Adaptive Multiscale Feature Detection Algorithm using Information Content Quantization

Yufu Qu*1,2, Dan Zhou1,2, Xiangjin Deng3, Yanhong Zheng3

1School of Instrument Science and Opto-Electronics Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100191, China
2Key Laboratory of Precision Opto-Mechatronics Technology of Education Ministry, Beijing University of Aeronautics and Astronautics, Beijing 100191, China
3Beijing Institute of Spacecraft System Engineering, Beijing 100094, China
*Corresponding author: qyf@buaa.edu.cn

A feature detection algorithm with adaptive scale selection using information content quantization in scale space representation is presented. The number of FAST keypoints is adopted as the information content of images in scale space. The scale parameter is adjusted according to differences in information content to obtain uniform changes in the details between adjacent Gaussian images. Finally, points are matched at existing image levels using construction with a matching instruction strategy to control the number of levels per pyramid and reduce the computational cost. Compared to SIFT, FAST, and ASIFT, our method generates sufficient features that densely cover the images under large variations in scale, illumination, and viewpoint.

OCIS codes: 100.3008, 100.2000, 330.5000.

1. Introduction

Interest point detectors based on image local invariant features have been widely used in computer vision owing to their excellent invariance and robustness against image changes. In visual applications such as image matching and object recognition and tracking, feature extraction, the first key stage, significantly affects the success or failure of the entire process. A good local feature extractor can greatly reduce the complexity of follow-up work and provide the appropriate prerequisites and a strong guarantee of obtaining reasonable results. Therefore, accurate extraction of effective keypoints and their adaptation invariant to image transformations in viewpoint, illumination, scaling, and rotation become the key to feature detection strategy research. In recent years, many excellent feature detection algorithms based on a uniform Gaussian scale space [1] have been described in the literature in discussions of the spatial correlation between keypoints and neighboring pixels. For example, both the Scale Invariant Feature Transform (SIFT) [2] and the Harris-Laplacian [3] find points of local extreme within the difference of Gaussian (DoG) [4] and scale space representation, respectively, by performing non-maxima suppression [5], while ignoring the relationship between the features themselves and the content of the entire image. As an important visual characteristic, local invariant keypoints provide a statistical method of image content representation. For this reason, the information in the image content can be associated with the distribution and density of features to provide a compact image representation [2] that determines the non-uniform distribution of features. Simple methods of processing scale parameters based on Gaussian scale space are currently used in many applications; however, excessive redundant features or insufficient features are obtained if the selected scale parameter is too small or too large, respectively. Consequently, the quality of the extracted local invariant features is heavily influenced by the scale parameters selected for each interval of scale space.

This paper defines a measure of the information content quantization for a scale space representation based on the classical scale space theory [1] and Marr’s vision theory [6]. Experimental results prove that this method satisfies the basic principles proposed for the information content of a scale space representation based on visual saliency and image content. Next, the information content of a Gaussian image is calculated as the basis for choosing appropriate scale parameters to generate each interval for the pyramid. An adaptive multiscale feature detection algorithm using the Gaussian pyramid is designed that achieves both uniform variations in image information among pyramid intervals and the full expression of image details. Thus, it lays a good foundation for effective feature extraction. In addition, construction with instructions for a keypoint matching strategy in which the intermediate results of information content calculation are directly involved in feature matching is used to improve the efficiency of the algorithm.

2. Related Work

The history of image local invariant feature extraction covers more than 30 years since Moravec [7] began to study corner points in 1977. Both the early basic corner extractors, such as Forstner [8], Harris [9], SUSAN [10], and the state-of-the-art efficient corner, blob and region detectors focus on achieving the high repeatability and various invariances [11] of keypoints to optimize their performance. Therein, the implementation of scale
invariance depends on the multi-scale representation of the Scale Theory, so do the local characteristics.

The idea of scale space was proposed for the first time by Lijima in 1962 [12], then gained wide attention and rapid development with the ground-breaking work of Witkin [13] and Koenderink [14] in the 1980s. Lindeberg put forward the scale-select theory for feature detection and gave detailed expression for scale space theory which is corresponding to image scaling transformation [15]. An object in different scales can be represented by changing the scale parameters in the scale space and the purpose of scale invariance can be achieved.

