

Research Article

Improved ORB-SLAM2 Algorithm Based on Information Entropy and Image Sharpening Adjustment

Kaiqing Luo ^{1,2,3}, Manling Lin ¹, Pengcheng Wang ¹, Siwei Zhou ¹, Dan Yin ¹,
and Haolan Zhang ⁴

¹School of Physics and Telecommunication Engineering, South China Normal University, Guangzhou 510006, China

²Guangdong Provincial Engineering Research Center for Optoelectronic Instrument, South China Normal University, Guangzhou 510006, China

³SCNU Qingyuan Institute of Science and Technology Innovation Co., Ltd., Qingyuan 511517, China

⁴The SCDM Center, NingboTech University, Ningbo 315100, China

Correspondence should be addressed to Kaiqing Luo; kqluo@scnu.edu.cn and Haolan Zhang; haolan.zhang@nit.zju.edu.cn

Received 31 July 2020; Revised 27 August 2020; Accepted 7 September 2020; Published 23 September 2020

Academic Editor: Sanghyuk Lee

Copyright © 2020 Kaiqing Luo 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.

Simultaneous Localization and Mapping (SLAM) has become a research hotspot in the field of robots in recent years. However, most visual SLAM systems are based on static assumptions which ignored motion effects. If image sequences are not rich in texture information or the camera rotates at a large angle, SLAM system will fail to locate and map. To solve these problems, this paper proposes an improved ORB-SLAM2 algorithm based on information entropy and sharpening processing. The information entropy corresponding to the segmented image block is calculated, and the entropy threshold is determined by the adaptive algorithm of image entropy threshold, and then the image block which is smaller than the information entropy threshold is sharpened. The experimental results show that compared with the ORB-SLAM2 system, the relative trajectory error decreases by 36.1% and the absolute trajectory error decreases by 45.1% compared with ORB-SLAM2. Although these indicators are greatly improved, the processing time is not greatly increased. To some extent, the algorithm solves the problem of system localization and mapping failure caused by camera large angle rotation and insufficient image texture information.

1. Introduction

Simultaneous Localization and Mapping [1] (SLAM) is described as follows: the robot enters an unknown environment, uses laser or visual sensors to determine its own posture information, and reconstructs a three-dimensional map of the surrounding environment in real time. The system is a hot spot in robot research at present, and it has important theoretical significance and application value for the realization of autonomous control and mission planning of robots.

The current SLAM system is mainly divided into two categories according to different sensor types: laser SLAM and visual SLAM. A SLAM that uses lidar as an external sensor is called a laser SLAM, and a SLAM that uses a camera as an external sensor is called a visual SLAM. The advantage

of a lidar is its wide viewing range, but the disadvantage is the high price. The angular resolution is not high enough, which affects the accuracy of modelling. The sensor camera of the visual SLAM has the characteristics of high cost performance ratio, wide application range, and rich information collection. Therefore, the visual SLAM has developed rapidly since the 21st century.

According to the classification of visual sensors, visual SLAM systems divided into monocular SLAM, binocular SLAM, and RGB-D SLAM. According to the image processing methods, they are divided into direct method and indirect method, such as feature point method and outline feature method. According to the construction of the map, the degree of sparseness can be divided into sparse, dense, semidense, etc. The iconic research results of visual SLAM are Mono-SLAM, PTAM, ORB-SLAM, ORB-SLAM2, etc.

In 2007, Andrew Davison proposed Mono-SLAM [2], which is the real-time implementation of SfM (Structure from Motion), so it is also known as Real-Time Structure from Motion. Mono-SLAM is a SLAM based on probabilistic mapping and has closed-loop correction function. However, Mono-SLAM can only handle small scenes in real time, and only has good effects in small scenes. In the same year, Georg Klein and David Murray proposed PTAM [3] (Parallel Tracking and Mapping), and the innovation is to divide the system into two threads: tracking and mapping, and the concept of key frames is proposed. PTAM is no longer a processing sequence, but a key frame that contains a large amount of information. In 2015, MUR-ARTAL R. et al. proposed ORB-SLAM. In 2016, ORB-SLAM2 was also proposed. ORB-SLAM [4] is an improved monocular SLAM system based on feature points of PTAM. The algorithm can run quickly in real time. As the indoor environment and the wide outdoor environment are robust to vigorous sports, ORB-SLAM2 [5] increases the scope of application on the basis of ORB-SLAM. It is a set that is based on monocular, binocular, and RGB-D. ORB-SLAM2 is more accurate than the previous solution and can work in real time on a standard CPU.

ORB-SLAM2 has improved greatly in running time and accuracy, but there are still some problems to be solved [6, 7]. The feature point extraction is poorly robust in an environment with sudden changing in lighting, too strong or too weak light intensity and too little texture information, which cause the problem of system tracking failure under dynamic environment; the feature points lose when the camera rotates at a large angle. Common cameras often produce image blur when moving quickly, and solving motion blur becomes an important direction in visual SLAM. Image restoration technology can reduce the ambiguity of the image and can restore it from the original image to a relatively clear image. The deblurring method is based on the maximum posterior probability and the norm of sparse representation is used for deblurring [8]. The computational efficiency is improved, but the computational cost is still very large. The image is segmented into many image blocks with the same fuzzy mode, and the point diffusion function of each image block is predicted by convolution neural network. Finally, the clear image is obtained by deconvolution, but the processing time is too long [9]. The literature [10] uses multiscale convolutional neural networks to restore clear images in end-to-end form and optimizes the results by multiscale loss function. Because the same network parameters are used for different fuzzy kernels, the network model is too large. The literature [11] reduces the network model and designs a spatial change cyclic neural network, which can achieve good deblurring effect in dynamic scene, but the generalization ability is general. It also has semantic segmentation architectures and deep learning methods based on deep Convolutional Neural Network (CNN) for semantic segmentation to identify relevant objects in the environment [12].

Because the SLAM system needs real time, the computation of neural network is large, the processing time is long, and the generality is lacking, and another method is

adopted in this paper. The algorithm in this paper starts from the perspective of increasing the image richness. This paper adds a sharpening adjustment algorithm based on the information entropy threshold in the feature point extraction part of the tracking thread. By preprocessing the image block, the information entropy of the image block is compared with the threshold. If the information entropy is below the threshold, it will be sharpened to enhance the richness of the information and facilitate subsequent processing. If the information entropy is above the threshold, the richness of the information is considered sufficient and no processing is performed. In order to increase the generality, this paper proposes the algorithm will automatically calculate the information entropy threshold of the scene according to different scenes. Experimental results show that the improved algorithm improves the accuracy and robustness of ORB-SLAM2 in the case of the rotation at a large angle when the texture information is not rich.

