



OPEN ACCESS

EDITED BY

Jia-Bao Liu,
Anhui Jianzhu University, China

REVIEWED BY

Hongxing Liu,
Nanjing University, China
Wenle Wang,
Jiangxi Normal University, China

*CORRESPONDENCE

Zhong Yang
YangZhong@nuaa.edu.cn

RECEIVED 06 September 2022

ACCEPTED 21 September 2022

PUBLISHED 09 November 2022

CITATION

Xue B, Yang Z, Liao L, Zhang C, Xu H
and Zhang Q (2022) High precision
visual localization method of UAV
based on feature matching.
Front. Comput. Neurosci. 16:1037623.
doi: 10.3389/fncom.2022.1037623

COPYRIGHT

© 2022 Xue, Yang, Liao, Zhang, Xu and
Zhang. This is an open-access article
distributed under the terms of the
[Creative Commons Attribution License
\(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or
reproduction in other forums is
permitted, provided the original
author(s) and the copyright owner(s)
are credited and that the original
publication in this journal is cited, in
accordance with accepted academic
practice. No use, distribution or
reproduction is permitted which does
not comply with these terms.

High precision visual localization method of UAV based on feature matching

Bayang Xue¹, Zhong Yang^{2*}, Luwei Liao², Chi Zhang², Hao Xu³
and Qiuyan Zhang⁴

¹Nanjing Taiside Intelligent Technology Co., Ltd., Nanjing, China, ²College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China, ³School of Mathematics and Physics, Anhui University of Technology, Ma'anshan, China, ⁴Guizhou Electric Power Research Institute, Guizhou Power Grid Co., Ltd., Guiyang, China

In this paper, the precision hovering problem of UAV operation is studied. Aiming at the diversity and complexity of the UAV operating environment, a high-precision visual positioning and orientation method based on image feature matching was proposed. The image feature matching based on the improved AKAZE algorithm is realized, and the optimal matching point pair screening method based on the fusion of Hamming distance and matching line angle is innovatively proposed, which greatly improves the robustness of the algorithm without affecting the performance of the algorithm. The real-time image is matched with the benchmark image for image feature matching. By reducing the deviation of image feature, the pose state correction of UAV hovering is achieved, and the precision hovering of the UAV is realized. Both simulation and real UAV tests verify the effectiveness of the proposed UAV high-precision visual positioning and orientation method.

KEYWORDS

UAV, AKAZE, image matching, localization and orientation, hovering

Introduction

In the inspection process of bridges, towers, transmission lines, and other facilities, unmanned aerial vehicles (UAVs) often need to hover at fixed points for key parts. The stability of the UAVs hovering will greatly affect the quality of the operation, and even affect the smooth completion of the operation. At present, UAVs mainly rely on the inertial navigation system and the global navigation satellite system's (GNSS) navigation and positioning system. Due to the accumulation of errors in the inertial navigation system, the GNSS is affected by signal strength and accuracy is limited, and the UAVs are hindered by external wind gusts, which often leads to pose deviation and is difficult to correct.

Using traditional navigation methods, it is difficult to achieve high-precision fixed-point hovering of UAVs. At present, many UAVs' pose guidance through machine vision has been carried out at home and abroad. The fixed-point hovering control of small quadrotor UAVs based on optical flow and ultrasonic module is mentioned in literature (Zhang et al., 2018). However, the optical flow algorithm will be disturbed by the slight shaking of ground objects and the change of light and shade, and the robustness of the

algorithm is poor. Literature (Yu et al., 2021) proposed the indoor fixed-point hovering method of UAVs based on the pyramid optical flow method. This method can only obtain the horizontal moving speed of the UAV, and its height needs to depend on the barometer, so the algorithm has poor robustness in corridors and other scenes. Wang et al. (2019) proposed a fixed-point hovering method for a quadrotor based on the Harris algorithm. The corner detection method was used to obtain the overlapping area of the image to calculate the UAV height. However, the corner detection method is not scale invariant and is only suitable for hovering control of the UAV under small disturbances. Therefore, this paper proposes an image matching method based on image scale-invariant features for UAV pose guidance during hovering operation.

As shown in Figure 1, the schematic diagram of the UAV visual guidance system is based on feature matching. In UAV operation, the reference image is set as the reference matching object, and the visual navigation computer preprocesses the real-time image, further extracts image features, and matches image features with the reference image. The positioning information of the real-time image relative to the reference image, including translation, rotation, and scaling, is calculated and entered into the flight control computer as the pose compensation parameters to correct the UAV pose. By reducing the feature deviation between the real-time image and the reference image, accurate visual guidance can be achieved when the UAV is hovering.

During the flight of the UAV, the dynamic images are often affected by rotation, scale scaling, brightness change, angle of view change, affine transformation, noise, and so on. In this case, to achieve better image matching, it is necessary to find an image matching algorithm that can overcome these disturbances. In order to satisfy the scale invariance of the feature matching algorithm, it is necessary to construct different scale spaces for feature extraction. The main construction methods of scale space are as follows: (1) convolution of gaussian kernel function with gray image; and (2) construct by using a nonlinear filtering function.

The matching algorithms that use the Gaussian kernel function to construct scale space include the SIFT algorithm (Bellavia, 2022; Liu et al., 2022), SURF algorithm (Liu et al., 2019a; Liu Z. et al., 2021; Fatma et al., 2022), and ORB algorithm (Liu et al., 2019b; Chen et al., 2022; Xie et al., 2022; Xue et al., 2022), etc. This kind of algorithm has good robustness and fast matching speed, but the Gaussian kernel convolution operation will lead to the loss of edge information of the image, which seriously affects the stability of feature points and descriptors. The main algorithms that use nonlinear filtering function to construct scale space include the KAZE algorithm (Khalid et al., 2021; Liu J.-B. et al., 2021; Roy et al., 2022), AKAZE algorithm (Sharma and Jain, 2020; Ji et al., 2021; Pei et al., 2021; Yan et al., 2021), etc. The scale space constructed by the nonlinear filtering function can better protect the information at the

edge of the image and increase the robustness of the matching algorithm. The feature point detection and descriptor of the KAZE algorithm are all borrowed from the SURF algorithm, but the robustness of the KAZE algorithm is stronger than that of the SURF algorithm, which is enough to prove the superiority of nonlinear scale space. AKAZE (Accelerated-KAZE) is an improved feature point detection and description algorithm proposed by KAZE at the 2012 ECCV Conference. AKAZE improves the KAZE algorithm in two main ways:

- (1) The fast explicit diffusion (FED) framework is introduced to solve partial differential equations. The scale space established by FED is faster than other nonlinear models, and more accurate than the airborne optical sectioning (AOS) method;
- (2) An efficient modified local difference binary descriptor (M-LDB) is introduced to improve the rotation and scale invariant robustness compared with the original LDB, and the scale spatial gradient information constructed by FED is combined to increase the uniqueness.

