gaussian (DoG) pyramid can be gotten by computing the difference of two images which have the same resolution and the nearby scales separated by a constant multiplicative factor k. Thus,

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) \\ &= L(x,y,k\sigma) - L(x,y,\sigma) \end{aligned} \tag{1}$$

wherein $L(x,y,\sigma)$ is the scale space with an input image $I(x,y)$, $*$ is the convolution operation, and $G(x,y,\sigma)$ is a variable-scale Gaussian function described as follows:

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-\left(x^2+y^2\right)}{2\sigma^2}} \tag{2}$$

Then, the extreme of each scale-space should be detected. In order to detect the local maxima and minima of the DoG images, each sample point is compared to its eight neighbors in the current image and nine neighbors in the scale above and below. The non-maximum suppression method is used to find all candidate features [21].

Once a candidate feature has been found, the next step is to perform a detailed fit to the nearby data for location, scale, and ratio of principal curvatures. The points which have low contrast or are poorly localized along an edge can be rejected by quadratic interpolation method and by computing the trace of the Hessian matrix, respectively, introduced in [21], [22]. In this way, the stability and the noise immunity of SIFT will be significantly enhanced.

Then, by assigning a consistent orientation to each feature based on local image properties, the feature descriptor can be represented relative to this orientation and therefore the rotation invariance can be achieved. For each image sample $L(x,y)$, at its scale, the gradient magnitude $m(x,y)$ and the orientation $\theta(x,y)$ can be computed using the differences between its neighbor pixels:

$$m(x,y) = \sqrt{(L_1 - L_2)^2 + (L_3 - L_4)^2} \tag{3}$$

$$\theta(x,y) = \tan^{-1}\frac{L_3 - L_4}{L_1 - L_2} \tag{4}$$

where $L_1 = L(x+1,y)$, $L_2 = L(x-1,y)$, $L_3 = L(x,y+1)$, $L_4 = L(x,y-1)$. An orientation histogram is formed from the gradient orientations of features in the neighborhood around a feature. The highest peak in the histogram is defined as the dominant direction of this feature. Around the feature, a circular Gaussian window, which is divided into 4×4 subarea, is selected and then gradient magnitudes of sample points in each subarea are summed in eight directions. Finally, a descriptor with $4 \times 4 \times 8 = 128$ dimensions is obtained and normalized. Here the SIFT method is invariant for scale and rotation transformations.

After the generation of the SIFT descriptors of two images, the extracted features can be matched by utilizing the Euclidean distance as the similarity measure.

Overall, the SIFT method has expected performance on scale and rotation transformations, but suffers from expensive computation due to its high descriptor dimension, especially in larger scale environment.

B. SURF

SURF, which is a robust local feature detector [25], was modified based on the SIFT. It is achieved by using a set of box-type filters to approximate Hessian matrix and by relying on integral images for image convolutions, which drastically reduce the computational complexity [16], [25].

Firstly, for an input image $I(x, y)$, the integral image is obtained by

$$I_{\Sigma}(x, y) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j) \tag{5}$$

Similarly to SIFT, the scale space is built for the scale invariance. However, the SURF method approximates the second-order Gaussian derivatives using a set of box-type filters, while different filter sizes corresponds different scales [25]. Based on the integral image theory, the scale space can be obtained quickly. And by subtracting two images which have nearby scales, the DoG will be built.

Then, the extremes of the DoG can be found using the Hessian matrix, which is given for an input image $I(x, y)$ with σ scale as follows:

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix} \tag{6}$$

where $L_{xx}(x, y, \sigma)$ is the convolution of the $I(x, y)$ and the second-order Gaussian derivatives; $L_{xy}(x, y, \sigma)$ and $L_{yy}(x, y, \sigma)$ are defined as similar as $L_{xx}(x, y, \sigma)$. In SURF, approximated by box-type filters, L_{xx}, L_{xy}, L_{yy} are denoted by D_{xx}, D_{xy}, D_{yy} , respectively. So the determinant of the Hessian matrix 6 is:

$$\det(H_{approx}) = D_{xx} D_{yy} - (\omega D_{xy})^2 \tag{7}$$

where ω is an adjusting factor to ensure that the Gaussian functions and the box filters can be approximated.

