Research on visual navigation based on remote sensing image

1st RunPing Xi
school of computing
N*orthwestern Polytechnical university*
Xian, China
xrp@163.com

2nd Yue Liu
school of computing
N*orthwestern Polytechnical university*
Xian, China
1508334860@qq.com

3rd Nan Li
school of computing
N*orthwestern Polytechnical university*
Xian, China
2897557416@qq.com

4th XueFeng Kou
Shaanxi Baocheng Aviation Instrument Co. LTD
Xian, China
kouxuefeng111@163.com

***Abstract*—At present, the positioning and navigation system covers global positioning and navigation system (GPS) and inertial navigation system, but GPS navigation and inertial navigation have their own shortcomings. Visual navigation can perceive and provide rich dynamic environment information, and can realize accurate and reliable positioning and navigation in more complex environments. In this paper, a remote sensing image feature matching algorithm suitable for image navigation and positioning is constructed. Aiming at the problems of difficult and time-consuming matching of remote sensing images obtained by aircraft sensors, the key points of fast feature are extracted, the main and secondary features are selected according to the standard, the direction intensity histogram is calculated by using the relative distance, azimuth and relationship intensity, and the descriptor is constructed through the calculation results. Cosine similarity is used as the similarity measure of directional intensity histogram to match the features in the two images to achieve feature matching. In the filtering error matching stage, the error matching on the hypothesis set is removed by calculating the good matching with high embedding ratio, and the matching result with high accuracy is obtained. Based on this algorithm, a navigation prototype system based on remote sensing image is designed and implemented to provide a visualization platform.**

***Keywords*—Remote sensing image, feature matching, descriptor, guidance strategy, navigation system**

I. INTRODUCTION (*HEADING 1*)

Feature point matching or indirect matching has been used as an effective alternative to template matching. The process includes two tasks: feature point detection and descriptor extraction. Feature point detection usually involves corner detectors, such as the famous Harris[1]and fast [2] detectors. The goal of feature point detection is to carry out two completely independent detection iterations on different images in the same region, and find significant points that may have great differences in illumination, proportion, rotation and viewpoint. Description extraction is the step of extracting feature vectors from the region around feature points.

In 2016, seemaetal. [3] and saranyaetal. [4] were very similar in their work. Their purpose was to compare the characteristics [5] of NCC with random sample consistency (RANSAC) [6] combined with surf characteristics [7], so as to match the aerial image with the reference image. When using NCC for template matching, firstly, the edge of aerial image and reference map image is detected. For each position in the reference map, the aerial image is used as the template, and the maximum response is used as the position of the UAV image in the reference map. In the feature point method, surf features are extracted and matched from UAV image and reference map respectively. RANSAC is used to remove outliers in matching. Then, by calculating the geometric transformation between the remaining feature points and the matching points, the position of the UAV image in the reference map is restored. Very limited simulation of images from Google maps. In the experiment, a reference map is arbitrarily selected, and a small image of the reference image is added with different noise changes, scale changes and rotation to simulate the aerial image as the image to be matched. The results show that the simulated aerial image extracted from the reference map has high positioning accuracy. The experimental results show that the execution time of NCC is better than that of RANSAC method. However, NCC algorithm is more sensitive than RANSAC under the conditions of scale, rotation, noise and ambiguity. The author's conclusion is that NCC algorithm is more suitable for filtering under the known conditions. At the same time, two disadvantages of the combination of RANSAC and surf are pointed out: the requirement for the minimum number of features and the non constant number of feature detection and extraction.

In 2019, mantellietal. [8] designed a 4-DOF absolute positioning system using satellite images. The system uses a down looking monocular camera, and its roll angle and pitch angle are considered to be close to zero. Match the UAV image with the satellite map. Another difference is the use of

quantization levels rather than Gaussian blur to reduce noise. This prevents pixel averaging and ensures that pixel differences are conserved. The particle filter method is used to estimate the space position of the aircraft. Motion update is based on the estimation provided by VO system. However, since the motion model is not the focus of this work, the VO solution is very basic and inaccurate. For example, the system does not consider the error model, and the author applies random Gaussian noise to the measurement model. The solution tested the changes of 3 tracks and 5 reference maps in a large environment. The longest trajectory is 2.4 km and the positioning is based on 1.1 km × 1.1 km map. The results reported show that the average error on the longest trajectory is 17.78 meters.

The algorithm based on feature matching has high positioning accuracy, rich texture information and more feature points of remote sensing image, which is suitable for positioning and navigation research. Therefore, this paper uses the feature matching algorithm as the core algorithm of positioning and navigation, trying to find a method to solve the difficulty of remote sensing image matching, We find a simultaneous interpreting method to solve the problem that the spatial reference of different sensors is inconsistent and that the altitude of the aircraft is not uniform.

This paper aims at the problems of difficult and time-consuming matching of remote sensing images obtained by aircraft sensors. Aiming at the features extracted by fast, the feature descriptor is improved, and a special feature similarity evaluator is designed to match the features in the two images to achieve fast and reliable feature description. In the filtering error matching stage, the error matching on the hypothesis set is removed by calculating the good matching with high embedding ratio, and the matching result with high accuracy is obtained.On this basis, the navigation system based on remote sensing image is designed and developed

II. IMAGE REGISTRATION METHOD

A. Feature extraction

Before obtaining fast features, Gaussian smoothing is used to reduce the influence of noise. When detecting features in the reference image $R(x,y)$, the fast detector will return a set of features $F^{(R)}=\{f_i^{(R)} \mid i\in[1...N_f]\}$, their positions $Loc(f_i^{(R)})$ and response amplitudes $Mag(f_i^{(R)})$, where $F^{(R)}$ represents a set of detection features in R and $f_i^{(R)}$ represents the second feature. Since the descriptors constructed in this chapter describe the features according to the spatial relationship, it is more likely to correctly describe the features close to the center of the image, because they can be described according to other features from all directions, which is impossible for the features close to the edge or corner. Therefore, the response amplitude should be adjusted according to the distance between the feature and the image center, as shown in Eq.(1).

$$Mag_m(f_i^{(R)})=Mag(f_i^{(R)})\cdot\exp[-\frac{(x_i-M/2)^2+(y_i-N/2)^2}{2\cdot\min(M,N)}] \quad (1)$$

Wherein, $Mag_m(f_i^{(R)})$ represents the modulation response amplitude of $f_i^{(R)}$, M and N represents the width and height of the image R. Intuitively, the stability feature has a stronger response $Mag_m(f_i^{(R)})$. Under this assumption, the $f_i^{(R)}$ in $F^{(R)}$ are sorted in $Mag_m(f_i^{(R)})$ descending order, and then get a secondary feature set $F^{(R,SF)}$ can be obtained, which contains the strongest half of these features. This operation can be expressed by Eq. (2).

$$F^{(R,SF)}=\{f_i^{(R)} \mid f_i^{(R)}\in F^{(R)}, i\in[1...N_f/2]\} \quad (2)$$

However, in some cases, some patterns (such as corners or spots) may have an uncertain number of secondary features, which may vary due to the random distribution of different parallax or noise. This will cause errors in the feature matching process. In order to reduce this effect, the main features are further determined from these secondary features according to the following steps.

