scientific reports

Check for updates

OPEN

A visual positioning model for UAV's patrolling video sequence images based on DOM rectification

Haojie Liu, Wei Fan✉ & Di Wu

With technological development of multi sensors, UAV (unmanned aerial vehicle) can identify and locate key targets in essential monitoring areas or geological disaster-prone areas by taking video sequence images, and precise positioning of the video sequence images is constantly a matter of great concern. In recent years, precise positioning of aerial images has been widely studied. But it is still a challenge to simultaneously realize precise, robust and dynamic positioning of UAV's patrolling video sequence images in real time. In order to solve this problem, a visual positioning model for patrolling video sequence images based on DOM rectification is proposed, including a robust block-matching algorithm and a precise polynomial-rectifying algorithm. First, the robust block-matching algorithm is used to obtain the best matching area for UAV's video sequence image on DOM (Digital Orthophoto Map), a pre-acquired digital orthophoto map covering the whole UAV's patrolling region. Second, the precise polynomial-rectifying algorithm is used to calculate accurate rectification parameters of mapping UAV's video sequence image to the best matching area obtained above, and then real time positioning of UAV's patrolling video sequence images can be realized. Finally, the above two algorithms are analyzed and verified by three practical experiments, and results indicate that even if spatial resolution, surface specific features, illumination condition and topographic relief are significantly different between DOM and UAV's patrolling video sequence images, proposed algorithms can still steadily realize positioning of each UAV's patrolling video sequence image with about 2.5 m level accuracy in 1 s. To some extent, this study has improved precise positioning effects of UAV's patrolling video sequence images in real time, and the proposed mathematical model can be directly incorporated into UAV's patrolling system without any hardware overhead.

Patrol inspection is an important work in many industries, and UAV (unmanned aerial vehicle) has gradually become a new tool for field patrol inspection due to its low cost and high efficiency. Shooting video to discover what has happened at sometime and somewhere in real-time is a common way of field patrol inspection by UAV, which is usually equipped with an image sensor and a POS (position and orientation system) unit. Accessing precise location of interest points in UAV's real time patrolling video sequence images is of great value for discovery and elimination of hidden safety hazards. At present, there are four different kinds of methods that can be used for positioning of UAV's patrolling video sequence images: photogrammetry method, binocular vision method, image feature matching method, and optical flow method.

Photogrammetry method includes forward intersection algorithm and block adjustment algorithm that can be both used for positioning of UAV's patrolling video sequence images. Forward intersection algorithm[1–3] can calculate out geodetic coordinates of homologous image points with assistance of POS data in real time. But the poor results cannot meet accuracy requirements of UAV's patrolling video sequence images positioning. Block adjustment algorithm[4–6] can precisely calculate out geodetic coordinates of homologous image points by using large overlapped sequence images under certain geometric conditions, but it is a post-processing algorithm which cannot meet the real-time requirements of UAV's patrolling video sequence images positioning.

Binocular vision method uses binocular cameras with precise 3D coordinates and 3D orientations to shoot two images of the same scene simultaneously, and then geodetic coordinates of homologous image points can be calculated out according to vertical parallax of the two images. Binocular vision method has high accuracy and efficiency, and mainly focus on precise calibration of the fixed binocular cameras at present[7–13], which cannot meet the dynamic requirement of UAV's patrolling video sequence images positioning.

Feature matching method realizes image matching by verifying consistency of descriptors that are obtained from surrounding pixels of corresponding key-points in two images. The famous SIFT (scale invariant feature

Yellow River Engineering Consulting Co., LTD., Zhengzhou 450003, China. ✉email: fanwei@yrec.cn

transform) algorithm realizes feature matching of scale-invariant, rotation-invariant and illumination-invariant by constructing Gaussian pyramid images and regional gradient distributions[14,15], which is widely employed in image registration[16,17] and image mosaic[18,19]. Ke proposed PCA-SIFT algorithm[20] by using PCA (Principal Component Analysis) method to reduce dimension of regional gradient distributions, which improved efficiency of feature matching to a certain extent. Following the idea of scale-invariant in SIFT, Morel proposed a so-called ASIFT algorithm of affine-invariant by simulating image geometric distortions caused by variations of camera optical axis[21]. Bay constructed multi-scale spaces by using box filters and integral images, constructed key-points and corresponding descriptors by using non-maximum suppression method[22] and Haar wavelet transform, and then proposed SURF (Speed Up Robust Features) algorithm[23,24]. SURF is one order magnitude faster than SIFT[25]. A more faster algorithm ORB (Oriented FAST and Rotated BRIEF)[25], further proposed by Bblee, used oriented FAST (Features from Accelerated Segment Test) algorithm[26] to detect key-points and used rotated BRIEF (Binary Robust Independent Elementary Features) algorithm[27] to construct descriptors. Other feature matching methods also exert certain influence on image matching, including BRISK (Binary Robust Invariant Scalable Keypoints) algorithm[29], KAZE algorithm[30], hardware acceleration algorithm[31], and etc. Literature[32,33] compare and analyze accuracy, efficiency, advantages and disadvantages of existing feature matching methods in details, and we will not go into much here. Feature matching method can be used for precise real-time image matching, while matching results only have relative positioning information, which cannot meet the absolute positioning requirements of UAV's patrolling video sequence images.

Optical flow method can obtain motion displacement of pixels between two adjacent sequence images through energy differential-difference equations which are constructed by certain assumptions and solved by certain optimization criteria. If all the image pixels are involved in this method, we call it dense optical flow, and if only part of the image pixels are involved in this method, we call it sparse optical flow. Two of the most classical optical flow algorithms are LK optical flow[34] and HS optical flow[35]. LK optical flow is constructed on three basic assumptions, namely, brightness constancy (projection of the same point looks the same in every frame), small motion (points do not move very fast) and spatial coherence (points move like their neighbors)[34]. LK optical flow can calculate out motion displacement of pixels between two adjacent sequence images accurately, but performs poor stability sometimes. Based on above mentioned three basic assumptions, HS optical flow adds a regularization term in the self-constructed differential-difference equations. By minimizing the self-constructed differential-difference equations with regularization term, HS optical flow obtains the optimal motion displacement between two adjacent sequence images' homologous points, which achieves a more stable performance. However, "brightness constancy" and "small motion" are two strong assumptions in LK optical flow and HS optical flow, which are difficult to be satisfied in practical applications. For this reason, lots of improved algorithms have been proposed. In Literature[36], gradient conservation is used to replace the assumption of brightness constancy, which improves robustness of optical flow algorithm against illumination variation. Literature[37] proposes multi-scale searching strategies, which has improved optical flow algorithm's tracking efficiency of objects with large motion and shortened calculating time. A coarse-to-fine process has been mentioned in literature[38], which further improves optical flow algorithm's tracking ability of objects with large motion. Literature[39] proposes an optical flow algorithm based on interpolation of correspondences, which has achieved good results in tracking objects with large displacement and significant occlusions. In literature[40,41], polynomials fitted by intensity of regional pixels are used for tracking objects with large motion, illumination variation and noise interference, and good results have also been achieved. With the development of artificial intelligence, optical flow algorithm based on neural network[42] has also emerged, but their robustness on unknown data sets remains to be verified. At present, optical flow algorithm has been widely used in medical image registration[44,45], remote sensing image registration[46], visual navigation[47] and many other industries. Optical flow method can be used for precise matching of sequence images, while the relative positioning results cannot meet the absolute positioning requirements of UAV's patrolling video sequence images.

To sum up, there is no method that can solve absolute positioning of UAV's patrolling video sequence images accurately and robustly in real time. For this reason, a series of visual positioning algorithms for UAV's patrolling video sequence images based on DOM rectification are proposed following the coarse-to-fine principle in this paper. All the proposed algorithms are analyzed and verified by three practical experiments, and results show that these algorithms are fast, effective and feasible.

Methodology

Technical flow

As shown in Fig. 1, number 1 is a UAV (unmanned aerial vehicle) in patrolling; Number 2 is a UAV's video sequence image, which is taken by the patrolling UAV (number 1) and is needed to be positioned in real time; Number 3 is named as region-DOM, which is a digital orthophoto map of UAV's patrolling region and is produced in advance; Number 4 is named as datum-DOM, which is a subarea of region-DOM (number 3); Number 5 is named as block-matched-DOM, which is further a subarea of datum-DOM (number 4) and is the best matching region for UAV's video sequence image (number 2) on datum-DOM (number 4). It should be noted that, UAV's patrolling video sequence image is abbreviated as video frame for convenience of subsequent work.

As shown in Fig. 1, the basic idea of this paper is to find out the best matching region (number 5) for video frame (number 2) on region-DOM (number 3) quickly and robustly, figure out the accurate rectification parameters for mapping video frame (number 2) to the best matching region (number 5), and finally realize real time positioning of video frame (number 2) by using accurate rectification parameters to obtain geodetic coordinates of each pixels in video frame (number 2). Following the basic idea and the coarse-to-fine principle, the technical flow of this study is described as follows.

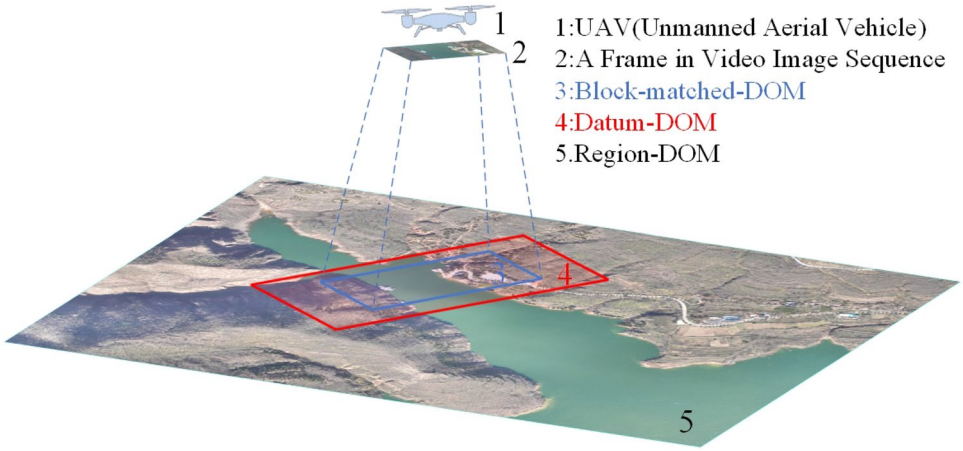


Figure 1. Key images involved in this study.

First, extract datum-DOM (number 4) from region-DOM (number 3) according to the POS data of video frame (number 2), and replace region-DOM (number 3) by datum-DOM (number 4) as a new matching area for video frame (number 2), so as to reduce matching area of video frame (number 2) on region-DOM (number 3) and increase matching speed.

Second, extract block-matched-DOM (number 5) from datum-DOM (number 4) by using the proposed robust block-matching algorithm. It should be noted that, video frame (number 2) and block-matched-DOM (number 5) have the same size in pixels, but the matching accuracy between these two images is still poor due to numerous negative factors. Therefore, a further optimization step is needed.