After the Hessian's determinant of each pixel is obtained and compared to its neighbors in $3 \times 3 \times 3$ neighborhood, the candidate features can be identified by utilizing the non-maximum suppression method. Then their locations and scales can be refined using quadratic interpolation in the same way as with the SIFT method.

In order for assigning orientation, the Haar wavelet responses in x and y directions are calculated within a circular neighborhood. Once the wavelet responses are weighted based on Gaussian function, the wavelet responses in horizontal and vertical directions are summed within a sliding orientation window covering an angle of $60°$, respectively. Since the orientation of the longest vector corresponds to the dominant direction, without loss of the generality, let assume the dominant direction in the x direction. Then, one can build a square region around the feature and then split it uniformly into 4×4 square sub-regions. For each sub-region, the Haar wavelet responses in x and y directions, denoted by d_x and d_y with their absolute values as $|d_x|$ and $|d_y|$, are computed at each sample point. Then, all these four values are summed, represented by the vector $\boldsymbol{v} = (\Sigma d_x, \Sigma d_y, \Sigma |d_x|, \Sigma |d_y|)$. Hence the dimension of the SURF descriptor is $4 \times 4 \times 4 = 64$. After the normalization, the SURF has the scale and rotation invariances.

At last, the features from different images can be matched by using the Euclidean distance.

The schematic flowchart of the SURF method is described in Fig.1.

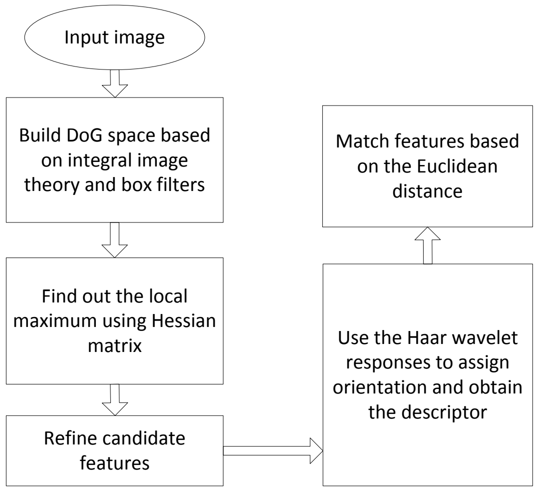


Fig. 1. The schematic flowchart of the SURF method

Therefore, compared with the SIFT, the SURF method has not only the scale and rotation invariance but also the faster computing speed.

C. FAST

Different from SIFT and SURF methods, the FAST method does not compute the gradients to extract features, but utilizes the comparison of the pixel gray values [28]. So it belongs to the high-speed feature extraction methods.

Firstly, the FAST method finds candidate features using the segment test [17], [28], which operates within a circle of sixteen pixels around a center point p. Fig. 2 gives an example of the segment test (right) using the Lena (left).

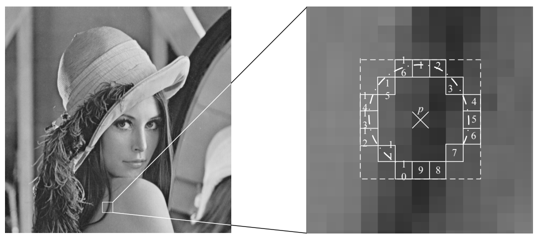


Fig. 2. An example of the segment test

Since the general test criteria have their weaknesses, Rosten et.al [17], [28] presented an approach which used machine learning to address the candidate features. For each center pixel point p, its related pixels on the circle (denoted by $p \to x$, $x \in \{1, \cdots, 16\}$) can have one of three states:

$$S_{p\to x} = \begin{cases} d, & I_{p\to x} \le I_p - t \quad \text{(darker)} \\ s, & I_p - t < I_{p\to x} \le I_p + t \quad \text{(similar)} \\ b, & I_p + t < I_{p\to x} \quad \text{(brighter)} \end{cases} \tag{8}$$

where I_p and $I_{p\to x}$ are the gray values of p and the pixel on the circle, respectively, and t is a threshold. By choosing an x and computing $S_{p\to x}$ for all points in a training image, each point has one of three states, darker, similar or brighter. So the training image is divided into three parts, denote by P_d, P_s and P_b.