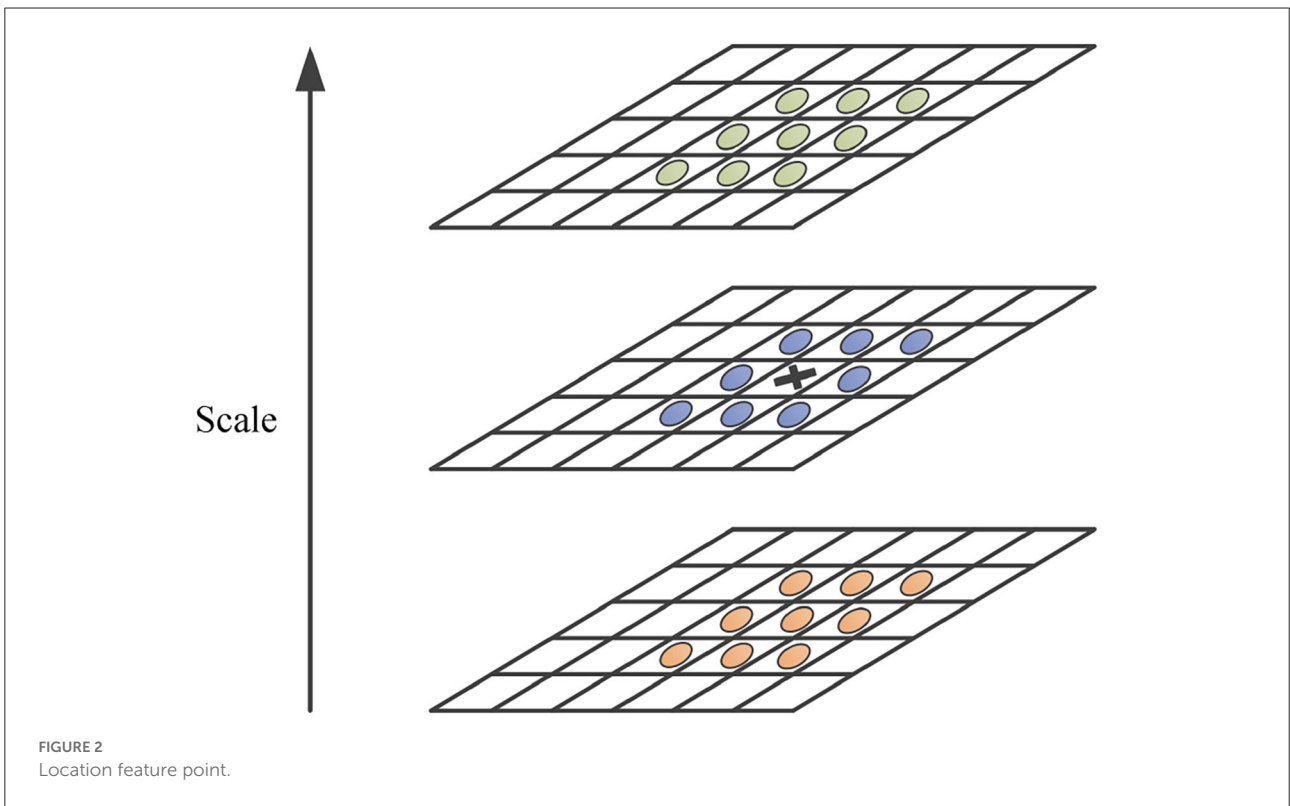
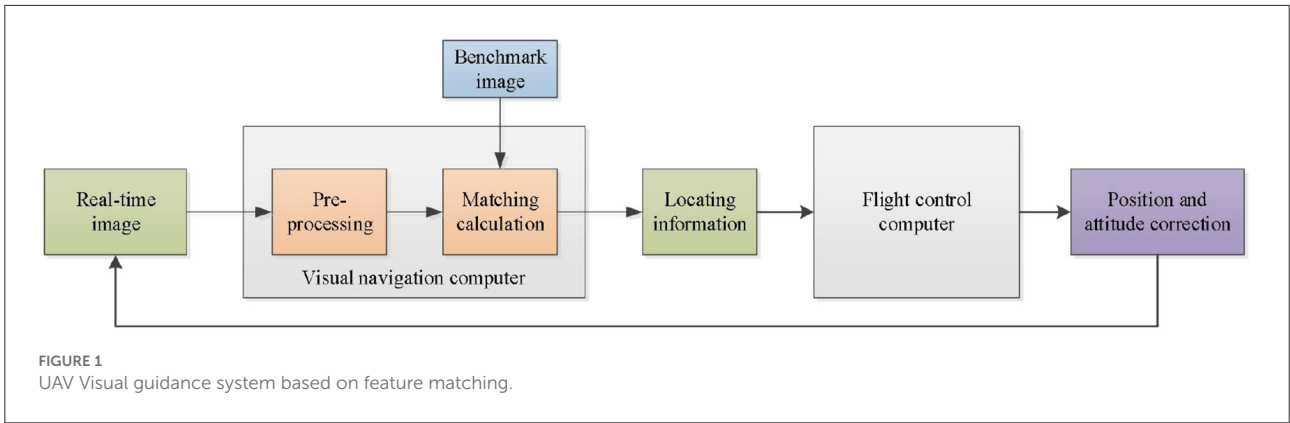
Compared with the SIFT and SURF algorithms, the AKAZE algorithm is faster. Meanwhile, compared with the BRISK (Niyishaka and Bhagvati, 2020; Singh and Singh, 2020; Shi et al., 2021), FAST (Feng et al., 2021; Yang et al., 2022; Zhang and Lang, 2022) and ORB algorithms, the repeatability and robustness are greatly improved. The descriptors obtained by the AKAZE's feature algorithm have rotation invariance, scale invariance, illumination invariance, space invariance, etc., and have high robustness, feature uniqueness, and feature accuracy. In view of the real-time requirements of the algorithm and the characteristics of rotation, scaling, and translation of the UAV operation images, this paper studies the UAV visual navigation (Cao et al., 2022; Guo et al., 2022; Qin et al., 2022; Zhao et al., 2022) algorithm based on the improved AKAZE algorithm, especially the precise hovering problem in UAV operation.

Visual navigation algorithm design

Visual navigation based on improved AKAZE algorithm

Construct nonlinear scale space

First, it is necessary to carry out nonlinear diffusion filtering on the image. The advantage of a nonlinear diffusion filtering (Li et al., 2016; Feng and Chen, 2017; Jubairahmed et al., 2019; Liu et al., 2019c) algorithm is that it can filter the image noise while preserving important boundary and other details. The nonlinear diffusion filtering algorithm is mainly a diffusion process in which the gray image changes at different scales are expressed



as flow functions. The nonlinear partial differential equation can be expressed by Equation (1).

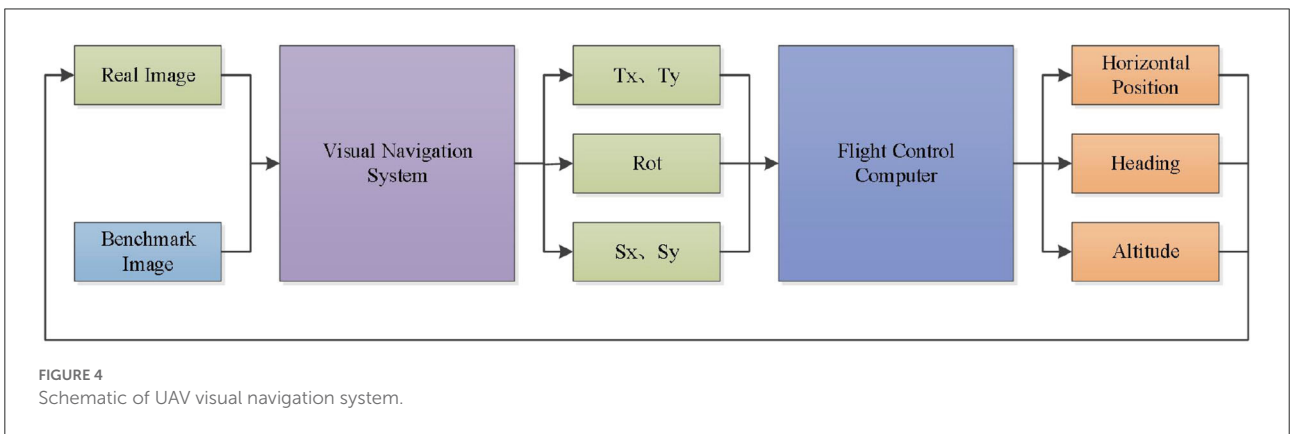
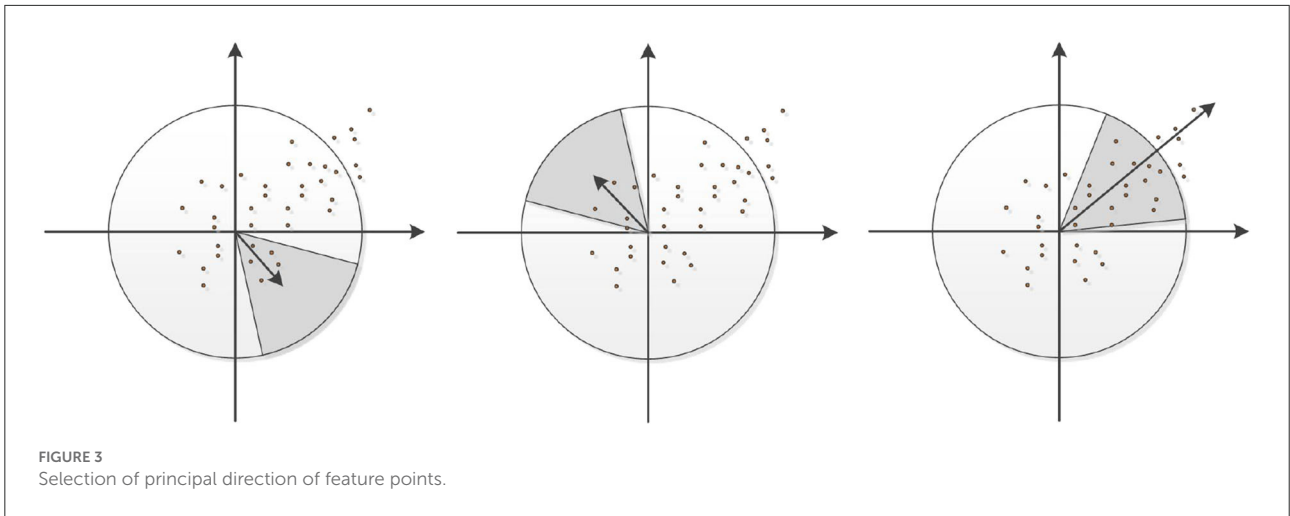
$$\frac{\partial L}{\partial t} = \text{div}(c(x, y, t)\nabla L) \tag{1}$$

where, L represents gray image information, div represents diffusion of flow function, ∇ represents image gradient, and $c(x, y, t)$ is conduction function. The time parameter t corresponds to the scale factor, which is controlled by the image gradient size during the diffusion process. The conduction function formula is defined by Equation (2).

$$c(x, y, t) = g(|\nabla L_\sigma(x, y, t)|) \tag{2}$$

where, ∇L_σ represents the gradient image of gray image L after Gaussian filtering. The conduction kernel function is selected optimally for regional diffusion smoothing, as shown in Equation (3).

$$g = \frac{1}{1 + \frac{|\nabla L_\sigma|^2}{\lambda^2}} \tag{3}$$



where, the parameter λ is the contrast factor controlling the diffusivity, and the decision factor determining the enhancement and flat region filtering in the edge region.

The AKAZE algorithm constructs nonlinear scale space in a similar way to the SIFT algorithm, both of which need to set groups o and layers s . The calculation formula of image scale parameters σ_i is expressed by Equation (4).

$$\sigma_i(o, s) = 2^{\frac{o+s}{s}} \tag{4}$$

The scope of the variable can be expressed by Equation (5).

$$o \in [0, 1, \dots, O - 1], s \in [0, \dots, S - 1], i \in [0, \dots, M - 1] \tag{5}$$

where, o is the number of groups, s is the number of layers in each group, and M is the total number of filtered images. The scale parameters σ of nonlinear scale space are converted into time units, and the mapping relationship can be expressed by Equation (6).

