Research Article

Homogenized ORB Algorithm Using Dynamic Threshold and Improved Quadtree

Chaoqun Ma , Xiaoguang Hu , Jin Xiao , and Guofeng Zhang

State Key Laboratory of Virtual Reality Technology and Systems, School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China

Correspondence should be addressed to Jin Xiao; xiaojin@buaa.edu.cn

Received 5 November 2020; Revised 11 December 2020; Accepted 20 December 2020; Published 5 January 2021

Academic Editor: Taoreed Owolabi

Copyright © 2021 Chaoqun Ma et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Oriented FAST and Rotated BRIEF (ORB) algorithm has the problem that the extracted feature points are overconcentrated or even overlapped, leading to information loss of local image features. A homogenized ORB algorithm using dynamic thresholds and improved quadtree method is proposed in this paper, named Quadtree ORB (QTORB). In the feature point extraction stage, a new dynamic local threshold calculation method is proposed to enhance the algorithm's ability to extract feature points at homogeneous regions. Then, a quadtree method is improved and adopted to manage and optimize feature points to eliminate those excessively concentrated and overlapping feature points. Meanwhile, in the feature points optimization process, different quadtree depths are set at different image pyramid levels to prevent excessive splitting of the quadtree and increase calculation speed. In the feature point description stage, local gray difference value information is introduced to enhance the saliency of the feature description. Finally, the Hamming distance is used to match points and RANSAC is used to avoid mismatches. Two datasets, namely, the optical image dataset and SAR image dataset, are used in the experiment. The experimental result shows that, considering accuracy and real-time efficiency, the QTORB can effectively improve the distribution uniformity of feature points.

1. Introduction

Image registration plays an important role in computer vision. Image registration is widely used in many aspects such as image matching [1–7], change detection [8, 9], 3D reconstruction [10–12], guidance [13–15], mapping sciences [16–21], and mobile robot [22, 23]. In general, image registration methods can be mainly divided into two kinds: gray-scale matching methods and feature-based matching methods. The image gray-scale matching method uses similarity measures such as correlation function, covariance function, and absolute judgement to determine the correspondence between two images. The image feature-based matching method mainly compares two images by analyzing the spatial location, color, texture, and shape to achieve the correct matching. Feature-based methods are widely concerned and adopted. Many improvements are also based on such algorithms, based on the reason that they are robust against variations in geometry and illumination [24].

Feature-based techniques are usually regarded as an issue of matching process between points due to the generality of point representation and easy extraction. This paper is also based on point-based methods.

Currently, there are many matching methods using image features. Feature-based description methods are mainly made up of two groups considering the different data types used to form the description: floating feature methods and binary feature methods. As for feature extraction algorithms based on floating-point description, there are many famous algorithms proposed in the past years. Among them, Scale-Invariant Feature Transform (SIFT) [25] is an excellent and outstanding algorithm. David Lowe firstly proposed the SIFT in 1999 and improved it furthermore in 2004. SIFT has strong matching ability, stable extracted feature points, and strong robustness under many conditions. SIFT adds a heavy burden to the estimate, however. The computational efficiency of SIFT is severely restricted by feature point detection method using extreme-scale space.

Meanwhile, the process of feature description on gradient histogram also causes a high time cost. Without compromising its efficiency too much, this weakness has led to comprehensive efforts to optimize its speed [26]. The Speeded-Up Robust Features, shortly named as SURF, algorithm was proposed in 2008 by Bay et al. [27] with reference to the SIFT algorithm. SURF has a high degree of appreciation for feature points, and it has well robustness under the conditions of illumination, viewing angle, and scale changes. At the same moment, to a considerable degree, it solves the shortcomings of high calculation complexity and long calculation time of the SIFT algorithm.

On the other hand, in recent years, numerous binary descriptor algorithms have been proposed [28–33]. The key to improving the performance of image matching algorithms is to generate descriptors with higher discrimination and less computation. Since binary descriptors are compact and can use bit operations to speed up matching, the required memory and calculations of binary descriptors are greatly reduced compared to nonbinary descriptors. Rublee et al. firstly [34] proposed an Oriented FAST and Rotated BRIEF (ORB) algorithm in 2011, which is essentially a combination of the Features from Accelerated Segment Test (FAST) [35] corner detection algorithm and the improvement of the Binary Robust Independent Elementary Features (BRIEF) [36] binary feature description algorithm. The ORB algorithm generates binary descriptors according to the point pairs selected by machine learning and introduces direction information to enhance the rotation invariance of the descriptor. ORB has fast calculation speed and low storage space consumption. Its calculation time is about one percent of SIFT and one tenth of SURF. However, the traditional ORB algorithm has poor matching quality and anti-interference ability. Based on the ORB algorithm, researchers have proposed many improved image matching algorithms. At the IEEE International Conference on Computer Vision (ICCV) conference in 2011, Leutenegger et al. proposed the Binary Robust Invariant Scalable Keypoints (BRISK) algorithm [29] which has better rotation invariance, scale invariance, and robustness. The BRISK algorithm mainly uses FAST9-16 for feature points detection and constructs an image pyramid to obtain scale invariance. For feature points description, the BRISK algorithm uses a circular sampling pattern to obtain binary descriptors. In 2015, Yang et al. proposed a local difference binary feature description method called Local Difference Binary (LDB) [37]. The LDB algorithm is improved based on the classic binary descriptor (just considering the regional intensity information), adding the gradient information in the horizontal and vertical directions to construct the descriptor. Meanwhile, LDB divides the feature point area into subblocks of different sizes, and the small subblocks can obtain enough local detailed information to improve the discrimination of the descriptor. Large subblocks can effectively remove noise, but are not sensitive to small changes, reducing the uniqueness of the descriptor. In 2017, Levi et al. proposed the Learned Arrangements of Three Patch Codes (LATCH) [38]. As an improvement on the original local binary feature description, the LATCH algorithm uses

multi-image block F norm comparison replacing the comparison of pixels' gray levels in ORB, which enhances the antinoise performance of the algorithm. In 2017, Mur-Arta [39, 40] uses the quadtree method to achieve the extraction of ORB feature homogenization. This method obviously improves the uniformity of the feature point distribution, but the algorithm operation time is obviously increased, and the efficiency needs to be improved. In 2019, Xin-Nan et al. used an adaptive threshold to replace the fixed threshold artificially set in the original ORB algorithm and were able to extract more feature points. However, this method requires the calculation of threshold value for each pixel, which results in the increase of algorithm operation time [41]. In 2020, Ma et al. proposed a feature description method combining multiple image block comparison method and added local gray difference information based on the original gray intensity information to improve the precision of the ORB algorithm [42].