One of the first systems to perform interest point detection in scale space was proposed in [7]. They built the Laplacian pyramid for coarse-to-fine feature selection. A normalized LoG(Laplace of Gaussian) function was applied in [1, 16] to build a scale space representation and search for 3D maxima. Another combination of local invariant features and scale space appeared in David G. Lowe's SIFT [4] which was improved and developed in 2004 and became a landmark work in the history of feature study. In SIFT, scale-space extrema in the difference-of-Gaussian function is utilized to detect stable keypoint locations, then a detailed fit is performed to reject points that have low contrast or are poorly localized along an edge [2]. SIFT is actually the only method that is fully scale invariant. With the concept of scale space, a multi-scale framework was utilized to detect points and then scale selection was applied to select characteristic points in Mikolajczyk and Schmid's article [17]. A scale-invariant corner detector, named Harris-Laplacian, and a scale-invariant blob detector, named Hessian-Laplacian were introduced to search points in an affine Gaussian scale space [18]. Another algorithm dealing with scale is ASIFT (Affine-SIFT) [19], which is inspired by the idea of combining simulation and normalization in SIFT. ASIFT simulates with enough accuracy all distortions caused by a variation of the camera optical axis direction, then applies the SIFT method. ASIFT is a full sense of affine invariant method.

The feature detection algorithms mentioned above are based on a scale space representation, and each image level is produced from the convolution of a variable-scale Gaussian function of the input image. The image will be blurred, and the high frequencies, which represent image details and features, will be lost with increasing scale parameter. (The details are often called features in some papers) Most applications currently apply a uniform change or proportional change to the Gaussian kernels; for instance, two nearby scales in SIFT are separated by a constant multiplicative factor $k$ [2]. However, the image details have a non-uniform distribution, and the number of them on each Gaussian image level in the pyramid cannot change uniformly or proportionally, and their distribution and intensity can be used to reflect the image content with the advantage of fewer calculations [11]. Under certain conditions, all of the original image information can be contained in such features. However, as a simplified representation of the original image, the use of scale space inevitably leads to loss of image information. The concept of metrics for information quantization was first proposed by Shannon in 1948 and extended by Sporring [20, 21], who presented a quantitative measurement of information. This method had theoretical and practical value for the quantitative description of the image information at each scale and the loss of information between scales. Nevertheless, it did not meet some basic requirements for describing the information content of smooth images in scale space from visual theory and intuition, and had high computational complexity. The technique for measuring the information content of discrete images suggested in [6] was defined as the number of first-class and second-class points. The principle of this method is simple, but the calculation cost of looking for two classes of points traversing the entire image is too expensive.

Another category of detectors is mainly concerned with the speed of algorithms. A simple and efficient detector named FAST (Features from Accelerated Segment Test) which compares pixels only on a circle of radius around the point was introduced in [22]. Compared time-consuming of early approach, FAST can achieve real-time implementation by an efficient classification based on a decision tree. It has become the base of many other efficient detectors owing to its advantages of sufficient points, high precision and good distribution. However, FAST is not invariant to scale changes. A non-hierarchical structure, Ferns, was proposed by Ozaysal in 2010 in order to reduce computation complexity [23]. A large number of training samples based on Bayesian classification are used in Ferns to achieve rapid and stable matching for wide baseline image. A robust feature matching scheme which features can be matched in 2.3 μs by combination of FAST points and Ferns training sample was presented in [24]. This method requires little memory and is robust against viewpoint just in small range.

3. Our Approach

In our approach, the two input images are defined as the query and model image, respectively, and have been smoothed by Gaussian functions and converted to grayscale images. The procedure consists of the following steps: first, the number of FAST keypoints is adopted as the information content of image in scale space. Then, the appropriate scale parameters are determined by the differences in information content, which are obtained by subtracting adjacent Gaussian images to generate each interval for the pyramid. In addition, keypoint matching is performed at the existing image levels, and a construction method with a matching instruction strategy is used to control the number of levels in one pyramid.