Then the FAST method employs the algorithm used in ID3 [17] to select the pixels which yield the most information about whether the candidate pixel is a feature. This is measured by the entropy of a Boolean variable named K_p:

$$H(P) = (c + \bar{c}) \log_2 (c + \bar{c}) - c\log_2 c - \bar{c}\log_2 \bar{c} \tag{9}$$

where $c = |\{p \,|K_p \; is \; ture\}|$ is the number of features and $\bar{c} = |\{p \,|K_p \; is \; false\}|$ is the number of the non-features.

The choice of x then yields the information gain:

$$H(P) - H(P_d) - H(P_s) - H(P_b) \tag{10}$$

Then this process is applied recursively on all three subsets and terminates when the entropy is zero.

After the defined response function as below:

$$V = \max(\sum_{x \in P_b} |I_{p\to x} - I_p| - t, \sum_{x \in P_d} |I_p - I_{p\to x}| - t) \tag{11}$$

the non-maxima suppression can be applied to extract features finally.

Since the FAST method can only detect features, in this paper the same description and matching methods are used as introduced in the SURF. Overall, it is a high efficient method, but sensitive to noise because only a small portion pixels have been operated.

IV. Evaluation Metrics and Image Samples

In this section, the criteria and the datasets used in the performance evaluation of the mentioned feature extraction methods were introduced.

A. Evaluation Criteria

In order to evaluate the performance of each feature extraction method, the following three metrics were used.

1) The running time: different feature extraction methods were used to process the same image in the same hardware environment. The less time a method takes, the more efficient it works. In order to compare the running time with SIFT, SURF and FAST clearly, the time ratios were calculated as follows

$$SURF/SIFT = \frac{SURF\_time}{SIFT\_time} \tag{12}$$

$$FAST/SIFT = \frac{FAST\_time}{SIFT\_time} \tag{13}$$

2) The number of the detected features: the number of the detected features reflects the capability of feature detection and extraction with each method.

3) The percentage of the correctly matched features: this criterion, calculated as the ratio of the number of the correct matched features to the number of all matched features, measures the correctness of the feature detection and extraction. A good feature extraction method should always have high correctness when it extracts features from each image in the same set. Here the "Correctness" describes the percentage of the correctly matched features.

$$Correctness = \frac{(Num\_correctmatches)}{(Num\_allmatches)} \times 100 \tag{14}$$

B. Data Sets

In order to analyze the performance of the SIFT, SURF, and FAST methods, five different datasets [1] of the transformed images were used (Fig.3-Fig.7). The transformations included rotation transformation (Fig.3), scale transformation (Fig.4), blur transformation (Fig.5), compression transformation (Fig.6) and illumination transformation (Fig.7). Each image set had six images, of which the last five images of each set had the corresponding transformation referred to their first images which were the original and reference images. In the rotated image set, the original image 3(a) was rotated 20, 30, 40, 50 and 60 degrees clockwise, respectively, of which the last one was shown in Fig.3(b). The scale transformed image set was obtained by varying the scale values contrasted to the Fig.4(a). And the scale values of the rest images in this image set were 1.2, 1.8, 2.4, 3.0 and 3.5, respectively. The blur transformation was acquired by adding some Gaussian noises with zero means and different variances which were 0.02, 0.04, 0.06, 0.08, and 0.1, respectively. The Fig.5(b), whose noise variance was 0.1, was present as an example. The image quality parameter of the Fig.6(a) was set as 50%, 20%, 10%, 5%, and 2%, and then the compression sequence was generated. The illumination transformation was introduced by varying the camera aperture.