This paper researched the feature points detection and feature points description stage. Through analyzing the principle of ORB algorithm, the following shortcomings were concluded from ORB algorithm:

- (1) For feature points detection, the ORB algorithm adopts the FAST point extraction algorithm. FAST compares the gray values of pixels with surrounding pixels to select the more prominent pixels in the image as feature points. The FAST algorithm, however, uses an arbitrarily set global fixed threshold, and the effect is that the detected feature points are too clustered or even overlapping. In areas where the image changes are not obvious, only a few or even no feature points can be extracted. Generally speaking, the more feature points, the more accurate the image matching obtained, but the dense distribution of feature points is very unfavorable for the subsequent feature description and will affect the accuracy of the system when applied to certain scenes [43], such as video tracking and image navigation.
- (2) The ORB adopts the enhanced BRIEF description method to describe the feature points. The BRIEF algorithm generates binary feature descriptions by randomly selecting pixels in the image window of feature points to compare gray values. But in the process of comparing the gray value of pixels, only the relationship between the pixel points gray value is used. However, the image information contained in the pixel points gray value difference itself is not used, which will cause the loss of image information.

This paper focuses on the problems of the ORB algorithm mentioned above and proposes an enhanced ORB algorithm. The following contributions are principally made:

- (1) Mur-Arta [39, 40] adopted the FAST feature point extraction method using double threshold and the feature point management and optimization method using quadtree. The algorithm improved in Mur-Arta [39, 40] significantly improves the distribution

uniformity of the feature points. However, this algorithm still uses artificially set thresholds and does not consider the specific conditions of the pixel neighborhood image, and it cannot truly achieve adaptive extraction. This paper makes an improvement, based on the FAST, by using dynamic local threshold. This method calculates the local threshold based on the neighborhood image block to improve the feature point extraction ability of the original ORB algorithm in the area where the image change is not obvious.

- (2) Mur-Arta [39, 40] used a quadtree-based feature point management and optimization method to eliminate excessively concentrated and overlapping feature points. However, there is no constraint on the depth of the quadtree when the quadtree nodes are split, which creates too many splits and decreases the computational efficiency of the algorithm. This paper sets a maximum depth of the quadtree for different pyramid levels. By this way, the estimation of redundant feature points is thus decreased, and the performance of feature extraction is increased.
- (3) In the feature point description stage, this paper converts the gray value difference information between pixel point pairs into a binary code string through a specified threshold. Then this binary code string and the binary code string generated by the original ORB algorithm are fused to obtain a combined binary final feature description. By this method, each descriptor obtains more spatially supported visual information to make feature description less sensitive to noise.

The structure of the remaining chapters of this article is shown below. The feature points detection and description used in ORB are explained in Section 2. The improved feature point detection method, feature point optimization method, and feature point description technique proposed in this paper are illustrated in Section 3. The experimental indicators and outcomes on the optical image dataset and SAR image dataset are discussed in Section 4. The topic is discussed in Section 5. Section 6 is the conclusion.

2. Feature Point Detection and Description of ORB Algorithm

The ORB algorithm, respectively, uses the FAST method and the BRIEF method in the feature point extraction and feature point description stages. And ORB algorithm has, respectively, made improvements to the defects of the FAST method and the BRIEF method.

In the feature point extraction stage, the theory of the FAST is that the gray difference values between a pixel point and sufficient pixel points in its neighborhood are measured separately. Count the number of the gray difference values that exceeds the selected threshold t from these gray differences. If this number exceeds a specific number, the central point is regarded as a feature point. On the one hand, FAST has the disadvantage of being sensitive to noise. Therefore,

FAST cannot calculate the direction information that is required for the description of the feature. It is necessary to assign a direction to each feature point in advance. Therefore, ORB calculates the main direction for the feature point by the gray centroid process. The gray centroid method assumes that, from the feature point, the gray intensity center of the feature point neighborhood patch is offset. The vector from the feature point to its neighborhood window gray intensity center is defined to calculate an orientation [25]. On the other hand, FAST algorithm cannot extract feature points with scale invariance. To overcome this shortcoming, the ORB algorithm first constructs a scale space image pyramid and then uses the FAST to generate feature points at every layer image of the pyramid. The feature points selected by this method have scale invariance.

In the feature point description stage, the main idea of the BRIEF method is making use of random selected pixels comparison to describe a point. A binary string can be obtained from the comparison between the gray values of selected pixel pairs. Considering the operation speed, the significance and discrimination, the ORB algorithm uses a 256-dimensional descriptor. In general, the BRIEF method is very noise sensitive. Some improvement has been made by the ORB algorithm to enhance the antinoise ability of the BRIEF. First, define an image window centered on the feature point. The size of the window is set to 31×31 pixels. Then, randomly select enough number of 5×5 pixels size image blocks pairs in the image window. In the generation of the binary description vector, the gray integral of these 5×5 pixels image size block pairs are used instead of pixel pairs. Meanwhile, the BRIEF algorithm does not have rotational invariance. The gray centroid method is adopted in ORB algorithm to measure the directions of the feature point. Since that both the FAST method and the BRIEF method have the advantage of fast computing speed, the ORB algorithm is very prominent in computing speed.

3. Homogenized ORB Algorithm Based on Dynamic Threshold and Improved Quadtree

Our work mainly improves the feature point extraction and feature description stage. First, this paper proposes to calculate thresholds based on the gray values of different pixel neighborhoods instead of manually set thresholds, to improve the ability of our algorithm to extract feature points in homogeneous image regions. Secondly, an improved quadtree method for feature points management and optimization is adopted to improve the distribution uniformity of the detected feature points. Meanwhile, this paper set a specific limit quadtree splitting depth for each image pyramid to prevent excessive node splitting. In the feature description stage, the image information contained in the gray value difference is adopted to obtain more information support to enhance the discrimination of description.

3.1. FAST Feature Point Extraction Based on Dynamic Local Threshold. In feature point extraction, the FAST makes use of the gray difference values between a pixel point and its

surrounding 16-pixels. The number of points extracted largely depends on the threshold t mentioned in Section 2. Therefore, for different image regions, different t values should be selected based on concrete image environment. The threshold set by the ORB algorithm for FAST is an artificially set global threshold, without considering the specific situation of the local neighborhood of the pixel points. Therefore, the feature points detected by the ORB algorithm are concentrated in the areas where the image changes are obvious, and there are a lot of redundant feature points. For the uniform area of the image, the FAST algorithm can only extract a few feature points because of the similar gray-scale features of the image. Based on the ORB algorithm, Mur-Artal [39, 40] adopted a double-threshold for the FAST corner detection method. A high threshold is used in image areas with obvious features, while a low threshold is used in image areas where feature information is not obvious. By this method, the algorithm proposed by Mur-Artal [39, 40] can extract more feature points than ORB. However, its two thresholds are also artificially set, which has certain limitations and cannot be admirably adapted to all areas of the image.

In this paper, we adopt the dynamic local threshold method. First, the entire image is divided into grids according to the set grid size. Each grid serves as a separate image area. In each grid, the threshold t is calculated as follows:

$$t = \alpha \times \left(\frac{1}{m} \sum_{i=1}^m |f(Q_i) - \overline{f(Q)}| \right), \quad (1)$$

where α is a scale factor, m is the number of pixels points in the grid, $f(Q_i)$ is the gray value of the pixel point Q_i in the grid, and $\overline{f(Q)}$ is the average gray value of all pixel points in the grid. This paper verifies that when the value of α is 1.2, our algorithm can make a better matching effect.