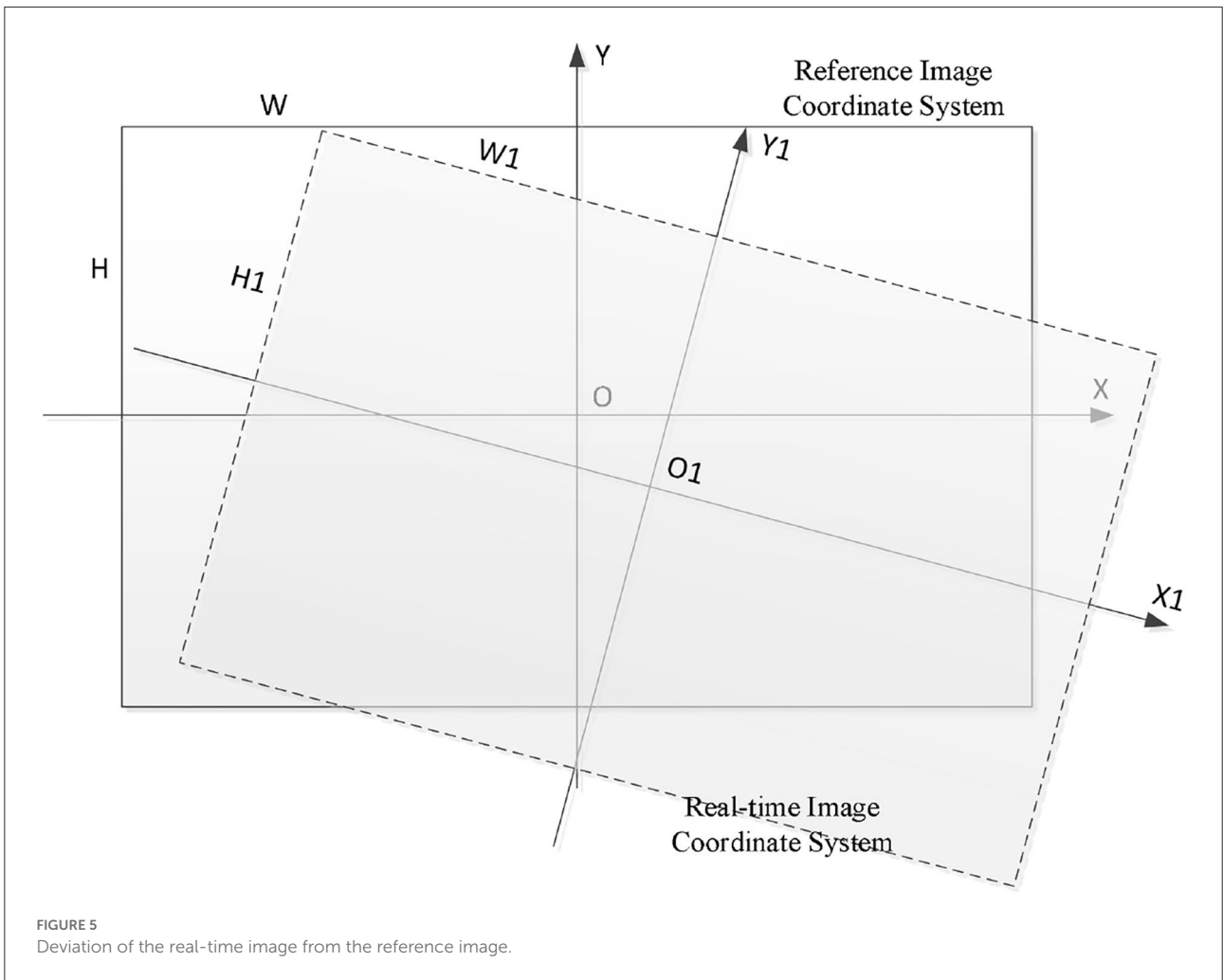
$$t_i = \frac{1}{2} \sigma_i^2, i = 0 \dots M \tag{6}$$

The core idea of the FED algorithm is to change the step size τ_j of n explicit diffusion processes to carry out M steps of the cycle. The formula for calculating the step size τ_j is expressed by Equation (7).

$$\tau_j = \frac{\tau_{\max}}{2 \cos^2 \frac{\pi(2j+1)}{(4n+2)}} \tag{7}$$

In the formula, τ_{\max} is the maximum step size threshold, and in order to ensure the stability of the constructed scale space, τ is smaller than τ_{\max} , and τ_{\max} represents the maximum iteration step that does not destroy the stability of the explicit equation. Equation (1) above can be expressed by a vectorization matrix, as shown in Equation (8).

$$\frac{L^{(i+1)} - L^i}{\tau} = A \left(L^i \right) L^i \tag{8}$$



where, $A(L^i)$ is the conduction matrix for image coding, and τ is a constant time step. In the explicit method, the solution of $L^{(i+1)}$ will be directly calculated through the previous image evolution L^i and image conduction function $A(L^i)$, as shown in Equation (9).

$$L^{(i+1,j+1)} = (I + \tau_j A(L^i)) L^{(i+1,j)}, j = 0, 1, \dots, n - 1 \quad (9)$$

The matrix $A(L^i)$ remains the same throughout the FED loop. When the FED loop ends, the algorithm recalculates the matrix $A(L^i)$.

Feature point detection and localization

After building the scale space, it is necessary to further detect the feature points. By calculating the determinant of the Hessian for each filtered image L^i in the non-linear scale space, the normalized differential multiscale operator σ_i^2 is used to find the maximum point of the determinant of the Hessian matrix.

As shown in Figure 2, the response value of the sampling point is compared with eight neighborhood points in the same scale and 2×9 neighborhood points in adjacent scales to determine whether it is the maximum value.

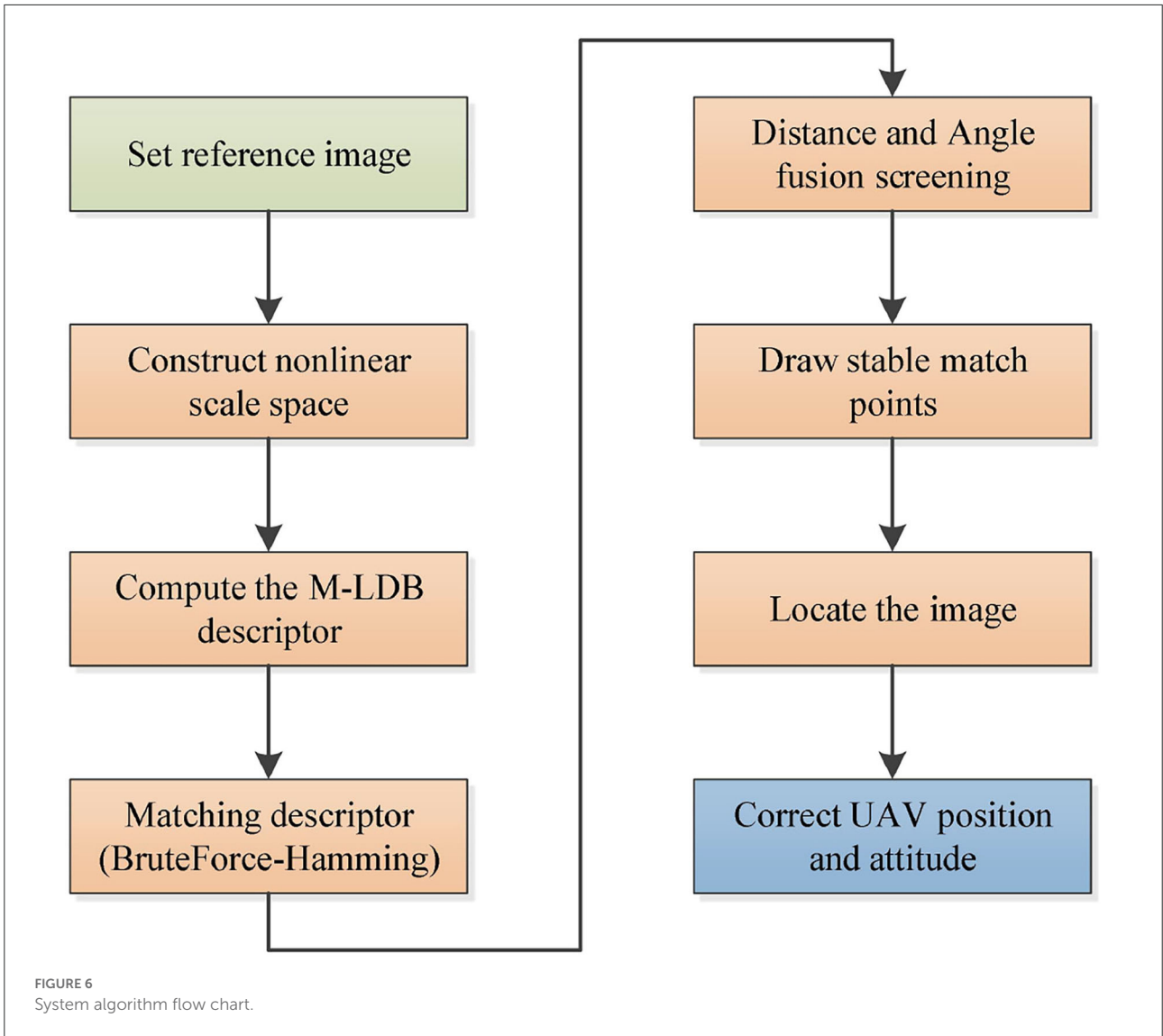
The determinant of the Hessian matrix is expressed by Equation (10).

$$L_{Hessian}^i = \sigma_i^2 (L_{xx}^i L_{yy}^i - L_{xy}^i L_{xy}^i) \quad (10)$$

where, σ_i is the initial value of scale factor; L_{xx}^i is the second transverse derivative; L_{yy}^i is the second longitudinal derivative; L_{xy}^i is the second cross differential.