Third, figure out accurate rectification parameters for mapping video frame (number 2) to block-matched-DOM (number 5) by using the proposed precise polynomial-rectifying algorithm.

Finally, obtain geodetic coordinates of each pixel in video frame (number 2) by using the accurate rectification parameters calculated above, so as to realize the real time positioning of video frame (number 2).

Algorithm framework

The algorithm flow of this study is shown in Fig. 2. Advantages lie in the proposed robust image-block-matching algorithm and precise polynomial-rectifying algorithm, which can solve geodetic coordinates of all pixels in a UAV's real-time video frame with about 2.5 m level accuracy in 1 s.

The visual positioning model

Extraction of datum-DOM

Following the basic idea of this paper, datum-DOM should be extracted from region-DOM at the beginning, so as to reduce matching area of video frame on region-DOM and increase matching speed. As shown in Fig. 1, Central point's coordinates of datum-DOM is determined by geodetic coordinates of UAV's POS data; Azimuth of datum-DOM is determined by yaw angle of UAV's POS data; Length and width of datum-DOM in pixels is determined by equations as:

$$\begin{cases} L_{pixels} = n\frac{L_{dist}}{gsd_D} \\ W_{pixels} = n\frac{W_{dist}}{gsd_D} \end{cases} \tag{1}$$

where, L_{pixels} and W_{pixels} are length and width of datum-DOM in pixels respectively; $L_{dist} = H_{fly} \times L_{CMOS}/f$; $W_{dist} = H_{fly} \times W_{CMOS}/f$; L_{CMOS} and W_{CMOS} are physical length and width of UAV's CMOS (Complementary Metal Oxide Semiconductor) sensor respectively; f is focal length of UAV's camera; gsd_D is spatial resolution of datum-DOM; n is scaling coefficient, ranging from 1.5 to 2.

Finally, datum-DOM can be extracted from region-DOM according to the already known parameters $(L_{POS}, B_{POS}, Yaw_{pos}, L_{pixels}, W_{pixels})$. Where, (L_{POS}, B_{POS}) are central point's coordinates of datum-DOM; Yaw_{pos} is yaw angle of UAV's POS data; L_{pixels} and W_{pixels} are obtained from Eq. (1).

Construction of robust block-matching algorithm

Follow the basic idea of this paper, the best matching area for video frame on datum-DOM should be extracted. However, existing image feature matching methods are all difficult to match video frame and datum-DOM automatically, since illumination conditions, surface specific features, projection modes and spatial resolution of these two kinds images are greatly different. Therefore, a robust block-matching algorithm is constructed for the purpose of finding out the best matching area for video frame on datum-DOM.

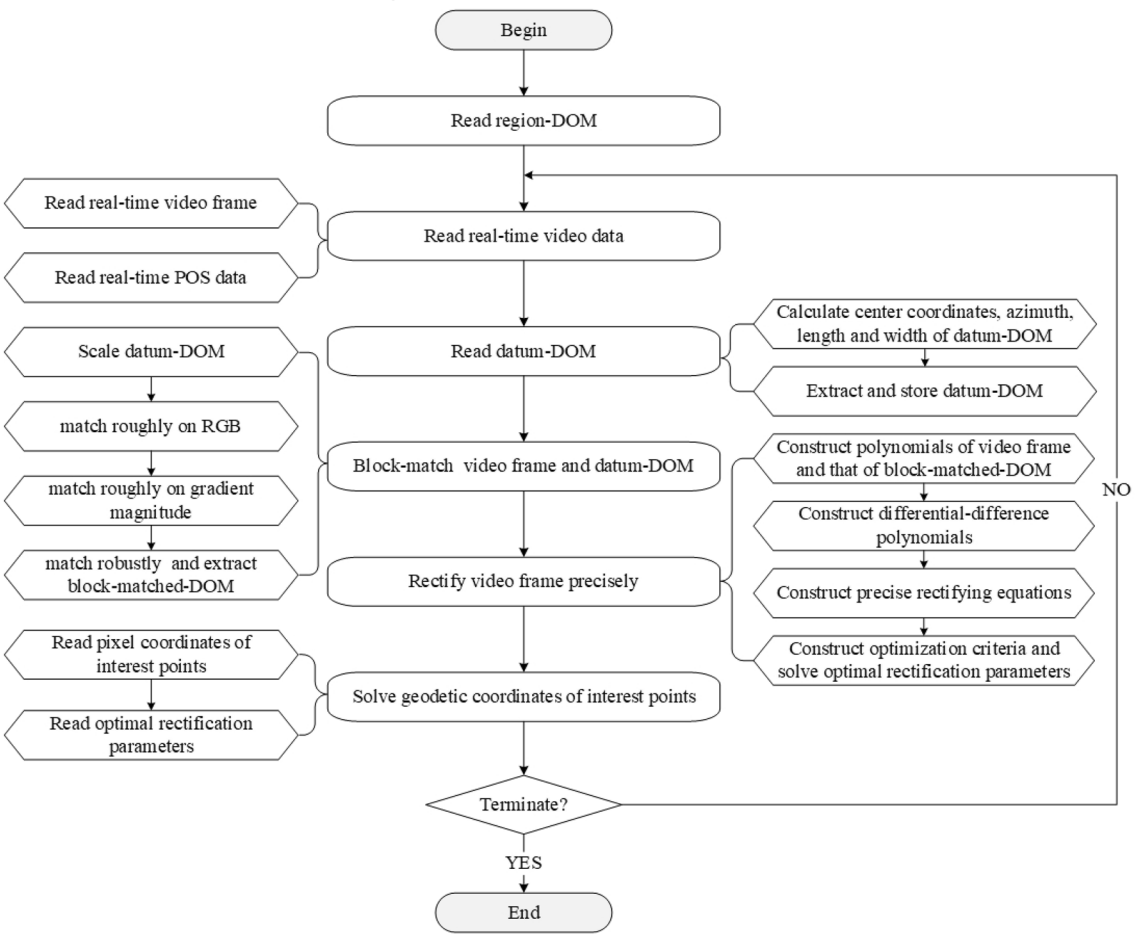


Figure 2. Algorithm framework.

Scaling datum-DOM

It is necessary to unify spatial resolutions of datum-DOM and video frame, so as to facilitate the subsequent matching work. To ensure the spatial resolution of datum-DOM is the same as video frame, datum-DOM is scaled as:

$$img_{D_S} = img_D \cdot Scale_D \tag{2}$$

where img_{D_S} represent the size of scaled datum-DOM; img_D represent the size of original datum-DOM; $Scale_D = gsd_D/gsd_F$; gsd_D is spatial resolution of the original datum-DOM; gsd_F is spatial resolution of video frame;

Block-matching roughly based on RGB color

At this step, the best matching area for video frame on datum-DOM can be found out based on the similarity of these two images in RGB color space. As shown in Fig. 3, (x_{L1}, y_{L1}) are pixel coordinates of the top left corner of the best matching area for video frame on datum-DOM in RGB color space, and (x_{L1}, y_{L1}) can be obtained as:

$$(x_{L1}, y_{L1}) = \underset{arg(\Delta x_1, \Delta y_1)}{max} \frac{\sum_{x,y} F_R D_R + \sum_{x,y} F_G D_G + \sum_{x,y} F_B D_B}{\sqrt{\sum_{x,y} F_R^2 \sum_{x,y} D_R^2} + \sqrt{\sum_{x,y} F_G^2 \sum_{x,y} D_G^2} + \sqrt{\sum_{x,y} F_B^2 \sum_{x,y} D_B^2}} \tag{3}$$

$$\text{where, } \begin{cases} F_R = F_{R_{src}}(x,y) - \sum_{x,y} F_{R_{src}}(x,y)/N_F \\ D_R = D_{R_{src}}(\Delta x_1 + x, \Delta y_1 + y) - \sum_{x,y} D_{R_{src}}(\Delta x_1 + x, \Delta y_1 + y)/N_F \\ F_G = F_{G_{src}}(x,y) - \sum_{x,y} F_{G_{src}}(x,y)/N_F \\ D_G = D_{G_{src}}(\Delta x_1 + x, \Delta y_1 + y) - \sum_{x,y} D_{G_{src}}(\Delta x_1 + x, \Delta y_1 + y)/N_F \\ F_B = F_{B_{src}}(x,y) - \sum_{x,y} F_{B_{src}}(x,y)/N_F \\ D_B = D_{B_{src}}(\Delta x_1 + x, \Delta y_1 + y) - \sum_{x,y} D_{B_{src}}(\Delta x_1 + x, \Delta y_1 + y)/N_F \end{cases}$$

$F_{R_{src}}(x,y)$, $F_{G_{src}}(x,y)$ and $F_{B_{src}}(x,y)$ are intensity of R, G and B channel of video frame respectively; $D_{R_{src}}(\Delta x_1 + x, \Delta y_1 + y)$, $D_{G_{src}}(\Delta x_1 + x, \Delta y_1 + y)$ and $D_{B_{src}}(\Delta x_1 + x, \Delta y_1 + y)$ are intensity of R, G and B channel of datum-DOM respectively; (x, y) are pixel coordinates in video frame, $x = (1, 2, \cdots, N_{LF})$, $y = (1, 2, \cdots, N_{WF})$; $(\Delta x_1, \Delta y_1)$ are pixel coordinates of video frame's top left corner in datum-DOM, $\Delta x_1 = (1, 2, \cdots, N_{LD} - N_{LF})$,

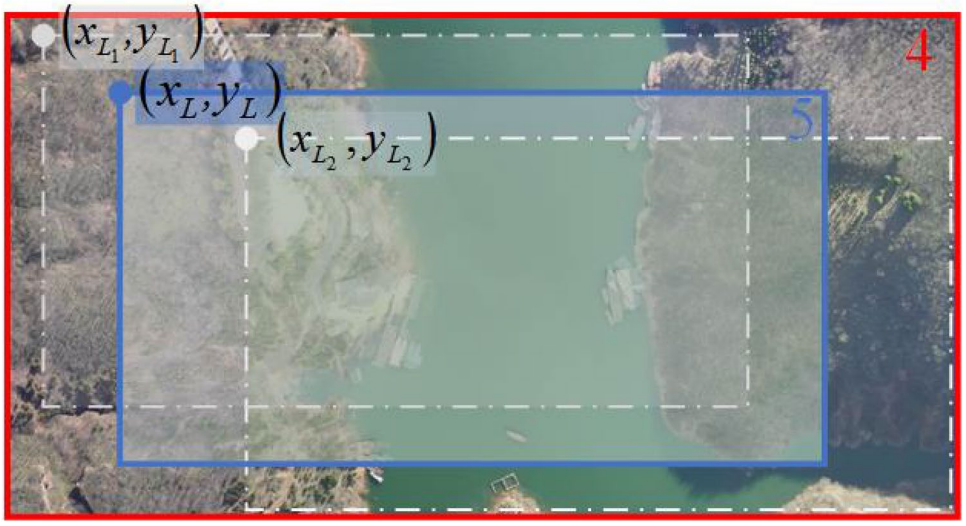


Figure 3. The robust block-matching of datum-DOM and video frame.

