Remote Sens. **2015**, *7*, 12606-12634; doi:10.3390/rs71012606

OPEN ACCESS

remote sensing

ISSN 2072-4292

www.mdpi.com/journal/remotesensing

Article

Metadata-Assisted Global Motion Estimation for Medium-Altitude Unmanned Aerial Vehicle Video Applications

Hongguang Li [1], Xinjun Li [2], Wenrui Ding [1,*] and Yuqing Huang [3]

[1] Research Institute of Unmanned Aerial Vehicle, Beihang University, Beijing 100191, China; E-Mail: lihongguang@buaa.edu.cn

[2] Asia-Pacific Space Cooperation Organization, Beijing 100191, China; E-Mail: lixinjun@apsco.int

[3] School of Electronic and Information Engineering, Beihang University, Beijing 100191, China; E-Mail: mickyqing@126.com

***** Author to whom correspondence should be addressed; E-Mail:dwr_buaa@163.com;
Tel.: +86-010-8233-9906; Fax: +86-010-8231-7755.

Academic Editors: Gonzalo Pajares Martinsanz and Prasad S. Thenkabail

Received: 5 July 2015 / Accepted: 17 September 2015 / Published: 25 September 2015

Abstract: Global motion estimation (GME) is a key technology in unmanned aerial vehicle remote sensing (UAVRS). However, when a UAV's motion and behavior change significantly or the image information is not rich, traditional image-based methods for GME often perform poorly. Introducing bottom metadata can improve precision in a large-scale motion condition and reduce the dependence on unreliable image information. GME is divided into coarse and residual GME through coordinate transformation and based on the study hypotheses. In coarse GME, an auxiliary image is built to convert image matching from a wide baseline condition to a narrow baseline one. In residual GME, a novel information and contrast feature detection algorithm is proposed for big-block matching to maximize the use of reliable image information and ensure that the contents of interest are well estimated. Additionally, an image motion monitor is designed to select the appropriate processing strategy by monitoring the motion scales of translation, rotation, and zoom. A medium-altitude UAV is employed to collect three types of large-scale motion datasets. Peak signal to noise ratio (PSNR) and motion scale are computed. This study's result is encouraging and applicable to other medium- or high-altitude UAVs with a similar system structure.

Keywords: global motion estimation; unmanned aerial vehicle; metadata; remote sensing

1. Introduction

1.1. Background

1.1.1. Unmanned Aerial Vehicle Remote Sensing

Unmanned aerial vehicle remote sensing (UAVRS) as a means of aerospace remote sensing is a strong complement to satellite and aerial remote sensing of manned aircrafts. With the exponential development of the sensors and instruments to be installed onboard, UAVRS applications with new potential are continuously increasing [1]. Owing to its real-time video transmission, detection of high-risk areas, low cost, high resolution, flexibility, and other advantages, UAVRS has been widely utilized in military and civilian areas in the past decade [2,3]. Equipped with various imaging equipment of visible light, infrared, and synthetic aperture radar to obtain remote sensing images, unmanned aerial vehicles (UAVs) utilize aerial and ground control systems for automatic video shooting, data compression and transmission, video preprocessing and post-processing, and other functions and can be utilized in many applications, such as national environmental protection [4], mineral resource exploration, land use survey, marine environmental monitoring [5], water resource development, crop growth monitoring and assessment [6], forest protection and monitoring [7], natural disaster monitoring and evaluation [8], target surveillance [9], and digital Earth.

Worldwide applications have contributed to the development of several scientific studies on remote sensing [10]. As an auxiliary means that employs indispensable visual information, video processing has been widely utilized in guidance, navigation, and control [11] and is eliciting an increasing amount of attention because of its special characteristics, namely, moving imaging, long-distance transmission, and complex atmosphere. Mai *et al.* [12] summarized five characteristics of small UAV airborne videos. Brockers [13] and Kanade *et al.* [14] studied computer vision for micro air vehicles. These studies have contributed significantly to the UAV vision in the field of UAVRS.

Compared with popular low-altitude UAVs, medium-altitude UAVs play a more important role in UAVRS because of their longer endurance, higher altitude, and more powerful imaging sensor loading capability. However, published studies on vision technologies and applications for medium- or high-altitude UAVs remain scarce.