1. The initial $F^{(R,PF)}$ is the main feature set of null, i.e $F^{(R,PF)}=\varnothing$;

2. For each secondary feature $f_i^{(R,SF)}$, define the feature domain $D(f_i^{(R,SF)})$ with $(x_i,y_i)=Loc(f_i^{(R,SF)})$ as the center:

$$D(f_i^{(R,SF)})=\{(x,y) \mid (x-x_i)^2+(y-y_i)^2\le R^2\} \quad (3)$$

If there is no other secondary feature $f_j^{(R,SF)}$ in the $D(f_i^{(R,SF)})$, the response $Mag(f_j^{(R)})$ is stronger than $Mag(f_i^{(R)})$,then there is $f_i^{(R,SF)}\in F^{(R,SF)}$. If $f_i^{(R,SF)}\in F(R,PF)$, the standard can be expressed as Eq. (4).

$$\{f_j^{(R,SF)} \mid Loc(f_j^{(R,SF)}\in D(f_j^{(R,SF)}), Mag(f_j^{(R,SF)})>Mag(f_j^{(R,SF)}), i\ne j)\}=\varnothing \quad (4)$$

The overall standard is shown in Fig. 1. It is assumed that there are 14 secondary features in the image (represented by bold points a – n), and this number represents the response strength of each feature. Circles (only four are shown for clarity) represent the fields of the corresponding features. When determining the domain radius r, according to formula (4), features a, B, C, F, J, K, m and N are determined as the main features.

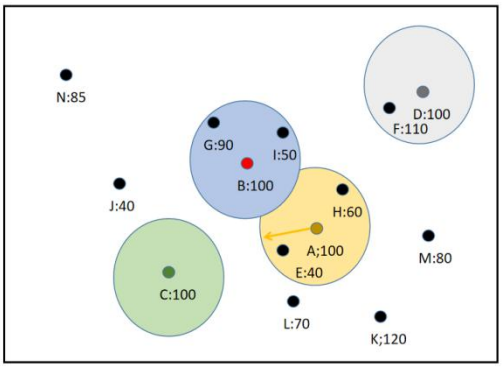


Fig. 1. Determination of main characteristics

The relationship between one feature and other features includes relative azimuth and distance (RAD). Suppose there is a primary feature $f_i^{(R,PF)}$ whose relative azimuth and distance relative to the secondary feature $f_j^{(R,SF)}$ can be obtained in the following ways:

$$Azim(f_j^{(R,SF)} \mid f_i^{(R,PF)}) = \arctan\frac{y_j - y_i}{x_j - x_i} + s\pi, s = \begin{cases} 0, x_j - x_i < 0 \\ 1, x_j - x_i \geq 0 \end{cases} \quad (5)$$

$$Dist(f_j^{(R,SF)} \mid f_i^{(R,PF)}) = (x_j - x_i)^2 + (y_j - y_i)^2 \quad (6)$$

Where, $(x_i, y_i) = Loc(f_i^{(R,PF)})$ and $(x_i, y_i) = Loc(f_j^{(R,SF)})$, in Eq. (5) $s\pi$ are used to distinguish $f_j^{(R,SF)}$ the quadrants relative to $f_i^{(R,PF)}$. When the image has a certain degree of distortion, the farther away the secondary features $f_i^{(R,PF)}$ change in spatial relationship. Therefore, the strength of this relationship is expressed by Eq.(7).

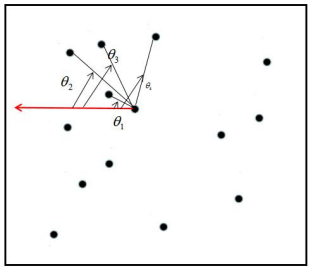
$$S(f_j^{(R,SF)} \mid f_i^{(R,PF)}) = \frac{1}{Dist(f_j^{(R,SF)} \mid f_i^{(R,PF)})} \quad (7)$$

B. Construct descriptor

In order to keep the rotation of the algorithm unchanged, the dominant direction of each main feature needs to be determined. For this purpose, calculate the distance of all secondary features relative to the primary features, collect them into a list, and then sort them according to the descending order of relationship strength (ascending order of relative distance), as shown in Fig. 2.

Then, the dominant direction of the main feature is the relative azimuth of its nearest secondary feature (which also has the strongest relationship).

$$Ori(f_i^{(R,PF)}) = Azim(f_k^{(R,SF)} \mid f_i^{(R,PF)}) \quad (8)$$



Secondary characteristics	Relative azimuth	Relationship strength
H	θ_1	$\frac{1}{\lvert AH\rvert}$
D	θ_2	$\frac{1}{\lvert AD\rvert}$
F	θ_3	$\frac{1}{\lvert AF\rvert}$
M	θ_4	$\frac{1}{\lvert AM\rvert}$
.....	.....	.....