$\Delta y_1 = (1, 2, \cdots, N_{WD} - N_{WF})$; N_{LF} and N_{WF} are length and width of video frame in pixels respectively; N_{LD} and N_{WD} are length and width of datum-DOM in pixels respectively; N_F is the total pixel numbers of video frame, $N_F = N_{LF}N_{WF}$.

Block-matching roughly based on gradient magnitude

At this step, the best matching area for video frame on datum-DOM can be found out based on the similarity of these two images in gradient magnitude space. As shown in Fig. 3, (x_{L2}, y_{L2}) are pixel coordinates of the top left corner of the best matching area for video frame on datum-DOM in gradient magnitude space, and (x_{L2}, y_{L2}) can be obtained as:

$$(x_{L2}, y_{L2}) = \max_{arg(\Delta x_2, \Delta y_2)} \frac{\sum_{x,y}\left[I_F(x,y) I_D(x_{L2}+x, y_{L2+y})\right]^2}{\sqrt{\sum_{x,y} I_F^2(x,y) \sum_{x,y} I_D^2(x_{L2}+x, y_{L2+y})}} \tag{4}$$

where,

$$\begin{cases} I_F(x,y) = \sqrt{F_x^2(x,y) + F_y^2(x,y)} \\ I_D(\Delta x_2 + x, \Delta y_2 + y) = \sqrt{D_x^2(\Delta x_2 + x, \Delta y_2 + y) + D_y^2(\Delta x_2 + x, \Delta y_2 + y)} \end{cases}$$; $F_x(x,y)$ and $F_y(x,y)$ are first partial derivative of video frame in x and y direction respectively; $D_x(\Delta x_2 + x, \Delta y_2 + y)$ and $D_y(\Delta x_2 + x, \Delta y_2 + y)$ are first partial derivative of datum-DOM in x and y direction respectively; (x, y) are pixel coordinates in video frame, $x = (1, 2, \cdots, N_{LF})$, $y = (1, 2, \cdots, N_{WF})$; $(\Delta x_2, \Delta y_2)$ are pixel coordinates of video frame's top left corner in datum-DOM, $\Delta x_2 = (1, 2, \cdots, N_{LD} - N_{LF})$, $\Delta y_2 = (1, 2, \cdots, N_{WD} - N_{WF})$; N_{LF} and N_{WF} are length and width of video frame in pixels respectively; N_{LD} and N_{WD} are length and width of datum-DOM in pixels respectively;

Block-matching robustly

In practice, it has been found that the above proposed RGB based block-matching method exhibits better performance in video frame with large color difference and complicate texture, while the above proposed gradient magnitude based block-matching method exhibits better performance in video frame with small color difference and simple texture.Therefore, it is necessary to further construct a robustly block-matching method by considering both color difference and texture complexity of video frame.

In the robustly block-matching method, symbol TH is proposed to comprehensive represent color difference amplitude and texture complexity of video frame, and a threshold number 20 is selected to judge TH. If $TH \leq 20$, the video frame is considered to have large color difference and complicate texture, and the matching result in section "Block-matching Roughly Based on RGB color" should have a lager weight. On the contrary, if TH > 20, the video frame is considered to have small color difference and simple texture, and the matching result in section "Block-matching Roughly Based on Gradient Magnitude" should have a larger weight. TH is calculated in Eq. (5), and the threshold number 20 is selected by numerous practical experiments.

As shown in Fig. 3, (x_L, y_L) are coordinates of the top left corner of the best matching area obtained by the proposed robustly block-matching method, and (x_L, y_L) can be calculated as:

$$(x_L, y_L) = \begin{cases} \frac{\left[\binom{x_{L_1}}{y_{L_1}} + \omega_L \binom{x_{L_2}}{y_{L_2}}\right]}{1+\omega_L} & TH \le 20 \\ \frac{\left[\omega_L \binom{x_{L_1}}{y_{L_1}} + \binom{x_{L_2}}{y_{L_2}}\right]}{1+\omega_L} & TH > 20 \end{cases} \tag{5}$$

$$TH = \sum_{x,y} \left| I_F(x,y) - I_D(x_{L_2}+x, y_{L_2}+y) \right| / N_F$$

where, ω_L is a weight, $\omega_L = \begin{cases} 1, & r \le 1/2 \\ 1/2r, & 1/2 < r \le 1 \\ 0, & 1 < r \end{cases}$, r represents a distance between (x_{L_1}, y_{L_1}) and (x_{L_2}, y_{L_2}), $r = \frac{\sqrt{(x_{L_1}-x_{L_2})^2+(y_{L_1}-y_{L_2})^2}}{\sqrt{(N_{LF}/10)^2+(N_{WF}/10)^2}}$, equations of r, ω_L and TH are all constructed by numerous practical experiments; (x_{L_1}, y_{L_1}) and (x_{L_2}, y_{L_2}) are obtained by Eqs. (3) and (4) respectively; N_{LF} and N_{WF} are length and width of video frame in pixels respectively; N_F is the total pixel numbers of video frame; TH represents color difference and texture complexity of video frame; the threshold number 20 is selected by numerous practical experiments; Meaning of the rest parameters can refer to Eqs. (3) and (4).

Extracting block-matched-DOM

According to parameters $(x_L, y_L, N_{LF}, N_{WF})$ calculated in Eq. (5), Block-matched-DOM can be extracted from datum-DOM. As shown in Fig. 3, block-matched-DOM is the area in blue box marked by number 5, and is the best matching area for video frame on datum-DOM ultimately found.

It should be noted that, video frame and its corresponding block-matched-DOM have the same size in pixels, and geodetic coordinates of each pixel on video frame can be obtained directly from the geodetic coordinates of pixels at the same position on block-matched-DOM. That is to say, positioning of UAV's patrolling video frame can be realized by directly assigning geodetic coordinates of each pixel in block-matched-DOM to pixels at the same position in UAV's patrolling video frame.

Construction of precise polynomial-rectifying algorithm

Unfortunately, there is a high probability that pixels in video frame are not homologous with pixels in block-matched-DOM at the same position, due to numerous negative factors, such as illumination variation, inconsistent spatial resolution, diverse surface specific features, topographic relief, camera distortion, different projection modes and etc. That is to say, the positioning accuracy of video frame is still poor, if we assign geodetic coordinates of each pixel in block-matched-DOM directly to pixels at the same position in video frame. In order to realize accurate positioning of UAV's patrolling video sequence images, a precise polynomial-rectifying algorithm is further constructed.

The basic idea of the proposed precise polynomial-rectifying algorithm is to find out homologous regions in block-matched-DOM for regions in video frame, so as to figure out accurate rectification parameters for mapping video frame to block-matched-DOM. And finally, accurate positioning of video frame can be realized by using accurate rectification parameters to calculate geodetic coordinates of each pixel in video frame. It should be noted that, we are committed to find out homologous regions between video frame and block-matched-DOM, instead of finding out the homologous points. Because homologous regions are more stable and reliable than homologous points under numerous negative influences. Where, homologous regions refer to the most similar local areas between two images.

Through in-depth study of common characteristics between block-matched-DOM and video frame, the precise polynomial-rectifying algorithm is constructed based on three assumptions: (1) video frame and block-matched-DOM can be regarded as two adjacent sequence images. (2) Overall surface features are similar between video frame and block-matched-DOM. (3) Pixels in a local area of the video frame share a same deformation law.

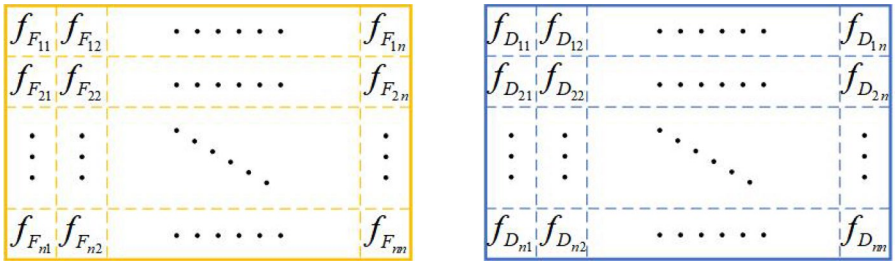
Constructing polynomials of video frame and that of block-matched-DOM

In order to reduce negative influence of illumination variation, gradient magnitude images of video frame and that of block-matched-DOM are used for image matching. In order to further reduce negative influence of diverse surface specific features, gradient magnitude images of video frame and that of block-matched-DOM are represented by second-order polynomials respectively, and the second-order polynomials of these two images are used for image matching ultimately.

As shown in Fig. 4, gradient magnitude images of video frame and that of block-matched-DOM are evenly divided into $n \times n$ local areas respectively, and each of the local areas is represented by a second-order polynomial as:

$$\begin{cases} f_{F_{ij}}(X_{\mathrm{I}}, T_{\mathrm{I}}) = X_{\mathrm{I}}^T A_{\mathrm{I}} X_{\mathrm{I}} + B_{\mathrm{I}}{}^T X_{\mathrm{I}} + C_{\mathrm{I}} \\ f_{D_{ij}}(X_{\Pi}, T_{\Pi}) = X_{\Pi}^T A_{\Pi} X_{\Pi} + B_{\Pi}{}^T X_{\Pi} + C_{\Pi} \end{cases} \tag{6}$$

where, $f_{F_{ij}}(X_{\mathrm{I}}, T_{\mathrm{I}})$ and $f_{D_{ij}}(X_{\Pi}, T_{\Pi})$ are intensity of local area of row i and column j in Fig. 4a,b respectively, $i = (1, \cdots, n)$, $j = (1, \cdots, n)$; $X_{\mathrm{I}} = (x_{\mathrm{I}}, y_{\mathrm{I}})^T$, $X_{\Pi} = (x_{\Pi}, y_{\Pi})^T$, T represent transpose of a matrix (vector),$(x_{\mathrm{I}}, y_{\mathrm{I}})$ and (x_{Π}, y_{Π}) are pixel coordinates in local areas of Fig. 4a,b respectively; T_{I} and T_{Π} are production time of video frame and that of block-matched-DOM respectively; $A_{\mathrm{I}} = \begin{pmatrix} m_4^{\mathrm{I}} & m_6^{\mathrm{I}}/2 \\ m_6^{\mathrm{I}}/2 & m_5^{\mathrm{I}} \end{pmatrix}$, $A_{\Pi} = \begin{pmatrix} m_4^{\Pi} & m_6^{\Pi}/2 \\ m_6^{\Pi}/2 & m_5^{\Pi} \end{pmatrix}$, A_{I} and A_{Π}



(a) All divided local areas in video frame's gradient magnitude image

(b) All divided local areas in block-matched-DOM's gradient magnitude image

Figure 4. Local areas divided in video frame and block-matched-DOM.