2. The Framework of the Algorithm

The ORB-SLAM2 system is divided into three threads, namely, Tracking, Local Mapping, and Local Closing. In the entire process, the ORB algorithm is used for feature point detection and matching, and the BA (Bundle Adjustment) algorithm is used to perform nonlinear iterative optimization of the three procedures to obtain accurate camera pose and 3D map data. Since the ORB-SLAM2 system relies heavily on the extraction and matching of feature points, running in an environment with rich textures or dealing with video will fail to obtain enough stable matching point pairs. Thus, the beam adjustment method lacks sufficient input information. The posture deviation cannot be effectively corrected [13]. The algorithm of this paper preprocesses the image before extracting the feature points of the tracking thread and adds a sharpening adjustment algorithm based on the information entropy threshold.

2.1. Overall Framework. The overall system framework of the improved algorithm is shown in Figure 1. There are two steps in our program. The first step automatically determines the information entropy threshold of the scene based on the adaptive information entropy algorithm. The second step compares the threshold and the information entropy of each image block and performs the following operations.

The system first accepts a series of image sequences from the camera, passes the image frames to the tracking thread after initialization, extracts ORB features from the images in this thread, performs pose estimation based on the previous frame, tracks the local map, optimizes the pose, determines the key frame according to the rules, and passes it to the Local Mapping thread to complete the construction of the local map. Finally, Local Closing performs closed-loop detection and closed-loop correction. In the algorithm of this paper, the sharpening adjustment based on the information entropy threshold is added to the image preprocessing part before the ORB feature detection in the Tracking thread.

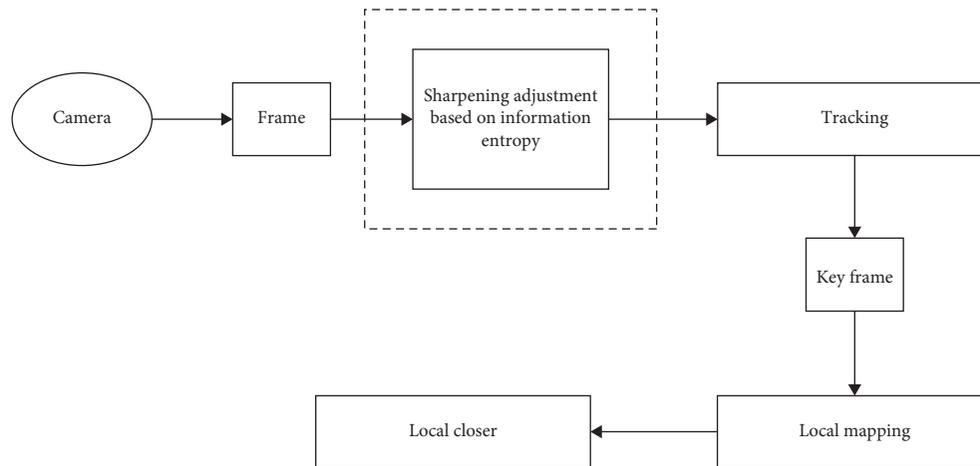


FIGURE 1: Algorithm system framework.

2.2. Tracking Thread. The input to the tracking thread is each frame of the video. When the system is not initialized, the thread tries to initialize using the first two frames, mainly to initialize and match the feature points of the first two frames. There are two initialization methods in the ORB-SLAM2 system, namely, the initialization of monocular camera, binocular camera, and RGB-D camera. The experiment in this article mainly uses monocular camera.

After the initialization is complete, the system starts to track the camera position and posture. The system first uses the uniform velocity model to track the position and posture. When the uniform velocity model fails, the key frame model is used for tracking. In the case when both types of tracking fail, the system will trigger the relocation function to relocate the frame. After the system obtains each frame, it will use feature points to detect and describe and use feature descriptors for tracking and pose calculation. In the process of feature point extraction, image quality plays an important role. Therefore, this article creatively adds a preprocessing algorithm to improve image quality before the image feature point extraction step.

At the same time, the system will track the local map, match the current frame with related key frames to form a set, and find the corresponding key points. The beam adjustment method is used to track the local map points to minimize the reprojection error, thereby optimizing the camera posture of the current frame. Finally, according to the conditions of whether to generate a key frame, a judgment is made to generate a key frame. Specify the current frame that meets certain conditions as a key frame. The key frames selected in the “tracking” thread need to be inserted into the map for construction. The key frame contains map points as feature points. When more than a certain number of key frames are collected, their key points will be added to the map and become map points. Delete the map points that do not meet the conditions.

2.3. Local Mapping Thread. The input of the thread is the key frame inserted by the tracking thread. On the basis of newly added key frames, maintain and add new local map points. The

key frame input by the thread is generated according to the rules set during tracking, so the result of Local Mapping thread is also closely related to the quality of the image used for feature point extraction in the tracking process. Image adaptive preprocessing algorithm also improves Local Mapping thread when optimizing the extraction of feature points.

During the construction of the map, local BA optimization will be performed. Use BA to minimize reprojection errors and optimize Map points and poses. Because BA requires a lot of mathematical operations, which are related to key frames, Local Mapping will delete redundant key frames in order to reduce time consumption.

2.4. Local Closing Thread. In the continuous movement of the camera, it is completely consistent with the actual that the camera pose calculated by the computer and the map points is obtained by the triangulation. There is a certain error between them. While the number of frames increases, the error gradually accumulates. In order to reduce these accumulated errors, the most effective method is closed-loop correction. ORB-SLAM2 uses the closed-loop detection method. When the camera re-enters the previous scene, the system detects Closed-loop, and global BA optimization is performed to reduce the cumulative error. Therefore, when the ORB-SLAM2 system is applied to a large-scale scene, it shows higher robustness and usability. The input of the thread is the key frame screened by the Local Mapping thread. The word bag vector of the current key frame is stored in the database of the global word bag vector to speed up the matching of subsequent frames. At the same time, it detects whether there is a loopback, and if it occurs, it passes the pose Graph optimization to optimize the pose of all key frames and reduce the accumulated drift error. After the pose optimization is completed, a thread is started to execute the global BA to obtain the most map points and key pose results of the entire system.

2.5. The Process of Sharpening Adjustment Algorithm Based on Information Entropy. Motion-blurred images are more disturbed by noise, which impact the extraction of

feature points in the ORB-SLAM2 system. The number of stable feature points found on these images is insufficient. The feature points cannot even be found in the images with too much blurs. The insufficient number will then affect the accuracy of the front-end pose estimation. When the feature points are less than the number specified by the system, the pose estimation algorithm cannot be performed so that the back-end cannot obtain the front-end information and the tracking fails. Based on this background, this paper proposes a sharpening adjustment algorithm based on information entropy, preprocessing the image before the tracking thread.