Fig. 2. Calculation of azimuth, distance and relationship strength

When f the following conditions are met.

$$S(f_k^{(R,SF)} \mid f_i^{(R,PF)}) = \max_j[S(f_j^{(R,SF)} \mid f_i^{(R,PF)})] \quad (9)$$

On this basis, the directional intensity histogram can be obtained through the following steps:

1. Divide the image into n sector regions according to the direction of the feature, where n is an adjustable parameter;

2. The directional intensity is estimated by calculating the sum of all relationship intensities in each sector. This operation can be expressed as Eq. (10).

$$O_k(f_i^{(R,PF)} \mid F^{(R,SF)}) = \sum_j S(f_j^{(R,SF)} \mid f_i^{(R,PF)}) \cdot \Pi_k(f_j^{(R,SF)} \mid f_i^{(R,PF)}) \quad (10)$$

Where $\Pi_k(f_j^{(R,SF)} \mid f_i^{(R,PF)})$ is a piecewise function.

$$\Pi_k(f_j^{(R,SF)} \mid f_i^{(R,PF)}) = \begin{cases} 1, \frac{2(k-1)\pi}{n} \leq Azim_{pm}(f_j^{(R,SF)} \mid f_i^{(R,PF)}) < \frac{2k\pi}{n} \\ 0, other \end{cases} \quad (11)$$

Through the above steps, the directional intensity histogram of each main feature is formed and can be expressed as a vector, which will be used as a descriptor vector.

$$V(f_i^{(R,PF)} \mid F^{(R,SF)}) = [O_1(f_i^{(R,PF)} \mid F^{(R,SF)}), ..., O_n(f_i^{(R,PF)} \mid F^{(R,SF)})]^T \quad (12)$$

C. Feature matching

In the process of feature matching, appropriate similarity measurement is the key factor to improve the accuracy of matching. For gradient based feature descriptors, such as sift, surf, SSD or sad are often used as similarity measures. Another similarity measure, called Hamming distance, is also used for binary feature descriptors [9]. The rotation invariance of the descriptor used in this chapter is realized by remapping the relative azimuth to a fixed range, and the main direction is set to 0. In addition, the descriptor vector also has the potential of scale invariance, because the change of scale will change the distance between features simultaneously and proportionally, which means that the "shape" of the directional intensity histogram will not change. However, these measures cannot take advantage of this potential. Therefore, cosine similarity is used as the similarity measure of directional intensity histogram.

Suppose there are two similar features. One is in the reference image. For simplicity, its directional intensity histogram is recorded as $V_i^{(R)} = V(f_i^{(R,PF)} \mid F^{(R,SF)})$. Similarly, $V_j^{(R)} = V(f_j^{(S,PF)} \mid F^{(S,SF)})$ represents another feature in the image to be matched, the cosine distance of the two features (the smaller the better) is expressed as Eq. (13).

$$d_{\cos}(f_i^{(R,PF)}, f_j^{(S,PF)}) = 1 - \frac{V_i^{(R)} \cdot V_j^{(S)}}{\left\| V_i^{(R)} \right\| \cdot \left\| V_j^{(S)} \right\|} = 1 - \frac{\sum_{k=1}^n a_k b_k}{\sqrt{\sum_{k=1}^n a_k^2} \sqrt{\sum_{k=1}^n b_k^2}} \quad (13)$$

Among them $a_k = O_k(f_i^{(R,PF)} \mid F^{(R,SF)})$, $a_k = O_k(f_i^{(R,PF)} \mid F^{(R,SF)})$, using cosine similarity can improve scale invariance, and using cosine similarity can improve scale invariance.

When matching feature pairs, because different views describe the same feature differently, it is inevitable that some features will not match. This will produce a larger proportion

Identify applicable funding agency here. If none, delete this text box.

of outliers (mismatched feature pairs) to estimate view differences. In order to reduce the impact, it is necessary to limit the matching conditions. When $d_{\cos}(f_i^{(R,PF)}, f_j^{(S,PF)})$ is the smallest in all $d_{\cos}(\cdot, f_j^{(S,PF)})$ and, $d_{\cos}(f_i^{(R,PF)}, \cdot)$ the features $f_i^{(R,PF)}$ and $f_j^{(S,PF)}$ matching.

D. Filter error matching

Suppose that a set of N hypothetical feature correspondences are extracted from two given remote sensing images (for example, according to the similarity of the above feature descriptors), where the sum is a two-dimensional vector representing the spatial position of the feature points. Then, the goal is to eliminate the outliers in the hypothesis set, retain the correct matching points, and establish a good matching relationship. If the matching relationship is a simple transformation, the two will be retained, otherwise they will be eliminated and represented as a signed unknown embedded set,The optimal solution is Eq. (14).

$$\Omega^* = \arg(\min_{\Omega} C(\Omega; S; \lambda)) \tag{14}$$

The cost function C is defined as Eq.(15).

$$C(\Omega; S; \lambda) = \sum_{i \in \Omega} \sum_{j \in \Omega} (d(x_i, x_j) - d(y_i, y_j)^2 + \lambda(N - |\Omega|)) \tag{15}$$

Where the parameter d represents the distance between (x_i, y_i) and represents the cardinality of the set. In Eq. (15), the first value of the function can be used to eliminate matching inconsistencies, and the second value can be used to eliminate matching exceptions. Parameter λ is used as trade-off factors.

However, the ideal rigid transformation is usually not suitable for the actual remote sensing task. In this case, the description distance of the above formula is not suitable for the actual remote sensing image, especially the point matching with great matching difficulty. The neighborhood information between the feature key points extracted from the remote sensing image does not change greatly, and the general neighborhood structure remains similar in the nearby local area . Therefore, Eq. (15) is changed into Eq. (16) by retaining the local neighborhood structure.

$$C(\Omega, S; \lambda) = \sum_{i \in \Omega} (\sum_{j | x_j \in N_{x_i}} (d(x_i, x_j) - d(y_i, y_j)^2 + \sum_{j | y_j \in N_{y_i}} (d(x_i, x_j) - d(y_i, y_j)^2)) + \lambda(N - |\Omega|) \tag{16}$$

Where N_x represents the neighborhood of the point x. A simple strategy is adopted to search the K(default = 4) nearest neighbor of each point in the corresponding feature set under the Euclidean distance. Assume s that the set is associated with a $N \times 1$ binary vector q , which $q_i \in \{0,1\}$ represents whether the corresponding relationship (x_i, y_i) matches correctly or not. The q_i specific values represent different meanings. When the value is 0, the key points representing the matching relationship belong to embedded values. When the value is 1, the key points representing the matching relationship belong to outliers, and the value represents outliers. If the remote sensing image is deformed such as scale transformation, the matching distance between feature point pairs will change. Based on this, the distance is quantified into two levels.

$$d(x_i, y_i) = x \begin{cases} 0 & x_j \in N_{x_i} \\ 1 & x_j \notin N_{x_i} \end{cases} \tag{17}$$

$d(y_i, y_j)$ the same is true. In this case, the cost function is transformed into:

$$C(\Omega, S; \lambda) = \sum_{i=1}^{N} q_i (\sum_{j | x_j \in N_{x_i}} d(y_i, y_j) + \sum_{j | y_j \in N_{y_i}} d(x_i, x_j)) + \lambda(N - \sum_{i=1}^{N} q_i) \tag{18}$$

The task of eliminating matching exceptions and constructing correct matching relationship can be solved by searching the optimal solution to minimize the cost function Eq. (18).