1.1.2. Utility of Global Motion Estimation in UAVRS

Global motion estimation (GME) is the process of determining the motion of the camera and is a key technology in the UAV vision for remote sensing data acquisition and various applications. Data acquisition is the process of imaging, compression, and transmission; it provides the image data source for remote sensing applications. During data acquisition, GME could be utilized to estimate the camera's motion in video stabilization [15] to obtain a stable and smooth video. It can also be utilized to calculate the redundancy information between frames in video encoding [16] to achieve high-resolution image compression and transmission despite the limited bandwidth of the data link.

GME has important contributions to the other four classes of video processing as shown in Table 1, namely, target detection and tracking [17], video shot segmentation and retrieval [18], super-resolution reconstruction [19], and structure from motion [20]. These four classes have important applications in remote sensing. In target detection and tracking, the global motion presents the background, and the different local motions indicate the moving targets. Several remote sensing applications for monitoring and surveillance are performed with this fact. In video shot segmentation and retrieval, accurate global motion is a reliable indicator to extract several important sequences or images from a remote sensing database. In super-resolution reconstruction, robust GME is utilized to complete image registration, which is highly useful for forest and agriculture applications. Another important application is the popular structure from motion for 3D mapping, which is based on camera motion estimation.

Table 1. Uses of GME in UAVRS.

Key Technology	Video Processing of UAV Vision	Applications in UAVRS
GME	Video encoding	Data acquisition
	Video stabilization	Data acquisition
	Target detection and tracking	Target monitoring and surveillance
	Video shot segmentation and retrieval	Data retrieval and production from a video database
	Super-resolution reconstruction	Crop and forest monitoring, change detection
	Structure from motion	3D reconstruction and 3D mapping for disaster areas

1.1.3. Problems in GME

In outdoor video applications, conventional GME based on image block matching has nearly evolved to maturity. However, in a UAV environment, performance is poor when the UAV's motion or behavior changes significantly. The cause of this problem is that conventional image-based GME is only applicable to the narrow baseline condition between frames and not to the wide baseline condition when a large-scale motion is prevalent in the video. Owing to the combined motion of the vehicle and camera, the image block moves out of the researching window or image distortion results in the same content in different sizes and shapes; either of these conditions results in image block matching failure. Several other methods utilize some prior information from special sensors to deal with this problem. Although these methods provide improved results, they cannot be applied to other UAVs with different structures. Furthermore, when images have minimal information (e.g., deserts, rivers, and other special landforms), image-based methods become completely unreliable.

The problems in GME that need to be solved are summarized below; these problems provide future research directions for our work.

(1) How to improve precision under a large-scale motion condition?
(2) How to reduce the dependence on image information to adapt to several special landforms?
(3) How to enhance adaptability to different UAVs?

1.2. Related Work

Traditional GME methods can generally be divided into pixel based [21], feature based [22], and vector based [23] according to different analysis objects. The performances of these methods were evaluated in [24,25]. To achieve high precision, a large number of pixels, features, and vectors are involved in the computation.

In recent years, several excellent GME algorithms [26–30] have been developed in an attempt to achieve both precision and computation. Okade *et al.* [26] proposed the use of discrete wavelet transform in the block motion vector field to estimate the global motion parameters in the compressed domain. A key assumption was that the LL sub-band provides the average motion, which is predominantly due to the background (camera) motion. The algorithm proposed by Yoo *et al.* [27] independently conducted motion estimations in both forward and backward directions and selected the more reliable vector between forward and backward motion vectors by evaluating motion vector reliability from the perspective of the interpolated frame. In [28], a new class of prediction algorithms based on region prediction was introduced. This new class of algorithms can be applied in conventional fixed-pattern algorithms to predict the region in which the best matched block is located, thereby increasing the speed of the algorithm. Sung and Chung [29] presented a robust real-time camera motion estimation method that employs a fast detector with a multi-scale descriptor; the method entails minimal computation but exhibits high precision. Krutz *et al.* [30] proposed the concept of global motion temporal filtering for the emerging video coding standard HEVC. All of these methods can be considered image-based GME.