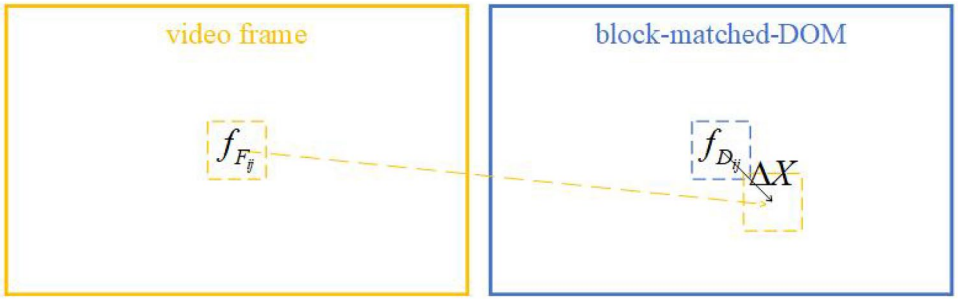


Figure 5. Small motion displacement from a local area of video frame to the corresponding local area of block-matched-DOM.

are second-order coefficient matrix of their polynomials respectively; $B_{\mathrm{I}} = (m_2^{\mathrm{I}}, m_3^{\mathrm{I}})^T$,$B_{\Pi} = (m_2^{\Pi}, m_3^{\Pi})^T$, B_{I} and B_{Π} are first-order coefficient vectors of their polynomials respectively; $C_{\mathrm{I}} = m_1^{\mathrm{I}}$, $C_{\Pi} = m_1^{\Pi}$, C_{I} and C_{Π} are scalars of their polynomials respectively; $m_1^{\mathrm{I}}, m_2^{\mathrm{I}}, m_3^{\mathrm{I}}, m_4^{\mathrm{I}}, m_5^{\mathrm{I}}, m_6^{\mathrm{I}}, m_1^{\Pi}, m_2^{\Pi}, m_3^{\Pi}, m_4^{\Pi}, m_5^{\Pi}, m_6^{\Pi}$ are parameters of polynomials.

Constructing differential-difference polynomials

Block-matched-DOM is the best matching area for video frame on datum-DOM. However, there are still irregular motion displacements between homologous regions of these two images due to numerous negative factors. Therefore, finding out homologous regions of these two images is important for precise positioning of video frame.

Based on the assumption that video frame and block-matched-DOM can be regarded as two adjacent sequence images, the second-order polynomials of video frame and that of block-matched-DOM can also be regarded as two adjacent sequence images. And then, differential-difference polynomials can be constructed based on Eq. (6), and further can be rewritten by using Taylor expansion for ΔX to the first order derivative as:

$$\begin{aligned} d &= f_D(X_{\Pi}, T_{\Pi}) - f_F(X_{\mathrm{I}}, T_{\mathrm{I}}) \\ &= f_D(X_{\mathrm{I}} + \Delta X, T_{\Pi}) - f_F(X_{\mathrm{I}}, T_{\mathrm{I}}) \\ &= f_D(X_{\mathrm{I}}, T_{\Pi}) + \frac{\partial f_D(X_{\mathrm{I}}, T_{\Pi})}{\partial X_{\mathrm{I}}}\Delta X - f_F(X_{\mathrm{I}}, T_{\mathrm{I}}) \\ &= X_{\mathrm{I}}^T A_{\Pi} X_{\mathrm{I}} + B_{\Pi}^T X_{\mathrm{I}} + C_{\Pi} + 2X_{\mathrm{I}}^T A_{\Pi}\Delta X + B_{\Pi}^T \Delta X - (X_{\mathrm{I}}^T A_{\mathrm{I}} X_{\mathrm{I}} + B_{\mathrm{I}}^T X_{\mathrm{I}} + C_{\mathrm{I}}) \\ &= X_{\mathrm{I}}^T (A_{\Pi} - A_{\mathrm{I}}) X_{\mathrm{I}} + (B_{\Pi} - B_{\mathrm{I}} + 2A_{\Pi}\Delta X)^T X_{\mathrm{I}} + \left(C_{\Pi} + B_{\Pi}^T \Delta X - C_{\mathrm{I}}\right) \end{aligned} \tag{7}$$

where, $f_F(X_{\mathrm{I}}, T_{\mathrm{I}})$ and $f_D(X_{\Pi}, T_{\Pi})$ are intensity of the corresponding local areas in Fig. 4a,b respectively; X_{I} and X_{Π} are pixel coordinates in local areas of Fig. 4a,b respectively; T_{I} and T_{Π} are production time of video frame and that of block-matched-DOM respectively;$A_{\mathrm{I}} = \begin{pmatrix} m_4^{\mathrm{I}} & m_6^{\mathrm{I}}/2 \\ m_6^{\mathrm{I}}/2 & m_5^{\mathrm{I}} \end{pmatrix}$, $A_{\Pi} = \begin{pmatrix} m_4^{\Pi} & m_6^{\Pi}/2 \\ m_6^{\Pi}/2 & m_5^{\Pi} \end{pmatrix}$; $B_{\mathrm{I}} = (m_2^{\mathrm{I}}, m_3^{\mathrm{I}})^T$,$B_{\Pi} = (m_2^{\Pi}, m_3^{\Pi})^T$; $C_{\mathrm{I}} = m_1^{\mathrm{I}}$, $C_{\Pi} = m_1^{\Pi}$; $m_1^{\mathrm{I}}, m_2^{\mathrm{I}}, m_3^{\mathrm{I}}, m_4^{\mathrm{I}}, m_5^{\mathrm{I}}, m_6^{\mathrm{I}}, m_1^{\Pi}, m_2^{\Pi}, m_3^{\Pi}, m_4^{\Pi}, m_5^{\Pi}, m_6^{\Pi}$ are parameters of polynomials; $X_{\Pi} = X_{\mathrm{I}} + \Delta X$.

As shown in Fig. 5, ΔX is a small motion displacement from a local area of video frame to the corresponding local area of block-matched-DOM. That is to say, homologous regions between video frame and block-matched-DOM can be obtained by finding out ΔX that can minimizes d in Eq. (7).

In Eq. (7), let d be exactly equal to zero, we can obtain as:

$$\begin{cases} A_\Pi = A_\mathrm{I} \\ A_\Pi \Delta X = (B_\mathrm{I} - B_\Pi)/2 \\ {B_\Pi}^T \Delta X = C_\mathrm{I} - C_\Pi \end{cases} \tag{8}$$

Further, we can obtain equations of ΔX as:

$$A_{\Delta X} \Delta X = L_{\Delta X} \tag{9}$$

where, $A_{\Delta X} = \begin{bmatrix} (A_\mathrm{I} + A_\Pi)/2 \\ {B_\Pi}^T \end{bmatrix}; L_{\Delta X} = \begin{bmatrix} (B_\mathrm{I} - B_\Pi)/2 \\ C_\mathrm{I} - C_\Pi \end{bmatrix}$.

Constructing precise rectifying equations

ΔX In Eq. (9) can be also regarded as registration errors between video frame and block-matched-DOM. These registration errors are supposed to be caused by video frame's scaling, displacement, rotation, distortion and etc. And then, ΔX can be also represented by second-order polynomials as:

$$\Delta X = \begin{pmatrix} a_0 + a_1 x_f + a_2 y_f + a_3 x_f^2 + a_4 x_f y_f + a_5 y_f^2 \\ b_0 + b_1 x_f + b_2 y_f + b_3 x_f^2 + b_4 x_f y_f + b_5 y_f^2 \end{pmatrix} \tag{10}$$

where, (x_f, y_f) are coordinates of a local area in video frame; $a_0, a_1, a_2, a_3, a_4, a_5, b_0, b_1, b_2, b_3, b_4, b_5$ are parameters of polynomials.

According to Eqs. (9) and (10), precise rectifying equations can be constructed ultimately as:

$$At = L \tag{11}$$

Where, $A = A_{\Delta X} \begin{pmatrix} 1, x_f, y_f, x_f^2, x_f y_f, y_f^2, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, 1, x_f, y_f, x_f^2, x_f y_f, y_f^2 \end{pmatrix}$, $A_{\Delta X} = \begin{bmatrix} (A_\mathrm{I} + A_\Pi)/2 \\ {B_\Pi}^T \end{bmatrix}$, x_f and y_f are column and row numbers of a local area in video frame respectively;$t = (a_0, a_1, a_2, a_3, a_4, a_5, b_0, b_1, b_2, b_3, b_4, b_5)^T$, t is a vector of unknown parameters to be resolved; $L = \begin{bmatrix} (B_\mathrm{I} - B_\Pi)/2 \\ C_\mathrm{I} - C_\Pi \end{bmatrix}$.

Constructing optimal estimation model

As shown in Eq. (11), the task of finding out ΔX is converted to find out t, and each pair of local areas in Fig. 4a,b can construct 3 equations. That is to say, $3n^2$ equations can be constructed in the form of Eq. (11), as there are n^2 pairs of local areas in Fig. 4a,b.

According to the presumption that the minimum energy difference should exist between video frame and block-matched-DOM in homologous regions, the optimization criteria for the $3n^2$ equations that are constructed in the form of Eq. (11) can be proposed as:

$$\underset{arg\ t}{min}\, V^T \Omega V \tag{12}$$

where, $V = At - L$, V is a vector of residual errors; Ω is a weight matrix;t is a vector of unknown parameters; Meaning of the remaining parameters refer to Eq. (11).

In order to obtain the optimal estimation of t, following iteration process are recommended.

① Down-sample images and construct k-layer image pyramids for video frame and block-matched-DOM.

② Set $\Omega = I$, I is an identity matrix; Set $i = k$ and $t = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T$.

③ Construct matrix A and L according to the ith layer images of pyramid.

④ Calculate correction vector for t as: $\Delta t = (A^T \Omega A)^{-1} A^T \Omega (L - At_0)$.

⑤ Calculate vector of residual errors as: $V = A(t_0 + \Delta t) - L$.

⑥ Redefine weight matrix as: $\Omega = \begin{pmatrix} \Omega_1 & 0 & \cdots & 0 \\ 0 & \Omega_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \Omega_n \end{pmatrix}$, $\Omega_j = \begin{cases} \Omega_j, \left|V_j\right| \le 1.5\sigma \\ \Omega_j \frac{\sigma_e}{\left|e_j\right|}, 1.5\sigma < \left|V_j\right| \le 3\sigma, \sigma = \sqrt{\frac{V^T \Omega V}{n^2 - 12}}, \\ 0, 3\sigma < \left|V_j\right| \end{cases}$

$j = 1, \cdots, n$.

⑦ Set $t = t + \Delta t$.

⑧ Repeat steps ④–⑦ m times, and we set $m = 3$ in this study.

⑨ Set $i = k - 1$. Repeat steps ③–⑧ until i equals zero, and the optimal estimates of t is calculated out from the last iteration.