3.2. Feature Point Management and Optimization Using the Improved Quadtree Method. The FAST algorithm based on dynamic local threshold proposed in Section 3.1 is first used to complete the feature point extraction. To further homogenize the feature points in the image, the quadtree method is used to divide the two-dimensional space of the image into blocks. Then, the feature points in each block are processed. The detailed steps are as follows:

Step 1. Take the entire image as the initial node of the quadtree segmentation to get the initial quadtree structure.

Step 2. Judge all nodes in the image. Assume that N_{kp} is the number of feature points contained in the node. If N_{kp} is equal to 0, delete this node. If N_{kp} is equal to 1, the node is no longer divided. If N_{kp} is greater than 1, the node continues to split into four child nodes.

Step 3. Repeat the second step until the number of nodes reaches the set number of required feature points, then the split of the quadtree ends.

Step 4. Select the feature points contained in each node. If N_{kp} is greater than 1, the feature point with the largest Harris response value is selected to present this node.

However, in the above steps, since the depth of the quadtree is not specifically set, the number of splits is too much, which brings a heavy computing load. This paper sets a different splitting depth for the quadtree based on the expected feature points in each pyramid image layer, to reduce the calculation of redundant features. The detailed process is shown in Algorithm 1. In Algorithm 1, d refers to the depth of the current quadtree; d_{max} refers to the maximum depth specifically set for current pyramid layer. N_{kp} refers to the number of feature points in the node. N_{store} refers to the number of stored nodes, and N_{set} refers to the expected number of feature points for the current pyramid layer. The maximum depth of the quadtree d_{max} should satisfy the following formula:

$$4^{d_{max}} \geq N_{set}. \quad (2)$$

3.3. Feature Description Method Fused with Gray Difference Information. As for the feature point description stage, the ORB adopts the steer BRIEF algorithm. As described in Section 2, in the feature point neighborhood image block, the BRIEF uses the gray value relationship of random pixel pairs to decide if the corresponding bit of the final binary feature definition is 0 or 1. But this method only uses the size of the gray value, and the quantization information of the difference between the gray values is not used, resulting in the loss of part of the image information. This paper proposes to use both gray value size information and gray value difference quantization information to describe feature points. The specific principle is expressed in Figure 1.

In Figure 1, this paper is built on basis of the improved BRIEF in ORB. We introduce the image information contained in the gray difference value of the pixel block pairs. Define a feature point Q ; according to the description method in Section 2, the ORB feature description method selects T groups of pixel block pairs for gray value comparison and obtains a binary code string, which is assumed to be D_W . Then, the gray difference value of each blocks pair is further recorded to generate another binary code string. A dataset G is defined to record the gray difference values:

$$G = \{G_t\}_{t=1\dots T} = \{|f(A_t) - f(B_t)|\}_{t=1\dots T}, \quad (3)$$

where f is the mean gray value of the image block.

For each feature point, T gray difference values can be obtained. Since the gray difference value is a floating-point value, it cannot be directly converted to a binary code. Therefore, the average value $G_{average}$ of these T gray difference values is calculated as a threshold to binary-code these gray difference values:

$$G_{average} = \frac{G}{T} = \frac{1}{T} \sum_{1 \leq t \leq T} G_t. \quad (4)$$

Compare the T gray difference values with the average $G_{average}$:

- (1) Initialize node
- (2) Split child node
- (3) **if** $d < d_{max}$ **then**
- (4) **if** $N_{kp} > 0$ **then**
- (5) **if** $N_{kp} > 1$ **then**
- (6) Split child node
- (7) **else then**
- (8) Store child node
- (9) **else then**
- (10) Store child node
- (11) **else then**
- (12) **if** $N_{store} > N_{set}$ **then**
- (13) Store the point which has the largest response in each node
- (14) **end**

ALGORITHM 1: Algorithm for improved quadtree method.

FIGURE 1: Binary description combined with gray difference value information.

$$h(Q, G_t) = \begin{cases} 1, & G_{average} < G_t, \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

Based on equation (5), a binary string $D_w^\wedge$ is generated:

$$D_w^\wedge = \sum_{1 \leq t \leq T} 2^t h(Q, G_t). \quad (6)$$

Finally, as the process in Figure 1, D_W and $D_w^\wedge$ are combined to generate a new binary description:

$$\text{Description}_{\text{keypoint}} = D_W + D_w^\wedge. \quad (7)$$

3.4. The Overall Workflow of the Improved ORB Algorithm. QTORB algorithm proposed in this work still follows the relevant steps of the ORB algorithm in the overall process as shown in Figure 2.

The detailed steps of QTORB algorithm are as follows in Algorithm 2. T is set as 128. f is the gray mean function.

4. Experiments and Results

A desktop computer with 64-bit Windows 10 system was used. The specific parameters are as follows: i7-7700, 3.60 GHz, and 16 GB. The simulation environment is Microsoft Visual Studio 2019. Simulation language adopts C++ combined with OpenCV 3.4.0. The test image datasets are the Oxford dataset [44] and the SAR dataset.

The comparative test algorithms include two types of matching algorithms: floating-point feature description and binary feature description. Algorithms based on floating-point feature description include SIFT [25] and SURF [27]. Algorithms based on binary feature description include BRISK [29], LATCH [38], LDB [37], TPLGD proposed in [42], ORB [34], the algorithm proposed in [39, 40] (in this paper, the symbol MRA refers to this algorithm), and the QTORB proposed in this paper. The detailed steps information of each algorithm is shown in Table 1. To further avoid mismatches, the RANSAC [45] algorithm is used for all tested algorithms. If the true match error is within 3 pixels, the keypoint pair is deemed to be the right match.

FIGURE 2: The overall workflow of the QTORB.

4.1. Evaluation Metrics. Four measurement criteria are used in experiment. Evenness is the first criterion. The Evenness [46] calculation setups are shown in Figure 3. Its basic principle is to use the mathematical characteristics between the number of feature points in the same area of various shapes of the image to measure the distribution of feature points. First, the image is divided into ten blocks from five directions: vertical, horizontal, 45 degrees, 135 degrees, internal, and external. By the division method in Figure 3, ten different regions can be obtained.

Count the number of feature points in each area as a dataset. Calculate the variance V of this dataset. The calculation of Evenness is shown in equation (8). The smaller the Evenness value, the better the uniform distribution of feature points.

$$\text{Evenness} = 101 \times \log(V). \quad (8)$$

The second criterion is Precision [44]. The calculation formula is equation (9). $\text{Num}_{\text{correct matches}}$ refers to the number of correct matches while $\text{Num}_{\text{all matches}}$ refers to the number of all matched points.

$$\text{Precision} = \frac{\text{Num}_{\text{correct matches}}}{\text{Num}_{\text{all matches}}} \times 100\%. \quad (9)$$

The third criterion is the Root-Mean-Square Error (RMSE). RMSE represents the positional distance between the feature point obtained by the matching algorithm and the real feature point. Assume that feature point $Ap(x_{ap}, y_{ap})$ and $Bq(x_{bq}, y_{bq})$ is a matched point pair obtained by the image matching algorithm. Feature point $Ap(x_{ap}, y_{ap})$ is located in image A , while $Bq(x_{bq}, y_{bq})$ is located in image B . The true position of point $Ap(x_{ap}, y_{ap})$ on image B is $Bp(x_{bp}, y_{bp})$. Then, the true position distance between point $Bq(x_{bq}, y_{bq})$ and point $Bp(x_{bp}, y_{bp})$ is as follows:

$$\text{RMSE} = \sqrt{\left(x_{bp} - \hat{x}_{bp}\right)^2 + \left(y_{bq} - \hat{y}_{bp}\right)^2}. \quad (10)$$

Image operation time is the fourth criterion. In this paper, algorithm operation time includes feature point extraction time, description time, and matching time. Input process time and indicators calculation time are not considered.