In images with poor texture and blurred images, the sharpening of the image can make the corner information of the image more prominent. Because when the ORB feature is extracted in the tracking thread, it is easier to detect the feature point and enhance the system's stability. And the screening based on the information entropy threshold is added, and the image sharpening adjustment is performed only when the information entropy of the image block is less than the threshold. It can reduce the time of sharpening adjustment, ensure the real-time performance of the system, and maintain the integrity of the image information. If the image block information entropy is bigger than the threshold, it will not be processed.

The process of sharpening adjustment algorithm based on information entropy threshold is as follows, and the algorithm flow chart is shown in Figure 2.

- (1) At first, an image input by the frame is converted into a grayscale image, and the image is expanded into an 8-layer image pyramid under the effect of the scaling factor, and then each layer of the pyramid image is divided into image blocks.
- (2) Calculate the information entropy E of the image block, and compare the obtained information entropy E with the information entropy threshold E_0 . The image block with the information entropy E less than the threshold E_0 indicates that the image block contains less effective information and the effect of ORB feature extraction is poor, so you need to sharpen first to enhance the detail in the image.
- (3) After the sharpening process is performed on the image block with the information entropy less than the threshold, ORB feature points are extracted with the image block with the information entropy greater than the threshold. The feature points are extracted using the FAST feature point extraction algorithm in the pyramid. The quadtree homogenization algorithm is used for homogenization processing, which makes the distribution of extracted feature points more uniform. It avoids the phenomenon of clustering of feature points; thus, the algorithm becomes more robust.
- (4) Then, perform a BRIEF description of the feature points to generate a binary descriptor of the feature points after the homogenization process. The feature points generated with the BRIEF descriptor at this time are called ORB features, which have invariance

of viewpoints and invariance of lighting ORB features which are used in the later graph matching and recognition in the ORB-SLAM2 system.

3. The Novel Image Preprocessing Algorithm

We propose and add a sharpening adjustment algorithm based on information entropy. Before the tracking thread, the image is preprocessed. The above algorithm framework roughly describes the specific operation of the algorithm. The following will explain in detail the relevant principles and details of the improved algorithm. Algorithms used in this article are proposed including the ORB algorithm in ORB-SLAM2, the sharpening adjustment algorithm, the principle and algorithm of information entropy, and the adaptive information entropy threshold algorithm.

3.1. ORB Algorithm. As explained in the previous article, the ORB-SLAM2 system relies heavily on the extraction and matching of feature points. Therefore, running in an environment with rich texture or video processing will not be able to obtain enough stable matching point pairs, resulting in loss of system position and attitude tracking. Case: in order to better explain the improved system in this article, we first introduce the feature point extraction algorithm used by the system.

The focus of image information processing is the extraction of feature points. The ORB algorithm combines the two methods: FAST feature point detection and BRIEF feature descriptor [14]. To make further optimization and improvement on their basis, in 2007, the FAST corner detection algorithm was proposed. Compared with other feature point detection algorithms, the FAST algorithm has better real-time detection and robustness. The core of the FAST algorithm is to take a pixel and compare it with the gray value of the points around it. For comparison, if the gray value of this pixel differs from most of the surrounding pixels, it is considered to be a feature point.

The specific steps of FAST corner detection feature points: each pixel is detected, with the detected pixel as the center p ; 3 pixels with a radius of 16 pixels on a circle, set a gray value threshold t . Comparing the 16 pixels on the circle, when there are consecutive n (12) pixels, the gray value is greater than $I_p + t$ or less than $I_p - t$ (I_p is the gray value of point p); then, p is determined as a feature point. In order to improve the efficiency of detection, a simplified judgment is first performed. It is only necessary to detect whether the gray values of 1, 5, 9, and 13 satisfied the above conditions; when at least 3 points are met, continue to detect the remaining 12 points (Figure 3).

The concept of mass points is introduced to obtain the scale and rotation invariance of FAST corner points. The centroid refers to the gray value of the image block as the center of weight. The specific steps are as follows.

- (1) In a small image block B , the moment defining the image block B is

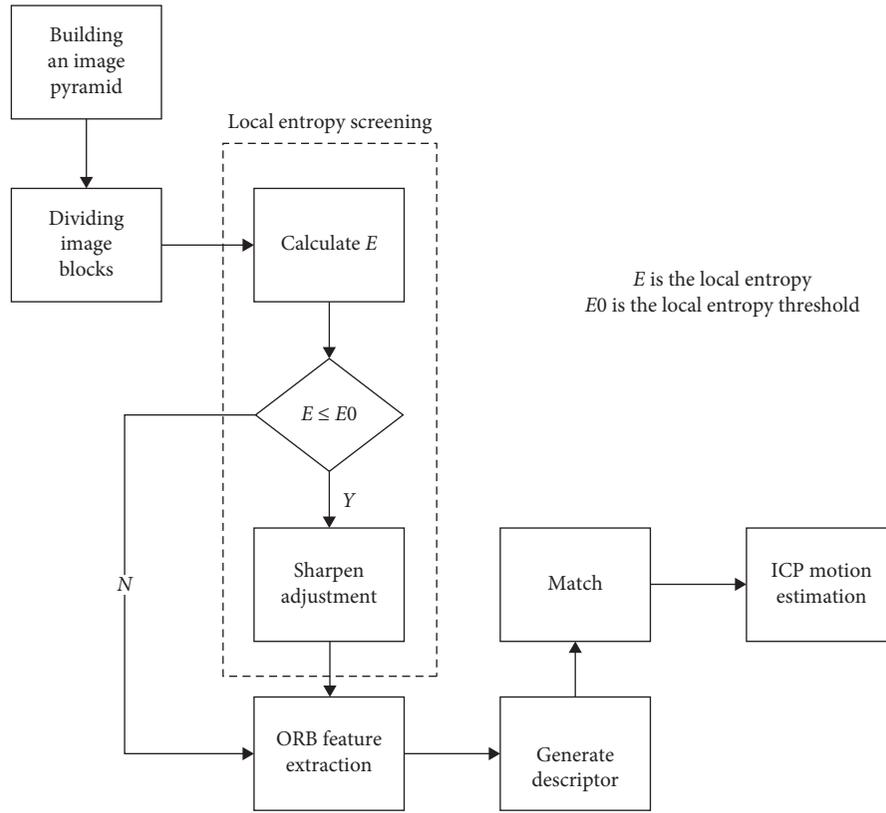


FIGURE 2: Process of sharpening adjustment algorithm based on information entropy.