In order to facilitate understanding, the mathematical expression of the cost function Eq.(18) can be more concise, and the Eq. (19) can be obtained by optimizing the common term.

$$C(q; S; \lambda) = \sum_{i=1}^{N} q_i (c_i - \lambda) + \lambda N \tag{19}$$

$$c_i = \sum_{j | x_j \in N_{x_i}} d(y_i, y_j) + \sum_{j | y_j \in N_{y_i}} d(x_i, x_j) \tag{20}$$

c_i is to estimate the constraint cost of whether the first matching relationship conforms to the local neighborhood structure or not. Obviously, the correct matching that conforms to the local neighborhood structure will consume less cost, on the contrary, the wrong matching that does not conform to the local neighborhood structure will consume a lot of cost.

Based on the invariable structure information on the known assumption set, this feature is convenient for early calculation, and saves the calculation time. If there is only unknown in the formula, any cost less than will generate a negative term, thus reducing the objective function, while any cost greater than will generate a positive term, will increase the objective function. Therefore, the optimal solution to minimize the objective function of Eq. (19) is determined by the following simple criteria:

$$q_i = \begin{cases} 0 & c_i \le \lambda \\ 1 & c_i > \lambda \end{cases} \tag{21}$$

Therefore, the optimal solution of the unknown embedded set is:

$$\Omega^* = \{ i \mid q_i = 1, i = 1, \ldots, N \} \tag{22}$$

In Eq.(16), the neighborhood of each point is constructed based on the entire feature set containing outliers. If the outlier ratio is not high, the strategy works well for the following reasons: on the one hand, for an outlier, its local neighborhood structure cannot be saved between two images, resulting in a high cost, so it is easy to be identified as an outlier. On the other hand, for an embedded point, even if its neighborhood or contains some outliers, in most cases, its main component is the embedded model conforming to geometric constraints, so its cost will not be very large.

It is preferable to construct neighborhoods only based on embedded sets. In this case, the calculation of the cost of embedded set will be more accurate and will not be affected by outliers; Therefore, the boundary between embedded points and outliers will be significantly expanded. This helps to accurately classify hypothetical matches, especially when the outliers in the hypothetical set are relatively high. However, real embedded sets Ω cannot be budgeted. In order to deal with this problem, explore similar sets Ω_0 . A correspondence set can be generated, which can remove most of the outliers while maintaining most of the embedded values. Obviously, this set is a good approximation of the real inline set. The neighborhood is constructed based on the whole set s, i.e

$$\Omega_0 = \arg\min_{\Omega} C(\Omega; S, \lambda) \tag{23}$$

Then, in order to preserve the potential real corresponding local neighborhood structure between the two images, the neighborhood is constructed for each corresponding point in s, and the optimal solution is:

$$\Omega^* = \arg\min_{\Omega} C(\Omega; \Omega_0, S, \lambda) \tag{24}$$

III. Overall navigation architecture

The position information is obtained through the image registration drive control process, and the UAV is recursively guided to align with the scene of the geo reference image. As shown in Fig. 3, within a sampling interval, the aerial camera on the UAV takes aerial images, and compares the similarity with the geographic reference image through the online image registration algorithm. When the common scene of two images is found through image registration, the corresponding transformation matrix is estimated. The latter contains information about how to manipulate the UAV in order to minimize the dissimilarity between image pairs. The UAV control signal generator interprets the estimated transformation matrix as a series of UAV steering control, and then drives the UAV to move to the scene of geo reference image. The process can be regarded as first-order feedback control and will terminate if the desired image matching quality is stable and effective. The UAV of the system has two flight modes. When in "search mode", UAV is guided by airborne inertial navigation system (INS). Once a common scene is detected, the system releases the UAV steering control to the UAV control signal generator and enters the "locked self positioning mode".

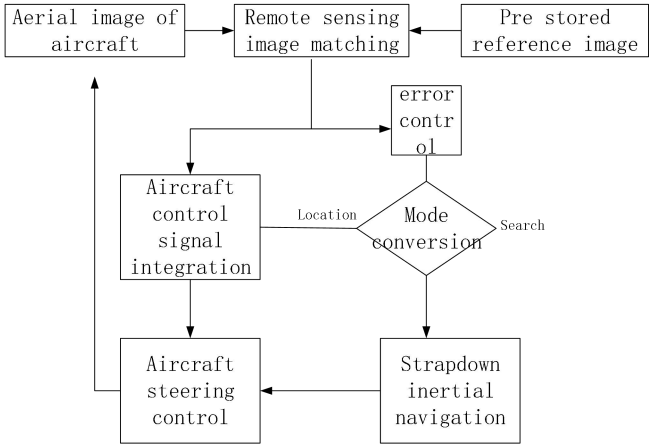


Fig. 3. Positioning and navigation flow charts

In the general scene, the positioning and navigation system based on remote sensing image works on a set of geo reference images taken on the predefined flight path, and uses the positioning results to recalibrate the airborne inertial navigation system. It is more suitable for applications where geo referenced images contain many recognizable features, such as 3D positioning / tracking in complex buildings, or in low altitude urban environments without GNSS.

A. Coarse to fine matching algorithm

Since it is impossible to know the approximate position of the aerial image in the reference image in advance, if the fine matching algorithm is adopted at the beginning, the matching time may be too redundant. Therefore, based on this, the matching algorithm adopts the coarse matching method to conduct a global search on the reference image in the first frame. However, the global search may lead to a large number of outliers in the matching, and the third remote sensing image matching method based on guidance strategy is used to remove them. Then, the local search is carried out at the approximate position by using the fine matching method, which greatly shortens the time required for image matching.