Aside from image-based GME, several scholars have employed camera sensor parameters as auxiliary information to solve the problem. This class of GME can be referred to as sensor-assisted GME. In [31], a novel approach called sensor-assisted motion estimation was developed to estimate the linear displacement of a mobile device through the use of built-in sensors. A built-in three-axis accelerometer was utilized to determine linear displacement on the X-, Y-, and Z-axes. Another method called sensor-assisted video encoding (SaVE) was introduced in [32] to reduce the computational complexity of video encoding. The method calculates the movement of a camera (on mobile devices) and then infers the global motion in a video. SaVE utilizes readings from a single accelerometer attached to the video camera to compute the vertical angle. For the horizontal angle, either absolute angle readings from a single digital compass or a pair of accelerometers are utilized. Wang *et al.* introduced a sensor-assisted GME method for H.264/AVC video encoding [33]. By leveraging location (GPS) and digital compass data, the method exploits the geographical sensor information to detect transitions between two video sub-shots based on the variations in both camera location and shooting direction. Strelow *et al.* [34] introduced an algorithm that computes optimal vehicle motion estimated by simultaneously considering all measurements from the camera, rate gyro, and accelerometer. Sensor-assisted GME takes advantage of the position and behavior information of sensors, such as accelerometers, GPS devices, digital compasses, and rate gyros. With information from sensors, good performance is achieved to some extent. The success of these methods lies in that they upgrade the GME problem to the system level and then exploits useful information from the system.

Owing to the characteristics of moving imaging and dual-platform (vehicle and camera) behavior change, UAV video GME becomes a system issue that cannot be simply solved by image processing algorithms. For example, in the compression domain, MPEG-4 or H.264 generally exhibits low effectiveness when ground speed and dual-platform behavior change significantly. In this case, translation between frames is larger than the search radius of the block matching algorithm, or rotation and zoom motions create image distortion; either one of these conditions results in an error in block matching. The compression ratio becomes large with the limited bandwidth of the data link, leading to data loss during transmission. The worst effect is the possible generation of mosaics in the ground video receiver, thereby seriously affecting video applications, such as target detection and recognition. The key to solving this problem is to develop a fast and accurate GME method.

To achieve effective GME for UAV/aerial video applications, several scholars worked to solve the problem from the system level, similar to sensor-assisted GME. Rodr ǵuez *et al.* [35] presented an efficient algorithm to solve the motion estimation problem. The algorithm requires minimal computation and is thus suitable for implementation in a mini-UAV. Computation is reduced by using prior knowledge on camera locations (from available mini-UAV sensor data) and projective geometry of the camera. Based on the algorithm in [35], Gong *et al.* [36] proposed a low-complexity image-sequence compressing algorithm for UAVs. Bhaskaranand *et al.* [37] designed a video encoding scheme suitable for applications in which encoder complexity needs to be low (e.g., UAV video surveillance). Encountering the same problem, Angelino *et al.* proposed a novel motion estimation scheme that employs the global motion information provided by the onboard navigation system [38,39]. The homography between two images was utilized to initialize the block matching algorithm, allowing for a more robust motion estimation and a smaller search window to reduce complexity. Bhaskaranand and Gibson proposed a low-complexity encoder [40] with no block level motion estimation, global motion compensated prediction, and spectral entropy-based coefficient selection and bit allocation.

These studies promoted the development of UAVRS technologies and applications under certain conditions. However, they also have several limitations (indicated below).

(1) The research objects were predominantly small, and low-altitude UAVs that have different structures were employed. This condition leads to poor expansibility of the method.
(2) The information used was not the bottom data measured from the UAV system, and some information was assumed to be known. Thus, the process of GME was not completed from the bottom level.
(3) The motion of the dual platform was often assumed to be smooth and stable, which confines GME to a narrow baseline condition. However, even the same contents (e.g., house, bridges) of two adjacent frames differ in geometric features (shape and size), location, and orientation when the vehicle’s translation or the dual platform’s behavior changes considerably.

1.3. Present Work

The current work aims to solve the three problems in GME mentioned in Section 1.1.3. First, according to the theory of coordinate transformation, GME is converted from a wide baseline condition to a narrow baseline one to improve the precision of GME under a large-scale motion condition. Second, bottom metadata are utilized to reduce the dependence on image information and derive an information and contrast feature to maximize the use of reliable image information. Third, a medium-altitude UAV with a common structure is investigated. The proposed scheme can also be applied to medium- or high-altitude UAVs with a similar system structure.