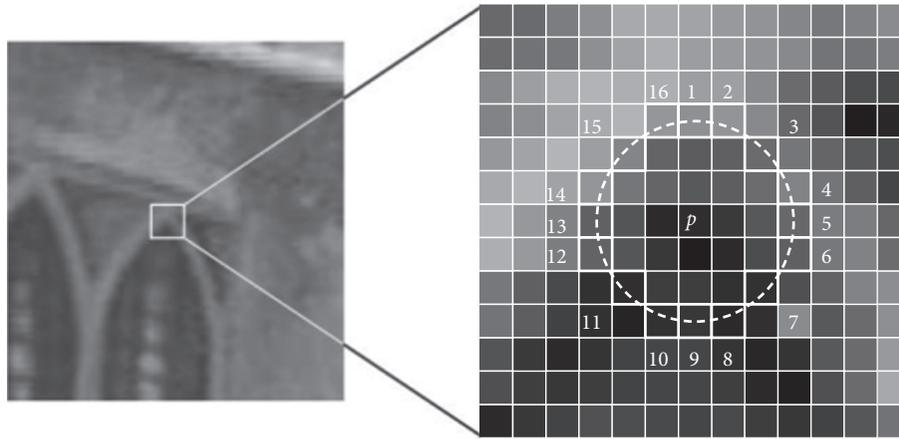


FIGURE 3: FAST feature point detection.

$$m_{pq} = \sum_{x,y \in B} x^p y^q I(x, y), \quad p, q = \{0, 1\}. \quad (1)$$

$$C = \begin{pmatrix} \frac{m_{10}}{m_{00}} & \frac{m_{01}}{m_{00}} \\ \frac{m_{00}}{m_{00}} & \frac{m_{00}}{m_{00}} \end{pmatrix}. \quad (2)$$

(2) You can find the centroid of the image block by moment:

(3) Connect the geometric center O and the centroid C of the image block to get a direction vector \vec{OC} . Then, the feature point direction can be defined as follows:

$$\theta = \arctan\left(\frac{m_{01}}{m_{10}}\right). \quad (3)$$

Through the above methods, FAST corners have a description of scale and rotation, which greatly improves the robustness of their representation between different images. So, this improved FAST is called Oriented FAST in the ORB.

After the feature point detection is completed, the feature description needs to be used to represent and store the feature point information to improve the efficiency of the subsequent matching. The ORB algorithm uses the BRIEF descriptor whose main idea is to distribute a certain probability around the feature point. Randomly select several point pairs and then combine the gray values of these point pairs into a binary string. Finally use this binary as the descriptor of this feature point.

The BRIEF feature descriptor determines the binary descriptor by comparing the pair of gray values. As shown in Figure 4, if the gray value at the pixel x is less than the gray value at y , the binary value of the corresponding bit is assigned to 1, otherwise it is 0:

$$\tau(p; x, y) = \begin{cases} 1, & I(x) < I(y), \\ 0, & I(x) \geq I(y). \end{cases} \quad (4)$$

Among them, $I(x)$ and $I(y)$ are the gray values at the corresponding corners after image smoothing.

For n (256) (x, y) test point pairs, the corresponding n -dimensional BRIEF descriptor can be formed according to the following formula:

$$f_n(p) = \sum_{1 \leq i \leq n} 2^{i-t} \tau(p; x, y). \quad (5)$$

However, the BRIEF descriptor does not have rotation invariance, the ORB algorithm improves it. When calculating the BRIEF descriptor, the ORB will solve these corresponding feature points in the main direction, thereby ensuring that the rotation angle is different. The point pairs selected for the same feature point are the same.

3.2. Sharpening Adjustment. In the process of image transmission and conversion, the sharpness of the image is reduced. In essence, the target contour and detail edges in the image are blurred. In the extraction of feature points, what we need to do is to traverse the image, and the pixels with large difference in surrounding pixels are selected as the feature points, which requires the image's target contour and detail edge information to be prominent. Therefore, we introduced image sharpening.

The purpose of image sharpening is to make the edges and contours of the image clear and enhance the details of the image [15]. The methods of image sharpening include statistical difference method, discrete spatial difference method, and spatial high-pass filtering method. Generally, the energy of the image is mainly concentrated in the low frequency. The frequency where the noise is located is mainly in the high-frequency band. The edge information of the image is mainly concentrated in the high-frequency band. Usually smoothing is used to remove the noise at high frequency, but this also makes the edge and contour of the

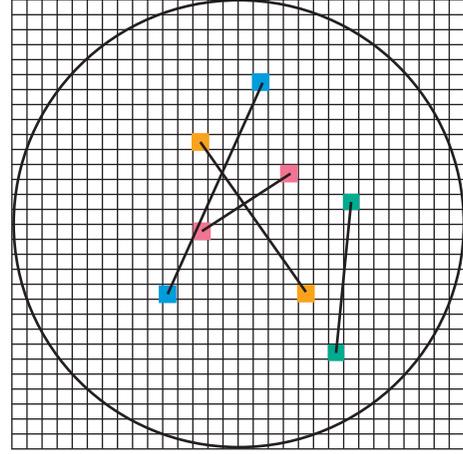


FIGURE 4: BRIEF description.

image blurred, which affects the extraction of the feature points. The root cause of the blur is that the image is subjected to an average or integral operation. So, performing an inverse operation (a differential operation) on the image can restore the picture to clear.

In order to reduce the adverse effects, we adopt the method of high-pass filtering (second-order differential) in the spatial domain and use the Laplacian operator to perform a convolution operation with each pixel of the image to increase the variance between each element in the matrix and the surrounding elements to achieve sharp image. If the variables of the convolution are the sequence $x(n)$ and $h(n)$, the result of the convolution is

$$y(n) = \sum_{i=-\infty}^{\infty} x(i)h(n-i) = x(n) * h(n). \quad (6)$$

The convolution operation of the divided image blocks is actually to use the convolution kernel to slide on the image. The pixel gray value is multiplied with the value on the corresponding convolution kernel. Then, all the multiplied values are added as the gray value of the pixel on the image corresponding to the middle pixel of the convolution kernel. The expression of the convolution function is as follows:

$$\begin{aligned} \text{dst}(x, y) = & \sum_{\substack{0 \leq x' < \text{kernel.cols} \\ 0 \leq y' < \text{kernel.rows}}} \text{kernel}(x', y') \\ & * \text{src}(x + x' - \text{anchor.x}, y + y' - \text{anchor.y}), \end{aligned} \quad (7)$$

where anchor is the reference point of the kernel; kernel is the convolution kernel, and the convolution template here is the matrix form of the Laplacian variant operator, as shown below:

$$H = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}. \quad (8)$$

The Laplacian deformation operator uses the second derivative information of the image and is isotropic.