The basic gradient vector $\left[G_x, G_y\right]^T$ is calculated according to the following equation:

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \end{bmatrix} * I(x, y) \tag{25}$$

Where G_x and G_y are gradients in the x and y directions, * represents convolution, I(x,y) represents gray image, and f_x and f_y are Sobel operations in X and Y directions.

However, the basic gradient is very sensitive to noise and the discretization of digital image, and the structural direction estimated by the basic gradient is quite unreliable. Therefore, this chapter develops the average square gradient in a multi-

scale framework, which effectively enhances the robustness of structural direction estimation.

For each scale, calculate the sum of G_x squares, the sum of G_y squares and the product sum of G_x and G_y at each pixel according to the following .

$$G_{xx} = G_x^2 \tag{26}$$

$$G_{yy} = G_y^2 \tag{27}$$

$$G_{xy} = G_x G_y \tag{28}$$

The calculation formula of the final structural direction is as follows:

$$o = \frac{1}{2}\angle\left(G_{xx} - G_{yy}, 2G_{xy}\right) \tag{29}$$

Where $o = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$

In the case of gradient inversion, even if the two gradient vectors are close to parallel, their directions may be very different. In order to solve this problem, the direction similarity measurement method based on cosine absolute value is improved, which measures the parallelism of two vectors. When two vectors are collinear, the similarity reaches the maximum value of 1. When the product of two vectors is zero, the similarity reaches the minimum value of 0. The basic form of this similarity measure is:

$$S_{basic} = \left|\cos\left(O_1 - O_2\right)\right| \tag{30}$$

Where O_1 and O_2 are the directions of the two gradient vectors.

Finally, using the matching method of line matching in the remote sensing image, the error of line matching is extracted and eliminated. The goal is to reject the exception value while obtaining the internal exception value. The probability that the algorithm never selects m sets with all embedded points is q:

$$q = \left(1 - p^m\right)^k \tag{31}$$

Where, m is the minimum number of points required to estimate the model,k is the number of samples required, and p is the probability that the filtering algorithm selects embedded points from the input data set.

In the first n frame images collected by the aircraft, the above coarse image matching algorithm is used to conduct global search in the reference map. At this time, the approximate position of the aircraft can be obtained. Then, after obtaining the approximate position of the aircraft, the fine image matching of local search is used, and the above registration method based on remote sensing image is used for fine matching.

B. *Aircraft steering control*

$$\begin{bmatrix} x_r \\ y_r \\ 1 \end{bmatrix} = \begin{bmatrix} \varphi\cos\theta & -\varphi\sin\theta & b_x \\ \varphi\sin\theta & \varphi\cos\theta & b_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ 1 \end{bmatrix} \tag{32}$$

The transformation matrix H is:

$$H = \begin{bmatrix} \varphi\cos\theta & -\varphi\sin\theta & b_x \\ \varphi\sin\theta & \varphi\cos\theta & b_y \\ 0 & 0 & 1 \end{bmatrix} \tag{33}$$

The formula contains all deformation information between image pairs. In the positioning mode, the matching algorithm is used to estimate the parameters θ 、 φ 、 b_x and b_y of the system and the corresponding transformation matrix H. Then, the UAV steering control is generated by estimating the value of the transformation matrix $\hat{H}$. The purpose of generating the control signal is to guide the UAV towards the scene of the geo referenced image, increase the overlapping area between the UAV aerial map and the geo referenced image, and update the conversion matrix $\hat{H}$ by registering from the new aerial map. This control process is expected to iterate step by step until the estimated transformation matrix approaches the identity matrix.

Under similar mapping, the steering process of the aircraft is expected to have four degrees of freedom control obtained from the image registration output. As shown in Fig 4, these controls are controlling the aircraft to move left or right, move forward or backward, rotate clockwise or counterclockwise, and fly up or down.

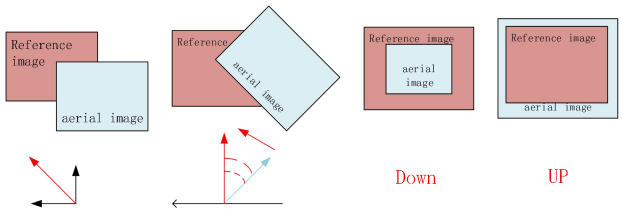


Fig. 4. Steering control

Table 1 lists the control types derived from the transformation matrix parameter values estimated by the image registration algorithm. When multiple controls appear at the same time, the correction of rotation, proportion and displacement will take priority in turn. The relationship between the control parameters in the table and the estimated transformation matrix is approximately as follows.

TABLE I. AIRCRAFT CONTROL SIGNAL

Paramenter	***num***	***Flight control***	***Processing instruction***
θ	>0	Turn counterclockwise	1
	<0	Turn clockwise	2
φ	>1	Up	3
	>1	down	4
b_x	>0	towards the left	5

Paramenter	*num*	*Flight control*	*Processing instruction*
b_y	<0	towards the right	6
	>0	forward	7
	<0	back off	8

The derived control is to drive the first-order feedback controller to turn the UAV to the direction of minimizing the image registration error. They can be implemented by many existing flight controllers, such as Px4 autopilot platform [10]. The self positioning process of aircraft is an optimization process based on control cost, which is executed iteratively in a feedback control cycle. After receiving the control signal interpreted by the estimated transformation matrix, guide the aircraft to approach the scene of the reference image. Then, a new aerial frame is compared with the reference image to generate a new set of control. Driven by these controls, the aircraft gradually approaches the scene of the reference image. This process is repeated until the control cost is no longer reduced, or in other words, the scene of the aerial frame completely overlaps the scene of the reference image.

IV. Positioning and Navigation System

Based on the research of the above methods, this paper designs and develops a positioning and navigation system based on remote sensing image. The main functions include aircraft track planning, remote sensing image matching and positioning and aircraft navigation control. Aircraft route planning mainly realizes the uploading and planning of aircraft route in advance. Remote sensing image matching and positioning mainly realizes the acquisition of the geographical location of the aircraft. Aircraft navigation control is to make the aircraft fly according to the route uploaded in advance on the basis of positioning. The system visualizes the system through Python language and QT desiner development tool. Through the visual interface, it can more conveniently and dynamically display and operate visual positioning and navigation, strengthen interaction and improve the user's sense of experience. The software mainly includes main interface, track planning module, positioning module and navigation module. The design scheme of the whole positioning and navigation system based on remote sensing image is shown in Figure 5.