(a) (b)

Fig. 3. Rotation transformation

Fig. 4. Scale transformation

[1]Available at http://www.robots.ox.ac.uk/vgg/research/affine/.

Fig. 5. Blur transformation

Fig. 6. Compression transformation

V. Experimental Results

In order to evaluate their performance, the datasets introduced in Section 4.2 were processed on the same computer equipped with an Intel Core 2 T6570, 2.1 GHz processor and 3 GB RAM under Windows XP. The evaluation metrics were calculated for all of the images using SIFT, SURF and FAST, respectively.

The results of the first set of images as in Fig.3 were graphically represented in Fig.8, of which the running time, the number of detected features and the correctness were compared in Fig.8(a), Fig.8(b) and Fig.8(c), respectively.

As can be seen from Fig.8, compared with SIFT, the SURF ran faster at a ratio of 0.1-0.15 whilst the FAST ran faster at a ratio of 0.02-0.05. Therefore, the FAST method was the fastest one in rotation transformation image sequences which was 20-50 times faster than SIFT and 3-6 times faster than SURF. The number of the detected features from SURF was largest among these three feature extraction methods, followed by SURF, and then FAST. For the correctness, SIFT had the highest correctness (about 78%) and SURF and FAST were almost of equal reliable (nearly 45%). Therefore, SIFT showed the best performance in rotation invariance among these three methods. The rotation invariance of SURF and FAST were almost of equivalent.

The results of the image set as in Fig.4 were shown in Fig.9. The running time, the number of detected features and the correctness were compared using bar graphs in Fig.9(a), Fig.9(b) and Fig.9(c), respectively.

Compared with SIFT, SURF ran faster at the ratio of 0.08-0.13 and FAST was faster at the ratio of 0.01-0.1. Therefore, the FAST method had the shortest running time with the scale transformation, which was 10-100 times faster than SIFT and 1-9 times faster than SURF. With the scale increment, the number of detected features by SIFT remained stable. However, the one by SURF was increased whilst the one

Fig. 7. Illumination transformation

by FAST was decreased. For the correctness of the feature detection, SIFT performed correctly at about 90% which was maintained stably while the feature detection correctness of SURF and FAST were both unsteady. But in general the correctness of FAST was better than the one of SURF. Therefore, SIFT had the best performance in scale invariance among these three method, followed by FAST.

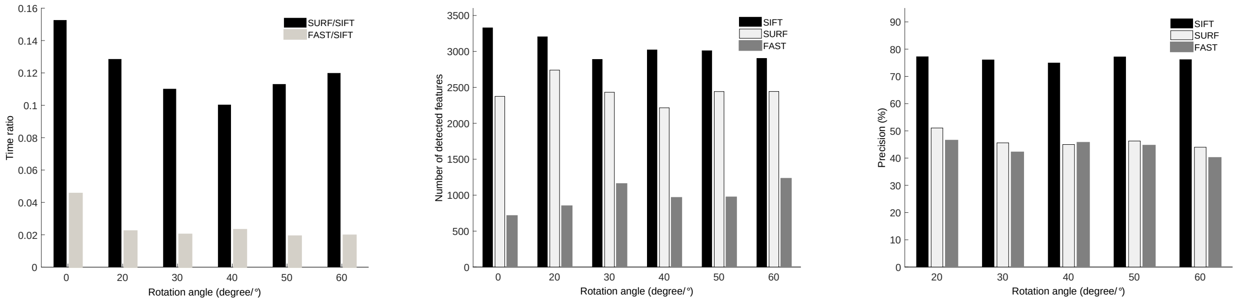
The results of the image set as in Fig.5 were shown in Fig.10. The running time, the number of detected features and the correctness were compared in Fig.10(a), Fig.10(b) and Fig.10(c), respectively.