FIGURE 5: Sharpening adjustment.

The results of image discrete convolution on the classic Lenna image using the sharpening algorithm adopted in this paper are shown in Figure 5.

From the processing results, it can be seen that the high-pass filtering effect of the image sharpening algorithm is obvious, which enhances the details and edge information of the image, and the noise becomes small, so this sharpening algorithm is selected.

3.3. Screening Based on Information Entropy. In 1948, Shannon proposed the concept of “information entropy” to solve the problem of quantitative measurement of information. In theory, entropy is a measure of the degree of disorder of information, which is used to measure the uncertainty of information in an image. The larger the entropy value is, the higher the degree of disorder. In image processing, entropy can reflect the information richness of the image and display the amount of information contained in the image. The information entropy calculation formula is

$$H(x) = - \sum_0^{255} p(x_i) \log_2 p(x_i), \quad (9)$$

where $p(x_i)$ is the probability of a pixel with grayscale i ($i = 0 \dots 255$) in the image. When the probability is closer to 1, the uncertainty of the information is smaller, and the receiver can predict the transmitted content. The amount of information in the belt is less, and vice versa.

If the amount of information contained in the image is expressed by information entropy, then the entropy value of an image of size $m \times n$ is defined as follows:

$$p_{ij} = \frac{f(i, j)}{\sum_{i=1}^M \sum_{j=1}^N f(i, j)}, \quad (10)$$

$$H = - \sum_{i=1}^M \sum_{j=1}^N p_{ij} \log_2 p_{ij}, \quad (11)$$

where $f(i, j)$ is the gray level at the point (i, j) in the image, p_{ij} is the gray distribution probability at the point, and H is the entropy of the image. If $m \times n$ is taken as (i, j) a local neighborhood in the center, then H is called the information entropy value of the image [16].

From the perspective of information theory, when the probability of image appearance is small, the information entropy value of the image is large, indicating that the amount of information transmitted is large. In the extraction of feature points, the information entropy reflects the texture contained in the partial image information richness or image pixel gradient change degree. The larger the information entropy value is, the richer the image texture information is and more obvious the changing of the image pixel gradient is [17].

Therefore, the effect of ORB feature point extraction is good, and the image block does not require detail enhancement. The local information entropy value is low, so the effect of ORB feature point extraction is poor which is shown in Figure 6. After sharpening adjustments to enhance details, we optimize the effect of feature point extraction [18], which is shown in Figure 7. By comparing the feature point extraction of ORB-SLAM2, it can be intuitively seen that the optimized extraction algorithm accuracy is better.

3.4. Adaptive Information Entropy Threshold. Because the number of the information entropy is closely related to the scene, different video sequences in different scenes have different information richness, so the information entropy threshold of different scenes must also be different. In each different scene, it requires repeated experiments to set the information entropy threshold multiple times for matching calculations to get the corresponding threshold value. However, the experience value in different scenes differs greatly. The threshold value is not universal, which will affect the image preprocessing and feature extraction. It will result in the failure of quickly obtaining better matching results. So, the adaptive algorithm of information entropy is particularly important [19].

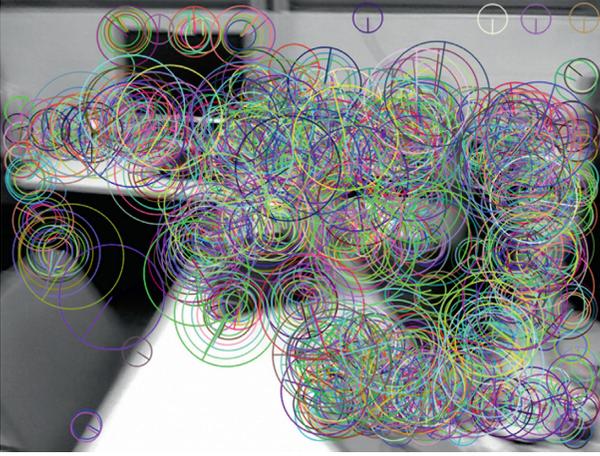


FIGURE 6: Feature point extraction based on ORB-SLAM2.

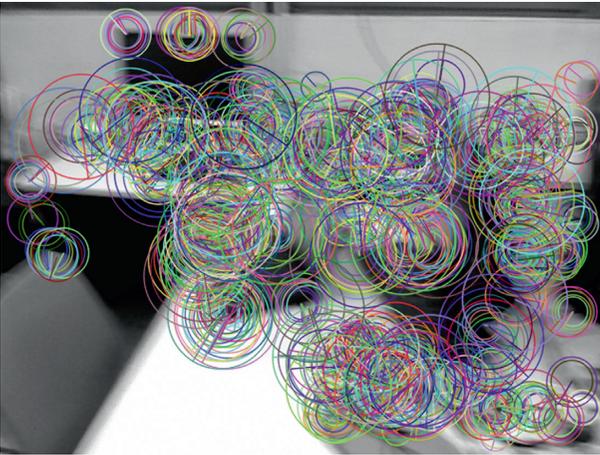


FIGURE 7: Extraction of sharpening adjustment feature points based on information entropy.

In view of the above problems, we propose an adaptive method of information entropy threshold, which adjusts the threshold according to different scenarios. The self-adjusting formula is

$$E_0 = \frac{H(i)_{\text{ave}}}{2} + \delta, \quad (12)$$

where $H(i)_{\text{ave}}$ is the average value of the information entropy in this scene, which can be obtained by obtaining the information entropy of each frame in a video in the scene under the first run and then divided by the number of frames, i is the number of frames in the video sequence, and δ is the correction factor; after experiment, it is 0.3 which works best. Through the above formula, the calculated E_0 is the information entropy threshold of the scene.

4. Experimental Results and Analysis

All the experiments are conducted on a laptop computer. The operating system is Linux 16.04 64 bit, the processor Intel (R) Core (TM) i5-7300 U CPU @ 2.50 GHz, the operating environment is CLion 2019 and opencv3.3.0, and the

program uses C++ Compilation. The back-end uses G2O for posture optimization based on posture maps to generate motion trajectories. The image sequence reading speed of this algorithm is 30 frames per second, compared with the open source ORB-SLAM2 system.