Find the principal direction of the feature points

After locating the feature points, the principal direction of the feature points should be further solved. As shown in Figure 3, the circular area is determined with the feature points as the center and 6σ as the radius, and the first-order differential values



L_x and L_y of the points in the neighborhood of the feature points are calculated respectively, and the Gaussian weighting operation is carried out. Rotate the fan window with angle $\pi/3$, superimpose the vectors of points in the neighborhood, and select the direction of the longest vector in the sum of unit vectors as the main direction. Then rotate the sector window around the origin, recalculate all the Gaussian weighted vectors in the sector area after rotation, and repeat this operation until the whole circular area is counted. Finally, the direction represented by the sector region with the highest superposition value is taken as the principal direction of the feature point.

Construct the M-LDB descriptor

After extracting the feature points in the nonlinear scale space and establishing the principal direction of the feature

points, the feature points are described by the M-LDB descriptor. The specific construction method is as follows:

In the first step, the M-LDB descriptor is constructed by selecting an appropriate sampling region P centered on the feature points, and the sampling region P is divided into $n \times n$ meshes. The average value of all pixels in a $n \times n$ single grid is calculated by Equation (11).

$$I_{avg}(i) = \frac{1}{m_i} \sum_{k=1}^{m_i} I(k) \tag{11}$$

where, $I(k)$ is the pixel value of the gray image; m is the number of pixels in each partition grid; i is the number of grids that the sampling area is divided into.

In the second step, the gradients of pixels in each grid in the x and y directions are calculated and encoded according to the

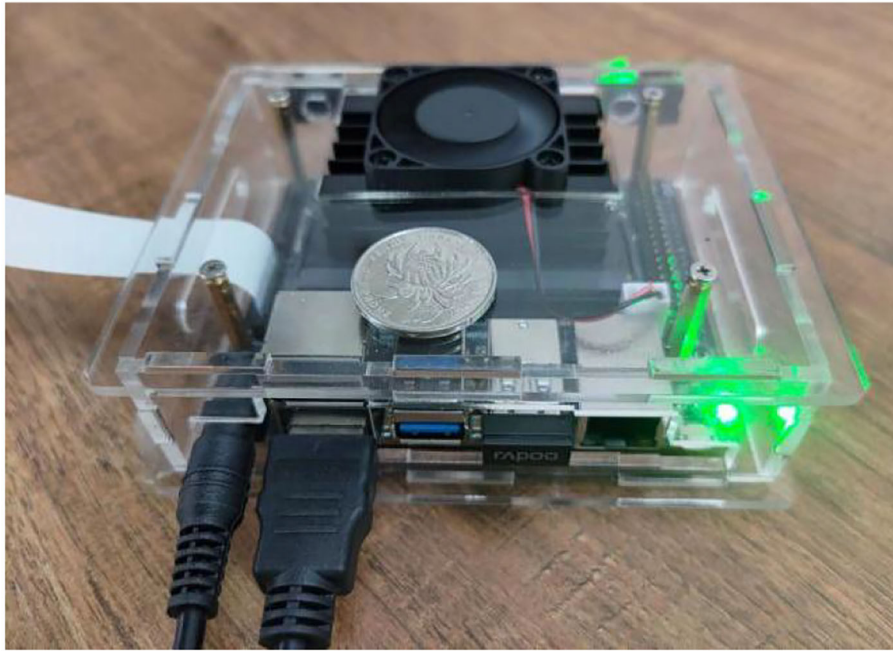


FIGURE 7
JETSON NANO B01.

TABLE 1 The algorithm verifies the platform performance parameters.

Item	Value
Computing Power:	472 GFLOPS
CPU:	Quad-core ARM® Cortex®-A57 MPCore processor
GPU:	NVIDIA Maxwell™ architecture with 128 NVIDIA CUDA® cores 0.5 TFLOPS(FP16)
Memory:	4 GB 64 bits LPDDR4 1600 MHz-25.6 GB/s
Operating System:	ubuntu 18.04 LTS

$$\tau (Func (i), Func (j)) = \begin{cases} 1 & (Func (i) - Func (j)) > 0 \\ 0 & \text{other} \end{cases} \quad (13)$$

Then the M-LDB descriptor can be expressed by Equation (14).

$$f_n = \sum_{1 \leq i < N} 2^{i-1} \tau (Func (i), Func (j)) \quad (14)$$

average pixel value of each grid and the gradient value of grid pixels, which can be expressed by Equation (12).

$$Func (\bullet) = \{Func_{intensity} (i), Func_{dx} (i), Func_{dy} (i)\} \quad (12)$$

where, $Func_{intensity} (i) = I_{avg} (i)$, $Func_{dx} (i) = Gradient_x (i) = dx$, $Func_{dy} (i) = Gradient_y (i) = dy$.

In the third step, it can be obtained from the above equation that the LDB descriptor compares the average intensity and gradient between paired grid cells respectively and is set to 0 or 1 according to the comparison results. The formula is expressed by Equation (13).

Descriptor matching and image localization

Since the M-LDB is a binary descriptor, Hamming distance is used to calculate the similarity of descriptors. Hamming distance can calculate the distance of binary descriptors only through XOR operation, so the matching efficiency of feature points is high. In addition, in order to improve the matching accuracy of feature points, the homography matrix between two images is calculated by the RANSAC algorithm, and the wrong matching is eliminated to obtain accurate matching results of feature points. The Hamming distance can be calculated by the following Equation (15).

$$dist (\vec{a}, \vec{b}) = \sum_i (a_i \neq b_i) \cdot 1 : 0 \quad (15)$$

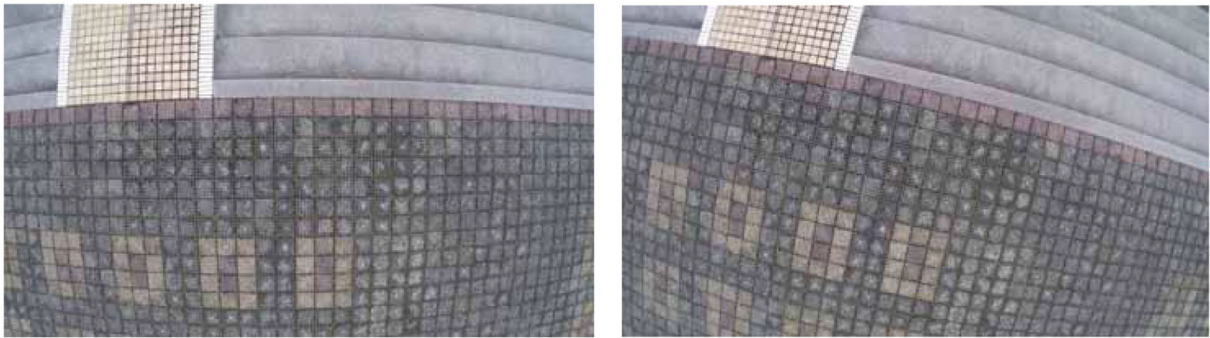


FIGURE 8
Images taken while the UAV is rotating.

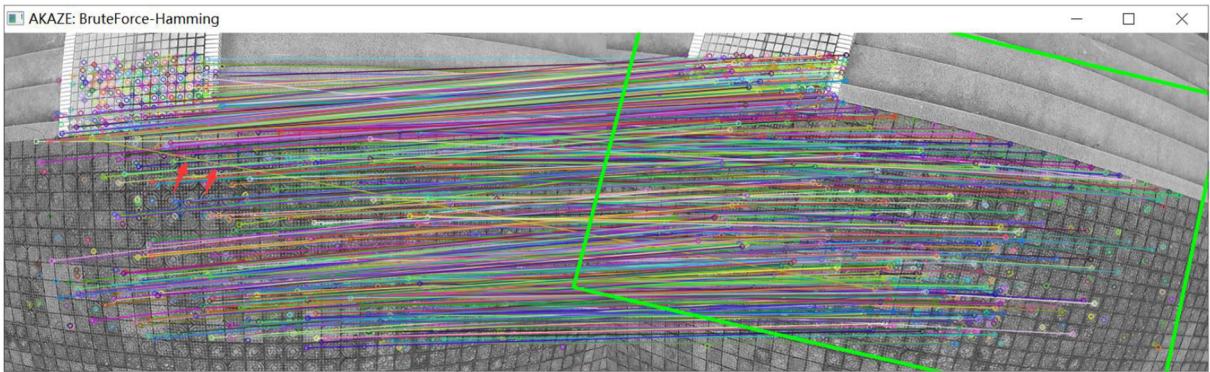


FIGURE 9
Matching effect of original algorithm.

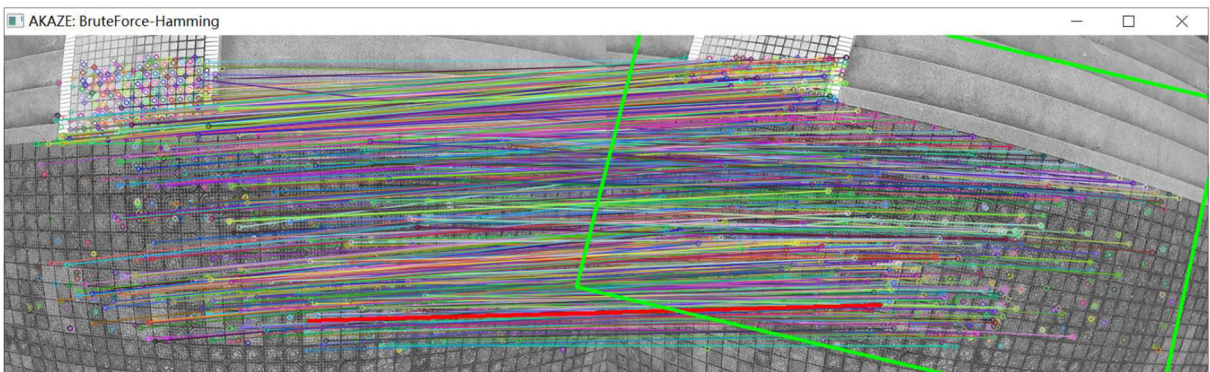


FIGURE 10
Matching effect of the optimized algorithm.