From Fig.10, it is obvious that compared with SIFT, the SURF ran faster at a ratio of 0.06-0.12 whilst the FAST ran faster at a ratio of 0.01-0.04. Therefore, the FAST method was the fastest method in blur transformation image sequences which was 25-100 times faster than SIFT and 3-6 times faster than SURF. SIFT detects more features than the other two methods, followed by SURF. Besides, the number of detected features by these three methods kept stable as the noise covariance increased. For the correctness, SIFT and SURF had almost the same and stable performance (nearly 70%), but the one of FAST was lower and unstable. Therefore SIFT and SURF had nearly the same performance in blur invariance which was much better than FAST.

The results of the image set as in Fig.6 were represented in Fig.11, of which the running time, the number of detected features and the correctness were compared using bar graphs in Fig.11(a), Fig.11(b) and Fig.11(c), respectively.

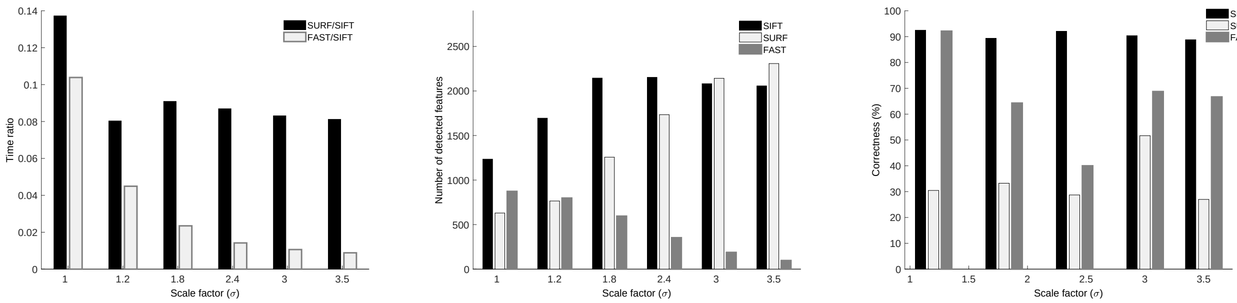
Compared with the SIFT method, SURF ran faster at the ratio of 0.1-0.15 and FAST was faster at the ratio of 0.02-0.06. Therefore, the FAST method was the fastest method in compression transformation image sequences which was 15 50 times faster than SIFT and 2-6 times faster than SURF. SIFT detected more features than the other two method, followed by SURF, and then FAST. With the compression ratio increment, the number of detected features by SIFT and SURF increased. But there was no obvious regularity for FAST. In general, the correctness of these three methods decreased as the compression ratio increased. But the correctness of these three algorithms did not change obviously unless the compression ratio was larger than 90%. When the compression ratio was larger than 90%, all of their correctness decreased rapidly. The correctness of SIFT decreased the least and the one of FAST decreased the most (when the compression ratio was 98%, its correctness was only 6%). Therefore, SIFT had the best performance on the compression invariance, followed by SURF, and then FAST.

The results of image set as in Fig.7 were shown in Fig.12. The running time, the number of detected features and the correctness were compared using bar graphs presented in



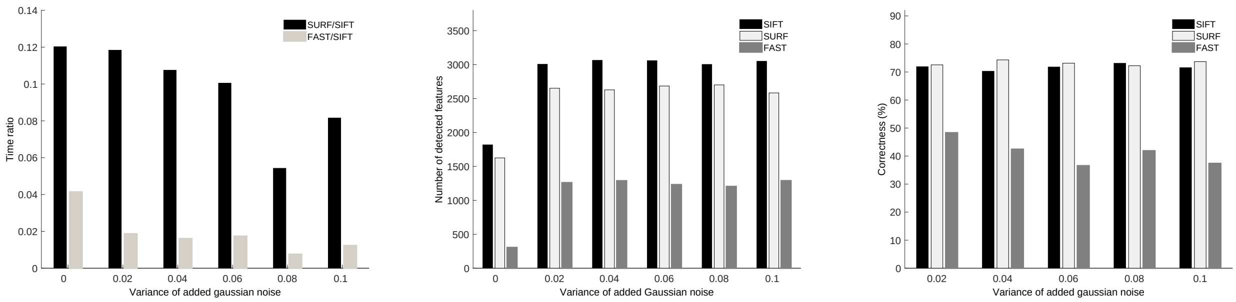
(a) (b) (c)