Positioning of UAV's patrolling video frame

By using the optimal estimates of t above resolved, precise geodetic coordinates of each pixel in video frame can be obtained as below:

$$\begin{pmatrix} L \\ B \end{pmatrix} = P\widehat{X} \tag{13}$$

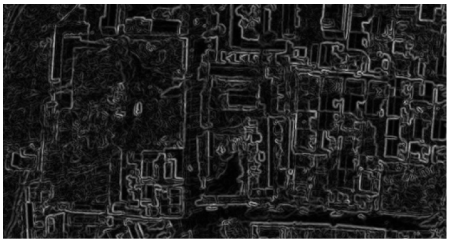
(a) Region-DOM

(b) Video frame (town area)

(c) Datum-DOM

(d) Block-matched-DOM

(e) Gradient magnitude image of video frame



(f) Gradient magnitude image of block-matched-DOM

(g) Accurate positioning result of video frame

Figure 6. The first experiment.

where, (L, B) are geodetic coordinates of a pixel in video frame;$P = \begin{pmatrix} P_A & P_B & P_C \\ P_D & P_E & P_F \end{pmatrix}$, P is a transformation matrix provided by producer of region-DOM; $\widehat{X} = X + X_f t$; $X = (x, y, 1)^T$, (x, y) are pixel coordinates of a pixel in video frame; $X_f = \begin{pmatrix} 1, x_f, y_f, x_f^2, x_f y_f, y_f^2, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, 1, x_f, y_f, x_f^2, x_f y_f, y_f^2 \\ 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 \end{pmatrix}$, x_f and y_f are column and row numbers of the local area where the pixel is located; t is the optimal rectifying parameters calculated above.

Finally, according to Eq. (13), precise positioning of UAV's patrolling video sequence images can be realized by calculating geodetic coordinates of each pixel in UAV's patrolling video sequence images.

Case study

Three practical experiments are designed in this study, which includes 3 videos and 2 region-DOMs. Among them, 3 videos are shot by 3 sorties fly of UAV in different areas, including town area, river area and high relief amplitude area. 2 region-DOMs have different spatial resolutions, one of the 2 region-DOMs has a lower spatial resolution, and the other one has a higher spatial resolution.

(h) Hybrid image of figure (b) and (a)

(I) Hybrid image of figure (g) and (a)

Figure 6. (continued)

The first experiment

As shown in Fig. 6, Fig. 6a is region-DOM used in this experiment, which was made on January 31, 2020, with length of 4096 pixels, width of 1792 pixels, and spatial resolution of 0.493663 m/pixels. Video used in this experiment was shot by 1 sortie fly of UAV at an altitude of about 250 m on November 24, 2021, including 3154 frames, with fps (frames per second) of 23.98, spatial resolution of 0.0684932 m/pixels, length of 4096 pixels and width of 2160 pixels in a single frame. In addition, the video was shot in town area. Figure 6b is the 301st frame of the video used in this experiment, and is picked out for algorithm demonstration without loss of generality. POS data of the 301st frame are obtained by IMU (Inertial Measurement Unit) mounted on UAV, where, the center geodetic coordinates are (111.2661504°, 34.2428275°), flight altitude is 250.1 m, pitch angle is − 8.3°, roll angle is − 1.3°, and yaw angle is 82.5°.

According to the theory proposed in section "Extraction of datum-DOM", Fig. 6c is datum-DOM that is extracted from Fig. 6a on the basis of POS data of Fig. 6b.

According to the theory proposed in section "Construction of robust block-matching algorithm", Fig. 6d is block-matched-DOM that is extracted from Fig. 6c, and Fig. 6d is the best matching area for Fig. 6b on Fig. 6c. And by timekeeping in the program,

Point number	1	2	3	4	5	6	7	8	9	10
Deviation of homologous points in Fig. 6h/m	2.834	1.807	1.622	1.942	5.698	3.149	5.511	3.537	4.133	3.111
Deviation of homologous points in Fig. 6i/m	1.234	3.307	1.268	2.519	5.517	2.735	2.523	1.184	0.555	1.857
Point number	11	12	13	14	15	16	17	18	19	20
Deviation of homologous points in Fig. 6h/m	5.729	5.082	5.537	3.617	4.802	5.956	8.365	6.487	6.587	6.764
Deviation of homologous points in Fig. 6i/m	1.638	1.491	1.128	1.303	0.746	1.409	6.489	2.154	2.096	2.287
Mean deviation of homologous points in Fig. 6h/m	4.614									
Mean deviation of homologous points in Fig. 6i/m	2.172									

Table 1. Distance deviations between 20 red homologous points in Fig. 6h,i.

According to the theory proposed in section "construction of precise polynomial-rectifying algorithm", Fig. 6e,f are gradient magnitude images that are calculated from Fig. 6b,d respectively. And, the optimal estimation t calculated out from Fig. 6e,f is, $t = (-7.8973, 0.1650, -0.0083, 0.0002, 0.0004, -0.0002, 2.1185, -0.1622, 0.2590, 0.0003, 0.0006, 0.0017)$.

According to the theory proposed in section "Positioning of UAV's patrolling video frame", Fig. 6g is the accurate positioning result of video frame. Figure 6g is obtained by using parameter t and P to calculate geodetic coordinates of each pixel in Fig. 6b. Where, t is obtained by optimal estimation model mentioned above, P is provided by producer of region-DOM, and $P = \begin{pmatrix} 0.0000053644 & 0 & 111.2558010221 \\ 0 & -0.0000053644 & 34.2457553744 \end{pmatrix}$.

Figure 6h is a hybrid image formed by superimposing Fig. 6b on Fig. 6a according to their geodetic coordinates. Where, geodetic coordinates of Fig. 6a are pre-acquired, and geodetic coordinates of Fig. 6b are directly assigned from the block-matched-DOM. Among Fig. 6h, the gray area is Fig. 6b and the 20 red points are interest points on Fig. 6b. Distance deviations between the 20 red homologous points in Fig. 6a,b are measured in ArcGIS and listed in Table 1, and the average distance deviation is 4.614 m.

Figure 6i is a hybrid image formed by superimposing Fig. 6g on Fig. 6a according to their geodetic coordinates. Where, geodetic coordinates of Fig. 6a are pre-acquired, and geodetic coordinates of Fig. 6g are obtained by using parameter t and P to calculate geodetic coordinates of each pixel in video frame. Among Fig. 6i, the gray area is Fig. 6g and the 20 red points are interest points on Fig. 6b. In order to improve reliability and generality of the experiment, all the 20 red homologous points are evenly selected from distinctive terrain points and building points without any deliberate adjustment. Distance deviations between the 20 red homologous points in Fig. 6a,g are measured in ArcGIS and listed in Table 1, and the average distance deviation is 2.172 m.

By timekeeping in our program, it takes about 0.206 s to complete extracting of the block-matched-DOM, it takes about 0.330 s to complete calculating of the optimal estimation t, and it takes about 0.101 s to complete calculating of the precise geodetic coordinates of video frame pixel by pixel. That is to say, the total positioning time of this UAV's patrolling video frame is less than 1 s.

The second experiment

As shown in Fig. 7, Fig. 7a is region-DOM used in this experiment, and is same as Fig. 6a. Video used in this experiment was shot by 1 sortie fly of UAV at an altitude of about 250 m on November 25, 2021, including 4687 frames, with fps (frames per second) of 23.98, spatial resolution of 0.0684932 m/pixels, length of 4096 pixels and width of 2160 pixels in a single frame. In addition, the video was shot in river area. Figure 7b is the 3547st frame of the video used in this experiment, and is picked out for algorithm demonstration without loss of generality. POS data of the 3547st frame are obtained by IMU mounted on UAV, where, the center geodetic coordinates are (111.2658703°, 34.2406338°), flight altitude is 250.1 m, pitch angle is − 7.1°, roll angle is 2.9°, and yaw angle is − 94.8°.

According to the theory proposed in section "Extraction of datum-DOM", Fig. 7c is datum-DOM that is extracted from Fig. 7a on the basis of POS data of Fig. 7b.

According to the theory proposed in section "construction of robust block-matching algorithm", Fig. 7d is block-matched-DOM that is extracted from Fig. 7c,d is the best matching area for Fig. 7b on Fig. 7c.

According to the theory proposed in section "Construction of precise polynomial-rectifying algorithm", Fig. 7e,f are gradient magnitude images that are calculated from Fig. 7b,d respectively. And, the optimal estimation t calculated out from Fig. 7e,f is, $t = (-17.6265, 0.0919, -0.0814, 0.0004, 0.0001, 0.0001, -36.3304, 0.0845, 0.0535, 0.0005, 0.0000, -0.0006)$.

According to the theory proposed in Section "Positioning of UAV's patrolling video frame", Fig. 7g is the accurate positioning result of video frame. Figure 7g is obtained by using parameter t and P to calculate geodetic coordinates of each pixel in Fig. 7b. Where, t is obtained by optimal estimation model mentioned above, P is provided by producer of region-DOM, and $P = \begin{pmatrix} 0.0000053644 & 0 & 111.2558010221 \\ 0 & -0.0000053644 & 34.2457553744 \end{pmatrix}$.

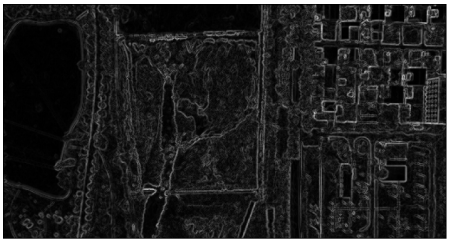
Figure 7h is a hybrid image formed by superimposing Fig. 7b on Fig. 7a in software according to their geodetic coordinates. Where, geodetic coordinates of Fig. 7a are pre-acquired, and geodetic coordinates of Fig. 7b are directly assigned from the block-matched-DOM. Among Fig. 7h, the gray area is Fig. 7b and the 20 red points are

(a) Region-DOM

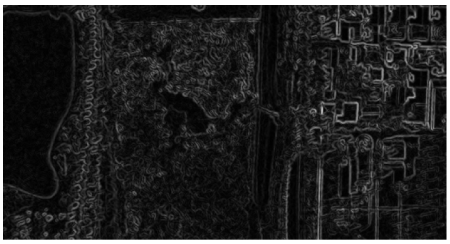
(b) Video frame (river area)

(c) Datum-DOM

(d) Block-matched-DOM



(e) Gradient magnitude image of video frame



(f) Gradient magnitude image of block-matched-DOM

(g) Accurate positioning result of video frame

Figure 7. The second experiment.