After the aforementioned feature point matching and screening, the correctly matched feature point pairs can be obtained, and the motion parameters of the real-time image and the benchmark image can

be correctly estimated through these feature point pairs. In this paper, affine transformation is used to model the motion between images, which is defined by Equation (16).

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = H \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \tag{16}$$

where, $H = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{pmatrix}$ is the affine transformation matrix, also known as the homography matrix, with 6 degrees of freedom; a_{13} and a_{23} are used to describe the translation change between images; a_{11} , a_{12} , a_{21} , and a_{22} are used to describe rotation and scaling changes between images; (x', y') represents the pixels of the previous image; (x, y) represents the pixels of adjacent images after affine transformation.

Matching point selection based on hamming distance and matching line angle fusion

In this paper, the Hamming distance is considered as a preliminary method to eliminate mismatches, and an adaptive threshold method is proposed for different scenes. The specific steps are as follows:

- (1) First, the matching points are sorted according to their Hamming distance to obtain the minimum Hamming distance.
- (2) The threshold is set as $d = n \times d_{\min}$, and all feature point matching points are traversed. The Hamming distance of each feature point is compared with the threshold d . The feature matching point pairs with a Hamming distance less than d are returned and saved as sub-optimal matching points. Here, the value of n needs to be adjusted according to different scenes. In this paper, by establishing the threshold library of scenes, the algorithm automatically loads the adaptive threshold parameters through scene analysis to realize the effective compatibility of the algorithm in multiple scenes.

After the screening of the Hamming distance, the screening effect is often not good, and there will be some obviously wrong matching points. Through a large number of experiments, it is found that the wrong matching lines often have a large deviation angle from most of the correct matching lines. This paper innovatively proposes that the matching line with the shortest Hamming distance is taken as the benchmark to calculate the angle between other matching lines and the reference line and filters the wrong matching point pairs by setting ang_{\max} . The correct matching condition is set by Equation (17).

$$(ang < ang_{\max} \ \& \ \& \ dit < d) \tag{17}$$

Through repeated experimental verification, the robustness of the algorithm is greatly improved without affecting the

TABLE 2 The output calculated by the visual navigation module.

<i>Tx</i>	<i>Ty</i>	<i>Rot</i>	<i>Sx</i>	<i>Sy</i>
-1.66501	-13.6392	0.234247	0.945235	0.950384

Tx and *Ty* are measured in px, and others have no units of measurement.

performance of the algorithm, which lays a theoretical foundation for the engineering application of the algorithm. The angle between the matching line l_2 and the reference line l_1 is deduced as follows:

The equation of the line l_1 is expressed by Equation (18).

$$a_1x + b_1y + c_1 = 0 \tag{18}$$

The equation of the line l_2 is expressed by Equation (19).

$$a_2x + b_2y + c_2 = 0 \tag{19}$$

Then the intersection point of l_1 and l_2 is calculated by Equation (20).

$$\begin{cases} a_1x + b_1y + c_1 = 0 \\ a_2x + b_2y + c_2 = 0 \end{cases} \Rightarrow \begin{cases} x = \frac{b_1c_2 - b_2c_1}{a_1b_2 - a_2b_1} \\ y = \frac{a_1c_2 - a_2c_1}{a_2b_1 - a_1b_2} \end{cases} \tag{20}$$

The intersection point $p_0(x_0, y_0)$, the end point $p_2(x_2, y_2)$ of l_1 , and the end point $p_4(x_4, y_4)$ of l_2 form a triangle, then the length of each side is expressed by Equation (21), Equation (22) and Equation (23).

$$a = \sqrt{(x_4 - x_2)^2 + (y_4 - y_2)^2} \tag{21}$$

$$b = \sqrt{(x_4 - x_0)^2 + (y_4 - y_0)^2} \tag{22}$$

$$c = \sqrt{(x_2 - x_0)^2 + (y_2 - y_0)^2} \tag{23}$$

The angle θ is derived from the law of cosines, as shown in Equation (24).

$$\begin{aligned} a^2 = b^2 + c^2 - 2bc \cos \theta \Rightarrow \cos \theta &= \frac{b^2 + c^2 - a^2}{2bc} \Rightarrow \theta \\ &= \arccos \left(\frac{b^2 + c^2 - a^2}{2bc} \right) \end{aligned} \tag{24}$$

UAV position and attitude modification

The image matching and positioning are realized, and the deviation of image features needs to be further calculated to form data connection with the flight control



FIGURE 11
Images taken when the UAV panned and climbed.

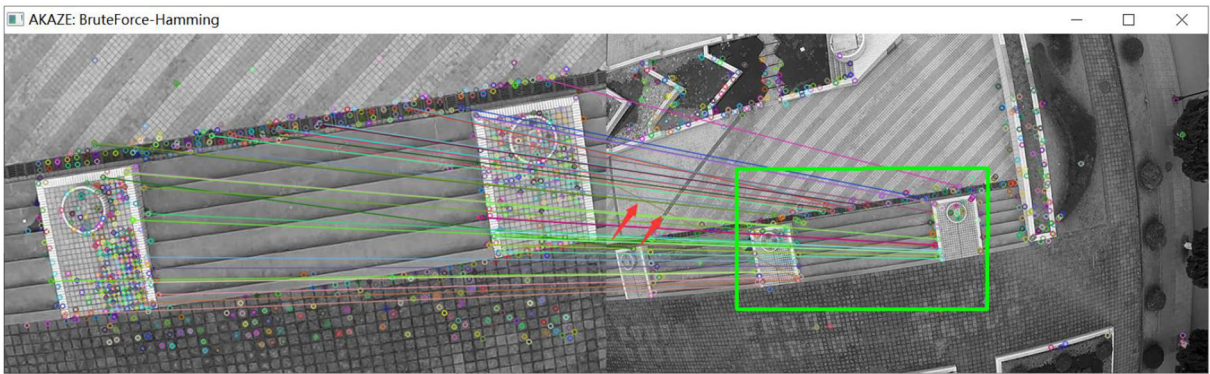


FIGURE 12
Matching effect of original algorithm.

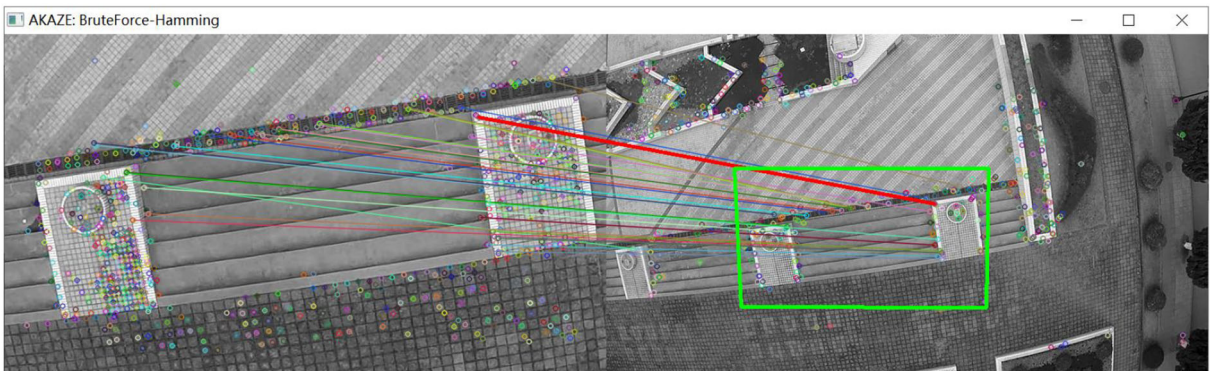


FIGURE 13
Matching effect of the optimized algorithm.

computer. Figure 4 shows the schematic diagram of the high-precision visual positioning system in this paper. The real-time image collected by the UAV and the set benchmark image are used as the input end of the visual navigation

module. After the image matching and positioning of the visual navigation system, the translation, rotation, and scaling of the real-time image relative to the benchmark image are output. Taking the amount of translation,

TABLE 3 The output calculated by the visual navigation module.

<i>Tx</i>	<i>Ty</i>	<i>Rot</i>	<i>Sx</i>	<i>Sy</i>
-62.5681	45.6894	0.00135039	0.422023	0.409761

Tx and *Ty* are measured in px, and the others have no units of measurement.

rotation, and scaling as the input parameters of the flight controller, the deviation between the real-time image and the reference image can be reduced by modifying the horizontal position, heading, and altitude of the UAV, so as to realize the closed-loop control of the UAV visual positioning and orientation.