2. Methodology

2.1. Scheme of Metadata-Assisted GME

2.1.1. Study Hypotheses

A novel metadata-assisted GME method (MaGME) for medium-altitude UAV video applications was developed. According to the imaging characteristics of medium-altitude UAVs, three hypotheses were established.

(1) Central projection model hypothesis

The camera imaging model conforms to the central projection model. On this basis, MaGME is applicable to CCD and infrared video. The central projection model is a common image model that is similar to the pinhole camera model assumed in [41]. It is utilized to solve collinear equations in coordinate transformation. However, selecting the wrong type of camera would lead to an erroneous result.

(2) Field depth consistency hypothesis

Terrain fluctuations and man-made buildings can be ignored relative to the long imaging distance. Thus, all pixels of an image are assumed to be on the same plane. Field depth consistency is a basic hypothesis in most studies on GME [35–39]. The lower the terrain fluctuations and buildings are, the more accurate the result of GME is.

(3) Content of interest hypothesis

Users are more interested in an area with strong contrast and rich information (e.g., houses, overpasses) than in an area with hue/brightness consistency and minimal information (e.g., lakes, wheat fields, grasslands). The image information in the contents of interest is rich and reliable. The purpose of the content of interest hypothesis is to indicate which regions of the image contain valuable information. Based on this hypothesis, this study attempts to detect meaningful blocks and discard worthless blocks from images in block matching.

2.1.2. Metadata

To improve the precision of GME under large-scale motion conditions and reduce the dependence on image information, full information mining was implemented to build a model from bottom metadata to global motion. A medium-altitude UAV with a CCD camera was employed. Equipped with GPS and INS, the UAV can measure its position and behavior by itself. The two-DOF camera mounted on the front belly can complete attitude measurement independently. The metadata associated with image global motion are shown in Table 2.

2.1.3. Workflow of MaGME

A GME method based on the theory of coordinate transformation was designed with the metadata provided above. The method completely relies on bottom metadata that UAV systems produce initially rather than on known data of camera position, orientation, and projective geometry provided in [35]. In addition, the camera calibration process in [35,38,39] was not considered in the current study because the camera mounted on a medium-altitude UAV usually implements this process before image compression.

As shown in the light blue box of Figure 1, MaGME performs in both wide and narrow baseline conditions, as indicated by the image motion monitor. To improve performance under a wide baseline condition, the following two steps were conducted. First, based on the theory of coordinate transformation, the coarse GME between image $F(t)$ and auxiliary image $F'(t)$ was computed using metadata (shown as M_1). Second, auxiliary image $F'(t)$ was built to convert the wide baseline condition to a narrow baseline one; then, the residual GME was solved by big-block matching method (shown as M_2). These two steps were combined to obtain the final global motion. The narrow baseline condition requires only big-block matching, the process of which is similar to the residual GME under the wide baseline condition.

Table 2. Metadata list.

Index	Name	Notation	Description
1	Longitude	*Lng*	Measured by GPS, unit: degree
2	Latitude	*Lat*	Measured by GPS, unit: degree
3	Altitude	*Alt*	Measured by altimeter, unit: meter
4	Terrain height	*Ter*	Obtained from GIS, unit: meter
5	Vehicle heading	*H*	Angle between the UAV's nose and the North measured by INS, unit: degree
6	Vehicle roll	*R*	Measured by INS, unit: degree
7	Vehicle pitch	*P*	Measured by INS, unit: degree
8–10	Camera installation Translation	t_{CX}、t_{CY}、t_{CZ}	Translation from camera to GPS on X-, Y-, and Z-axes, unit: meter
11	Camera pan	*pan*	Angle between the camera's optical axis and the UAV's nose, unit: degree
12	Camera tilt	*tilt*	Angle between the camera's optical axis and the UAV body plane, unit: degree
13	Resolution	*Row*Col*	Row: image row, Col: image column
14	Focal length	*f*	Unit: meter
15	Pixel size	*u*	Size of each pixel, unit: meter

