GEORGE BROWN PRESS
JOURNALS, BOOKS & PROCEEDINGS

Journal of Computer, Signal, and System Research

Article

Application of an Improved Orb Algorithm in Image Matching

Xiaofei Deng [1,*] and Joan Lazaro [1]

[1] University of the East, Manila, Philippines
* Correspondence: Xiaofei Deng, University of the East, Manila, Philippines

Abstract: An improved ORB feature matching algorithm is proposed to solve the problem of high mismatching rate of non-uniform distribution of feature points extracted by traditional ORB algorithms. First of all, to extract feature points of image block, and then calculate each block information entropy, the entropy value dynamically setting the threshold of each block to extract the feature points, make to extract the feature points of distribution more uniform, improve the subsequent registration accuracy. Secondly, for the problem of high mismatch rate caused by insufficient use of local information by the rBRIEF descriptor in the ORB algorithm, on the basis of introducing gradient information to improve the LBP algorithm, an improved descriptor combining rBRIEF and LBP is proposed. The 128-bit descriptor generated by the improved LBP algorithm is used to replace the 128-bit descriptor with the smallest variance in the original rBRIEF to improve the stability of the feature point description. Finally, RANSAC was used to delete the mismatched points. Results Compared with the traditional ORB algorithm, the improved algorithm reduces the time overhead by 8% and the average mismatch rate by about 10%. The improved orb algorithm has lower mismatch rate and higher registration speed.

Keywords: orb algorithm; information entropy; feature point; LBP algorithm; gradient

Received: 02 December 2025
Revised: 24 January 2026
Accepted: 04 February 2026
Published: 11 February 2026

1. Introduction

During image acquisition for printing, the limited field of view of imaging devices such as cameras makes it difficult to capture a complete large-scale image in a single shot. Therefore, multiple images are typically combined into a single large-scale image using image stitching techniques. A critical step in image stitching is image matching. Common feature point-based matching algorithms include SIFT, SURF, and ORB [1-3].

The SIFT algorithm is invariant to rotation and scale and exhibits strong resistance to noise, providing reliable registration performance. However, its computational complexity makes it unsuitable for real-time applications. SURF was developed as an improvement over SIFT. By replacing the Difference of Gaussian (DoG) pyramid with a Hessian matrix and using integral images to accelerate the detection of local extrema, SURF reduces the 128-dimensional descriptor to a 64-dimensional one. This modification improves computational efficiency while maintaining good robustness. ORB further improves computational speed by employing an enhanced FAST algorithm for feature detection, a gray-scale centroid method for orientation estimation, and the rBRIEF descriptor for feature description. Its processing speed is significantly higher than that of both SIFT and SURF. Nevertheless, ORB still suffers from uneven feature point distribution and relatively low matching accuracy [4].

To address the problem of uneven feature point distribution in ORB, several studies have introduced strategies to improve spatial uniformity. For example, the quadtree method has been applied to achieve a more balanced distribution of feature points, leading to moderate improvements in matching accuracy [5-6]. Other studies have adopted image partitioning methods, extracting feature points from individual blocks to avoid redundancy and overlapping. However, when a fixed threshold is used during feature extraction, feature points still tend to cluster in local regions [7-9].

Regarding the high mismatch rate of ORB, some researchers have enhanced descriptor discriminability. An improved retinal-like descriptor with 512-dimensional binary features has been proposed to increase distinctiveness, although this approach significantly increases computation time [10-12]. Another approach replaces the rBRIEF descriptor with a 128-bit descriptor based on the Local Binary Pattern (LBP) algorithm, which strengthens local texture representation and reduces mismatches [13]. Similarly, LATCH has been used instead of rBRIEF to achieve better matching performance than the traditional ORB algorithm, but at the cost of increased computational time [14-16].

Based on the above studies, this paper proposes an improved ORB algorithm. First, the image is divided into blocks, and the information entropy of each block is calculated. The feature extraction threshold for each block is then dynamically adjusted according to its entropy value, resulting in a more uniform distribution of feature points and improved registration accuracy. Second, the ORB-LBP method is further enhanced by incorporating pixel gradient information in addition to gray-level comparisons, thereby improving the accuracy of feature description and effectively reducing the mismatch rate.