4.1. Comparison of Two-Frame Image Matching. In this paper, the `rgbd_dataset_freiburg1_desk` dataset is selected, from which two images with blurry images and large rotation angles are extracted, and the image detection and matching processing are performed using the ORB algorithm, ORB + information entropy, and ORB + information entropy + sharpening extraction algorithm. And compare the data such as extraction effect, key points, matching number, matching number, and matching rate after RAN-SAC filtering. The matching result statistics of the image are shown in Table 1, and the matching effect is shown in Figures 8, 9, and 10.

Table 1 shows the results of the feature point detection and matching through the ORB, ORB + information entropy, and ORB + information entropy + sharpening. It proves that the correct matching points and the matching rate of ORB and ORB + information entropy are basically the same. The matching effect diagram of the latter shows a mismatch with a large error. For ORB + information entropy + sharpening, the number of correct matching points of the extraction algorithm has increased significantly, and its matching rate, compared to ORB + information entropy and ORB + information entropy + sharpening extraction algorithms, have been, respectively, improved by 36.7% and 38.8%.

The results show that adding the information entropy screening and sharpening process proposed by this algorithm can effectively increase the number of matches and improve the matching rate. The accuracy of image feature detection and matching are improved, which provide more accurate and effective for subsequent process as tracking, mapping, and loop detection.

4.2. Comparison of Image Matching in Visual Odometer. For the timestamp sequence of different blur degrees and rotation angles of the image (that is, a sequence of frames 170–200), ORB-SLAM2 algorithm and the feature detection and matching algorithm in the improved algorithm of this paper are introduced to perform two adjacent frame's feature detection and matching. To compare the matching points of the two algorithms in different images, the image is clear between 170 and 180 frames, and the motion blur starts between 180 and 190 frames. Maximum rotation angle is between 190 and 200 frames. The most dramatic motion blur is shown in Figure 11.

It can be seen from Figure 12 that the improved algorithm can match more feature points than the original algorithm. The number of matching points of the improved algorithm is more than 50, and tracking loss does not occur, even if the image is blurred due to fast motion. When the resolution of the image is reduced, the

TABLE 1: Comparison of matching rates of three feature detection and matching algorithms.

Model	Key points	Matches	RANSCK	Match rate (%)
ORB	1000/1004	254	87	34.25
+information entropy	1000/1002	255	86	33.73
+sharpening	1000/983	284	133	46.83



FIGURE 8: Matching effect of ORB.

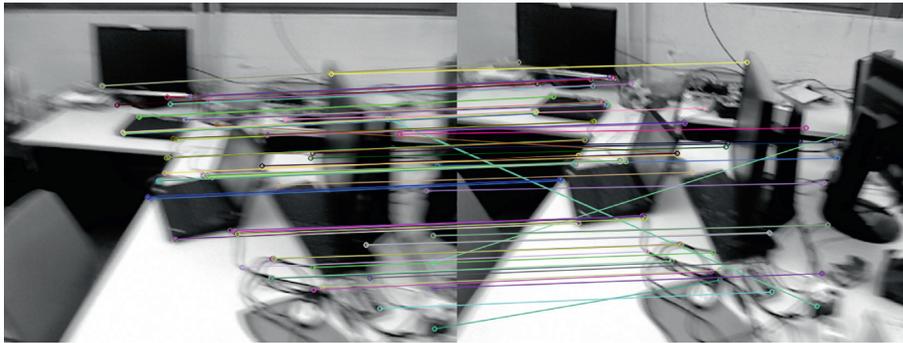


FIGURE 9: Matching effect of ORB + information entropy.



FIGURE 10: Matching effect of ORB + information entropy + sharpening.

improved algorithm still has more matching feature points and has strong anti-interference ability and robustness.

4.3. Analysis of Trajectory Tracking Accuracy. The `gbd_dataset_freiburg1_desk` dataset in the TUM data is selected in the experiment. We use the real trajectory of the robot

provided by TUM to compare it with the motion trajectory calculated by the algorithm to verify the improvement effect of the proposed algorithm. Here, we take the average absolute trajectory error. The average relative trajectory error and the average tracking time are standard. The comprehensive trajectory tracking accuracy and real-time performance are compared between the ORB-SLAM2 system and the improved SLAM system.



FIGURE 11: 200th frame image.

We use the information entropy of image blocks to determine the amount of information. The algorithm sharpens the image blocks with small information entropy, enhances local image details, and extracts local feature points that can characterize image information for the back-end as adjacent frames. The correlation basis of key frame matching enhances robustness and reduces the loss of motion tracking caused by the failure of interframe matching at the same time. Based on the matching result, the R and t transformation relationship between frames is calculated, and the back-end uses g2o to optimize the pose based on the position and attitude map. Finally the motion trajectory is generated, which is shown in Figure 13.

In terms of trajectory tracking accuracy, the absolute trajectory error reflects the difference between the true value of the camera's pose and the estimated value of the SLAM system. The absolute trajectory error root mean square RMSE (x) is defined as follows:

$$\text{RMSE}(x) = \sqrt{\frac{\sum_{i=1}^n \|x_{e,i} - x_{s,i}\|^2}{n}}. \quad (13)$$

Among them, $x_{e,i}$ represents the estimated position of the i th frame in the image sequence and $x_{s,i}$ represents the standard value of the position of the i th frame in the image sequence. Taking the `rgbd_dataset_freiburg1_desk` dataset as an example, the error between the optimized motion trajectory and the standard value is shown in Figure 14.

The average absolute trajectory error refers to directly calculating the difference between the real value of the camera's pose and the estimated value of the SLAM system. The program first aligns the real value and the estimated value according to the time stamp of the pose and then calculates each pair of poses. The difference is very suitable for evaluating the performance of visual SLAM systems. The average relative trajectory error is used to calculate the difference between the pose changes on the same two timestamps. After alignment with the timestamps, both the real pose and the estimated pose are calculated at every same period of time; then, the difference is calculated. It is suitable

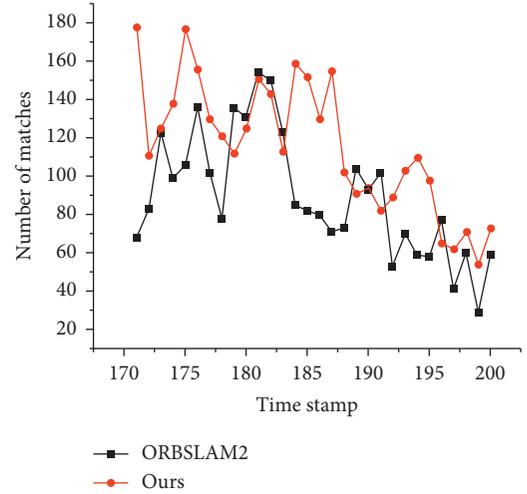


FIGURE 12: Comparison between the algorithm of this paper and ORB-SLAM2 motion blur.