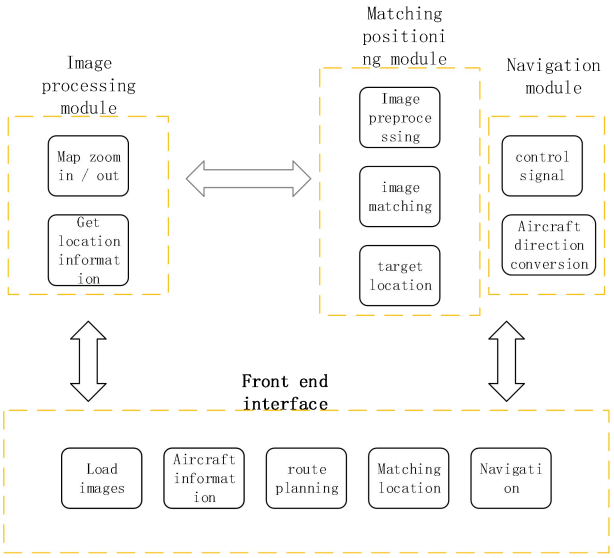


Fig. 5. Overall system design

A. System module

1) Main interface

The interface is visually displayed through the QT designer development tool. The displayed contents include the pre stored reference image of the aircraft, the real-time aerial image of the aircraft, the information of the current position of the aircraft, the entrance of the track planning module, the entrance of the matching positioning module and the entrance of the navigation module. For the pre stored reference image of the aircraft, this chapter uses the remote sensing satellite image of 70 square kilometers in Chang'an District, Xi'an City, Shaanxi Province for simulation. The current position information of the aircraft includes the altitude, flight speed, pitch angle rate, roll angle rate, heading and GPS positioning longitude and latitude information. In the image loading module, clicking image loading can transform the real-time aerial image of the aircraft as the object to be registered, clicking the next group can change the pre stored remote sensing reference image of the aircraft as the matching object, clicking feature extraction can extract the feature points of the reference image and aerial image for display, and clicking the track planning module can enter the track planning module to design the scheduled flight path of the aircraft, Click the image matching and positioning module to enter the image matching and positioning module for image matching and fusion, so as to calculate the longitude and latitude of the aircraft's position at this time based on the matching results, so as to achieve the purpose of aircraft visual positioning, and then issue the command of steering control according to the calculated results to realize aircraft steering control, so that the aircraft can fly with a pre-designed track, so as to achieve the purpose of navigation.

Fig. 6. Schematic diagram of main

2) route planning

The interface is visually displayed through development tools, and the displayed contents include map type, map operation and additional waypoints. The map type is divided into administrative map, satellite map and vector map. The map operation includes reducing map, enlarging map, downloading waypoint, uploading waypoint and moving map. The additional waypoint includes the flight pitch angular velocity, roll angular rate, azimuth rate, longitudinal acceleration and flight duration of the aircraft. The downloaded waypoint includes the waypoint name, longitude and latitude and altitude of the waypoint.

Fig. 7. Schematic diagram of track planning

3) Positioning module

The visual display of the interface is carried out through the development tool. The displayed contents include real-time data, loading, visualization, others, and the display of remote sensing image registration results and positioning results. The real-time data includes the height, resolution and rotation angle information of the real-time aerial image of the aircraft; Loading is mainly responsible for selecting the path of aircraft real-time aerial images for replacement; Visualization mainly includes the adjustment of grid size, satellite remote sensing map type and transparency; Satellite remote sensing map types mainly include goole satellite image source, baidu satellite image source, ARCIS satellite image source, Gaode satellite image source, ARCIS Street image source, baidu Street image source, Google Street image source and Gaode Street image source; Remote sensing image registration adopts the above algorithm for matching. The result image is displayed in the visual interface. The green box shows the position of the aerial image in the reference image, and the specific longitude and latitude information is displayed in the lower right corner.

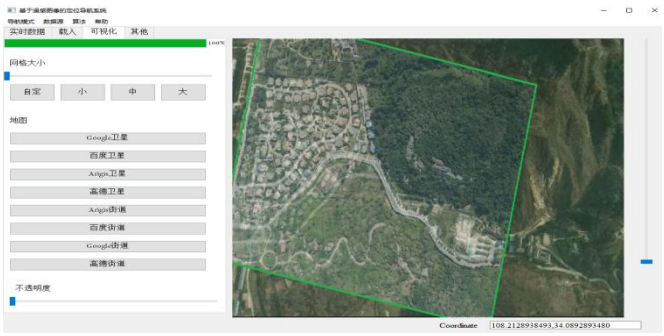


Fig. 8. Matching positioning module

4) Navigation module

The interface is visually displayed through the development tool, and the displayed contents include the preset track display, control command, aircraft track, position error and speed error in the reference image. The control command module includes serial number, control command, behavior made by the aircraft and flight duration; The aircraft track module contains some real-time data such as speed and time; Position error, using the generated real-time data including longitude, latitude and flight altitude to calculate the position error trajectory; The velocity trajectory module uses the generated real-time data, that is, the flight speed of the aircraft, to calculate the velocity error trajectory. The aircraft obtains the real-time position of the aircraft through image matching and positioning, so as to make up for the disadvantage of large error in pure inertial navigation for a long time, so as to achieve the purpose of accurate navigation.

Fig. 9. Navigation module

B. *Positioning experiment*

In this section, two groups of aerial images are taken as the images to be matched, and the image matching and positioning experiment is carried out by using the combination of coarse matching algorithm and accurate matching algorithm. The pixel experimental coordinate position is calculated by calculating the coordinate transformation formula, and the PNP algorithm is used to estimate the position of the aircraft. Then, the error distance between the positioning point and the real position is calculated by using the semi positive vector formula, Observe the difference between the positioning point and the real distance through the error distance.

(1) Areas with obvious geomorphic features

In areas with obvious geomorphic features, such as areas with obvious features such as streets, schools and lakes, more

correct matching feature points are extracted, and the positioning effect is good if the error distance is small. This experiment mainly uses the images to be matched in the school area for positioning. The data shown in Table Ⅱ is the effect of using the above research algorithm for positioning.