2. ORB Algorithm

The ORB algorithm was proposed by Ethan Rublee in 2011. The FAST algorithm is used to extract features, the gray-scale centroid algorithm is used to obtain the main direction, rBRIEF is used to acquire the feature descriptor, and finally RANSAC is used to delete matching error points.

2.1. Feature Point Extraction

In ORB algorithm, FAST algorithm is used to extract feature points. First, select any point in the image as the circle center and use 3 pixels as the radius to calculate the absolute difference between the gray value of the 16 circle pixels and that of the circle center. If it is greater than the set threshold, it is 1, otherwise it is 0, as shown in formula (1):

$$f_{def} = \begin{cases} 1 , & |I_{x,\ y} - I_p| > t \\ 0 , & \text{others} \end{cases} \tag{1}$$

Where, I_p is the gray value of the circle center P, $I_{x\ ,\ y}$ is the gray value of the pixel point(x, y), $x, y \in [-3,3]$ and t is the set threshold.

Secondly, count the number of continuous f_{def} with a value of 1. If it is greater than the set threshold T, P is a feature point.

Since the feature points extracted by the FAST algorithm do not possess rotation invariance, the ORB algorithm makes improvement on this. Take the feature point as the center, find the gray centroid of its neighborhood to construct a vector from the feature point to the gray centroid. Then, use the direction of this vector as the direction of the feature point. The image block moment is calculated using formula (2).

$$M_{p,q} = \sum_{x,y\in S} x^p\, y^q I(x, y) \qquad p, q = \{0,1\} \tag{2}$$

Where, $I(x, y)$ is the gray value of the pixel point (x, y), and the gray centroid is calculated as shown in formula (3):

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right) \tag{3}$$

Where, $m_{00}$is the zero-order moment, m_{10} and m_{10} are the first-order moments.

Connect the feature point P with the gray centroid C to derive the direction vector $\overrightarrow{PC}$. The direction of the feature point is shown in formula (4):

$$\theta = arctan(\, m_{01}, m_{10}) \tag{4}$$

2.2. Description of Feature Points

In ORB algorithm, BRIEF algorithm is used to describe the extracted feature points. Randomly extract n pairs of pixel points in the neighborhood of the feature points and compare the gray values with formula (5).

$$\tau(p,x,y)=\begin{cases}1, p(x)<p(y)\\0, p(x)\geq p(y)\end{cases} \tag{5}$$

Where, $p(x)$ is the gray value of the feature point in the neighborhood $x=(u,v)^T$. After $n$comparisons, a binary string of length n is derived. The calculation method is shown in formula (6):

$$f_n=\sum_{1\leq i\leq n}2^{i-1}(p,x_i,y_i) \tag{6}$$

Since the descriptor derived from BRIEF algorithm does not have directionality, the main direction θ is used for correction in the ORB algorithm. Suppose Q is the coordinate matrix of random pixel pairs generated by the BRIEF algorithm, which can be expressed by formula (7):

$$Q=\begin{bmatrix}x_1,x_2,x_3\cdots x_n\\y_1,y_2,y_3\cdots y_n\end{bmatrix} \tag{7}$$

Use formula (8) to check the direction and construct a BRIEF descriptor with rotation invariance.

$$Q_\theta=R_\theta\cdot Q \tag{8}$$

Where, R_θ is shown in formula (9).

$$R_\theta=\begin{pmatrix}\cos\theta & \sin\theta\\-\sin\theta & \cos\theta\end{pmatrix} \tag{9}$$

3. Improved ORB algorithm

Although image segmentation has been used to alleviate the uneven distribution of feature points, feature clustering can still occur within individual segments [7-9]. Information entropy can be used to measure the complexity of image content: a higher entropy value indicates a more complex region with richer information.

Based on the entropy of each image block, this study adaptively adjusts the feature extraction threshold. Blocks with higher entropy are assigned higher thresholds, while blocks with lower entropy use lower thresholds. This adaptive strategy promotes a more uniform distribution of extracted feature points and improves the accuracy of subsequent image registration.