4.2. Oxford Optical Image Dataset. Provided by Mikolajczyk and Schmid [44], the Oxford dataset is a publicly measurable dataset. There are eight image classes in the dataset including image blur (Trees group and Bikes group), light (Leuven group), JPG compression (Ubc group), rotation scale conversion (Bark group, Boat group), and visual improvements (Wall group and Graf group). As shown in Figure 4, we will pick six image classes from the Oxford dataset.

There are six images prepared for experimentation in each image group. As an input image pair, the first image can be paired with the other five images. The homography matrix between two images in each image pair is provided. The ground truth of the two images can be calculated based on this homography matrix.

4.3. SAR Dataset. For tests, a SAR image dataset acquired by the UAV-borne SAR platform is used. These SAR images include several typical scenes of objects on the ground, including cities, lakes, fields, roads, hills, and rivers, in Bedfordshire, southeast England. For a SAR image, rotation and zoom operations are used to create an image to be matched. The SAR image and its image to be matched are matched as an input image pair. As for the ground truth for each image pair, the homography matrix between two images in each image pair can be calculated according to the

Input: Image data

Output1: Feature points

Output2: Feature points descriptions

- (1) Build Gauss pyramid bases on input image. The number of layers is set to 8 and the scale is 1.2
- (2) Gridding pyramid image. The grid size is set to 30×30 . In each grid, calculate threshold using Equation (1). Traverse all meshes of all pyramid images to detect feature points set $Q = \{Q_1, Q_2, \dots, Q_i, \dots, Q_N\}$
- (3) Optimize the feature points using improved quadtree method
- (4) **for** $i = 1, 2, \dots, N$ **do**
- (5) Define image block M of 31×31 pixels size around point Q_i
- (6) In M , randomly select T patches pairs with the centers of patches are $RP = \{RP_t\}_{t=1\dots T} = \{(x_{t1}, y_{t1}), (x_{t2}, y_{t2})\}_{t=1\dots T}$
- (7) Define 5×5 size patches W_{t1}, W_{t2} centered on point (x_{t1}, y_{t1}) and (x_{t2}, y_{t2}) , $S = \{W_t\}_{t=1\dots T} = \{[W_{t1}, W_{t2}]\}_{t=1\dots T}$
- (8) **for** $t = 1, 2, \dots, T$ **do**
- (9) **if** $f(W_{t1}) < f(W_{t2})$ then Corresponding bit = 1
- (10) **else** then Corresponding bit = 0
- (11) **end if**
- (12) **end for** (generate binary code D_W)
- (13) Calculate the gray difference based on all pairs $G = \{G_t\}_{t=1\dots T} = \{|f(W_{t1}) - f(W_{t2})|\}_{t=1\dots T}$, calculate the mean $G_{average}$ of all differences
- (14) **for** $t = 1, 2, \dots, T$ **do**
- (15) **if** $G_{average} < G_t$ then Corresponding bit = 1
- (16) **else** then Corresponding bit = 0
- (17) **end if**
- (18) **end for** (generate binary code D_w^Δ)
- (19) D_W and D_w^Δ are combined to obtain final Description_{keypoint} for each feature point
- (20) **end for**

ALGORITHM 2: Detailed operation steps of QTORB.

TABLE 1: The detailed related steps information of each algorithm.

Tested algorithms	Detection method	Description method	Matching distance	Description size	Error threshold	Mismatch elimination
SIFT	DOG	SIFT	Euclidean	128-dimension	3 pixels	RANSAC
SURF	Hessian	SURF	Euclidean	64-dimension	3 pixels	RANSAC
BRISK	BRISK	BRISK	Hamming	64-byte	3 pixels	RANSAC
LATCH	FAST	LATCH	Hamming	32-byte	3 pixels	RANSAC
LDB	FAST	LDB	Hamming	64-byte	3 pixels	RANSAC
TPLGD	FAST	TPLGD	Hamming	64-byte	3 pixels	RANSAC
ORB	FAST	BRIEF	Hamming	32-byte	3 pixels	RANSAC
MRA	FAST	BRIEF	Hamming	32-byte	3 pixels	RANSAC
QTORB	FAST	QTORB	Hamming	64-byte	3 pixels	RANSAC

specific angle degree and scale ratio. In Figure 5, some SAR images are shown.

4.4. Experiment Result on Oxford Dataset. Six image groups from the Oxford dataset are selected for experiment including Bikes (blur), Boat (rotation and scale), Graf (viewpoint), Leuven (illumination), Trees (blur), and Ubc (JPG compression). To ensure the generality of the test results, each image pair is tested thirty times. The average value of each data is used as the final experimental data.

4.4.1. Feature Points Evenness and Matching Effect on Oxford Dataset. The SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB algorithms performed feature point extraction and image matching on the images selected from the Oxford dataset. From the distribution of feature

points before and after matching, the performance difference of uniform distribution of feature points of the nine algorithms tested can be intuitively found.

Figure 6 shows us the feature point extraction results of the tested algorithms using the Bikes images group from the Oxford dataset. It can be intuitively found that the feature points extracted by the LATCH, LDB, TPLGD, and ORB algorithms are concentrated in image areas including more prominent edges while almost no feature points are extracted in the homogeneous area. Although the SIFT, SURF, and BRISK algorithms can extract feature points in the homogeneous area of the image, the number of feature points in the homogeneous area is not much. Most of the feature points are still concentrated in the edge area of the image, resulting in a large amount of overlap. As for the MRA and QTORB, the feature points are obviously more uniform than the other tested algorithms. The uniform difference between the MRA

FIGURE 3: Divide the image in multiple directions.

FIGURE 4: Optical images from Oxford dataset. (a) Bikes (blur). (b) Boat (rotation and scale). (c) Graf (viewpoint). (d) Leuven (illumination). (e) Trees (blur). (f) Ubc (JPG compression).

FIGURE 5: Eight SAR images. (a) Hills. (b) Lakes. (c) Town. (d) Road. (e) River. (f) Field. (g) Field. (h) Field.

FIGURE 6: The distribution of feature points extracted by the (a) SIFT, (b) SURF, (c) BRISK, (d) LATCH, (e) LDB, (f) TPLGD, (g) ORB, (h) MRA, and (i) QTORB algorithms on the Bikes group image from Oxford dataset

and QTORB algorithm is not obvious, but the QTORB algorithm can extract some feature points that the MRA cannot extract in some homogeneous area.