A. Measurement of information content quantization for scale space representation

A measure of the information content quantization for scale space representation is supposed to reflect some properties of scalespace itself, such as non-negativity, causality, and grayscale inversion invariance, which are not only natural but also significant in visual behavior. According to the causality of scale space, images become smoothed and simplified gradually as Gaussian kernels of increasing size are used. Therefore, no more
new structures appear in the smoothed images in scale space, and points with small support regions tend to be smoothed out in content of images to decrease with increasing scale. Lowe stated that the information content of the image can be associated with the distribution and density of features [2]. For this reason, the number of features can be used as a measure of the visual information, and the existing efficient local invariant feature detectors are the first choice for achieving this challenge. In FAST, interest points are defined according to the Segment-Test algorithm [25, 26]. In Segment-Test, the structure of keypoints is analyzed directly in a relatively intuitive manner in accordance with the very nature of information content. In this paper, the number of FAST keypoints is applied as the information content of an image in scale space. A FAST point is classified according to Eq.(1):

\[
S_{p,q} = \begin{cases} 
  d, & I(q) \leq I(p) - \epsilon_d \\
  s, & I(p) - \epsilon_d \leq I(q) \leq I(p) + \epsilon_d \\
  b, & I(p) + \epsilon_d \leq I(q) \end{cases} \quad (darker)
\]

\[
\Rightarrow \epsilon \rightarrow \leq - \quad (2)
\]

A candidate point \( p \) is considered to be a FAST corner if one can find a sufficiently large set of pixels that includes \( N \) pixels on a circle of fixed radius around the point \( p \), where \( N \) is larger than the setting threshold \( n \), such that these pixels are all significantly brighter or darker than the central point.

For any discrete image \( I \), let \( P \) be the set of keypoints extracted from \( I \):

\[
P = \{ p \in P \mid N \geq n \}
\]

Then, the information content of the discrete image \( I \) can be defined as \( E = |P| \), where \(|P|\) indicates the number of elements in the set \( P \).

Next, a numerical test is conducted to verify that the proposed measure of information content quantization can meet the required causality for a scale space representation, in which the number of FAST-9 keypoints is used as the information content of the smoothed versions in a typical SIFT pyramid. The actual computation scheme of the SIFT LoG pyramid and the Gaussian images at various scales are illustrated in Fig. 1.

![Fig. 1. Overview of the SIFT LoG (Laplace of Gaussian) pyramid. (a) Input Butterfly image (resolution: 400 × 300) [27]. (b) and (c) Smoothed versions of Butterfly image in SIFT, where \( O = 3 \), \( S = 3 \), \( \sigma_0 = 1.6 \).](image)

For each octave of scale space, the initial image is smoothed several times with a Gaussian convolution mask to produce the set of scale space images. Each octave of scale space is divided into an integer number \( S \) of intervals. After each octave, the Gaussian image is down-sampled by a factor of 2, and the process is repeated. The scale parameter \( t \) describes the current image level of scale resolution, and \( t = \sigma^2 \), where \( \sigma^2 \) is the variance of the Gaussian function. Parameter \( t \) is sometimes replaced by \( \sigma \) to describe the scale of the current image level [11]. In SIFT, the original image has a blur of \( \sigma = 0.5 \), and two nearby scales are separated by a constant multiplicative factor \( k \) [2]; here \( \sigma_{s+1} = k\sigma \), where \( \sigma \) is treated as the relative scale parameter. As displayed in Fig.1 (c), the input Butterfly image [27] becomes blurred with proportionally changed scales, and small-scale features tend to merge gradually and be smoothed out in the coarse-to-fine processing, which is in conformance to the requirement of causality in scale space representation.

The information content of each of the six continuous blurred images in the first octave are calculated to examine the relation between the information content and the located scale of the corresponding image, as shown in Fig. 2. There, \( \sigma \) is the absolute scale parameter.