The ORB-LBP approach enhances texture representation and reduces mismatches to some extent [13]. However, because LBP constructs descriptors solely from gray-level comparisons, different feature points may still produce identical descriptors, which can degrade matching accuracy.

Building on this method, the present study incorporates pixel gradient information into the LBP process. By combining gray-level and gradient comparisons, the descriptor captures richer local characteristics around each feature point, leading to improved distinctiveness and a further reduction in mismatch rate.

3.1. Improvement of feature point extraction threshold

The image information entropy reflects distribution characteristics of gray information in the image. The more information an image contains, the greater the entropy value is. For a gray-scale image, formula (9) can be used to calculate the amount of information it contains in formula (10).

$$H(x)=-\sum_{x\in X}p(x)\cdot log(p(x)) \tag{10}$$

Where, x represents the gray value of each pixel in the image, X represents the gray level, and $p(x)$ represents the probability that the gray value x is in X.

When extracting feature points, first construct a four-layer image pyramid to solve the scale invariance. Second, divide each layer of images into $N\times M$ blocks, and calculate the information entropy of each block. Then, according to formula (10), determine the threshold for extracting feature points of each block as in formula (11).

$$t=\frac{E_o,i}{\bar{E}_o}\times T \tag{11}$$

Where, t represents the threshold for extracting feature points in the block i on the layer o, $E_{o,i}$ represents the information entropy of the image in the block i on the layer o. T is a fixed value, which is 40 in the experiment? $\bar{E}_o$ represents the mean value of the image information entropy on the layer o.

To verify the adaptive threshold algorithm based on image information entropy proposed herein, a standard Lena image is taken as an example to compare the distribution of the extracted feature points. The experimental results are shown in Figure 1. (a) shows distribution of the feature points extracted by the traditional ORB algorithm, (b) shows distribution of the feature points extracted by 3×3 image block, (c) shows distribution of feature points extracted by the adaptive threshold algorithm after 3×3 image block is divided. It can be clearly seen from Figure 1 that feature points extracted by this algorithm have more uniform distribution.

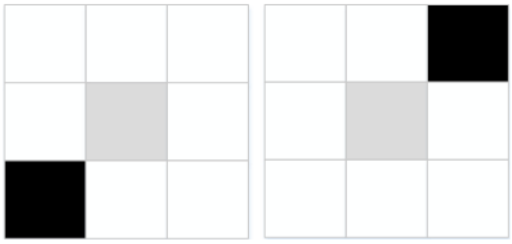
(a) Traditional ORB algorithm (b) Block algorithm (c) Algorithm herein.

Figure 1. Comparison of distribution of feature points.

3.2. Improvement of feature point description

The traditional ORB algorithm does not fully exploit local image information, which can limit registration accuracy. To address this issue, the ORB-LBP method first generates a 128-bit descriptor using the LBP algorithm and then replaces the 128 bits with the smallest variance in the 256-bit rBRIEF descriptor [13]. Because LBP provides a stronger representation of local texture patterns, this hybrid descriptor improves matching accuracy to a certain extent.

However, the LBP algorithm only compares the gray value of each pixel in the 3×3 area around the center pixel with that of the center pixel If it is greater than the gray value of the center pixel, the comparison result is 1, otherwise it is 0. In comparison, start with a certain pixel around the center point, select order or reverse order for 8 comparisons, and finally get an 8-bit binary descriptor. Due to the different starting positions and selection order of the selected pixels during the comparison, a total of 28 descriptors can be derived. To enable rotation invariance in description of feature points by LBP algorithm, the smallest LBP value is selected as the descriptor of the feature points. However, for the two different areas a and b on the same image as shown in Figure 2, after selecting the smallest LBP, the descriptors formed are the same, both are 00000001, which affects the subsequent registration accuracy.



(a) Area a **(b) Area b.**

Figure 2. Different regions with the same descriptor.