As shown in Figure 5, is the deviation diagram between the real-time image and the reference image. *Tx* is the X-axis offset of UAV in the reference image coordinate system; *Ty* is the Y-axis offset of UAV in the reference image coordinate system; *Tx* and *Ty* are input quantities for UAV plane position correction. *Rot* is the rotation angle of the real-time image relative to the reference image, and *Rot* is the input of the UAV heading correction. *Sx* and *Sy* are the scaling ratio of the real-time image relative to the benchmark image, and *Sx* and *Sy* are the input quantities for UAV height correction. By reducing the offset of image features, the UAV 3D pose correction can be realized, and high-precision visual positioning and orientation can be realized. *Tx* and *Ty* can be calculated by Equation (25).

$$(Tx, Ty) = O_1(x, y) - O(x, y) \tag{25}$$

From the formula of the angle between vectors, it can be known that, as shown in Equation (26).

$$\cos Rot = \frac{\vec{OX} \times \vec{O_1X_1}}{|\vec{OX} \times \vec{O_1X_1}|} \tag{26}$$

Then, the rotation angle of the real-time image relative to the reference image can be calculated by Equation (27).

$$Rot = \arccos \frac{\vec{OX} \times \vec{O_1X_1}}{|\vec{OX} \times \vec{O_1X_1}|} \tag{27}$$

Sx is the scaling quantity of the real-time image relative to the reference image in the width direction, and *Sy* is the scaling quantity of the real-time image relative to the reference image in the length direction. The calculation formula is expressed by Equation (28).

$$Sx = W_1/W, Sy = H_1/H \tag{28}$$

System algorithm flow

This system uses the improved AKAZE image matching algorithm for the precise hovering of UAV operation. The algorithm flow of the system is as follows:

- (1) Set the reference image of UAV operation;
- (2) The key points of the benchmark image and real-time image are detected respectively;
- (3) M-LDB descriptors of key points of two images are calculated;
- (4) BruteForce-Hamming is used to match descriptor vectors;
- (5) The best matching points are selected based on the fusion of Hamming distance and matching line angle;
- (6) The RANSAC algorithm is used to calculate the homography matrix between two images;
- (7) The feature deviation and position information of the real-time image relative to the reference image are calculated;
- (8) The rotation, translation, and scaling of the real-time image relative to the reference image are calculated and input to the flight controller for UAV pose correction.

Above is the algorithm flow of this experiment, and the algorithm flow chart is shown in Figure 6.

Algorithm embedded device verification

As shown in Figure 7, the JETSON NANO B01 embedded computer serves as the verification platform for algorithm implementation (Kalms et al., 2017; Bao et al., 2022; Edel and Kapustin, 2022; Zhang et al., 2022). The UAV is equipped with a vision computer, and the camera is used as the image acquisition terminal to calculate the features of the real-time image and the benchmark image. The real-time image is located in the benchmark image by feature matching, and the translation, rotation, and scaling of the real-time image are calculated and input into the flight controller as compensation parameters of flight control, so as to realize the pose correction of the UAV. Based on the OpenCV 4.5.5 computer vision library, this paper applies the Qt 5.14.2 software development platform to develop visual algorithm software.

As shown in Table 1, the performance parameters of the JETSON NANO B01 are embedded into the computer for algorithm verification.

In this paper, a large number of algorithms have been verified for various scenes such as ordinary road surface and grass, and the image matching localization method has been fully verified in the process of translation, rotation, and climbing of the UAV. The input image size of the visual navigation module is 640×480,



FIGURE 14
Images taken while the UAV is rotating.

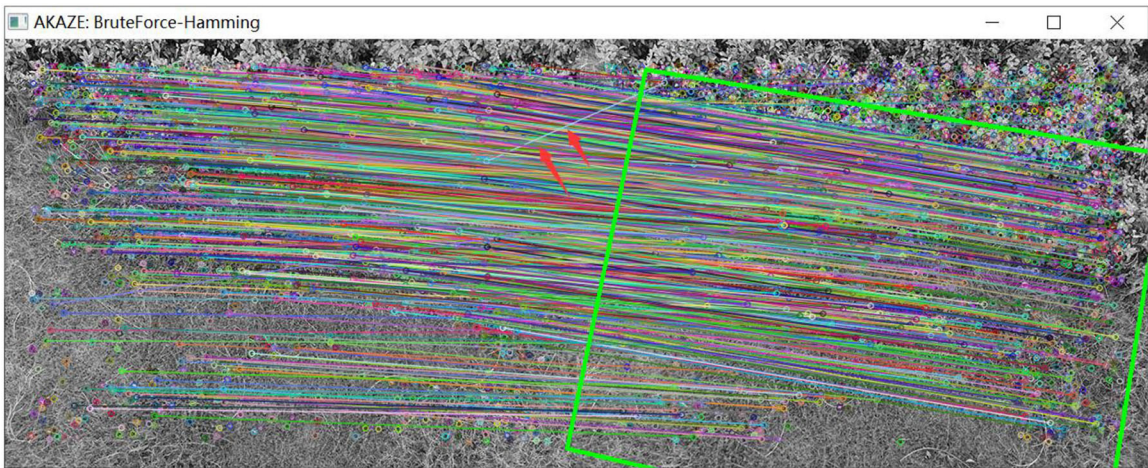


FIGURE 15
Matching effect of original algorithm.

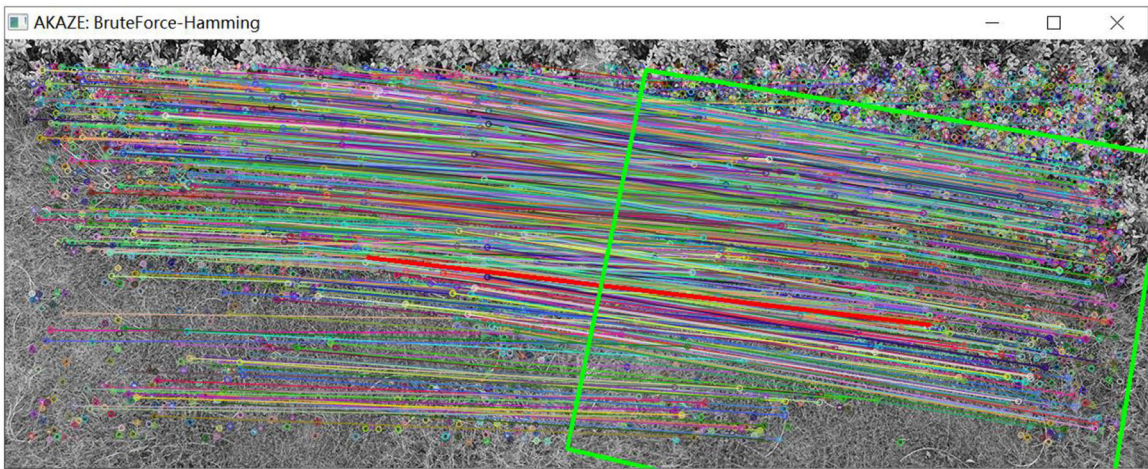


FIGURE 16
Matching effect of the optimized algorithm.

which can realize real-time image processing (Jiang et al., 2015; Soleimani et al., 2021) and ensure the real-time demand of the flight control response cycle.

Scenario 1: Common pavement scenario verification

(1) The UAV rotates, that is, the UAV heading changes.

As shown in Figure 8, a set of images is captured by the camera when the UAV rotates. As shown in Figure 9, the matching and positioning results of the original AKAZE algorithm are shown. It is obvious that some wrong matching points with too large an angle deviation are seen in the figure. In this paper, an innovative method based on Hamming distance and matching line angle fusion is proposed to screen the best matching points. As shown in Figure 10, in order to obtain matching effect, the improved algorithm greatly improves the robustness of the algorithm without affecting the performance of the algorithm. The bold red line in Figure 10 is the matching line with the minimum Hamming distance, which is used as the reference line. A matching point that pairs with a too large deviation angle is eliminated through the algorithm to obtain more accurate matching results.

TABLE 4 The output calculated by the visual navigation module.

T_x	T_y	Rot	S_x	S_y
10.4906	75.4823	0.162642	0.910785	0.883817

T_x and T_y are measured in px, and the others have no units of measurement.

As shown in Table 2, are the translation (T_x , T_y), rotation (Rot), and scaling (S_x , S_y) of the two groups of images in Figure 8 calculated by the visual navigation module.

(2) The UAV panned and climbed.

As shown in Figure 11, a group of pictures is collected by the UAV under the condition of translation and height climbing. As shown in Figure 12, through the matching point results of the original AKAZE method, the matching points with too large a deviation angle can also be obviously found, which further verifies the necessity of improving the direction of the algorithm in this paper. By fusing the angle deviation condition, the matching result as shown in Figure 13 is obtained, and the wrong matching points are eliminated effectively. The premise of UAV visual guidance is to obtain more accurate matching results. The algorithm in this paper has been repeatedly verified and has strong practicability and effectiveness, which greatly improves the robustness of the algorithm. The bold red line in Figure 13 is the matching line with the minimum Hamming distance, which is used as the base line for angle elimination condition.

As shown in Table 3, the translation (T_x , T_y), rotation (Rot), and scaling (S_x , S_y) of the two groups of images in Figure 11 are calculated by the visual navigation module.

The algorithm is verified by a large number of ground scenes. The above are typical processing effects of several groups of algorithms, which can achieve good image matching effects under the conditions of UAV rotation, translation, and height climbing, indicating that the improved algorithm can achieve better application effects. By calculating the deviation between the images, the navigation parameters required by the UAV can be further calculated. When used as the motion compensation parameters of the UAV, it can effectively assist the UAV to complete the visual navigation and achieve accurate fixed-point hovering.

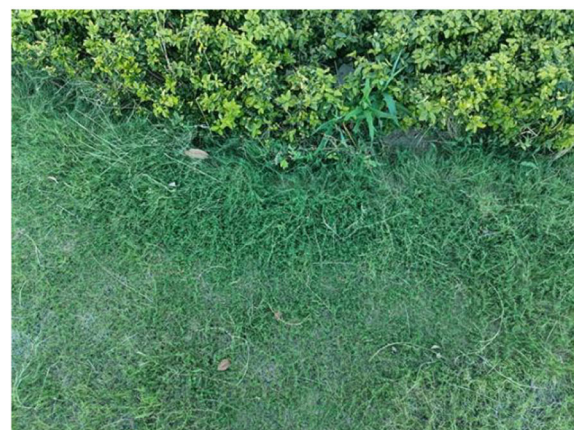
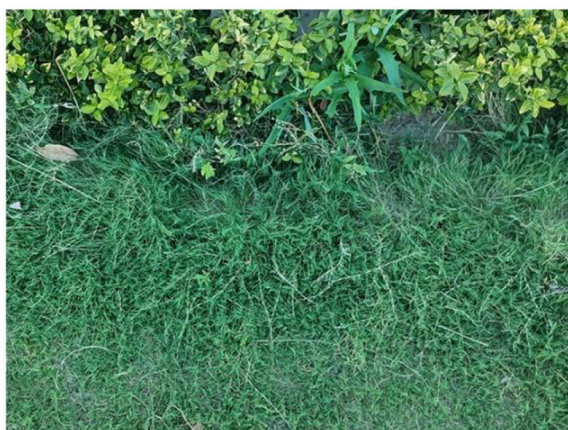


FIGURE 17
Images taken when the UAV panned and climbed.