Fig. 8. Comparison of the three algorithms in rotation transformation



(a) (b) (c)

Fig. 9. Comparison of the three algorithms in scale transformation



(a) (b) (c)

Fig. 10. Comparison of the three algorithms in blur transformation

Fig.12(a), Fig.12(b) and Fig.12(c), respectively.

It is easy to know that compared with the SIFT method, SURF ran faster at the ratio of 0.1-0.18 and FAST was faster at the ratio of 0.04-0.06. Therefore, the FAST method was the fastest method with the illumination transformation which was about 25 times faster than SIFT and nearly 3 times faster than SURF. In general, the detected features decreased with the image illumination weakening. For the same image, the features detected by SIFT was more than the ones detected by the other two methods and FAST detected the least features. From Fig.12(c), we can see that the correctness of SIFT decreased with the image illumination intensity weakening (decreased from 74% to 50%), but the correctness of SURF maintained stably (about 60%). The correctness of FAST was always lower than 45% and it changed with the illumination randomly. Therefore, SURF had the best performance in illumination invariance, followed by SIFT.

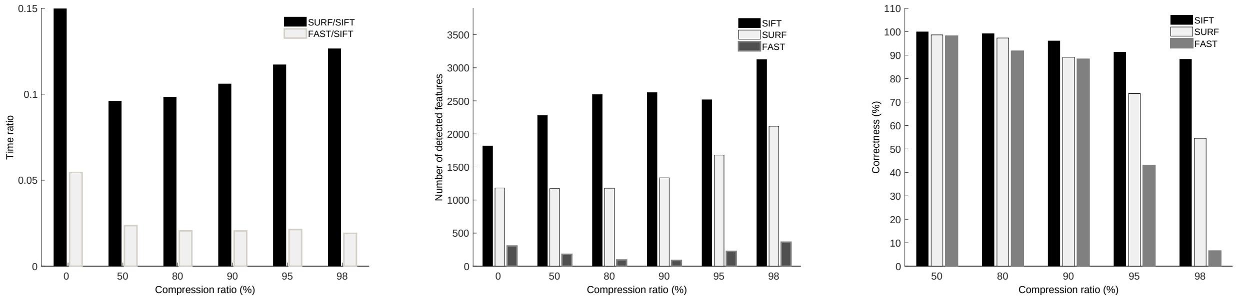
In summary, the comparison results of the three feature extraction methods were shown in Table I, in which more stars correspond to better performance.

TABLE I
THE PERFORMANCE COMPARISON OF THE SIFT, SURF AND FAST

Methods	Invariance				
	Rotation	Scale	Blur	Compression	Illumination
SIFT	★★	★★★	★★	★★★	★
SURF	★	★	★★	★★	★★★
FAST	★	★★	★	★	★