Based on this, this paper introduces the gradient information [17,18] of pixels to improve the LBP algorithm. Without loss of generality, suppose the gray value of the central pixel is 100 and that of the black pixel is 255, then the gradient value matrix in the two directions x, y in area a in Figure 2 is shown in equations (12) and (13):.

$$g_{ax} = \begin{bmatrix} 0 & 0 & 0 \\ 100 & 0 & -100 \\ -255 & -127.5 & 0 \end{bmatrix} \tag{12}$$

$$g_{ay} = \begin{bmatrix} 0 & 100 & 0 \\ 127.5 & 0 & 0 \\ 255 & -100 & 0 \end{bmatrix} \tag{13}$$

The gradient value matrices in the two directions x, y in area b in Figure 2 are shown in equations (14) and (15):

$$g_{bx} = \begin{bmatrix} 0 & 127.5 & 255 \\ 100 & 0 & -100 \\ 0 & 0 & 0 \end{bmatrix} \tag{14}$$

$$g_{by} = \begin{bmatrix} 0 & 100 & -255 \\ 0 & 0 & -127.5 \\ 0 & -100 & 0 \end{bmatrix} \tag{15}$$

When each pixel in the 3×3 neighborhood is compared with the center pixel, each pixel is compared with the center pixel in the order of gray value, gradient value in x direction, and gradient value in y direction. Each comparison result is a 3-bit binary descriptor, and finally an 8×3=24-bit binary descriptor is derived. Similarly, because the starting pixel position and sequence in the comparison differ, 28 24-bit binary descriptors can be derived, and then the smallest LBP value is selected as the descriptor. The descriptor of the feature point of area a in Figure 2 is 000000000000101011000001, and that of area b in Figure 2 is 000000000000001000001111, so the two are different. It can be seen that after making full use of the relevant pixel information, feature point description will have higher discrimination.

However, if the area a and area b shown in Figure 2 belong to two different images, and area b is derived by rotating the area a, then after the minimum LBP descriptor is selected as above, the two descriptors are different, and mismatch will occur during registration of the two images.

In this regard, this paper no longer determines the descriptor based on the minimum LBP value, but uses the gray centroid in the image to determine the starting position of the contrast pixel. The specific method is as follows: First, use formula (2), (3) to calculate the gray centroid of the global image. Secondly, calculate the distance from the 8 pixels around the feature point to the gray centroid, and select the pixel with the smallest distance as the starting position. Finally, select pixels in a clockwise direction for comparison. When the image is rotated, the direction of the gray-scale centroid also changes, so the descriptor generated by the improved LBP algorithm features rotation invariance.

To avoid noise interference, this paper uses a 3×3 pixel area to replace a single pixel when designing the algorithm. The specific steps of the improved algorithm are as follows:

Step1: Construct a 2×2 main block in the neighborhood of the feature point, each main block is divided into 3×3 sub-blocks, and each sub-block is composed of 3×3 pixels.

Step2: Calculate the average gray value of each sub-block, the average gradient in the x direction and the y direction. Since the gradient has a positive value and a negative value, to express the magnitude of the gradient value change, calculate $g=\frac{1}{n}\sum_n\sqrt{{g_x}^2+g_y^2}$. g_x, g_y are respectively the gradient in the x, y directions of each pixel in the sub-block, and n is the number of pixels in the sub-block.

Step3: In each main block, select the sub-block closest to the image gray centroid as the starting position, and compare it with the average gray value, the average gradient in the directions x and y, and the value g of the center area one by one in a clockwise direction. The result of the comparison generates a 4-bit binary, and a total of 32-bit binary descriptors are generated for 3×3 sub-blocks as a description of the main block.

Step4: Starting from the main block closest to the image gray centroid, the description of the 2×2 main block is connected clockwise to form a 128-bit descriptor.

Step5: Replace 128-bit descriptor with a smaller variance in 256 descriptors generated by the rBrief algorithm with 128-bit descriptor generated in Step4, which is used as the final descriptor of the feature point.

4. Experimental results and analysis

To verify this algorithm, the test data adopts Mikolajczyk public data set [19], and 4 sets of images are selected for 4 experiments on illumination invariance, blur invariance, scale rotation invariance, and affine invariance. 5 images are randomly selected from each set of images to record average consumed time and matching accuracy. Figure 3 shows the registration results of the traditional ORB algorithm and the improved algorithm in this paper in one of the four experiments. Where, the first picture of each group represents the registration result of the traditional ORB algorithm, and the second picture represents the registration result of the improved algorithm.