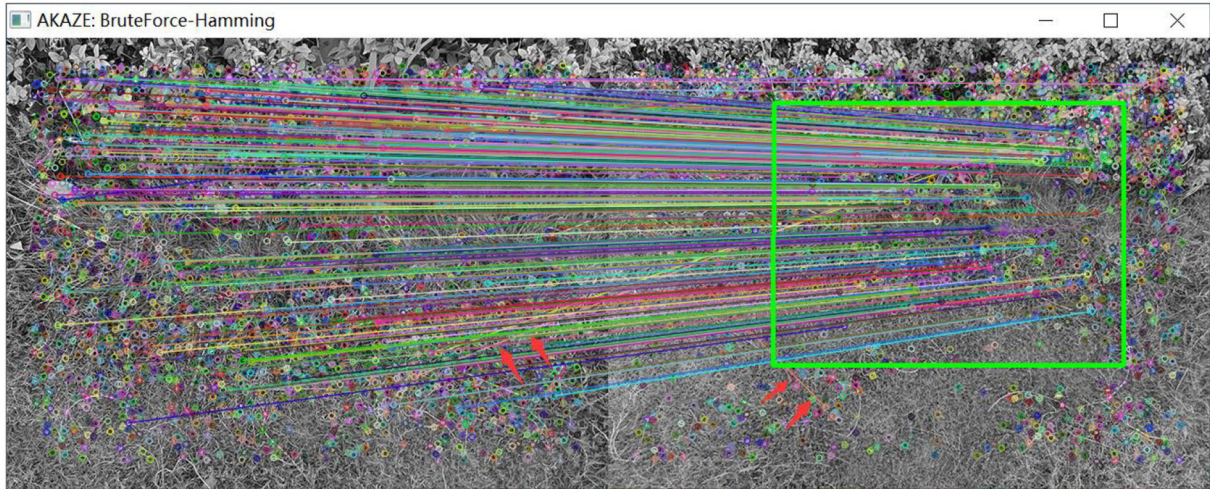


FIGURE 18
Matching effect of original algorithm.

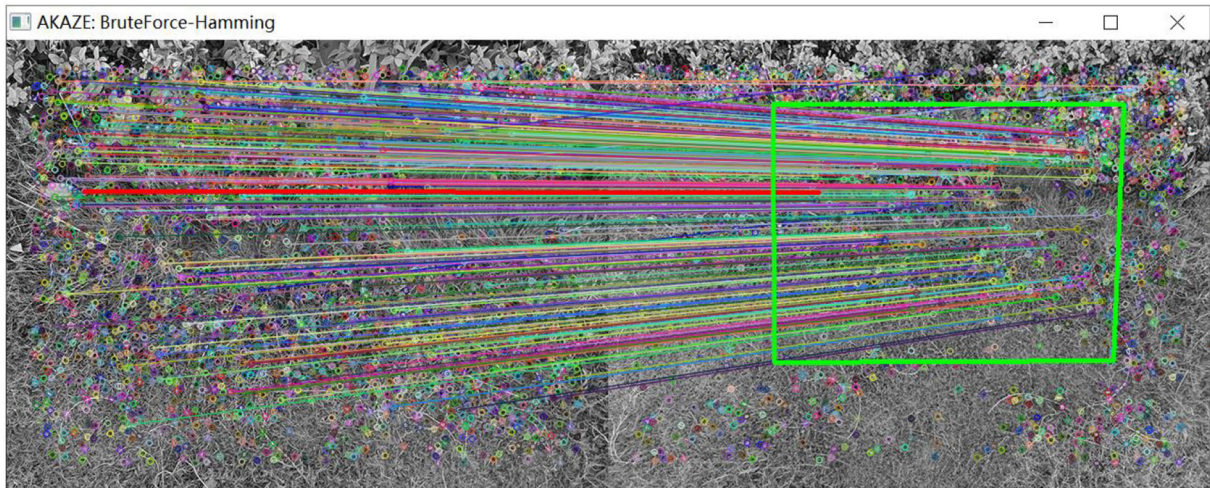


FIGURE 19
Matching effect of the optimized algorithm.

Scenario 2: Grassland scenario verification

(1) The UAV rotates, that is, the UAV heading changes.

As shown in Figure 14, a set of images is captured by the camera when the UAV rotates. By detecting the features of the two images, matching of the two images is further realized. As shown in Figure 15, the matching results of the original AKAZE algorithm can obviously find some wrong matching points with too large a deviation angle, which will affect the accuracy of the subsequent calculation of motion parameters. In order to solve this problem, this

paper proposes the best matching point screening method based on Hamming distance and matching line angle fusion, which achieves very good results. The matching effect of the improved algorithm is shown in Figure 16, where the bold red line is the matching line with the shortest Hamming distance, which is regarded as the base line. The matching point pairs that deviate too much from the baseline line are often wrong.

As shown in Table 4, the translation (T_x , T_y), rotation (Rot) and scaling, (S_x , S_y) of the two groups of images in Figure 14 are calculated by the visual navigation module.

(2) The UAV panned and climbed.

TABLE 5 The output calculated by the visual navigation module.

<i>T_x</i>	<i>T_y</i>	<i>Rot</i>	<i>S_x</i>	<i>S_y</i>
36.2779	−35.5712	≤0.001	0.583277	0.572635

T_x and *T_y* are measured in px, and the others have no units of measurement.

As shown in Figure 17, a group of pictures is collected by the UAV under the condition of translation and altitude climbing. As shown in Figure 18, matching results are obtained by image feature matching for the original AKAZE algorithm. The red arrow in the figure indicates that some wrong matching points with too large a deviation angle are excluded. In view of these shortcomings of the original algorithm, the matching effect as shown in Figure 19 is obtained following improvements outlined in this paper, which can effectively eliminate these obvious wrong matching points and greatly improve the robustness of the algorithm, thus laying a foundation for UAV visual guidance. In this way, more accurate matching results can be obtained, and the amount of feature deviation can be calculated more accurately, which makes the guidance of UAV hovering more accurate. In Figure 19, the bold red line is the matching line with the shortest Hamming distance, which is the benchmark reference line innovatively proposed in this paper and provides a reference for eliminating erroneous matching points.

As shown in Table 5, the translation (*T_x*, *T_y*), rotation (*Rot*), and scaling (*S_x*, *S_y*) of the two groups of images in Figure 17 are calculated by the visual navigation module.

Through the repeated test of the grassland scene, good experimental results are also obtained, which fully verifies that the proposed algorithm can achieve good image matching effects in multi-scenes and is suitable for multi-scene UAV visual guidance. This paper has achieved good results in the direction of improving the robustness of the AKAZE algorithm, and innovatively proposed the best matching point screening method based on Hamming distance and matching line angle fusion. A large number of experiments have been carried out on ordinary ground, grassland, and other scenes, which fully verified the feasibility and effectiveness of the algorithm.

Conclusion

Aiming at the high-precision hovering problem of UAV operation, this paper proposes a high-precision visual positioning and orientation method of UAVs based on image feature matching. The image feature matching based on the improved AKAZE algorithm is realized, and the optimal matching point pair screening method based on the fusion of Hamming distance and matching line angle is innovatively proposed, which greatly improves the robustness of the algorithm without affecting the performance of the algorithm.

Aiming at the diversity and complexity of the UAV operating environment, the application of image scale invariant features for UAV visual navigation is proposed. By setting the reference image of UAV operation, the real-time image is matched with the reference image for feature matching, and the translation, rotation angle, and scaling ratio of the image are calculated as the input parameters of the aircraft pose control, so as to reduce the image feature deviation and achieve the UAV pose state correction. A lot of tests and verification are carried out. Finally, the multi-scene UAV flight environment verification is carried out, and good application results are obtained, which shows that the algorithm has high engineering application value.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

BX and ZY contributed to conception and design of the study and performed the statistical analysis. BX organized the database and wrote the first draft of the manuscript. BX, LL, CZ, HX, and QZ wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

Funding

This research was funded by the Guizhou Provincial Science and Technology Projects under Grant Guizhou-Sci-Co-Supp[2020]2Y044 and in part by the Science and Technology Projects of China Southern Power Grid Co. Ltd. under Grant 066600KK52170074.