TABLE II. LOCATION DATA OF AREAS WITH OBVIOUS GEOMORPHIC FEATURES

Serial number	***Experimental location***	***Real location***	***Error distance / m***
1	（34.03159601，108.6546655）	（108.85457264，108.75357264）	22.23
2	（34.06943414，108.74748119）	（34.07043987,108.94823749）	22.32
3	（34.17043414，108.77930899）	（34.29943414，109.010529066）	19.60
4	（34.06944560，108.84068996）	（34.73944560，108.99393366）	29.59
5	（34.06943987，108.87857966）	（34.06923987，109.11782336）	26.60
average error			24.068

(2) Areas with insignificant geomorphic features

In areas with no obvious geomorphic features, such as areas with no obvious special texture such as mountains, lawns and forests, the extracted correct matching feature points are less, and the error distance is large, the positioning effect is not as good as that in areas with obvious geomorphic features. This experiment mainly uses the images to be matched in the Qinling region for positioning. The data shown in table Ⅲ is the effect of using the above research algorithm for positioning.

TABLE III. AIRCRAFT CONTROL SIGNAL

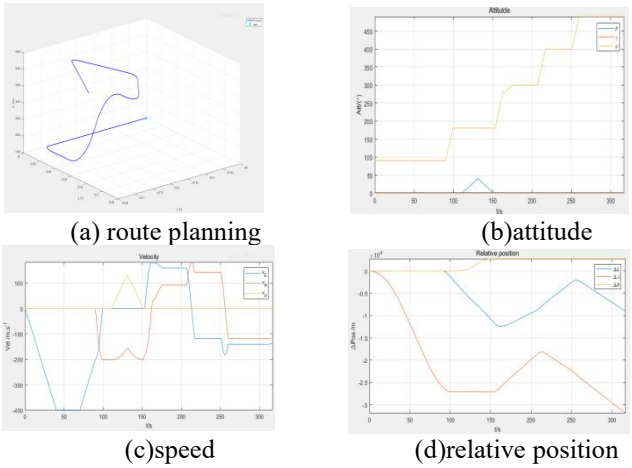
Serial number	***Experimental location***	***Real location***	***Error distance / m***
1	（32.599855,107.323423）	（33.594315,108.352391）	119.32
2	（32.957832,108.987422）	（33.592298,108.352060）	74.17
3	（32.610258，107.98742）	（33.623120,108.873231）	104.73
4	（33.687541,108.998547）	（34.984794,108.345948）	86.01
5	（32.393848,108.878463）	（33.584794,108.351361）	72.26
average error			91.298

It can be observed from the above two experiments that the positioning experiment has good positioning effect for areas with obvious features, and can assist the pure inertial navigation system to make up for the error of pure inertial navigation. However, for areas with unclear features, the positioning error may be greater than the error of inertial navigation, which is difficult to make up for the inconsistency of routes caused by inertial navigation due to long range. The feature-based scene matching method has limitations, The effect is good only in areas with obvious characteristics. Therefore, this positioning and navigation system is not suitable for areas with obvious characteristics such as desert and no man's land.

C. Navigation simulation experiment

a) simulation data

The pitch angle of the initial attitude of the simulated aircraft is set to 1, the yaw angle is set to 1 and the roll angle is set to 30, the East speed, North speed and heading angle in the three directions of the initial attitude of the aircraft are set to 0 respectively, and the latitude of the initial position of the aircraft is set to 34 degrees north latitude, 108 degrees east longitude and 1500 flight altitude. The preset trajectory sets the flight pitch angular velocity, aircraft roll angular rate, aircraft azimuth rate, aircraft longitudinal acceleration and aircraft flight duration as the simulation trajectory, which is specifically manifested in aircraft acceleration, uniform speed, deceleration, turning and acceleration. In this section, the IMU module is simulated, and the static inertial navigation device is simulated in pure inertial navigation mode, and the misalignment angle error is added on this basis, Add errors to the inertial navigation sensor data to continuously update the speed, attitude and position of the aircraft, track planning results, attitude results, correlation speed and relative position results, as shown in Figure 10.

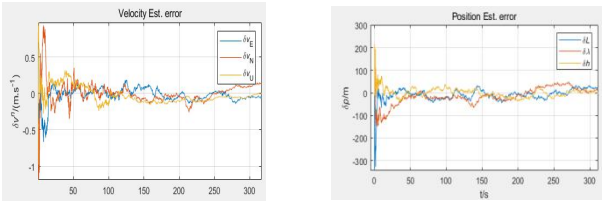


(a) route planning (b)attitude

(c)speed (d)relative position

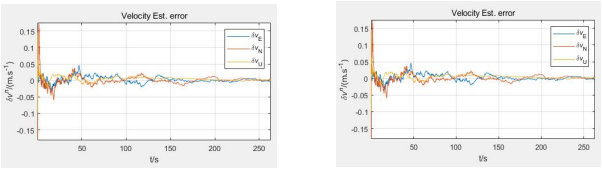
Fig. 10. Schematic diagram of simulation data setting

Based on this simulated aircraft trajectory, in the pure inertial navigation mode, FIG. 11 shows the curve of speed error and position error of aircraft flight, and with the aid of matching positioning, FIG. 12 shows the curve of speed error and position error of aircraft.



(a)Velocity error (b)Position error

Fig. 11. Pure inertial navigation error curve of simulation data



(a)Velocity error (b)Position error

Fig. 12. Aided inertial navigation error curve of simulation data

This section analyzes the speed error and position error caused by navigation in different modes during flight, and draws the errors generated by East speed, North speed and heading angle with time, as well as the errors generated by the longitude, latitude and flight altitude of the aircraft with time. Based on the above error data, the error comparison under different modes is counted, as shown in table Ⅳ.