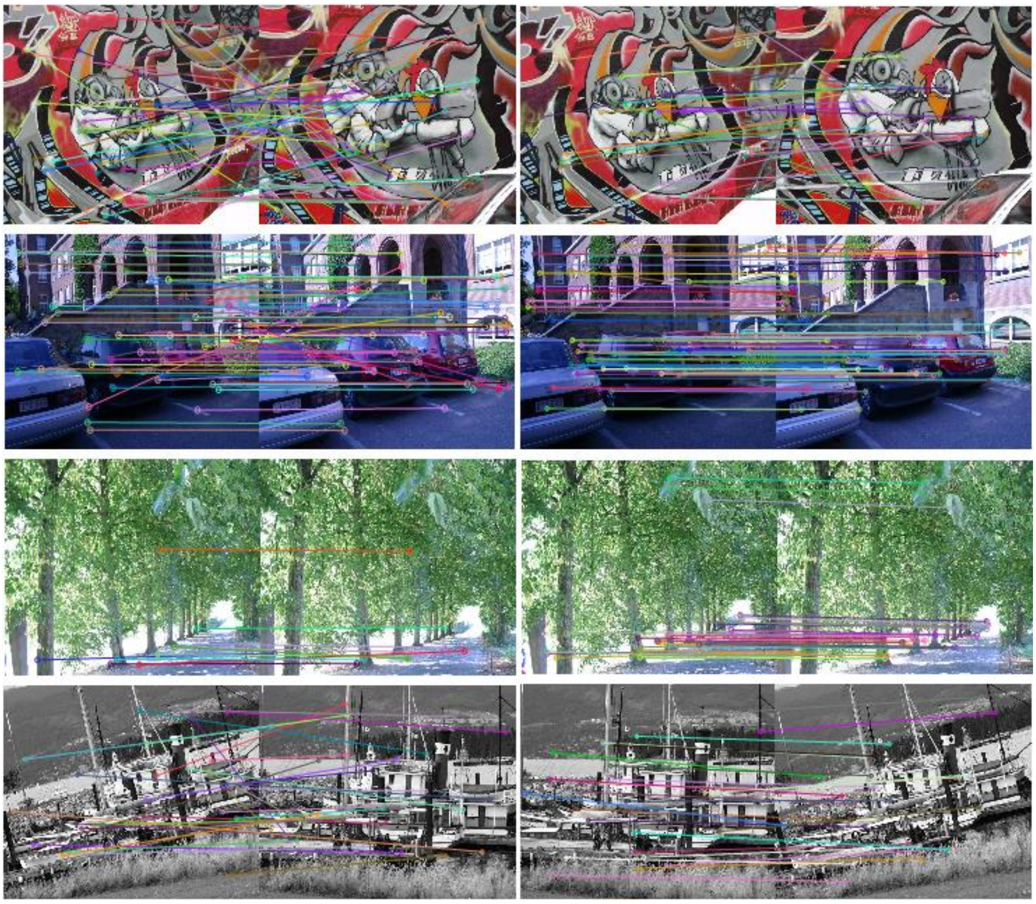


Figure 3. Matching results of feature points between our algorithm and traditional orb algorithm.

The first group in the figure shows the comparison result of affine invariance. After introduction of gradient information to the improved algorithm, it strengthens the description of the local texture. At the same time, gray centroid is used to determine the starting position of the generated descriptor, which shows certain affine invariance and reduces the mismatch rate. The second group is the comparison result of illumination invariance. Since the average gray value of 3×3 pixel area and average gradient in direction x, y are used, there is illumination invariance, with number of matched feature points and mismatch rate significantly superior. The third group shows the comparison result of blur invariance. Also, because single pixel is replaced with 3×3 pixel area for comparison, it can be seen that the number of correctly matched feature points is significantly higher compared with the traditional ORB algorithm. The fourth group shows the comparison result of rotation invariance. Since the gray centroid is used for direction correction and the local information description is more accurate, the number of registrations has been greatly improved. At the same time, due to the adoption of block and adaptive threshold method, the extracted feature points are distributed more uniformly, with registration accuracy improved to a certain extent.