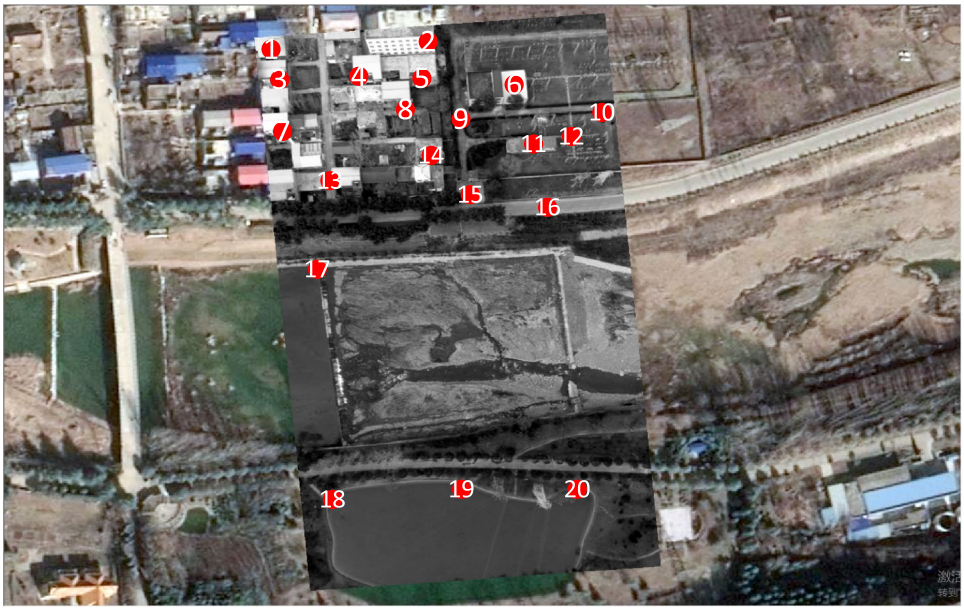
interest points on Fig. 7b. Distance deviations between the 20 red homologous points in Fig. 7a,b are measured in ArcGIS and listed in Table 2, and the average distance deviation is 5.240 m.

Figure 7i is a hybrid image formed by superimposing Fig. 7g on Fig. 7a according to their geodetic coordinates. Where, geodetic coordinates of Fig. 7a are pre-acquired, and geodetic coordinates of Fig. 7g are obtained by using parameter t and P to calculate geodetic coordinates of each pixel in video frame. Among Fig. 7i, the gray area is Fig. 7g and the 20 red points are interest points on Fig. 7b. In order to improve reliability and generality of the experiment, all the 20 red homologous points are evenly selected from distinctive terrain points and building points without any deliberate adjustment. Distance deviations between the 20 red homologous points in Fig. 7a,g are measured in ArcGIS and listed in Table 2, and the average distance deviation is 2.253 m.

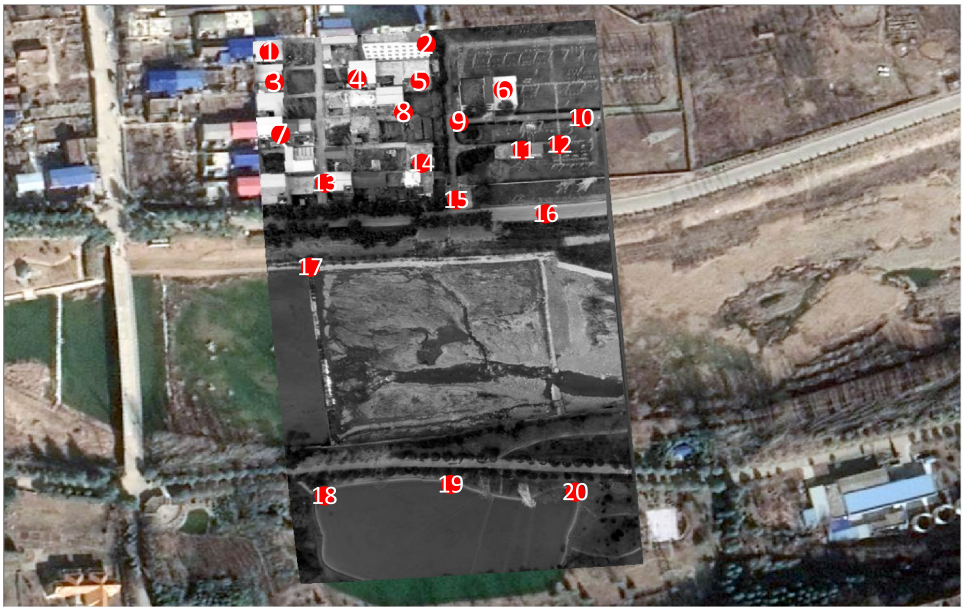
By timekeeping in our program, it takes about 0.119 s to complete extracting of the block-matched-DOM, it takes about 0.118 s to complete calculating of the optimal estimation t, and it takes about 0.053 s to complete calculating of the precise geodetic coordinates of video frame pixel by pixel. That is to say, the total positioning time of this UAV's patrolling video frame is less than 1 s.

The third experiment

As shown in Fig. 8, Fig. 8a is region-DOM used in this experiment, which was made on May 13, 2021, with length of 19,266 pixels, width of 14,483 pixels, and spatial resolution of 0.08 m/pixels. Video used in this experiment was shot by 1 sortie fly of UAV at an altitude of about 250 m on November 26, 2021, including 3788 frames,



(h) Hybrid image of figure (b) and (a)

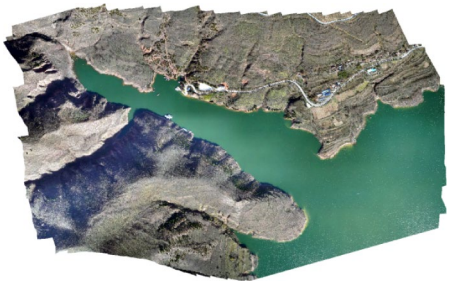


(I) Hybrid image of figure (g) and (a)

Figure 7. (continued)

Point number	1	2	3	4	5	6	7	8	9	10
Deviation of homologous points in Fig. 7h/m	6.262	5.44	6.344	5.725	7.465	3.538	5.698	5.383	6.675	5.501
Deviation of homologous points in Fig. 7i/m	1.602	7.240	1.556	1.118	1.622	1.831	3.778	1.478	1.646	1.029
Point number	11	12	13	14	15	16	17	18	19	20
Deviation of homologous points in Fig. 7h/m	3.728	4.763	5.720	5.305	4.763	3.406	4.622	6.152	4.051	4.262
Deviation of homologous points in Fig. 7i/m	0.539	0.354	2.157	0.720	0.411	0.307	0.219	5.540	5.811	6.106
Mean deviation of homologous points in Fig. 7h/m	5.240									
Mean deviation of homologous points in Fig. 7i/m	2.253									

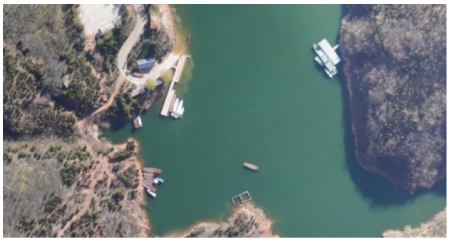
Table 2. Distance deviations between 20 red homologous points in Fig. 7h,i.



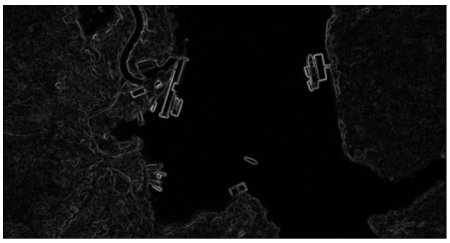
(a) Region-DOM

(b) Video frame (high relief amplitude area)

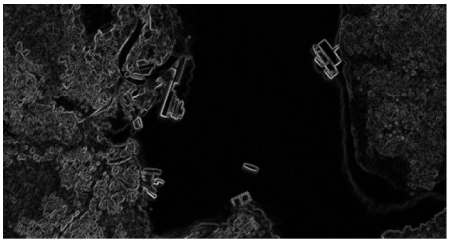
(c) Datum-DOM



(d) Block-matched-DOM



(e) Gradient magnitude image of video frame



(f) Gradient magnitude image of block-matched-DOM

(g) Accurate positioning result of video frame

Figure 8. The third experiment.

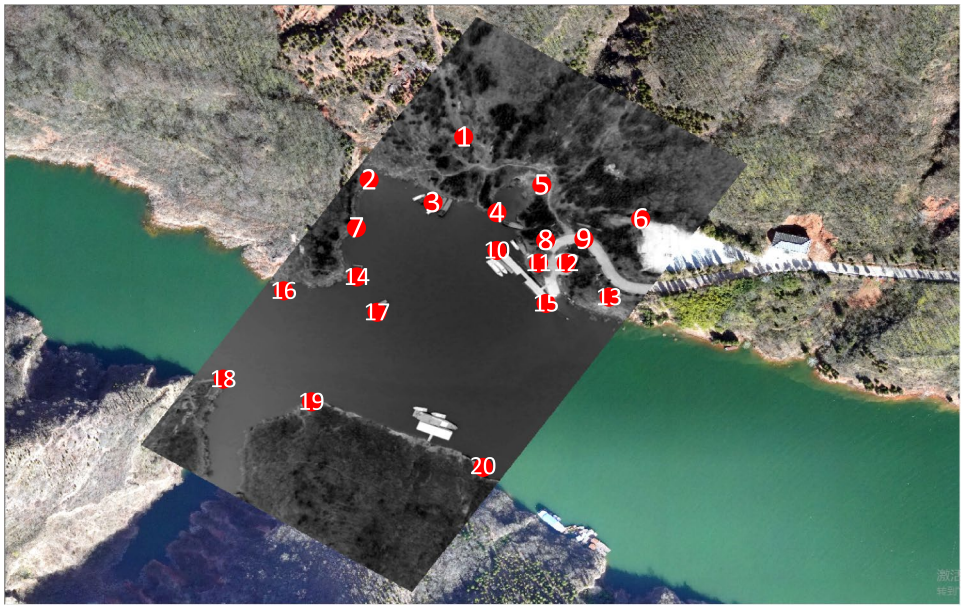
with fps (frames per second) of 23.98, spatial resolution of 0.0684932 m/pixels, length of 4096 pixels and width of 2160 pixels in a single frame. In addition, the video was shot in high relief amplitude area. Figure 8b is the 901st frame of the video used in this experiment, and is picked out for algorithm demonstration without loss of generality. POS data of the 901st frame are obtained by IMU mounted on UAV, where, the center geodetic coordinates are (111.2504477°, 34.2280547°), flight altitude is 250.5 m, pitch angle is − 22.7°, roll angle is − 9.7°, and yaw angle is 123.2°.

According to the theory proposed in section "Extraction of datum-DOM", Fig. 8c is datum-DOM that is extracted from Fig. 8a on the basis of POS data of Fig. 8b.