![Fig. 2. Information content versus located scale of six continuous blurred Butterfly images in the first octave of the SIFT pyramid.](image)
As illustrated, the information content first decreases dramatically and then gently drops toward zero, which indicates that this method satisfies the basic principles proposed for the information content of a scale space representation according to visual saliency and image content. It is feasible to use the number of FAST keypoints as the information content in the scale space representation; this provides a reliable basis for achieving adaptive multiscale feature selection.

However, the large differences in information content between adjacent Gaussian images is problematic. More detailed data can be found in Table 1: here $\sigma$ is the relative scale parameter. The information content difference between the first and second image levels is as high as 2416, which results in a huge number of features with scales under 1.226 that cannot be reflected. Beginning with the fourth image level, the differences in information content between adjacent Gaussian images are less than 100. If an image level that contains so little information is generated, the problems of increasing computational complexity and the repeatability of interest points will arise. These will not be conducive to feature extraction.

### B. Adaptive multiscale feature detection algorithm

To solve the existing problems with Gaussian filtering using fixed scale spaces that are separated by a constant multiplicative factor $k$, here we propose an adaptive multiscale feature detection algorithm. As described above, the number of FAST keypoints is used as the information content in the scale space representation, and the differences in the information content of adjacent Gaussian images are applied as the criteria for the degree of changes in details in order to adjust the parameter $\sigma$ and achieve uniform changes in details between adjacent Gaussian images.

The SIFT scheme is referenced to build a pyramid having $O$ octaves, and each octave is divided into $S$ intervals. For convenience, only the procedure for the stack of blurred images in the first octave is presented here. For the remaining octaves, the Gaussian image at the first level of each octave $L(x, y; \sigma(a, 0))$ must be down-sampled by a factor of 2 from the Gaussian image $L(x, y; \sigma(o-1, S-1))$ in the previous octave, and the process is repeated. For a given initial grayscale image $I$, the procedure is applied as follows:

**Step1** Compute the information content $inf_0$ of the initial image $L_0$ using Eqs. (1) and (2); here, $L_0 = I$.

**Step2** Generate $L_{s+1}$ by convolving the initial image $L_0$ with Gaussian scale parameter $\sigma_{s+1}$, with $s \in [0, S-2]$; when $s = 0$, then $\sigma_{s+1} = \sigma_0$, where $\sigma_0$ is the amount of prior smoothing.

**Step3** Compute the information content $inf_{s+1}$ of $L_{s+1}$ using Eqs. (1) and (2).

**Step4** Compute the difference between two adjacent Gaussian images, $diff_{inf} = inf_{s} - inf_{s+1}$, with $s \in [0, S-2]$.

**Step5** Determine the value of $\sigma_{s+1}$. The threshold of the difference between two adjacent Gaussian images is set to $\varepsilon_1$ and $\varepsilon_2$, respectively. If $\varepsilon_2 \leq diff_{inf} < \varepsilon_1$, which indicates that the current value of $\sigma_{s+1}$ can guarantee uniform changes in the image details, then $s = s + 1$; proceed to Step 2. Otherwise, go to the following steps, in which the different values of $\sigma_{s+1}$ and $diff_{inf}$ in the adjustment process are expressed as $\sigma_{s+1}(i)$ and $diff_{inf}(i)$, respectively, $i \geq 1$:

1. Go to Step 2 with $\sigma_{s+1}(i+1) = \sigma_{s+1}(i) + 0.1$ if $diff_{inf}(i) < \varepsilon_1$, and gradually increase $\sigma_{s+1}$.
2. Go to Step 2 with $\sigma_{s+1}(i+1) = 0.5 \times \sigma_{s+1}(i)$ if $diff_{inf}(i) \geq \varepsilon_1$, and reduce $\sigma_{s+1}$.
3. Go to Step 2 with $\sigma_{s+1}(i+1) = 0.5 \times (\sigma_{s+1}(i-1) + \sigma_{s+1}(i))$; all of the following steps are then limited to Steps 2, 3, 4, 5 (4), and 5 (5) if the condition of the difference in information content $diff_{inf}$ changes from $diff_{inf}(i-1) < \varepsilon_1$ to $diff_{inf}(i) \geq \varepsilon_1$, or changes from $diff_{inf}(i-1) \geq \varepsilon_1$ to $diff_{inf}(i) < \varepsilon_1$ in the process of $\sigma_{s+1}$ variation.