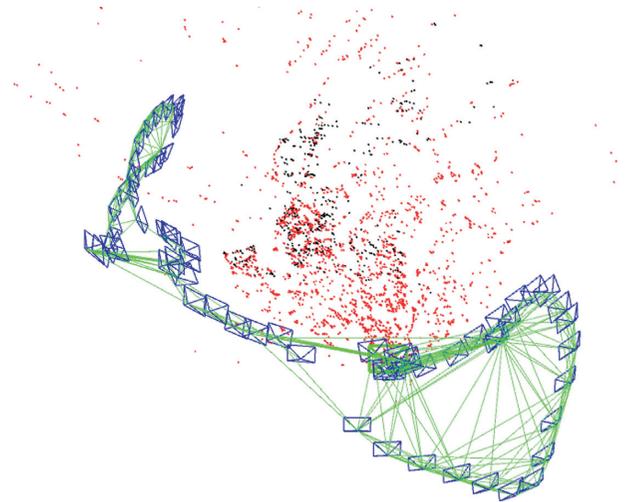


FIGURE 13: Movement track.

for estimating the drift of the system. The definition of the absolute trajectory error in frame i is as follows:

$$F_i = Q_i^{-1} S p_i. \quad (14)$$

Among them, Q_i represents the real pose of the frame, and p_i represents the algorithm estimated pose of the frame, $S \in \text{Sim}(3)$ is the similarity transformation matrix from estimated pose to true pose.

This experiment comprehensively compares the three datasets `rgbd_dataset_freiburg1_desk/room/360`, of which the desk dataset is an ordinary scene; the 360 dataset is the dataset collected when the camera performs a 360-degree rotation motion, and the room dataset is a scene collected when the image sequence texture is not rich. The ORB-SLAM2 generates different trajectory results each time. In order to eliminate the randomness of the experimental results, this article uses 6 experiments on each dataset under the same experimental environment to calculate the average absolute trajectory error, the average relative trajectory

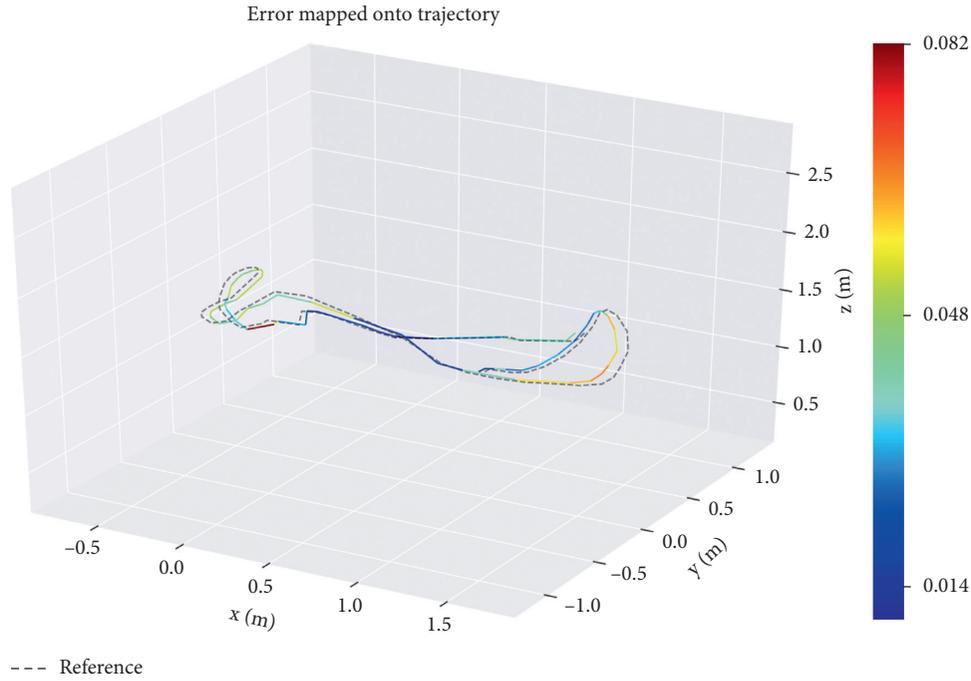


FIGURE 14: Absolute trajectory error.

TABLE 2: Average absolute trajectory error.

Average absolute trajectory error difference/m	ORB-SLAM2	The proposed algorithm
Desk	0.057630667	0.029899333
360	0.113049667	0.072683167
Room	0.229222167	0.111185333

TABLE 3: Average relative trajectory error.

Average relative trajectory error difference/m	ORB-SLAM2	The proposed algorithm
Desk	0.0362615	0.029899
360	0.1534532	0.078611
Room	0.1592775	0.092167

TABLE 4: Average tracking time.

Average relative trajectory error difference/m	ORB-SLAM2	The proposed algorithm
Desk	0.0244937	0.0289299
360	0.0186606	0.0225014
Room	0.0220836	0.0282757

error, and the average tracking time. Tables 2–4 are plotted in Figures 15–17 to compare the algorithm performance by analyzing the trajectory.

Under the evaluation standard of absolute trajectory error, the algorithm in this paper has certain advantages. In the scene where the camera rotates 360°, the absolute trajectory error is increased by 35%, and in the ordinary scene, the absolute trajectory error is increased by 48%. In the case of not rich texture, the absolute trajectory error improvement is the largest, indicating that the algorithm in this paper has a considerable improvement on scenes

with rich image sequence textures. There is also a certain improvement in the situation of large-angle camera rotation.

There is an advantage in trajectory error: in ordinary scenes, it is 17.5% smaller than the average relative trajectory error of the ORB-SLAM2 system. In the scene where the camera rotates 360° and the image sequence texture is not rich, the algorithm in this paper is better than the ORB-SLAM2 system, which has been improved by more than 40%. It proves that the algorithm in this article is indeed improved compared to the ORB-SLAM2 system.

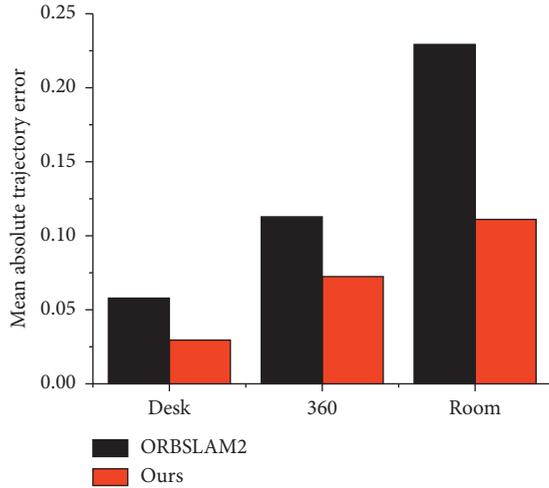


FIGURE 15: Average absolute trajectory error.