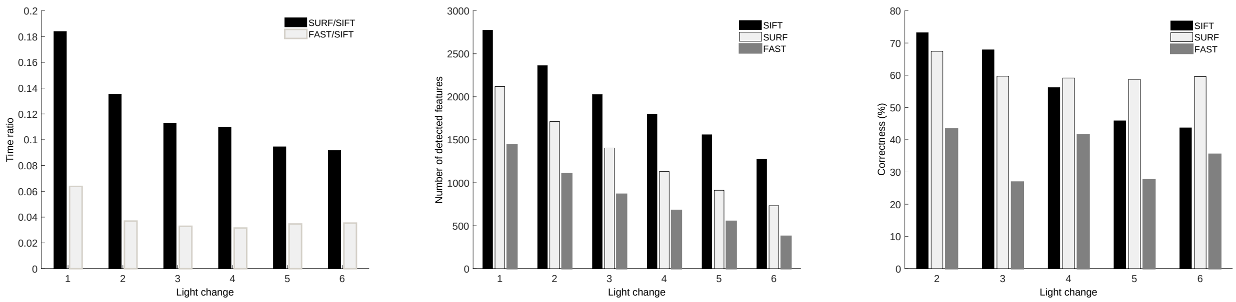
VI. CONCLUSION

Three features extraction methods, SIFT, SURF and FAST, are most widely used for visual measurements. In this manuscript, their performance were analyzed and compared comprehensively. In order to do this, five image sets were processed, respectively. Besides, their performance was systematically evaluated and compared through three criteria:



(a) (b) (c)

Fig. 11. Comparison of the three methods in compression transformation



(a) (b) (c)

Fig. 12. Comparison of the three methods in illumination transformation

the running time, the number of the detected features and the correctness. The experimental results showed that the FAST method had the fastest computing speed, the SURF method took the second place, and the SIFT method had the slowest computing speed among these three methods. The number of the detected features from the SIFT method was the largest while the one from FAST was the least. With different transformations, their performance varied. The rotation invariance, the scale invariance and the compression invariance with the SIFT method were all superior to the ones with other two methods. About the blur invariance, the performance from SIFT and SURF were similar, which was better than the FAST method. Besides, the FAST method had the worst illumination invariance. Additionally, SURF and FAST had a similar performance on rotation invariance while on scale invariance FAST was better than SURF. However, the FAST method had a worst performance on compression transformation. Furthermore, SURF was good with the illumination transformed and FAST had the worst illumination invariance. Since in vision-aided integrated navigation system a great many of images should be deal with, the SIFT method is unsatisfactory for its low computing speed. Besides, the FAST method has similar invariances with the SURF method in different transformations, but much faster than the SURF method. Thus, for visual measurements, the FSAT method is recommended. And this performance evaluation filled in a gap in the performance comparison among the SIFT, SURF and FAST methods and can be useful for choosing appropriate feature extraction methods for visual measurements in practice.

REFERENCES

[1] K. Abdulrahim, C. Hide, T. Moore, and C. Hill, “Integrating low cost imu with building heading in indoor pedestrian navigation,” *Journal of Global Positioning Systems*, vol. 10, no. 1, pp. 30–38, 2011.

[2] A. Amirsadri, J. Kim, L. Petersson, and J. Trumpf, “Practical considerations in precise calibration of a low-cost mems imu for road-mapping applications,” in *American Control Conference (ACC), 2012*. IEEE, 2012, pp. 881–888.

[3] I. P. Prikhodko, J. A. Gregory, C. Merritt, J. A. Geen, J. Chang, J. Bergeron, W. Clark, and M. W. Judy, “In-run bias self-calibration for low-cost mems vibratory gyroscopes,” in *Position, Location and Navigation Symposium-PLANS 2014, 2014 IEEE/ION*. IEEE, 2014, pp. 515–518.

[4] J. Georgy, A. Noureldin, M. J. Korenberg, and M. M. Bayoumi, “Modeling the stochastic drift of a mems-based gyroscope in gyro/odometer/gps integrated navigation,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 4, pp. 856–872, 2010.

[5] K.-W. Chiang, T. T. Duong, J.-K. Liao, Y.-C. Lai, C.-C. Chang, J.-M. Cai, and S.-C. Huang, “On-line smoothing for an integrated navigation system with low-cost mems inertial sensors,” *Sensors*, vol. 12, no. 12, pp. 17 372–17 389, 2012.

[6] M. Ilyas, Y. Yang, Q. S. Qian, and R. Zhang, “Low-cost imu/odometer/gps integrated navigation aided with two antennae heading measurement for land vehicle application,” in *Control and Decision Conference (CCDC), 2013 25th Chinese*. IEEE, 2013, pp. 4521–4526.