Based on the extracted feature points above, we continue to complete the image matching process. Figure 7 shows us the image matching effect of the tested algorithms using the Bikes images group from the Oxford dataset. The feature point pair connected by the green line is a correct matching pair, and the feature point pair connected by the red line is a wrong matching pair. Consistent with the distribution of feature points shown in Figure 6, Figure 7 shows that the correct matching feature points implemented by SIFT, SURF, BRISK, LATCH, LDB, TPLGD, and ORB algorithms also have a relatively serious centralized distribution. The matching results achieved by the MRA and QTORB algorithms are very uniform, and there are correct matches in both the homogeneous area and the edge area. The matching results achieved by the MRA and QTORB are relatively similar, making it difficult to distinguish the difference intuitively. To analyze the performance differences of the tested algorithms in more detail, the feature distribution uniformity of the selected six images groups of Oxford dataset is counted, as shown in Table 2.

From Table 2, under the six scene transformations provided by Oxford dataset, both MRA and QTORB can achieve more uniform feature point extraction than the other tested algorithms and provide a more reasonable distribution of feature points for subsequent image processing. Meanwhile, compared with MRA, the QTORB can better adapt to more scenarios.

4.4.2. Matching Precision Using Oxford Dataset Images. This paper has carried out image matching tests on the six images groups provided by Oxford dataset. In each group, five images pairs (from 1–2 to 1–6, the more backward the image pairs, the greater the change between images) need to be matched. The precision of each matching process is shown in Figure 8. The symbol “Mean” represents the average precision of the five matching processes in each group. The average precision data is counted in Table 3.

From Figure 8 and Table 3, for the image groups Bikes, Boat, Trees, and Ubc, SIFT is the best performer. The best performance on the Graf and Leuven image groups, respectively, is the MRA algorithm and the BRISK algorithm. The QTORB algorithm proposed in this paper has acceptable performance in terms of precision. The SIFT algorithm is the highest. The QTORB outperforms the SURF, LATCH, LDB, ORB, and MRA.

FIGURE 7: Image matching effect of the (a) SIFT, (b) SURF, (c) BRISK, (d) LATCH, (e) LDB, (f) TPLGD, (g) ORB, (h) MRA, and (i) QTORB algorithms on the Bikes group image from Oxford dataset.

TABLE 2: Mean evenness of feature points using Oxford dataset images.

Method	Bikes	Boat	Graf	Leuven	Trees	Ubc
SIFT	192.5424	203.3804	159.8065	234.0446	207.2986	253.872
SURF	254.0986	201.3406	159.55948	273.6258	231.2498	269.886
BRISK	196.2948	234.982	170.94194	251.4328	255.885	289.5048
LATCH	173.5252	137.4798	116.26842	174.9244	140.3472	181.9418
LDB	179.6288	143.53562	120.0294	177.5232	144.188	185.8752
TPLGD	174.0654	141.31146	116.88112	175.7902	151.1302	180.313
ORB	171.8528	148.43458	120.23718	175.1806	146.18394	180.0906
MRA	141.277	111.05174	84.45352	158.6698	112.15136	168.1602
QTORB	140.6476	92.43084	84.3705	168.1638	113.32876	166.0178

4.4.3. Matching RMSE Using Oxford Dataset Images.

Figure 9 shows the RMSE indicators of the SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB in different images change scenarios. In Table 4, among all the tested algorithms, the RMSE of the SIFT algorithm is the smallest, which means that the positioning is the most accurate. The performances of other algorithms are at the same level. The QTORB is inferior to the SIFT, SURF, and BRISK algorithms, while still better than LATCH, LDB, TPLGD, ORB, and MRA.

4.4.4. Operation Time Using Oxford Dataset Images.

Figure 10 visually shows the difference in operation time of SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB. Table 5 records the mean operation time and

shows that the running time of SIFT algorithm is the longest while ORB is the least time-consuming. The QTORB algorithm takes slightly longer than ORB, but still much shorter than MRA. At the same time, it also can be found that the real-time performance of the image matching algorithm based on the floating-point feature description is far lower than that based on the binary feature description.

4.5. Experiment on SAR Dataset. To check the performance on remote sensing SAR images of the tested algorithms, scaling, rotation, and mixing (scaling and rotation occur simultaneously) were evaluated for three types of image transformations. The scale ratio in the experiment is 0.9 and the angle of rotation is 10 degrees. In the experiment, a total of one hundred SAR images were used.

FIGURE 8: Matching precision based on the Oxford dataset. (a) Bikes (blur). (b) Boat (rotation and scale). (c) Graf (viewpoint). (d) Leuven (illumination). (e) Trees (blur). (f) Ubc (JPG compression).

TABLE 3: Mean precision using Oxford dataset images (%).

Method	Bikes	Boat	Graf	Leuven	Trees	Ubc	Mean
SIFT	81.7995	83.6288	44.5121	90.1414	73.1389	97.6497	78.4784
SURF	76.2795	58.6993	32.3778	81.0737	57.1606	81.4300	64.5035
BRISK	80.6069	74.9718	45.0690	93.7247	61.2645	94.1188	74.9593
LATCH	79.0836	66.1339	47.5753	84.3021	68.5101	96.4268	73.6720
LDB	75.1508	60.6045	43.5586	84.8627	65.3837	96.3330	70.9822
TPLGD	79.3059	79.4594	47.2453	87.0085	69.5843	96.7416	76.5575
ORB	78.0477	77.9540	42.6809	83.9069	67.3593	96.8749	74.4706
MRA	74.5541	74.3186	49.4520	87.8470	58.1138	94.9708	73.2094
QTORB	76.5233	76.3945	47.1812	88.9028	63.3976	95.3173	74.6195

FIGURE 9: Mean RMSE of the SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB algorithms based on Oxford dataset.

TABLE 4: Mean RMSE using Oxford dataset images (pixel).

Method	Bikes	Boat	Graf	Leuven	Trees	Ubc	Mean
SIFT	0.8991	0.8027	0.8451	0.5164	1.2837	0.5839	0.8218
SURF	1.0374	1.2399	1.2851	0.7172	1.2889	0.6797	1.0414
BRISK	1.1918	1.1204	1.2578	0.7383	1.3297	0.7363	1.0624
LATCH	1.2478	1.2461	1.2350	1.1041	1.4992	0.2999	1.1054
LDB	1.2872	1.2130	1.3515	1.0643	1.3777	0.3321	1.1043
TPLGD	1.2637	1.1817	1.2728	1.0898	1.4965	0.3122	1.1028
ORB	1.3030	1.0976	1.1418	1.1009	1.4386	0.3126	1.0658
MRA	1.2852	1.1061	1.1766	1.0050	1.5009	0.3295	1.0672
QTORB	1.2623	1.2487	1.0949	0.9953	1.4316	0.3515	1.0641

FIGURE 10: Mean operation time of the SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB algorithms based on Oxford dataset.