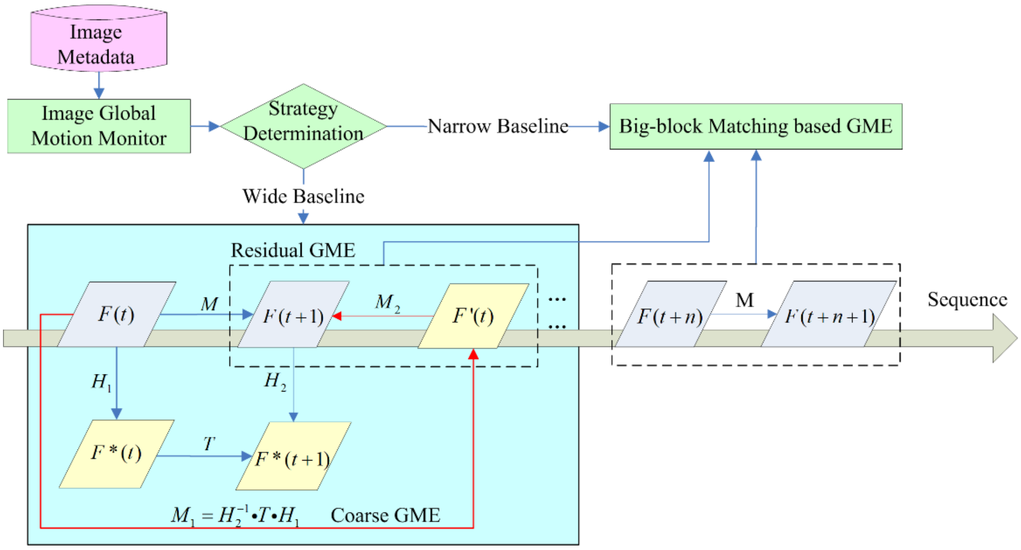

Figure 1. Workflow of MaGME.

Given that UAV video global motion suffers from the combined motion of the vehicle and camera, complex transformations, including translation, rotation, zoom, and shear, widely exist between two frames or between the image and real scene. This relationship needs to be represented by a perspective projection model [42]. Thus, video GME can be converted into a problem of solving the perspective projection transformation matrix between frames. A perspective projection model is described by eight parameters (m_0, m_1, m_2, m_3, m_4, m_5, m_6, m_7). At time *t*, one point of the frame is recorded as $P(x_t, y_t)$. At time *t+1*, the point is recorded as $P(x_{t+1}, y_{t+1})$. The relationship is shown in Equation (1). The purpose of video GME is to compute successive perspective projection models.

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ 1 \end{bmatrix} = \begin{bmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & 1 \end{bmatrix} \begin{bmatrix} x_t \\ y_t \\ 1 \end{bmatrix} \tag{1}$$

An image motion monitor was designed to determine whether the global motion between frames is under a wide baseline condition for guide strategy determination. When the monitor shows a large-scale motion between frames, coarse and residual GME need to be performed together. Otherwise, only residual GME based on big-block matching is required. This procedure makes the proposed method feasible under all motion conditions of the UAV system.

Under a wide baseline condition, the ground coordinate system (GCS) is introduced as an auxiliary coordinate system. Based on the central projection model hypothesis, frames $F(t)$ and $F(t+1)$ in the image coordinate system (ICS) are projected onto the ground plane in GCS. Corrected image planes $F^*(t)$ and $F^*(t+1)$ are then obtained with Equations (2) and (3).

$$F^*(t) = H_1 \bullet F(t), \tag{2}$$

$$F^*(t+1) = H_2 \bullet F(t+1), \tag{3}$$

where H_1 and H_2 denote the perspective projection transformations from the image planes in ICS to their projection image planes in GCS at time t and $t+1$, respectively. After transformation from ICS to GCS, the main motion between frames can be represented by translation T, which can be achieved according to the positional relationship of $F^*(t)$ and $F^*(t+1)$ in GCS. The transformation between $F(t)$ and $F(t+1)$ can be expressed as Equation (4).

$$F(t+1) = H_2^{-1} \bullet T \bullet H_1 \bullet F(t) \tag{4}$$

Based entirely on metadata, coarse global motion M_1 between two images is obtained.