Time comparison of the algorithm is shown in Table 1:

Table 1. Comparison of average running time of algorithms.

Experimental item	Time consumption of traditional ORB algorithm(s)	Time consumption of algorithm[13] (s)	Time consumption of the algorithm herein (s)
Affine invariance	1.049146	1.227501	0.954722

Illumination invariance	1.031889	1.201223	0.949338
Blur invariance	1.011814	1.170495	0.922647
Rotation invariance	1.100543	1.265473	1.056521

It can be seen from Table. 1 that, due to adoption of LBP algorithm, time consumption is increased by about 17% compared with the traditional ORB algorithm [13], while time consumption of the algorithm in this paper is reduced by about 8% compared with the traditional ORB algorithm. In analysis of the reason, the algorithm herein adopts image block, the extraction time of each block's feature point is 1/N of the original, and the registration time is $\frac{N+1}{2N}$, so the original running time will be greatly reduced. However, when determining the threshold for extracting feature points in each block, the information entropy of each block needs to be calculated, which increases the time overhead [18-20]. In addition, when the improved algorithm generates descriptor, as gradient information needs to be calculated, time overhead is increased to a certain extent. In overall, the algorithm herein still has a certain improvement in time consumption.

The average registration accuracy of the algorithm is compared as shown in Table. 2:

Table 2. Comparison of average registration accuracy.

Experimental item	registration accuracy of the traditional ORB algorithm (%)	registration accuracy of the algorithm [13] (%)	registration accuracy of the algorithm herein (%)
Affine invariance	64.3	73.6	80.2
Illumination invariance	85.7	90.5	94.5
Blur invariance	83.2	89.6	95.8
Rotation invariance	88.7	92.4	96.2
Average	80.48	86.53	91.68

It can be seen from Table. 2 that, as it adopts the descriptor generated by the LBP and rBRIEF hybrid algorithm to describe the feature points, the mismatch rate is reduced to a certain extent [13]. In the 4 sets of experiments, the average registration accuracy is 8% higher than that of the traditional ORB. The algorithm in this paper adds a comparison of gradient information on the basis of Ref [13], so description of the information around the feature point is more accurate, and it avoids the situation in section 2.2 that the traditional LBP algorithm may have the same descriptor in different image regions, thus further increasing registration accuracy. At the same time, after the image is divided into blocks, the extraction threshold of the feature points is set adaptively according to the information entropy of each block, so that distribution of the extracted feature points is more uniform, which lowers the mismatch rate to a certain extent. According to the experimental data in Table 2, the average registration accuracy of the algorithm in this paper is about 14% higher compared to the traditional ORB algorithm and 6% higher compared to the algorithm proposed [11].

5. Conclusion

In this paper, improvement is made in view of the problem of uneven distribution of feature points extracted from traditional ORB algorithm and low matching accuracy. First, after the image is divided into blocks, the threshold for feature point extraction is dynamically set using information entropy. Feature points can also be effectively extracted in blocks with small information entropy, which further improves the uniformity of the extracted feature points and reduces the subsequent mismatch rate. Secondly, the ORB-LBP algorithm is improved. To improve the discrimination of feature point descriptors in the LBP algorithm, gradient information is introduced. Experiments show that the improved ORB algorithm improves the registration accuracy with less time overhead and better robustness.

Funding: This work was supported in part by the Key R&D and Promotion Project of Henan Province (Science and Technology Research) No.202102310204.