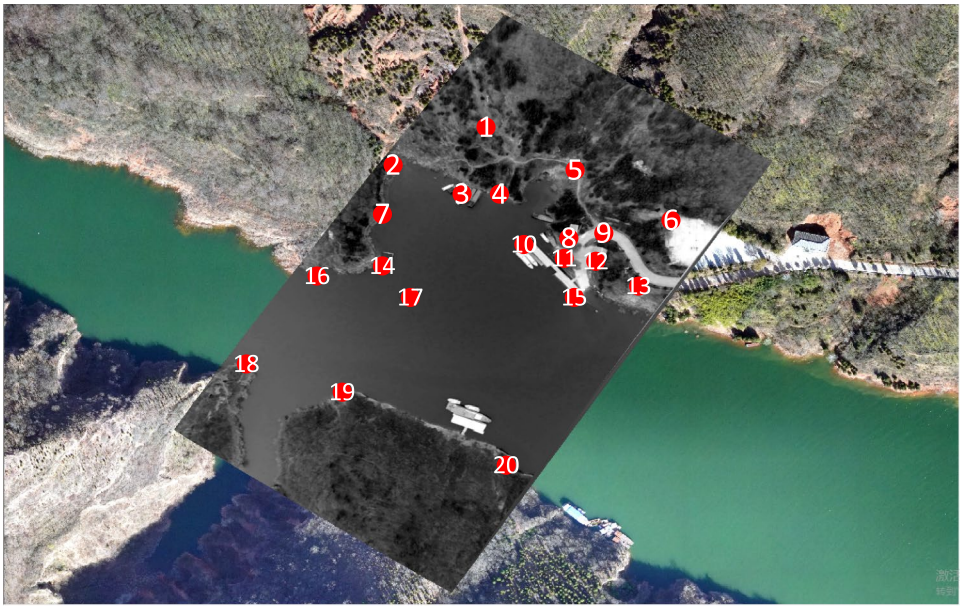
According to the theory proposed in section "construction of robust block-matching algorithm", Fig. 8d is block-matched-DOM that is extracted from Fig. 8c,d is the best matching area for Fig. 8b on Fig. 8c.

According to the theory proposed in section "Construction of precise polynomial-rectifying algorithm", Fig. 8e,f are gradient magnitude images that are calculated from Fig. 8b,d respectively. And, the optimal estimation t calculated out from Fig. 8e,f is, $t = (-1.3754, 0.0206, -0.1524, 0.0006, -0.0004, -0.0001, 6.9715, -0.0663, 0.2922, -0.0006, 0.0008, 0.0009)$.

According to the theory proposed in Section "Positioning of UAV's patrolling video frame", Fig. 8g is the accurate positioning result of video frame. Figure 8g is obtained by using parameter t and P to calculate geodetic



(h) Hybrid image of figure (b) and (a)



(I) Hybrid image of figure (g) and (a)

Figure 8. (continued)

coordinates of each pixel in Fig. 8b. Where, t is obtained by optimal estimation model mentioned above, P is provided by producer of region-DOM, and $P = \begin{pmatrix} 0.0000008683 & 0 & 111.2441313977 \\ 0 & -0.0000007212 & 34.2315524404 \end{pmatrix}$.

Figure 8h is a hybrid image formed by superimposing Fig. 8b on Fig. 8a according to their geodetic coordinates. Where, geodetic coordinates of Fig. 8a are pre-acquired, and geodetic coordinates of Fig. 8b are directly assigned from the block-matched-DOM. Among Fig. 8h, the gray area is Fig. 8b and the 20 red points are interest points on Fig. 8b. Distance deviations between the 20 red homologous points in Fig. 8a,b are measured in ArcGIS and listed in Table 3, and the average distance deviation is 7.105 m.

Figure 8i is a hybrid image formed by superimposing Fig. 8g on Fig. 8a according to their geodetic coordinates. Where, geodetic coordinates of Fig. 8a are pre-acquired, and geodetic coordinates of Fig. 8g are obtained by using parameter t and P to calculate geodetic coordinates of each pixel in video frame. Among Fig. 8i, the gray area is Fig. 8g and the 20 red points are interest points on Fig. 8b. In order to improve reliability and generality of the experiment, all the 20 red homologous points are evenly selected from distinctive terrain points and building points without any deliberate adjustment. Distance deviations between the 20 red homologous points in Fig. 8a,g are measured in ArcGIS and listed in Table 3, and the average distance deviation is 3.619 m.

Point number	1	2	3	4	5	6	7	8	9	10
Deviation of homologous points in Fig. 8h/m	8.594	12.119	8.255	8.470	8.483	2.828	9.881	5.656	6.587	5.497
Deviation of homologous points in Fig. 8i/m	1.431	4.612	2.136	3.272	4.114	0.358	3.967	0.870	1.485	1.69
Point number	11	12	13	14	15	16	17	18	19	20
Deviation of homologous points in Fig. 8h/m	4.460	5.352	3.039	7.484	4.483	10.303	7.137	10.812	6.426	6.236
Deviation of homologous points in Fig. 8i/m	0.400	0.738	1.204	4.968	0.684	6.99	3.66	11.339	9.921	8.547
Mean deviation of homologous points in Fig. 8h/m	7.105									
Mean deviation of homologous points in Fig. 8i/m	3.619									

Table 3. Distance deviations between 20 red homologous points in Fig. 8h,i.

By timekeeping in our program, it takes about 0.118 s to complete extracting of the block-matched-DOM, it takes about 0.122 s to complete calculating of the optimal estimation t, and it takes about 0.074 s to complete calculating of the precise geodetic coordinates of video frame pixel by pixel. That is to say, the total positioning time of this UAV's patrolling video frame is less than 1 s.

Experimental analysis

In the first experiment, spatial resolution of region-DOM is far less than that of video frame, region-DOM's surface universal features are similar with video frame's, and region-DOM's surface specific features and illumination condition are great different from video frame's. From the experimental results, we can see that average positioning deviation of all interest points in Fig. 6h is about 4.614 m, and average positioning deviation of all interest points in Fig. 6i is about 2.172 m. Among them, interest points that are located on roads and low-rise buildings have lower positioning deviations, while interest points that are located on high-rise buildings have higher positioning deviations.

In the second experiment, spatial resolution of region-DOM is still far less than that of video frame, region-DOM's surface universal features are similar with video frame's, region-DOM's surface specific features and illumination condition are greatly different from video frame's, and surface features on the left side of video frame is significantly less than those on the right side. From the experimental results, we can see that average positioning deviation of all interest points in Fig. 7h is about 5.2402 m, and average positioning deviation of all interest points in Fig. 7i is about 2.2532 m. Among them, interest points that are located on roads and low-rise buildings have lower positioning deviations, interest points that are located on high-rise buildings have higher positioning deviations, and interest points that are located on the left side of video frame have higher positioning deviations than those on the right side.

In the third experiment, spatial resolution of region-DOM is similar with that of video frame, region-DOM's surface universal features are similar with video frame's, region-DOM's surface specific features and illumination condition are a little different from video frame's, while there are extensive mountain body shadows on region-DOM. From the experimental results, we can see that average positioning deviation of all interest points in Fig. 8h is about 7.1051 m, and average positioning deviation of all interest points in Fig. 8i is about 3.6193 m. Among them, interest points that are located on roads and low-rise buildings have lower positioning deviations than the first two experiments, while interest points that are located on mountain edges have the highest positioning deviations.

By analyzing the above three experiments, following conclusions can be achieved.

(1) Geometrical shape of video frame deformed obviously after accurate positioning, as shown in Figs. 6g, 7g and 8g.
(2) The average positioning deviations of video frame by using the proposed robust bock-matching algorithm is 5.653 m, and the average positioning deviations of video frame by using the proposed precise polynomial-rectifying algorithm is 2.681 m. That is to say, positioning accuracy of video frame can be significantly increased by using the proposed precise polynomial-rectifying algorithm.
(3) The red homologous points located on roads and low-rise buildings have a higher positioning accuracy, while the red homologous points located on mountains and high-rise buildings have a lower positioning accuracy.
(4) Using region-DOM of high spatial resolution can significantly improve positioning accuracy of video frame, while extensive shadows that are similar to video frame's surface universal features will significantly decrease positioning accuracy of video frame.
(5) The proposed model can be applied in various areas, such as, town area, river area, high relief amplitude area and etc. And experiment results show that, the average positioning accuracy in town area and river area is gentle higher than that in high relief amplitude area, as high terrain relief will impose a negative influence on the distortion of imaging.
(6) By timekeeping in our program, the average time of extracting the block-matched-DOM is about 0.148 s, the average time of calculating the optimal estimation t is about 0.19 s, and the average time of calculating all pixels' precise geodetic coordinates in a video frame is about 0.076 s. That is to say, the total positioning time of a UAV's patrolling video frame is less than 1 s.

(7) The proposed methods can be also applied in the field of medical image registration, remote sensing image registration, visual navigation of other industries and etc. Subsequently, the current mathematical model can be optimized significantly by fusing with multi-source data, such as airborne LiDAR point cloud, and then can achieve a higher positioning accuracy and a broader application.

Conclusion

In order to realize real-time positioning of UAV's patrolling video sequence images, a visual positioning model is recommended, including a robust block-matching algorithm and a precise polynomial-rectifying algorithm.

First, the robust block-matching algorithm is constructed to realize roughly positioning of UAV's video patrolling video sequence images. The robust block-matching algorithm is divided into 5 steps, including scaling datum-DOM, block-matching roughly based on RGB, Block-matching roughly based on gradient magnitude, block-matching robustly, and extracting block-matched-DOM. Through the above 5 steps, the so-called block-matched-DOM can be obtained, and rough positioning of UAV's patrolling video sequence images can be realized by assigning geodetic coordinates of each pixel in block-matched-DOM to pixels at the same position in UAV's patrolling video sequence images.

Second, the precise polynomial-rectifying algorithm is constructed to realize accurate positioning of UAV's patrolling video sequence images. The precise polynomial-rectifying algorithm is divided into 5 steps, including constructing polynomials of video frame and that of block-matched-DOM, constructing differential-difference polynomials, constructing precise rectifying equations, constructing optimal estimation model, and calculating geodetic coordinates of interest points in video frame. Through the above 5 steps, the so-called accurate rectification parameters can be obtained, and accurate positioning of UAV's patrolling video sequence images can be realized by using accurate rectification parameters to calculate geodetic coordinates of each pixel in UAV's patrolling video sequence images.

Finally, all the proposed algorithms are verified by three practical experiments, and results indicate that the proposed robust block-matching algorithm can realize positioning of UAV's patrolling video sequence images with an average accuracy of 5 m, even if spatial resolution, surface specific features, illumination and topographic relief of region-DOM are greatly different from that of UAV's patrolling video sequence images. The proposed precise polynomial-rectifying algorithm can further improve positioning accuracy of UAV's patrolling video sequence images with an average accuracy of about 2.5 m. And calculation time of positioning a single UAV's patrolling video sequence image is less than 1 s.

Data availability

The data presented in this study are available on request from the corresponding author. The data are not publicly available due to another study related to this data is not yet publicly available.