4. Go to Step 2 with $\sigma_{s+1}(i+2) = 0.5 \times (\sigma_{s+1}(i+1) + \max(\sigma_{s+1}(i-1), \sigma_{s+1}(i)))$ if $diff_{inf}(i+1) < \varepsilon_1$.

5. Go to Step 2 with $\sigma_{s+1}(i+2) = 0.5 \times (\sigma_{s+1}(i+1) + \min(\sigma_{s+1}(i-1), \sigma_{s+1}(i)))$ if $diff_{inf}(i+1) \geq \varepsilon_1$.

Iterate through all possible image levels.

It is crucial to determine the values of $\varepsilon_1$ and $\varepsilon_2$, which control the information content alteration between two nearby Gaussian images, which is supposed to be uniform. Ideally, the information content of the current image level decreases by $\varepsilon_2 = diff_{inf} = inf_0 / S$ compared to that of the previous one, and then the information content of the final image level is reduced to $inf_0 / S$. In extreme cases, the information content of the last image level goes to zero, and the decrease between two adjacent smoothed images is $\varepsilon_1 \geq diff_{inf} = inf_0 / (S-1)$. Therefore, in a difference in information content $diff_{inf}$ in the range of $[\varepsilon_1, \varepsilon_2]$ is considered to be reasonable.

![Fig. 3. Relation between $\sigma$ and the average number of calculations required for determining the next scale $\sigma_i$ at different sampling frequencies $S$.](image-url)
The computational complexity depends in part on the selection of the prior scale of smoothing. Figure 3 shows an experimental determination of the amount of prior smoothing, \( \sigma \), that is applied to the first level of each octave to build the scale space representation with different numbers of scales sampled per octave. The abscissa and ordinate in Fig. 3 represent the value of \( \sigma \) and the average calculation times required for determining the next scale \( \sigma \), respectively. The database used for this test includes 50 real images drawn from a diverse range, including buildings, outdoor scenes, statues, and messy scenes. The average number of calculation reaches a minimum at the bottoms of the curves, which are denoted as \( \sigma_{\text{min}} \) and \( \sigma_{\text{max}} \), with \( \sigma_{\text{min}} < \sigma_{\text{max}} \). According to the causality of information content in scale space, \( \sigma_{\text{min}} \) represents the scale value when the information content varies in the ideal situation, and \( \sigma_{\text{max}} \) is the corresponding scale when the information content changes in extreme cases. Therefore, \( \sigma_{\text{min}} \) is selected as the optimal prior scale of smoothing. The curves in Fig. 3 show that when \( \sigma \in [0.1,0.5] \), the sampling frequency has little effect on the average number of calculations of \( \sigma \); however, the computational burden increases greatly with increasing number of scales sampled per octave when \( \sigma \in [0.5,2] \). It is easy to prove that the allowable scope of the difference in information content is narrowed with increasing sampling frequency, which increases the computation time.

For comparison with SIFT, our adaptive multiscale feature detection algorithm with \( S = 6 \) was applied to build the first octave of the Butterfly image's pyramid (Fig. 4). On the basis of

\[
\begin{align*}
\text{Relative scale parameter } \sigma \text{ } & \text{Location (o,s) } \\
0.500 & (0, 0) \\
0.453 & (0, 1) \\
0.503 & (0, 2) \\
0.603 & (0, 3) \\
0.903 & (0, 4) \\
1.603 & (0, 5)
\end{align*}
\]

Table 2. Details of data in Fig.5

For an initial grayscale image \( I \), the Gaussian image in the pyramid is expressed as \( L(x,y,\sigma(o,s)) \) with \( o \in [0, O - 1], s \in [0, S - 1] \). The initial image is \( L(x,y,\sigma(0,0)) = I \). For the remaining octaves, the Gaussian image at the first level of each octave \( L(x,y,\sigma(o,0)) \) must be down-sampled by a factor of 2 from the Gaussian image \( L(x,y,\sigma(o-1,S-1)) \) in the previous octave. The number of octaves is determined by the minimum number of pixels in the top-level smoothed image in the pyramid. For an initial grayscale image \( I \) with a size of \( width \times height \) pixels, the minimum number at the top level is \( 2^n \), \( n \in [0, \log_2(\min(width, height)) - 1] \). The number of octaves \( O \) can be computed by Eq. (3):

\[
O = \log_2(\min(width, height)) - n \tag{3}
\]