[7] F. Yu, Q. Sun, C. Lv, Y. Ben, and Y. Fu, “A slam algorithm based on adaptive cubature kalman filter,” *Mathematical Problems in Engineering*, vol. 2014, 2014.

[8] L.-H. Lin, P. D. Lawrence, and R. Hall, “Robust outdoor stereo vision slam for heavy machine rotation sensing,” *Machine vision and applications*, vol. 24, no. 1, pp. 205–226, 2013.

[9] J. Kim, S. Sukkarieh *et al.*, “6dof slam aided gnss/ins navigation in gnss denied and unknown environments,” *Positioning*, vol. 1, no. 09, 2005.

[10] C. Harris and M. Stephens, “A combined corner and edge detector.” in *Alvey vision conference*, vol. 15. Citeseer, 1988, p. 50.

[11] S. M. Smith and J. M. Brady, “Susana new approach to low level image processing,” *International journal of computer vision*, vol. 23, no. 1, pp. 45–78, 1997.

[12] A. Gil, O. M. Mozos, M. Ballesta, and O. Reinoso, “A comparative evaluation of interest point detectors and local descriptors for visual

slam," *Machine Vision and Applications*, vol. 21, no. 6, pp. 905–920, 2010.
[13] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International journal of computer vision*, vol. 65, no. 1-2, pp. 43–72, 2005.
[14] P. Panchal, S. Panchal, and S. Shah, "A comparison of sift and surf," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 1, no. 2, pp. 323–327, 2013.
[15] C. Valgren and A. J. Lilienthal, "Sift, surf & seasons: Appearance-based long-term localization in outdoor environments," *Robotics and Autonomous Systems*, vol. 58, no. 2, pp. 149–156, 2010.
[16] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.
[17] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 1, pp. 105–119, 2010.
[18] Ó. M. Mozos, A. Gil, M. Ballesta, and O. Reinoso, "Interest point detectors for visual slam," in *Current Topics in Artificial Intelligence*. Springer, 2007, pp. 170–179.
[19] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International journal of computer vision*, vol. 60, no. 1, pp. 63–86, 2004.
[20] Y. Ryu, Y. Park, J. Kim, and S. Lee, "Image edge detection using fuzzy c-means and three directions image shift method," *IAENG International Journal of Computer Science*, vol. 45, no. 1, pp. 1–6, 2018.
[21] 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.
[22] ——, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
[23] S. Se, D. G. Lowe, and J. J. Little, "Vision-based global localization and mapping for mobile robots," *Robotics, IEEE Transactions on*, vol. 21, no. 3, pp. 364–375, 2005.
[24] P. Vinod, "Unknown metamorphic malware detection: Modelling with fewer relevant features and robust feature selection techniques," *I-AENG International Journal of Computer Science*, vol. 42, no. 2, pp. 139–151, 2015.
[25] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer vision–ECCV 2006*. Springer, 2006, pp. 404–417.
[26] A. C. Murillo, J. J. Guerrero, and C. Sagüés, "Surf features for efficient robot localization with omnidirectional images," in *Robotics and Automation, 2007 IEEE International Conference on*. IEEE, 2007, pp. 3901–3907.
[27] Q. Sun, Y. Tian, and M. Diao, "Cooperative localization algorithm based on hybrid topology architecture for multiple mobile robot system," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4753–4763, 2018.
[28] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Computer Vision–ECCV 2006*. Springer, 2006, pp. 430–443.
[29] A. S. M. Murugavel and S. Ramakrishnan, "An optimized extreme learning machine for epileptic seizure detection," *IAENG International Journal of Computer Science*, vol. 41, no. 4, pp. 212–221, 2014.
[30] 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.
[31] S. Saleem, A. Bais, and R. Sablatnig, "A performance evaluation of sift and surf for multispectral image matching," in *Image Analysis and Recognition*. Springer, 2012, pp. 166–173.
[32] L. Juan and O. Gwun, "A comparison of sift, pca-sift and surf," *International Journal of Image Processing (IJIP)*, vol. 3, no. 4, pp. 143–152, 2009.