Received: 29 June 2023; Accepted: 2 December 2023
Published online: 07 December 2023

References

1. Li, Z., Bian, S. & Qu, Y. Robust total least squares estimation of space intersection appropriate for multi-images [J]. *Acta Geodaet. Cartograph. Sin.* **46**(5), 593–604 (2017).
2. Li, D. *et al.* A multi-slice mapping technique based on oblique images [J]. *Survey. Map Bull.* **7**, 83–87 (2018).
3. Yang, B. *et al.* Approaches for exploration of improving multi-slice mapping via forwarding intersection based on images of UAV oblique photogrammetry [J]. *Comput. Electr. Eng.* **92**, 107135. https://doi.org/10.1016/j.compeleceng.2021.107135 (2021).
4. Zhang, Y. *et al.* Multistrip bundle block adjustment of ZY-3 satellite imagery by rigorous sensor model without ground control point [J]. *IEEE Geosci. Remote Sens. Lett.* **12**(4), 865–869. https://doi.org/10.1109/LGRS.2014.2365210 (2015).
5. Zhang, G. *et al.* Block adjustment for satellite imagery based on the strip constraint [J]. *IEEE Trans. Geosci. Remote Sens.* **53**(2), 933–941. https://doi.org/10.1109/TGRS.2014.2330738 (2015).
6. Lalak, M., Wierzbicki, D. & Kędzierski, M. Methodology of processing single-strip blocks of imagery with reduction and optimization number of ground control points in UAV photogrammetry. *Remote Sens.* **12**(20), 3336. https://doi.org/10.3390/rs12203336 (2020).
7. Cui, Y. *et al.* Precise calibration of binocular vision system used for vision measurement [J]. *Optic Exp.* **22**(8), 9134–9149. https://doi.org/10.1364/OE.22.009134 (2014).
8. Liu, Z. *et al.* High precision calibration for three-dimensional vision-guided robot system. *IEEE Trans. Ind. Electron.* **70**(1), 624–634. https://doi.org/10.1109/TIE.2022.3152026 (2023).
9. Abdel-Aziz, Y., Karara, H. & Hauck, M. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry [J]. *Photogram. Eng. Remote Sens.* **81**(2), 103–107. https://doi.org/10.14358/PERS.81.2.103 (2015).
10. Li, S. & Xu, C. A stable direct solution of perspective-three-point problem [J]. *Int. J. Pattern Recogn. Artif. Intell.* **25**(05), 627–642. https://doi.org/10.1142/S0218001411008774 (2011).
11. Wang, P. *et al.* An efficient solution to the perspective-three-point pose problem [J]. *Comput. Vis. Image Understand.* **166**, 81–87. https://doi.org/10.1016/j.cviu.2017.10.005 (2018).
12. Li, S., Xu, C. & Xie, M. A robust on solution to the perspective-n-point problem [J]. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(7), 1444–1450. https://doi.org/10.1109/TPAMI.2012.41 (2012).
13. Zimiao, Z. *et al.* A simple and precise calibration method for binocular vision [J]. *Meas. Sci. Technol.* **33**(6), 1. https://doi.org/10.1088/1361-6501/ac4ce5 (2022).
14. Lowe, D. G. Object recognition from local scale-invariant features[C]. In *Proceedings of the seventh IEEE international conference on computer vision*. IEEE, **2**, 1150–1157 (1999). https://doi.org/10.1109/ICCV.1999.790410.
15. Lowe, D. G. Distinctive image features from scale-invariant keypoints [J]. *Int. J. Comput. Vis.* **60**(2), 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94 (2004).
16. LI, X., Zheng, L., & Hu, Z. SIFT based automatic registration of remotely-sensed imagery [J]. *J. Remote Sens.* **10**(6), 885–892 (2006). http://www.en.cnki.com.cn/Article_en/CJFDTOTAL-YGXB200606008.htm.
17. Ma, W. *et al.* Remote sensing image registration with modified SIFT and enhanced feature matching [J]. *IEEE Geosci. Remote Sens. Lett.* **14**(1), 3–7. https://doi.org/10.1109/LGRS.2016.2600858 (2016).

18. Yang, Z. L., & Guo, B. L. Image mosaic based on SIFT[C]. In *2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*. IEEE, 1422–1425 (2008). https://doi.org/10.1109/IIH-MSP.2008.335.
19. Zeng, L. *et al.* Dynamic image mosaic via SIFT and dynamic programming [J]. *Mach. Vis. Appl.* **25**(5), 1271–1282. https://doi.org/10.1007/s00138-013-0551-8 (2014).
20. Ke, Y., & Sukthankar, R. PCA-SIFT: A more distinctive representation for local image descriptors[C]. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004. CVPR 2004. IEEE, 2: II-II. https://doi.org/10.1109/CVPR.2004.1315206 (2004).
21. Morel, J. M. & Yu, G. ASIFT: A new framework for fully affine invariant image comparison [J]. *SIAM J. Imaging Sci.* **2**(2), 438–469. https://doi.org/10.1137/080732730 (2009).
22. Neubeck, A., & Van Gool, L. Efficient non-maximum suppression[C]. In *18th International Conference on Pattern Recognition (ICPR'06)*. IEEE, 3: 850–855 (2006). https://doi.org/10.1109/ICPR.2006.479.
23. Bay, H., Tuytelaars, T., Gool, L. V. Surf: Speeded up robust features[C]. In *European conference on computer vision. Springer, Berlin, Heidelberg*, 404–417 (2006). https://doi.org/10.1007/11744023_32
24. Bay, H. *et al.* Speeded-up robust features (SURF) [J]. *Comput. Vis. Image Understand.* **110**(3), 346–359. https://doi.org/10.1016/j.cviu.2007.09.014 (2008).
25. Tareen, S. A. K., & Saleem, Z. A comparative analysis of sift, surf, kaze, akaze, orb, and brisk [C]. In *2018 International conference on computing, mathematics and engineering technologies (iCoMET). IEEE*, 2018. https://doi.org/10.1109/ICOMET.2018.8346440.
26. Rosten, E., & Drummond, T. Machine learning for high-speed corner detection[C]. In *European conference on computer vision* 430–443 (Springer, Berlin, Heidelberg, 2006). https://doi.org/10.1007/11744023_34.
27. Calonder, M., Lepetit, V., Strecha, C., *et al.* Brief: Binary robust independent elementary features[C]. In *European conference on computer vision. Springer, Berlin, Heidelberg*, 778–792 (2010). https://doi.org/10.1007/978-3-642-15561-1_56.
28. Rublee, E., Rabaud, V., Konolige, K., *et al.* ORB: An efficient alternative to SIFT or SURF[C]. In *2011 International conference on computer vision. Ieee*, 2564–2571 (2011). https://doi.org/10.1109/iccv.2011.6126544.
29. Leutenegger, S., Chli, M., & Siegwart, R. Y. BRISK: Binary robust invariant scalable keypoints[C]. In *2011 International conference on computer vision. Ieee*, 2548–2555. https://doi.org/10.1109/iccv.2011.6126542 (2011).
30. Alcantarilla, P.F., Bartoli, A., & Davison, A. J. KAZE features. In *European conference on computer vision. Springer, Berlin, Heidelberg*, 214–227 (2012).https://doi.org/10.1007/978-3-642-33783-3_16.
31. Ouyang, P., Yin, S., & Liu, L., *et al.* A fast and power-efficient hardware architecture for visual feature detection in affine-sift [J]. *IEEE Trans. Circ. Syst. I Regul. Papers* **65**(10), 3362–3375. https://doi.org/10.1109/TCSI.2018.2806447 (2018).
32. Tareen, S. A. K., & Saleem, Z. A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 1–10 (2018). https://doi.org/10.1109/ICOMET.2018.8346440.
33. Bansal, M., Kumar, M. & Kumar, M. 2D object recognition: A comparative analysis of SIFT, SURF and ORB feature descriptors [J]. *Multimed. Tools Appl.* **80**(12), 18839–18857. https://doi.org/10.1007/s11042-021-10646-0 (2021).
34. Lucas, B. D., & Kanade, T. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence* **81**, 121–130 (1981). https://www.researchgate.net/publication/215458777.
35. Horn, B. K. P. & Schunck, B. G. Determining optical flow [J]. *Artif. Intell.* **17**(1–3), 185–203. https://doi.org/10.1016/0004-3702(81)90024-2 (1981).
36. Brox, T., Bruhn, A., Papenberg, N., & Weickert, J. High Accuracy Optical Flow Estimation Based on a Theory for Warping. Computer Vision—ECCV 2004, 25–36 (2004). https://doi.org/10.1007/978-3-540-24673-2_3.
37. Tzovaras, D., Strintzis, M. G. & Sahinoglou, H. Evaluation of multiresolution block matching techniques for motion and disparity estimation [J]. *Signal Process. Image Commun.* **6**(1), 59–67. https://doi.org/10.1016/0923-5965(94)90046-9 (1994).
38. Hu, Y., Song, R., & Li, Y. Efficient coarse-to-fine patchmatch for large displacement optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5704–5712 (2016). https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Hu_Efficient_Coarse-To-Fine_PatchMatch_CVPR_2016_paper.html.
39. Revaud, J., Weinzaepfel, P., Harchaoui, Z., *et al.* Epicflow: Edge-preserving interpolation of correspondences for optical flow[C]. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1164–1172 (2015). https://doi.org/10.1109/cvpr.2015.7298720
40. Farnebäck, G. Polynomial expansion for orientation and motion estimation [D] (Linköping University Electronic Press, 2002).
41. Farnebäck, G. Two-frame motion estimation based on polynomial expansion[C]. In *Scandinavian conference on Image analysis* 363–370 (Springer, Berlin, Heidelberg, 2003). https://doi.org/10.1007/3-540-45103-X_50.
42. Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., & Brox, T. FlowNet: Learning Optical Flow with Convolutional Networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*. https://doi.org/10.1109/iccv.2015.316 (2015).
43. Fortun, D., Bouthemy, P. & Kervrann, C. Optical flow modeling and computation: A survey [J]. *Comput. Vis. Image Understand.* **134**, 1–21. https://doi.org/10.1016/j.cviu.2015.02.008 (2015).
44. Sotiras, A., Davatzikos, C. & Paragios, N. Deformable medical image registration: A survey [J]. *IEEE Trans. Med. Imaging* **32**(7), 1153–1190. https://doi.org/10.1109/TMI.2013.2265603 (2013).
45. Keeling, S. L. & Ring, W. Medical image registration and interpolation by optical flow with maximal rigidity [J]. *J. Math. Imaging Vis.* **23**(1), 47–65. https://doi.org/10.1007/s10851-005-4967-2 (2005).
46. Feng, R. *et al.* Region-by-region registration combining feature-based and optical flow methods for remote sensing images [J]. *Remote Sens.* **13**(8), 1475. https://doi.org/10.3390/rs13081475 (2021).
47. Chen, Q. *et al.* Horticultural image feature matching algorithm based on improved ORB and LK optical flow [J]. *Remote Sens.* **14**(18), 4465. https://doi.org/10.3390/RS14184465 (2022).

Acknowledgements

We thank the editors for reviewing the manuscript, and the anonymous reviewers for providing suggestions that greatly improved the quality of the work.

Author contributions

H.L. proposed the basic idea of this paper; W.F. conducted all the experiments and wrote the main manuscript text; D.W. provided experimental data. All authors reviewed the manuscript.

Funding

This research was supported by China Postdoctoral Science Foundation (2021M701373).

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to W.F.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.