The procedure for construction with a matching instruction strategy is illustrated in Fig. 6, where \( P \) is the set of keypoints, and \( P(o,s) \) represents the set of FAST keypoints in each image level. \( P_{\text{match}} \) is the set of matching points, and \( N_{\text{match}} \) is the threshold.

Unlike previous algorithms that are applied throughout the entire pyramid, here construction with a matching instruction strategy is performed such that the building of the pyramid will end if an adequate number of correct pairs of matches have been obtained. Thus, the advantages of low computation complexity can be fully realized, and the problem of image roughness from bottom to top in the quantitative processing of the pyramid can be avoided.
In view of the selection of the threshold $N_{\text{Match}}$, 20 pairs of correct matches were used in [19] as the minimum judgment data in accordance with the matching effectiveness. The performance of SIFT and SURF [28] was evaluated by the matching ratio [9]: the greater the matching ratio is, the better the detectors will be able to discriminate in a large database. In a certain range of the scale, viewpoint, and illumination in structural images, SIFTs and SURF can achieve a relatively stable matching ratio of 40%, whereas outside of the range, the matching ratio dropped dramatically to about 10%. However, both SIFT and SURF provided robust matching to blurred images, with a matching ratio of up to 70%. Furthermore, little is known in advance about the degree of variance in the images to be matched. To reduce the difficulties in parameter selection, a large number of experiments were conducted. It was generally found that a good trade-off can be obtained between reliability and computational complexity when the threshold was set to 25% to 30% of the information content of the initial image in the pyramid.

$$\text{Match} \geq 1$$

$$s = s + 1$$

Fig. 6. Flow graph of construction with matching instruction strategy.

4. Experimental Results

The image matching performance of our approach was compared with that of state-of-the-art methods using the detectors SIFT [2], FAST [22], and ASIFT [19], all in combination with the most popular SIFT descriptor. Then the Euclidean distance between feature vectors was used as the similarity measure, and the nearest neighbor distance ratio method [15] was applied to obtain the matches. Finally, the RANSAC algorithm [29] was used to delete false matches. To provide a generic feature extraction algorithm, the parameters remain the same throughout the experimental process: the minimum number of pixels in the top-level smoothed image in the pyramid is set to 32. Then the number of octaves can be computed by Eq. (3). An experimental determination of the sampling frequency in [2] proved that there is a trade-off between the repetition rate of extracted features and the computational efficiency when the sample frequency of every group of the image pyramid is set to 3. In light of the experimental results displayed in Fig. 3, the sampling frequency was set to 3, and the corresponding prior scale of smoothing was $\sigma(a,0) = \sqrt{2}/2$. The threshold $N_{\text{Match}}$ can be determined using Eq. (4):

$$N_{\text{Match}} = 0.25 \times \min \left( P\_{\text{left}}(0,0), P\_{\text{right}}(0,0) \right)$$

In the experiments, the OpenCV 2.4.3 software was used for the SIFT and FAST detectors and the SIFT descriptor. The ASIFT detector was provided by the authors and was downloadable from [7]. In the following, the methods will be named after their detectors, namely our method, SIFT, FAST, and ASIFT. The experimental results are organized in four parts (4.1–4.4) in which the scale, illumination, viewpoint, and multiple conditions change, respectively.

A. Changes in scale

To demonstrate the robust performance of our method against scale variation, we tested four pairs of images to be matched that were drawn from a diverse range, including buildings, statues, and messy scenes at different scales using our method, SIFT, FAST, and ASIFT. Both the query and model images from [30] have a uniform size of 200×256 pixels.
Fig. 7. Performance of the methods under scale change. (a, f, k, p) Original images to be matched. Correspondences between (a, f, k, p) using our method (b, g, l, q), SIFT (c, h, m, r), FAST (d, i, n, s), and ASIFT (e, j, o, t).