Conflict of interest

Author BX is employed by Nanjing Taiside Intelligent Technology Co., Ltd. Author QZ is employed by Electric Power Research Institute of Guizhou Power Grid Co., Ltd.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated

organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or

claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Bao, X., Li, L., Ou, W., and Zhou, L. (2022). Robot intelligent grasping experimental platform combining Jetson NANO and machine vision. *J. Phys. Conf. Ser.* 2303, 012053. doi: 10.1088/1742-6596/2303/1/012053
- Bellavia, F. (2022). SIFT matching by context exposed. *IEEE Trans. Pattern Anal. Mach. Intell.* 2022, 1. doi: 10.1109/TPAMI.2022.3161853
- Cao, M., Tang, F., Ji, P., and Ma, F. (2022). Improved real-time semantic segmentation network model for crop vision navigation line detection. *Front. Plant Sci.* 13, 898131. doi: 10.3389/fpls.2022.898131
- Chen, J., Xi, Z., Lu, J., and Ji, J. (2022). Multi-object tracking based on network flow model and ORB feature. *Appl. Intell.* 52, 12282–12300. doi: 10.1007/s10489-021-03042-6
- Edel, G., and Kapustin, V. (2022). Exploring of the MobileNet V1 and MobileNet V2 models on NVIDIA Jetson Nano microcomputer. *J. Phys. Conf. Ser.* 2291, 012008. doi: 10.1088/1742-6596/2291/1/012008
- Fatma, M., El-Ghamry, W., Abdalla, M. I., Ghada, M., and Algarni, A. D., Dessouky, M. I. (2022). Gauss gradient and SURF features for landmine detection from GPR images. *Comput. Mater. Continua* 71, 22328. doi: 10.32604/cmc.2022.022328
- Feng, Q., Tao, S., Liu, C., Qu, H., and Xu, W. (2021). IFRAD: a fast feature descriptor for remote sensing images. *Remote Sens.* 13, 3774. doi: 10.3390/rs13183774
- Feng, W., and Chen, Y. (2017). Speckle reduction with trained nonlinear diffusion filtering. *J. Math. Imag. Vis.* 58, 162–178. doi: 10.1007/s10851-016-0697-x
- Guo, H., Wang, Y., Yu, G., Li, X., Yu, B., and Li, W. (2022). Robot target localization and visual navigation based on neural network. *J. Sens.* 202, 6761879. doi: 10.1155/2022/6761879
- Ji, X., Yang, H., Han, C., Xu, J., and Wang, Y. (2021). Augmented reality registration algorithm based on T-AKAZE features. *Appl. Opt.* 60, 10901–10913. doi: 10.1364/AO.440738
- Jiang, G., Liu, L., Zhu, W., Yin, S., and Wei, S. (2015). A 181 GOPS AKAZE accelerator employing discrete-time cellular neural networks for real-time feature extraction. *Sensors* 15, 22509–22529. doi: 10.3390/s150922509
- Jubairahmed, L., Satheeskumaran, S., and Venkatesan, C. (2019). Contourlet transform based adaptive nonlinear diffusion filtering for speckle noise removal in ultrasound images. *Clust. Comput.* 22, 11237–11246. doi: 10.1007/s10586-017-1370-x
- Kalms, L., Mohamed, K., and Göhringer, D. (2017). “Accelerated embedded AKAZE feature detection algorithm on FPGA,” in *Highly Efficient Accelerators and Reconfigurable Technologies*. New York, NY: Association for Computing Machinery. doi: 10.1145/3120895.3120898
- Khalid, N. A. A., Ahmad, M. I., Mandeel, T. H., and Isa, M. N. M. (2021). Palmprint features matching based on KAZE feature detection. *J. Phys. Conf. Ser.* 1878, 012055. doi: 10.1088/1742-6596/1878/1/012055
- Li, Y., Ding, Y., and Li, T. (2016). Nonlinear diffusion filtering for peak-preserving smoothing of a spectrum signal. *Chemomet. Intell. Lab. Syst.* 156, 157–165. doi: 10.1016/j.chemolab.2016.06.007
- Liu, J.-B., Bao, Y., Zheng, W. T., and Hayat, S. (2021). Network coherence analysis on a family of nested weighted n-polygon networks. *Fractals*. 29, 2150260. doi: 10.1142/S0218348X21502601
- Liu, J.-B., Zhao, J., He, H., and Shao, Z. (2019b). Valency-based topological descriptors and structural property of the generalized sierpinski networks. *J. Stat. Phys.* 177, 1131–1147. doi: 10.1007/s10955-019-02412-2
- Liu, J.-B., Zhao, J., Min, J., and Cao, J. D. (2019a). On the Hosoya index of graphs formed by a fractal graph. *Fract. Comp. Geometry Patterns Scal. Nat. Soc.* 27, 1950135. doi: 10.1142/S0218348X19501354
- Liu, Y., Chen, Y., Chen, P., Qiao, Z., and Gui, Z. (2019c). Artifact suppressed nonlinear diffusion filtering for low-dose CT image processing. *IEEE Access*. 7, 109856–109869. doi: 10.1109/ACCESS.2019.2933541
- Liu, Y., Tian, J., Hu, R., Yang, B., Liu, S., Yin, L., et al. (2022). Improved feature point pair purification algorithm based on SIFT during endoscope image stitching. *Front. Neurobot.* 16, 840594. doi: 10.3389/fnbot.2022.840594
- Liu, Z., Huang, D., Qin, N., Zhang, Y., and Ni, S. (2021). An improved subpixel-level registration method for image-based fault diagnosis of train bodies using SURF features. *Measur. Sci. Technol.* 32, 115402. doi: 10.1088/1361-6501/ac07d8
- Niyishaka, P., and Bhagvati, C. (2020). Copy-move forgery detection using image blobs and BRISK feature. *Multimedia Tools Appl.* 79, 26045–26059. doi: 10.1007/s11042-020-09225-6
- Pei, L., Zhang, H., and Yang, B. (2021). Improved Camshift object tracking algorithm in occluded scenes based on AKAZE and Kalman. *Multimedia Tools Appl.* 81, 2145–2159. doi: 10.1007/s11042-021-11673-7
- Qin, J., Li, M., Li, D., Zhong, J., and Yang, K. (2022). A survey on visual navigation and positioning for autonomous UUVs. *Remote Sens.* 14, 3794. doi: 10.3390/rs14153794
- Roy, M., Thounaojam, D. M., and Pal, S. (2022). Perceptual hashing scheme based on improved BRISK. *Int. J. Wireless Mobile Comput. Multimed. Tools Appl.* 81, 29045–29073. doi: 10.1007/s11042-022-12626-4
- Sharma, S. K., and Jain, K. (2020). Image stitching using AKAZE features. *J. Indian Soc. Remote Sens.* 48, 1389–1401. doi: 10.1007/s12524-020-01163-y
- Shi, Q., Liu, Y., and Xu, Y. (2021). Image feature point matching method based on improved BRISK. *Int. J. Wireless Mobile Comput.* 20, 132–138. doi: 10.1504/IJWMC.2021.114129
- Singh, N. P., and Singh, V. P. (2020). Efficient segmentation and registration of retinal image using gumble probability distribution and BRISK feature. *Traitement du Signal* 37, 370519. doi: 10.18280/ts.370519
- Soleimani, P., Capson, D. W., and Li, K. F. (2021). Real-time FPGA-based implementation of the AKAZE algorithm with nonlinear scale space generation using image partitioning. *J. Real-Time Image Process.* 18, 2123–2134. doi: 10.1007/s11554-021-01089-9
- Wang, Y., Chen, Z., and Li, Z. (2019). Fixed-point hovering method for quadrotor based on Harris algorithm. *Elect. Technol. Softw. Eng.* 2019, 82–83. Available online at: <https://kns.cnki.net/kcms/detail/10.1108.TP.20190115.1535.136.html>
- Xie, Y., Wang, Q., Chang, Y., and Zhang, X. (2022). Fast target recognition based on improved ORB feature. *Appl. Sci.* 12, 786. doi: 10.3390/app12020786
- Xue, P., Wang, C., Huang, W., and Zhou, G. (2022). ORB features and isophotes curvature information for eye center accurate localization. *Int. J. Pattern Recognit. Artif. Intell.* 36, 2256005. doi: 10.1142/S0218001422560055
- Yan, Q., Li, Q., and Zhang, T. (2021). “Research on UAV image mosaic based on improved AKAZE feature and VFC algorithm,” in *Proceedings of 2021 6th International Conference on Multimedia and Image Processing (ICMIP 2021)* (New York, NY: ACM), 52–57. doi: 10.1145/3449388.3449403
- Yang, L., Huang, Q., Li, X., and Yuan, Y. (2022). Dynamic-scale grid structure with weighted-scoring strategy for fast feature matching. *Appl. Intell.* 2022, 1–5. doi: 10.1007/s10489-021-02990-3
- Yu, X., He, Y., and Liu, S. (2021). Indoor fixed-point hovering design of UAV based on pyramid optical flow method. *Autom. Technol. Appl.* 2021, 16–28. Available online at: <https://kns.cnki.net/kcms/detail/detail.aspx?dbcode=CJFD&dbname=CJFDLAST2021&filename=ZDHJ202102004&uniplatform=NZKPT&v=phO58Qgp7cD6E79rjWBRumOQ41VBGRFBXn0llbnwGKgTQCQBF8ks3ugM0ciY9bufz>
- Zhang, S., and Lang, Z.-Q. (2022). Orthogonal least squares based fast feature selection for linear classification. *Pattern Recogn.* 123, 108419. doi: 10.1016/j.patcog.2021.108419
- Zhang, W., Zhang, W., Song, F., and Zhao, Y. (2018). Fixed-point hovering control of small quadrotor UAV based on optical flow and ultrasonic module. *Chem. Eng. Autom. Instrum.* 2018, 289–293. Available online at: <https://kns.cnki.net/kcms/detail/detail.aspx?dbcode=CJFD&dbname=CJFDLAST2018&filename=HGZD201804009&uniplatform=NZKPT&v=rGxwyQw4ETU0jZeAL1-OpkWxV9a9e7ju8TEKOcdJ9OByxE7e26bu0cY1AvLwA84>
- Zhang, Z. D., Tan, M. L., Lan, Z. C., Liu, H. C., Pei, L., and Yu, W. X. (2022). CDNet: a real-time and robust crosswalk detection network on Jetson nano based on YOLOv5. *Neural Comput. Appl.* 2022, 1–2. doi: 10.1007/s00521-022-07007-9
- Zhao, Y., Gupta, R. K., and Onyema, E. M. (2022). Robot visual navigation estimation and target localization based on neural network. *Paladyn J. Behav. Robot.* 13, 76–83. doi: 10.1515/pjbr-2022-0005