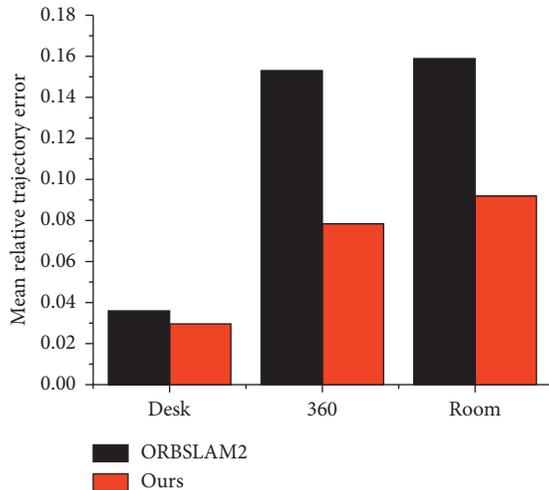


FIGURE 16: Average relative trajectory error.

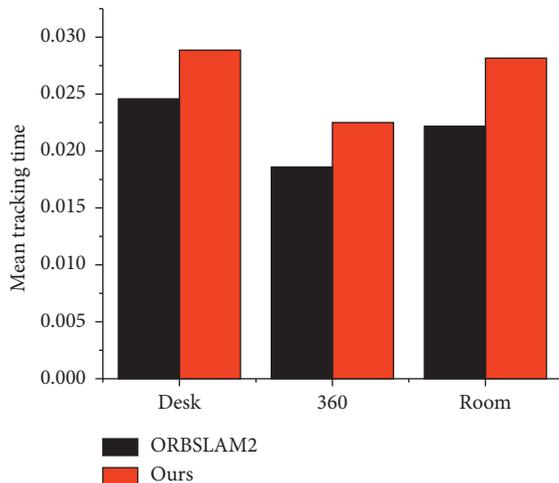


FIGURE 17: Average tracking time.

Due to the sharpening of the local image, the average tracking time is slightly increased but the increase is not large.

5. Conclusion

We propose an improved ORB-SLAM2 algorithm, which is based on adaptive information entropy screening algorithm and sharpening adjustment algorithm. Our advantage is that the information entropy threshold can be automatically calculated according to different scenes with generality. Sharpening the picture blocks below the information entropy threshold increases the clarity in order to better extract the feature points. The combination of information entropy screening image block and sharpening algorithm solves the camera's large-angle rotation and the system positioning and mapping failure caused by the lack of texture information. Although the processing time increases, it improves the processing accuracy. We compare the average absolute trajectory error, the average relative trajectory error, and the average tracing time in the trajectory error with ORB-SLAM2. The experimental results show that the improved model based on information entropy and sharpening improved algorithm achieve better results compared with ORB-SLAM2. This article innovatively combines the ORB-SLAM2 system with adaptive information entropy. The combination of sharpening algorithms improves the accuracy and robustness of the system without a noticeable gap in processing time.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC)-Guangdong big data Science Center Project under Grant U1911401 and in part by the South China Normal University National Undergraduate Innovation and Entrepreneurship Training Program under Grant 201910574057.

References

- [1] D. Schleicher, L. M. Bergasa, R. Barea, E. Lopez, and M. Ocana, "Real-time simultaneous localization and mapping using a wide-angle stereo camera," in *Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and its Applications (DIS'06)*, pp. 2090–2095, Beijing, China, October 2006.
- [2] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.
- [3] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Proceedings of the 2007 6th IEEE and*

- ACM International Symposium on Mixed and Augmented Reality*, pp. 225–234, Nara, Japan, November 2007.
- [4] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM: a versatile and accurate monocular SLAM system,” *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1052–1067, 2007.
 - [5] R. Mur-Artal and J. D. Tardós, “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.
 - [6] K. Di, W. Wan, and J. Chen, “SLAM visual odometer optimization algorithm based on information entropy,” *Journal of Automation*, vol. 47, no. 6, pp. 770–779, 2020.
 - [7] M. Quan, H. Wei, and J. Chen, “SLAM visual odometer optimization algorithm based on information entropy,” *Journal of Automation*, vol. 1, no. 8, pp. 180–198, 2020.
 - [8] L. Xu, S. Zheng, and J. Jia, “Unnatural L0 sparse representation for natural image deblurring,” in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1107–1114, Portland, OR, USA, June 2013.
 - [9] J. Sun, W. Cao, Z. Xu, and J. Ponce, “Learning a convolutional neural network for non-uniform motion blur removal,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 769–777, Boston, MA, USA, June 2015.
 - [10] S. Nah, T. H. Kim, and K. M. Lee, “Deep multi-scale convolutional neural network for dynamic scene deblurring,” in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 257–265, Honolulu, HI, USA, July 2017.
 - [11] J. Zhang, J. Pan, J. Ren et al., “Dynamic scene deblurring using spatially variant recurrent neural networks,” in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2521–2529, Salt Lake City, UT, USA, June 2018.
 - [12] G. Chen, W. Chen, H. Yu et al., “Research on autonomous navigation method of mobile robot based on semantic ORB-SLAM2 algorithm,” *Machine Tool & Hydraulics*, vol. 48, no. 9, pp. 16–20, 2020.
 - [13] X. Qiu and C. Zhao, “Overview of ORB-SLAM system optimization framework analysis,” *Navigation Positioning and Timing*, vol. 6, no. 3, pp. 46–57, 2019.
 - [14] E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” in *Proceedings of the European Conference on Computer Vision*, pp. 430–443, Graz, Austria, 2006.
 - [15] A. Foi, V. Katkovnik, and K. Egiazarian, “Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images,” *IEEE Transactions on Image Processing*, vol. 16, no. 5, pp. 1395–1411, 2007.
 - [16] Y. Yu and W. H. Tardós, “ORB-SLAM: a versatile and accurate monocular SLAM system,” *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.
 - [17] R. Xu, S. Liu, and J. Chen, “The rationality and application of information entropy description in images,” *Information Technology*, vol. 11, no. 5, pp. 59–61, 2005.
 - [18] Y. Zhu, “Overview of image enhancement algorithms,” *Information and Computer (Theoretical Edition)*, vol. 16, no. 6, pp. 104–106, 2017.
 - [19] Z. Yuan, “Image threshold automatic selection based on one-dimensional entropy,” *Journal of Gansu University of Technology*, vol. 1, no. 1, pp. 84–88, 1993.