The number of correct matches and the running time of each original pair of images for the four methods are shown in Table 3 for statistics and reports. Although SIFT achieves the highest speed, there are no effective matches between the query and model images in Fig. 7 (a, f, k, p). ASIFT works well, as shown in Fig. 7 (j, o), with a higher computational cost than our method, but it fails when the scale change is substantial [Fig. 7 (o)]. FAST exhibits similar performance to our method and high speed; however, our approach produces more matches with a good distribution than FAST.

Figure 7 (j) shows that our approach significantly improves the results under a strong scale transformation. It is very robust to the scale change and can achieve a large number of correct matches that densely cover the images; this features are inherited from the FAST detector. Our method is also relatively fast and will be very useful for various applications.

<table>
<thead>
<tr>
<th>Method</th>
<th>Our method</th>
<th>SIFT</th>
<th>FAST</th>
<th>ASIFT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image</td>
<td>(b)</td>
<td>(g)</td>
<td>(l)</td>
<td>(q)</td>
</tr>
<tr>
<td>Number of matches</td>
<td>178</td>
<td>150</td>
<td>106</td>
<td>50</td>
</tr>
<tr>
<td>Time (s)</td>
<td>7.06</td>
<td>5.15</td>
<td>15.2</td>
<td>6.04</td>
</tr>
</tbody>
</table>

B. Changes in illumination

Our method and the other three state-of-the-art algorithms (SIFT, FAST, ASIFT) were applied in an illumination-invariant manner to test the performance in two groups of images to be matched. The results are shown in Fig. 8, and the number of correct matches and running time of each pair of images analyzed by each algorithm are recorded in Table 4. The difference in light between the query and model images in Fig. 8(a) is caused by a change in the weather. Further, the contrast in the illumination intensity in Fig. 8(b) yields very different conditions of light and dark. Both groups of images (from [30]) have the same size, 256 × 200 pixels. Again, the SIFT matching procedure fails in both groups of images to be matched, as shown in Fig. 8(c) and (h). For the first group, the distribution of keypoints with ASIFT is good in Fig. 8(e); however, there are few correct matches and too many false matches between the query and model images, with large time consumption. FAST produces a modest number of valid matches at high speed, yet it is evident in Fig. 8 (d) that very few points are matched in the middle of the building.

For the second group of images, the ASIFT algorithm yields no valid matches. FAST works well and produces a reasonable number of correct matches, as shown in the image pair in Fig. 8(o).
Among the methods under comparison, our method provides the most robust matching across a substantial range of changes in illumination caused by weather and diurnal changes. Our method outperforms SIFT and ASIFT in terms of the number of valid matches, matching accuracy, and efficiency. In terms of speed, our method is slightly slower than FAST; nevertheless, the distribution of valid matches is better than that of FAST.

### Changes in viewpoint

The robustness of our method against changes in viewpoint was checked using four groups of images that have a uniform size of $256 \times 180$ pixels from [19, 30]. The four methods are applied to these image pairs. The results are shown in Fig. 9, and the number of valid matches and the running time of image pairs (a, f, k, p) are shown in Table 5.

SIFT can achieve correct matching when the number of matches is small. FAST produces a modest number of valid matches at high speed. ASIFT in particular shows excellent performance across a substantial range of changes in 3D viewpoint in (o) and (t), and obtains more valid matches than our method. We see that our method, FAST, and ASIFT are all robust to variations in viewpoint. However, our method is the only one that works well for all four groups of images. In image pairs (g, h, i, j), our method generates large numbers of features that densely cover the image over the entire range of viewpoints to find the same object in both the query and model images accurately, and the other three methods fail.