4.5.1. Feature Points Evenness and Matching Effect on SAR Dataset. This section tests the distribution uniformity of the feature points extracted by the SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB, applied to SAR images. In Figure 11, a SAR image containing a township scene is selected to display the distribution of feature points. It is intuitive to find that the feature points extracted by the SIFT, SURF, BRISK, LATCH, LDB, TPLGD, and ORB algorithms have obvious concentration in the town area at the bottom-left of the image. For other uniform regions, although SIFT, SURF, and BRISK can also extract feature points, most of the feature points are also concentrated on

the edge. The LATCH, LDB, TPLGD, and ORB basically cannot extract feature points in other uniform regions. Most of the feature points are concentrated in the towns and roads. Compared with the images provided by the Oxford dataset, the differences between the scenes contained in the SAR images are more significant, which makes the concentrated distribution of feature points more obvious. The MRA and QTORB algorithms greatly alleviate the concentrated distribution of feature points.

Figure 12 shows the image matching effect of SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB using SAR images. The feature point pair connected

TABLE 5: Mean operation time using Oxford dataset images (s).

Method	Bikes	Boat	Graf	Leuven	Trees	Ubc	Mean
SIFT	1.2626	2.9351	1.3366	1.0757	5.4473	2.1728	2.3717
SURF	1.4441	2.0138	1.6816	1.2808	3.1829	1.8514	1.9091
BRISK	0.3921	1.8734	0.6803	0.5525	2.2866	1.8438	1.2715
LATCH	0.3377	0.5061	0.3799	0.3295	0.7337	0.4505	0.4562
LDB	0.5474	0.7177	0.5917	0.5275	0.9340	0.6666	0.6642
TPLGD	0.2838	0.4718	0.3518	0.2472	0.7356	0.5093	0.4333
ORB	0.1737	0.2773	0.1932	0.1644	0.4079	0.2339	0.2417
MRA	0.3968	0.5823	0.3769	0.3300	0.8778	0.4623	0.5044
QTORB	0.2804	0.3437	0.2167	0.2609	0.4807	0.3400	0.3204

by the green line is a correct matching pair, and the point pair connected by the red line is a wrong matching pair.

From Figure 12, the distribution of the correct matched feature points by the SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB is still consistent with that of the feature points extracted in Figure 11. The performance of the MRA and QTORB algorithms is confirmed again. In SAR images experiment, three image transformation scenarios are set according to the scale ratio and rotation angle including scaling, rotation, and mixing (scaling and rotation exist simultaneously). From Table 6, the QTORB algorithm can achieve the maximum uniform distribution of feature points in all these three image transformation scenarios.

4.5.2. Matching Precision on SAR Dataset Images. In experiment, one hundred SAR images were used. Five SAR images were randomly selected to visually display the test results. Figure 13 shows the matching precision of SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB using the selected five SAR images. Figure 13 intuitively shows that the QTORB algorithm can achieve image matching well in the given three image scenarios. In most cases, it can achieve better matching precision than ORB and MRA.

Table 7 is calculated from all the one hundred tested SAR images. Under the three image transformation conditions specifically set, the highest matching precision is the SIFT. According to the mean data in Table 7, the QTORB algorithm is lower than the SIFT, BRISK, and TPLGD algorithms and higher than SURF, LATCH, LDB, ORB, and MRA.

4.5.3. Matching RMSE on SAR Dataset Images. Figure 14 and Table 8 show the RMSE indicators of the SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB tested on SAR images. Under the three image transformation conditions specifically set, the RMSE of the SIFT is the lowest, which means that the positioning accuracy of the SIFT is the most accurate. According to the mean data of the three cases in Table 8, the RMSE of SIFT and SURF is significantly lower than that of BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB. The RMSE of the QTORB algorithm is lower than that of the ORB algorithm.

4.5.4. Operation Time on SAR Dataset Images. Figure 15 and Table 9 record the operation time of SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB. It can be found that among all the algorithms tested, the SIFT algorithm takes the longest time and ORB algorithm takes the least time. The LATCH, LDB, TPLGD, MRA, and QTORB algorithms proposed based on the improvement of the ORB algorithm cost more time than ORB algorithm to different degrees. Compared with the MRA algorithm, the time-consuming increase of the QTORB algorithm is smaller. The operation time of QTORB is only slightly longer than that of ORB and still much less than that of MRA. It can also be concluded through experiments that the image matching algorithm based on binary feature description is much faster than the image matching algorithm based on floating-point feature description.

5. Discussion

The feature points detected by ORB algorithm and the matched feature points on subsequent matching are all concentrated and overlapping. To make the feature points evenly distributed, the MRA algorithm first adopted the quadtree method to manage and optimize the feature points. Its essence is to redistribute and optimize feature points by dividing the two-dimensional image space with the help of a quadtree. The algorithm achieves the goal of uniform distribution of feature points. In the feature point extraction stage, the MRA algorithm uses a lower threshold than the ORB algorithm, so MRA initially extracts more feature points than the ORB algorithm, and it should contain all the feature points extracted by the ORB. In fact, it can be understood that the feature points extracted by the ORB algorithm are the most prominent part of the feature points extracted by MRA. Therefore, in subsequent image matching, the matching precision of MRA is lower than the ORB algorithm. The QTORB algorithm proposed in this paper uses a dynamic threshold based on local pixels in the feature point extraction stage. Since the threshold value is not fixed, more high-quality feature points may be extracted in some areas than the ORB algorithm. Meanwhile, in feature description stage, gray difference value information is added to obtain more image space information, which improves the discriminative power of feature descriptors.

FIGURE 11: The distribution of points extracted by the tested algorithms using the selected SAR image containing a township scene. (a) SIFT. (b) SURF. (c) BRISK. (d) LATCH. (e) LDB. (f) TPLGD. (g) ORB. (h) MRA. (i) QTORB.

Therefore, the QTORB algorithm can improve the matching precision under the premise of ensuring the uniform distribution of feature points. In terms of operating speed, MRA does not limit the depth of the quadtree. The quadtree will be excessively split, which will reduce the computational efficiency. The QTORB algorithm limits the depth of the quadtree, which can end the continued splitting of deep nodes early. And if the node contains a lot of feature points, ending the splitting of the node early has a more significant effect on reducing the calculation time. The experiment uses a variety of comparative algorithms, including several other algorithms based on ORB improvements. The experimental results prove that the improved algorithms based on the ORB algorithm also have the problem of uneven distribution

of feature points, which is mainly caused by the feature point extraction algorithm adopted by the ORB algorithm. Therefore, the main method to improve the distribution of feature points is to improve the extraction algorithm and further carry out reasonable screening of the feature points after extraction. While realizing the homogenization of feature points, the matching precision also requires close attention. The matching precision is not only related to the extraction of feature points, but also more closely related to the description method. The stronger the discriminative power of the feature description vector, the better the precision. Therefore, while homogenizing the feature points, it is also necessary to improve the feature description method. On the premise of ensuring good matching

FIGURE 12: Matching result of the (a) SIFT, (b) SURF, (c) BRISK, (d) LATCH, (e) LDB, (f) TPLGD, (g) ORB, (h) MRA, and (i) QTORB algorithms using the selected SAR image containing a township scene.