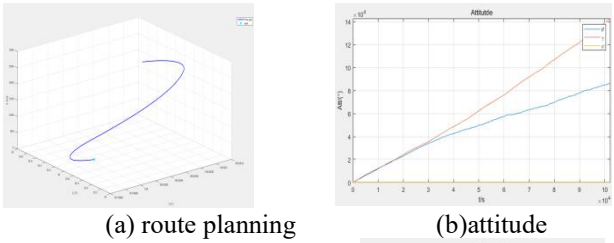
TABLE IV. ERROR STATISTICAL COMPARISON OF SIMULATION DATA

		Error distance / m		***Auxiliary mode***	
		Error range	Mean error	Error range	Mean error
Position error / M	longitude	[-356.72,42.76]	89.95	[-62.72,48.92]	20.98
	latitude	[-142.34,60.25]	78.61	[-38.42,289.9]	19.98
	flight altitude	[-42.74,214.5]	48.54	[-289.2,34.28]	12.64
Speed error / (M / s)	Eastward velocity	[-0.71,0.21]	0.17	[-0.027,0.048]	0.024
	Northward velocity	[-1.22,0.89]	0.32	[-0.053,0.156]	0.088
	Heading angle	[-0.25，0.28]	0.25	[-0.022,0.058]	0.037

From the navigation data of the above simulation data and real data experiment, it can be seen that the error mean and error standard deviation of integrated matched navigation are generally smaller than those of pure inertial navigation mode, which can achieve the purpose of accurate navigation.

b) Real data

The real data adopts the one-time navigation data of Northwest University of technology. Set the pitch angle of the initial attitude of the aircraft as 1, the yaw angle as 1 and the roll angle as 30, the East speed, North speed and heading angle in the three directions of the speed of the initial attitude of the aircraft are set as 0 respectively, and the latitude of the initial position of the aircraft is set as 34.01832630 degrees north latitude, 108.73510530 degrees east longitude and 461.8m flight altitude. The preset trajectory sets the flight pitch angular velocity, aircraft roll angular rate, aircraft azimuth rate, aircraft longitudinal acceleration, and aircraft flight duration as the real trajectory of the UAV. This voyage contains 2480 data, which are embodied in aircraft acceleration, uniform speed, deceleration, turning and acceleration. The same simulation IMU module is used to simulate the devices required for inertial navigation in pure inertial navigation mode, On the basis of this simulation, the misalignment angle error is added, and the error is added to the inertial navigation sensor data to continuously update the speed, attitude and position of the aircraft, as well as the schematic diagram of track planning results, attitude results, speed and relative position, as shown in Figure 13.

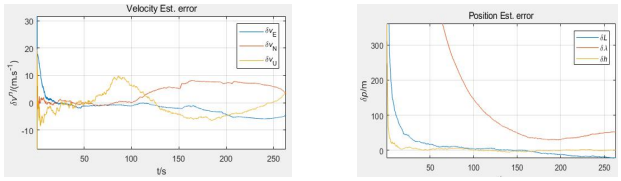


(a) route planning (b)attitude

(c)speed (d)relative position

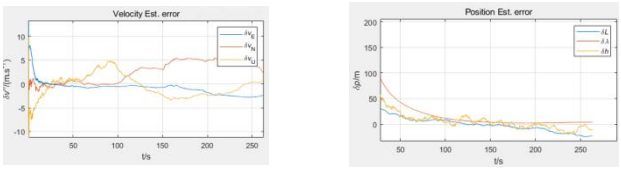
Fig. 13. Schematic diagram of real data setting

Based on this aircraft trajectory, in the pure inertial navigation mode, FIG. 14 shows the curve of speed error and position error of aircraft flight, and with the aid of matching positioning, FIG. 15 shows the curve of speed error and position error of aircraft.



(a)Velocity error (b)Position error

Fig. 14. Pure inertial navigation error curve of real data



(a)Velocity error (b)Position error

Fig. 15. Aided inertial navigation error curve of real data

This paper analyzes the speed error and position error caused by the navigation of real data in different modes in the

flight process, and draws the errors generated by the East speed, North speed and heading angle with time, as well as the errors generated by the longitude, latitude and flight altitude of the aircraft with time. Based on the above error data, the error comparison under different modes is counted, as shown in table Ⅴ.

TABLE V. ERROR STATISTICAL COMPARISON OF REAL DATA

		Error distance / m		***Auxiliary mode***	
		Error range	Mean error	Error range	Mean error
Position error / M	longitude	[-11.97,383.72]	80.72	[-45.78,38.72]	25.84
	latitude	[30.72,487.27]	98.47	[1.73,89.97]	27.76
	flight altitude	[0,150]	52.63	[-25.78,52.18]	23.75
Speed error / (M / s)	Eastward velocity	[-7.21,31.93]	7.37	[-0.027,0.048]	1.33
	Northward velocity	[-0.03,9.17]	5.62	[-0.053,0.156]	2.57
	Heading angle	[-9.83,20.13]	4.18	[-0.022,0.058]	1.85

From the navigation data of the above simulation data and real data experiment, it can be seen that the error mean and error standard deviation of integrated matched navigation are generally smaller than those of pure inertial navigation mode, which can achieve the purpose of accurate navigation.

REFERENCES

[1] Qi Jun, Tian Yimin, Wang Xin An improved Harris corner detection algorithm [J] Journal of Beijing Institute of printing, 2015, 23 (2): 5

[2] Chang Xujian, Han Xie, Xiong Fengjing Feature detection algorithm based on fast detection and sift description [J] Computer engineering and design, 2015, 36 (10): 5

[3] Seema B, Kumar H, Naidu V. Geo-Registration of Aerial Images using RANSAC Algorithm[J], 2014: pp.1-5..

[4] Saranya K C, Naidu V, Singhal V, et al. Application of vision based techniques for UAV position estimation[C]. 2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS), 2016: 1-5.

[5] Sun bujiao, Zhou Donghua Fast matching algorithm based on NCC [J] Sensors and Microsystems, 2007, 26 (9): 3

[6] Yu Qing Q, Qiao Yuqing, Ze Xun g, et al. Image matching based on SIFT and RANSAC algorithm [C] China Computer Society; China system simulation society; China image and graphics society, 2017: 21-25.

[7] Yang Jibin, Hu Zhanyu, Zhu Gao Remote sensing image registration based on improved surf algorithm [J] Electronic measurement technology, 2012, 35 (3): 5

[8] Mm A, Dp A, Rn A, et al. A novel measurement model based on abBRIEF for global localization of a UAV over satellite images[J]. Robotics and Autonomous Systems, 2019, 112: 304-319.

[9] Muja M, Lowe D G. Fast Matching of Binary Features[C]. Computer and Robot Vision (CRV), 2012 Ninth Conference on, 2012.

[10] Jiang qiongge, Wang Lifeng Research on obstacle avoidance system and path planning of ground unmanned vehicle based on Px4 [J] Powertrain and control, 2019, 8 (2): 14