We also note that our method has no overwhelming advantage compared to SIFT, FAST, and ASIFT in terms of the number of valid matches, which is strongly related to the selection of the threshold $N_{\text{match}}$. For all the experiments in this section, a good trade-off between the matching performance and computational cost can be achieved when the number of correct matches reaches the threshold $N_{\text{match}}$, at which no more Gaussian images will be generated.
Fig. 9. Performance of the methods under viewpoint change. (a, f, k, p) Original images to be matched. Correspondences between (a, f, k, p) using our method (b, g, l, q), SIFT (c, h, m, r), FAST (d, i, n, s), and ASIFT (e, j, o, t).

Table 5. Number of correct matches and running time of Fig. 9(a, f, k, p) analyzed by our method, SIFT, FAST, and ASIFT

<table>
<thead>
<tr>
<th>Method</th>
<th>Our method</th>
<th>SIFT</th>
<th>FAST</th>
<th>ASIFT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(b)</td>
<td>(g)</td>
<td>(l)</td>
<td>(q)</td>
</tr>
<tr>
<td>Number of matches</td>
<td>90</td>
<td>121</td>
<td>225</td>
<td>122</td>
</tr>
<tr>
<td>Time (s)</td>
<td>8.7</td>
<td>18.6</td>
<td>13.8</td>
<td>10.8</td>
</tr>
</tbody>
</table>

D. Comprehensive test

In practical application, the differences between the query and model images are not always due to a single type of change, and variations in scale, illumination, viewpoint, and rotation tend to exist simultaneously. In Fig. 10(a), each image is thus subjected to a range of transformations, including rotation, scaling, affine stretching, changes in brightness, and changes in viewpoint. The correspondences between any two images in (a) in terms of the number of correct matches using each of the four algorithms are summarized in Fig. 10(b). Note that there are also illumination, viewpoint, scale, and zoom changes between (a)1 and (a)5. The images in the top row in (a) are 200×256 pixels in size, and the others are 256×200 pixels; all are downloadable from [30].

As shown in (b), when the differences between the query and model images involve illumination, all the ASIFT matching procedures fail, for instance, the matches between (a)1 and each of (a)2, (a)3, (a)4, and (a)5. ASIFT is not designed to cover the entire illumination range, so its performance drops quickly under substantial illumination changes. ASIFT does not demonstrate its strength with changes in 3D viewpoint, which indicates that it adapts relatively poorly to the environment. SIFT can achieve effective matches only in a small range of scaling and viewpoint. FAST works well in most of the image-matching tasks, and even obtains more valid matches than our method. This is entirely because, as a corner detector, FAST’s ability to locate corners decreases as images become blurred, whereas in the matching procedures in our method, the query and model images are first smoothed by a Gaussian function and converted to grayscale images, which decreases the number of FAST keypoints.

Fig. 10. Performance of the methods under diverse changes. (a) Original images to be matched. (b) Summary of the performance of each algorithm in terms of the number of correct matches; the abscissa, \( \frac{m}{n} \), represents the number of matched images in (a).
To summarize, experiments were performed to show that the proposed algorithm can simultaneously adapt to changes in location as well as scale, illumination, and viewpoint, and can generate large numbers of features densely covering the images over a large range of these variations. Our approach significantly improves the results for complex environmental changes and will be very useful for various applications.

5. Conclusions

An adaptive multiscale feature detection algorithm using information content quantization is proposed in this paper. It can generate large numbers of features densely covering the images under variations in scale, illumination, and viewpoint. First, the number of FAST keypoints is adopted as the information content of the image in scale space: experimental results prove that this method satisfies the basic principles proposed for the information content of scale space representation according to visual saliency and image content. Then, the difference in the information content is applied as the criterion for adjusting the parameters and achieving uniform changes in details between adjacent Gaussian images. Finally, a matching procedure using FAST keypoints is performed at the existing image levels by construction with a matching instruction strategy to control the number of levels per pyramid and reduce the computational cost. Compared to three state-of-the-art feature extraction algorithms (SIFT, FAST, and ASIFT), our method yields significantly improved results under strong transformations in scale, illumination, and viewpoint with adaptive selected parameters. Furthermore, the features detected by our method can be easily combined with many other types of feature vectors, and they will be very useful for various applications in complex environments.

Acknowledgments

This work was supported by the National Natural Science Foundation of China, Grant No. 51105027.

References