TABLE 6: Mean evenness of feature points on SAR dataset.

Method	Scale = 0.9	Rotation = 10	Scale = 0.9 Rotation = 10	Mean
SIFT	259.9164	257.2572	255.6382	257.6039
SURF	242.1884	246.6370	236.4556	241.7603
BRISK	263.5500	267.2996	263.2762	264.7086
LATCH	182.0922	177.5690	185.4978	181.7197
LDB	180.8172	179.3682	173.2460	177.8105
TPLGD	184.1988	170.0662	167.3348	173.8666
ORB	179.6610	171.1720	185.9520	178.9283
MRA	162.9324	157.5000	146.1532	155.5285
QTORB	159.4978	153.2828	141.2876	151.3561

FIGURE 13: Continued.

FIGURE 13: Matching precision based on the selected SAR images. Three transformations are used: (a) scaling (scale = 0.9), (b) rotation (rotation = 10), and (c) mixing (scale = 0.9, rotation = 10).

TABLE 7: Mean precision based on SAR dataset (%).

Method	Scale = 0.9	Rotation = 10	Scale = 0.9 Rotation = 10	Mean
SIFT	99.7279	99.7616	99.7268	99.7388
SURF	96.1932	96.3977	94.3978	95.6629
BRISK	97.3116	97.9397	98.3298	97.8604
LATCH	96.7398	96.2728	96.7039	96.5721
LDB	97.2395	95.9803	95.8465	96.3554
TPLGD	97.9655	96.5666	96.5133	97.0151
ORB	94.7575	95.7366	95.5876	95.3606
MRA	95.4964	95.8518	97.2326	96.1936
QTORB	95.8321	95.9272	98.1136	96.6243

FIGURE 14: Mean RMSE of SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB using the SAR dataset images.

TABLE 8: Mean RMSE based on SAR dataset (pixel).

Method	Scale = 0.9	Rotation = 10	Scale = 0.9 Rotation = 10	Mean
SIFT	0.1090	0.1123	0.1242	0.1151
SURF	0.5088	0.4728	0.6219	0.5345
BRISK	0.6928	0.4994	0.6046	0.5989
LATCH	0.9015	0.8752	0.8837	0.8868
LDB	0.8025	0.8148	0.8154	0.8109
TPLGD	0.7740	0.9151	0.8807	0.8566
ORB	0.8776	0.8965	0.9201	0.8980
MRA	0.8780	0.9098	0.8519	0.8799
QTORB	0.8748	0.9214	0.8824	0.8929

FIGURE 15: Mean operation time of the SIFT, SURF, BRISK, LATCH, LDB, TPLGD, ORB, MRA, and QTORB based on the SAR dataset.

TABLE 9: Mean operation time based on SAR dataset (s).

Method	Scale = 0.9	Rotation = 10	Scale = 0.9 Rotation = 10	Mean
SIFT	1.1625	1.3485	1.2700	1.2603
SURF	0.6678	0.9232	0.7842	0.7917
BRISK	0.5391	0.5890	0.5611	0.5631
LATCH	0.4337	0.4374	0.4355	0.4355
LDB	0.4592	0.4609	0.4346	0.4516
TPLGD	0.3319	0.3143	0.2959	0.3140
ORB	0.1750	0.2047	0.1935	0.1911
MRA	0.3215	0.3688	0.3416	0.3439
QTORB	0.2257	0.2539	0.2342	0.2379

precision and real-time performance, the uniform distribution of feature points can be achieved as much as possible.

6. Conclusions

This paper proposes a uniform ORB algorithm based on local dynamic threshold and improved quadtree, defined as Quadtree ORB (QTORB). QTORB, based on the ORB algorithm, focuses on research and improvement in the feature point extraction stage and feature point description stage. First, a feature point extraction method based on local dynamic threshold is adopted. This method can extract feature points uniformly on the image. After extracting the feature points, this paper draws on the quadtree-based method proposed by MRA to manage and optimize the feature points, avoiding excessive concentration and overlap of feature points. At the same time, in order to solve the time-consuming problem of the quadtree method, a limit is set on the split depth of the quadtree, which effectively reduces the calculation time. In the feature point description stage, an improved feature description method is proposed, which binarizes the image gray level difference information and implements a new binary feature using the feature fusion method. Through the verification of Oxford and SAR datasets, the QTORB algorithm proposed in this paper can effectively realize the uniform distribution of feature points under the premise that the precision and real-time

performance are basically equivalent to the ORB algorithm and its other related improved algorithms.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Authors' Contributions

All four authors made excellent efforts to the paper. In collaboration with all writers, ideas and concepts were carried out; C. M., X. H., and J. X. conceived and designed the experiments; C. M., J. X., and G. Z. completed implementation of the method, realization of experiments, and review of data; and C. M. and J. X. completed writing the manuscript. The paper was reviewed and proofread by X. H., G. Z., and J. X.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (under Grant no. 51807003).

References

- [1] J. Jiang, Y. C. Hu, N. Tyagi et al., "PSIGAN: joint probabilistic segmentation and image distribution matching for unpaired cross-modality adaptation based MRI segmentation," *IEEE Transactions on Medical Imaging*, <http://arxiv.org/abs/2007.09465>, 2020.
- [2] J. Ma, X. Jiang, A. Fan et al., "Image matching from hand-crafted to deep features: a survey," *International Journal of Computer Vision*, no. 1, , 2020.
- [3] C. Han, W. Luo, H. Guo, and Y. Ding, "An image matching method for SAR orthophotos from adjacent orbits in large area based on SAR-moravec," *Remote Sensing*, vol. 12, no. 18, p. 2892, 2020.
- [4] Q. Wu, G. Xu, Y. Cheng, W. Dong, L. Ma, and Z. Li, "Histogram of maximal point-edge orientation for multi-source image matching," *International Journal of Remote Sensing*, vol. 41, no. 14, pp. 5166–5185, 2020.
- [5] J. Li, Q. Hu, and M. Ai, "RIFT: multi-modal image matching based on radiation-variation insensitive feature transform," *IEEE Transactions on Image Processing*, vol. 29, pp. 3296–3310, 2020.
- [6] Z. Song, S. Zhou, and J. Guan, "A novel image registration algorithm for remote sensing under affine transformation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, pp. 4895–4912, 2013.
- [7] Q. Guo, J. Xiao, X. Hu, and B. Zhang, "Local convolutional features and metric learning for SAR image registration," *Cluster Computing*, vol. 22, no. S2, pp. 3103–3114, 2019.
- [8] M. Jia, Z. Q. Zhao, L. Huo et al., "Incorporating global-local a Priori knowledge into expectation-maximization for SAR image change detection," *International Journal of Remote Sensing*, vol. 40, no. 1-2, pp. 734–758, 2019.