References

1. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91-110, 2004. doi: 10.1023/b:visi.0000029664.99615.94
2. H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," In *European conference on computer vision*, May, 2006, pp. 404-417. doi: 10.1007/11744023_32
3. 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, no. 1, pp. 105-119, 2008.
4. M. S. Shang, "An image registration method based on SIFT and ORB," In *2022 2nd Conference on High Performance Computing and Communication Engineering (HPCCE 2022)*, April, 2023, pp. 366-373.
5. C. Wang, W. Jiao, S. Zhang, and Y. Song, "Improved ORB Image Matching Algorithm Based on Local Information Entropy," In *2023 5th International Conference on Frontiers Technology of Information and Computer (ICFTIC)*, November, 2023, pp. 658-663. doi: 10.1109/icftic59930.2023.10456220
6. B. Li, J. Wang, J. Chen, and Z. Sun, "OGIL: An Improved ORB Algorithm Based on GMS, Information Entropy and LC," In *2023 35th Chinese Control and Decision Conference (CCDC)*, May, 2023, pp. 1164-1169.
7. W. Wang, Z. Xiong, and Z. Liu, "ORB Feature Uniform Distribution Algorithm Integrating Quadtree and Adaptive Thresholding," In *Journal of Physics: Conference Series*, May, 2025, p. 012016. doi: 10.1088/1742-6596/3024/1/012016
8. Z. Tang, Y. Dou, X. Wang, and L. Zhang, "Enhanced Feature Matching: Entropy-Guided ORB with Adaptive Descriptors," 2025. doi: 10.21203/rs.3.rs-7535082/v1
9. Q. Liu, X. Xi, W. Zhang, L. Yang, and N. Hanajima, "Improved image matching algorithm based on LK optical flow and grid motion statistics," *International Journal of Computer Applications in Technology*, vol. 68, no. 1, pp. 49-57, 2022.
10. L. Li, L. Wu, and Y. Gao, "Improved image matching method based on ORB," In 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), May, 2016, pp. 465-468. doi: 10.1109/snpd.2016.7515941
11. C. Liu, and S. Dang, "The research and application of improved ORB feature matching algorithm," In *International Conference on Guidance, Navigation and Control*, August, 2022, pp. 4246-4255. doi: 10.1007/978-981-19-6613-2_413
12. C. Ma, X. Hu, L. Fu, and G. Zhang, "An improved ORB algorithm based on multi-feature fusion," In *2018 IEEE 27th International Symposium on Industrial Electronics (ISIE)*, June, 2018, pp. 729-734. doi: 10.1109/isie.2018.8433586
13. Y. Zhang, C. Li, C. Cao, and Y. Gao, "An improved Orb feature point matching algorithm," In *Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence*, December, 2018, pp. 207-211. doi: 10.1145/3297156.3297184
14. L. Zhao, J. Yang, Y. Zhang, and J. Huang, "Research on feature matching of an improved orb algorithm," In *2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC)*, March, 2022, pp. 765-769. doi: 10.1109/itoec53115.2022.9734583
15. H. Sun, L. Wei, Y. Jiang, Y. Su, S. Deng, H. Sun, and H. Wang, "Research on Mobile Robot Localization Algorithm with Improved ORB Extraction Matching," In *2024 IEEE International Conference on Mechatronics and Automation (ICMA)*, August, 2024, pp. 351-356. doi: 10.1109/icma61710.2024.10633169
16. Q. Huang, X. Guo, Y. Wang, H. Sun, and L. Yang, "A survey of feature matching methods," *IET Image Processing*, vol. 18, no. 6, pp. 1385-1410, 2024. doi: 10.1049/ipr2.13032
17. C. Yao, H. Zhang, J. Zhu, D. Fan, Y. Fang, and L. Tang, "ORB feature matching algorithm based on multi-scale feature description fusion and feature point mapping error correction," *IEEE Access*, vol. 11, pp. 63808-63820, 2023. doi: 10.1109/access.2023.3288594
18. S. Wang, A. Zhang, and Y. Li, "Feature extraction algorithm based on improved ORB with adaptive threshold," In *2023 IEEE International Conference on Industrial Technology (ICIT)*, April, 2023, pp. 1-6. doi: 10.1109/icit58465.2023.10143089
19. 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. doi: 10.1109/tpami.2005.188
20. Y. Cong, "Image stitching technology for police drones using an improved image registration method incorporating ORB algorithm," *Informatica*, vol. 48, no. 2, 2024. doi: 10.31449/inf.v48i2.5877