- [9] B. Chen, P. Sun, and S. Fu, "Consciousness modulates the automatic change detection of masked emotional faces: evidence from visual mismatch negativity," *Neuropsychologia*, vol. 144, Article ID 107459, 2020.
- [10] Z. Lv, J. L. Mauri, and H. Song, "Editorial RGB-D sensors and 3D reconstruction," *IEEE Sensors Journal*, vol. 20, no. 20, pp. 11751–11752, 2020.
- [11] S. Hu, Z. Li, S. Wang, M. Ai, and Q. Hu, "A texture selection approach for cultural artifact 3D reconstruction considering both geometry and radiation quality," *Remote Sensing*, vol. 12, no. 16, p. 2521, 2020.
- [12] C.-V. Lopez-Torres, S. Salazar Colores, K. Kells, J.-C. Pedraza-Ortega, and J.-M. Ramos-Arreguin, "Improving 3D reconstruction accuracy in wavelet transform profilometry by reducing shadow effects," *IET Image Processing*, vol. 14, no. 2, pp. 310–317, 2020.
- [13] Z. Deng, D. Yang, X. Zhang, Y. Dong, C. Liu, and Q. Shen, "Real-time image stabilization method based on optical flow and binary point feature matching," *Electronics*, vol. 9, no. 1, p. 198, 2020.
- [14] C.-G. Lee, G. L. Dunn, I. Oakley, and J. Ryu, "Visual guidance for a spatial discrepancy problem of in encountered-type haptic display," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 4, pp. 1384–1394, 2020.
- [15] A. Marquardt, C. Trepkowski, T. D. Eibich, J. Maiero, E. Kruijff, and J. Schoning, "Comparing non-visual and visual guidance methods for narrow field of view augmented reality displays," *IEEE Transactions on Visualization and Computer Graphics*, IEEE Transactions on Visualization and Computer Graphics, vol. 26, no. 12, pp. 3389–3401, 2020.
- [16] E. G. Parmehr, C. S. Fraser, C. Zhang, and J. Leach, "Automatic registration of optical imagery with 3D LiDAR data using statistical similarity," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 88, pp. 28–40, 2014.
- [17] J. L. Lerma, S. Navarro, M. Cabrelles, A. E. Segui, and D. Hernandez, "Automatic orientation and 3D modelling from markerless rock art imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 76, pp. 64–75, 2013.
- [18] M. Favalli, A. Fornaciai, I. Isola, S. Tarquini, and L. Nannipieri, "Multiview 3D reconstruction in geosciences," *Computers and Geosciences*, vol. 44, pp. 168–176, 2012.
- [19] S. Ahmadi, M. J. V. Zoej, H. Ebadi, H. A. Moghaddam, and A. Mohammadzadeh, "Automatic urban building boundary extraction from high resolution aerial images using an innovative model of active contours," *International Journal of Applied Earth Observation and Geoinformation*, vol. 12, no. 3, pp. 150–157, 2010.
- [20] N. Ekhtari, M. J. V. Zoej, M. R. Sahebi, and A. Mohammadzadeh, "Automatic building extraction from LIDAR digital elevation models and WorldView imagery," *Journal of Applied Remote Sensing*, vol. 3, pp. 1–12, 2009.
- [21] N. Mohammadi and M. Malek, "VGI and reference data correspondence based on location-orientation rotary descriptor and segment matching," *Transactions in GIS*, vol. 19, no. 4, pp. 619–639, 2015.
- [22] Y. Wang and X. Yang, "Research on omnidirectional ORB-SLAM2 for mobile robots," in *Proceedings of the 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC)*, IEEE, Xiamen, China, August 2020.
- [23] Y. Wang, M. Shan, Y. Yue, and D. Wang, "Autonomous target docking of nonholonomic mobile robots using relative pose measurements," *IEEE Transactions on Industrial Electronics*, vol. 99, p. 1, 2020.
- [24] P. Xu, L. Zhang, K. Yang, and H. Yao, "Nested-SIFT for efficient image matching and retrieval," *IEEE Multimedia*, vol. 20, no. 3, pp. 34–46, 2013.
- [25] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [26] C. Guo, F. Jia, W. Tang et al., "A fast method for image matching and registration based on SIFT algorithm and image pyramid," *Journal of Physics: Conference Series*, vol. 1449, no. 1, 6pp, Article ID 012119, 2020.
- [27] H. Bay, A. Ess, T. Tuytelaars, and L. H. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.
- [28] A. Alahi, R. Ortiz, and P. F. Vanderghenst, "Fast retina keypoint," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 510–517, Providence, RI, USA, June 2012.
- [29] S. Leutenegger, M. Chli, and R. B. Siegwart, "Binary robust invariant scalable keypoints," in *Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV)*, pp. 2548–2555, Barcelona, Spain, November 2011.
- [30] T. Trzcinski, M. Christoudias, P. Fua, and V. Lepetit, "Boosting binary keypoint descriptors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2874–2881, Portland, Oregon, USA, June 2013.
- [31] C. Strecha, A. Bronstein, M. Bronstein, and F. P. LDAHash, "Improved matching with smaller descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 66–78, 2011.
- [32] T. Trzcinski, M. Christoudias, and V. Lepetit, "Learning image descriptors with boosting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, pp. 597–610, 2014.
- [33] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, "BRIEF: computing a local binary descriptor very fast," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 1281–1298, 2011.
- [34] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "An efficient alternative to SIFT or SURF," in *Proceedings of the ICCV*, pp. 2564–2571, Barcelona, Spain, November 2011.
- [35] E. Rosten, R. Porter, and T. Drummond, "Faster and better: a machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 105–119, 2008.
- [36] M. Calonder, V. Lepetit, C. Strecha, and F. P. Brief, "Binary robust independent elementary features," in *Proceedings of the European Conference on Computer Vision*, pp. 778–792, Heraklion, Crete, Greece, September 2010.
- [37] X. Yang and K. T. Cheng, "Local difference binary for ultrafast and distinctive feature description," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 188–194, 2013.
- [38] G. Levi and T. Hassner, "LATCH: Learned arrangements of three patch codes," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1–9, Lake Placid, NY, USA, March 2016.
- [39] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, pp. 1147–1163, 2017.
- [40] R. Mur-Artal, J. D. Tardos, and J. D. Tardos, "ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

- [41] X. Fan, Y. Gu, and J. Ni, "Application of improved ORB algorithm in image matching," *Computer and Modernization*, vol. 2, pp. 5–18, 2019, in Chinese.
- [42] C. Ma, X. Hu, J. Xiao et al., "Improved ORB algorithm using three-patch method and local gray difference," *Sensors (Basel, Switzerland)*, vol. 20, no. 4, 2020.
- [43] Y. Li, G. Li, and S. Gu, "Image mosaic algorithm based on area blocking and SIFT," *Optics and Precision Engineering*, vol. 24, pp. 1197–1205, 2016.
- [44] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.
- [45] Q. Tran, T. J. Chin, G. Carneiro, M. Brown, and D. Suter, "Defence of RANSAC for outlier rejection in deformable registration," in *Proceedings of the 12th European Conference on Computer Vision*, pp. 274–287, Berlin, Germany, October 2012.
- [46] H. Zhu and C. Zhao, "The evaluation method of image feature point distribution uniformity," *Journal of Daqing Normal University*, vol. 30, pp. 9–12, 2010, in Chinese.