$$M_1 = H_2^{-1} \bullet T \bullet H_1 \tag{5}$$

M_1 multiplied by image $F(t)$ results in compensated image $F'(t)$.

$$F'(t) = M_1 \bullet F(t) \tag{6}$$

If both the metadata and calculation are absolutely accurate, M_1 is the global motion from frame $F(t)$ to frame $F(t+1)$, and $F'(t)$ is similar to $F(t+1)$. However, affected by equipment installation and sensor measurement errors, M_1 cannot represent the real global motion. In fact, a residual global motion exists between $F'(t)$ and $F(t+1)$.

Although $F'(t)$ is not similar to $F(t+1)$, experiments show that much of translation, rotation, zoom, and shear were eliminated. With auxiliary image $F'(t)$, the wide baseline problem between $F(t)$ and $F(t+1)$ can be converted to a narrow baseline one between $F'(t)$ and $F(t+1)$. The residual global motion between $F'(t)$ and $F(t+1)$ is denoted as M_2.

Finally, as the core of MaGME, global motion M is expressed as Equation (7).

$$M = M_2 \bullet M_1 = M_2 \bullet H_2^{-1} \bullet T \bullet H_1 \tag{7}$$

2.2. Coordinate Transformation

Based on coordinate transformation theory and the central projection model, the image plane in ICS was converted to the ground plane in GCS by utilizing the metadata. After the transformation, the complex projective projection transformation between two image planes can be described by a simple translation transformation in GCS.

Coordinate transformation from ICS to GCS follows the order "Image Coordinate System → Camera Coordinate System → Plane Coordinate System → North-East-Up Coordinate System → Ground Coordinate System", as shown in Figure 2.

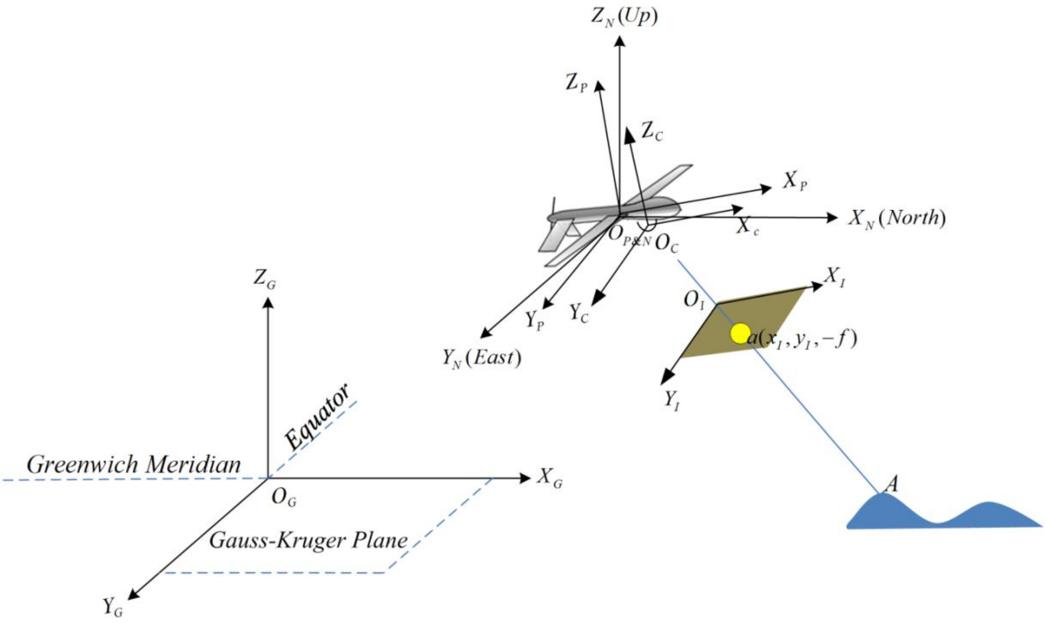

Figure 2. Five coordinate systems. GCS utilizes the Gauss–Kruger coordinate on the XOY plane and altitude on the Z-axis.

Owing to different origin definitions, translation T_I^C exists between ICS and the camera coordinate system (CCS). The origins of CCS and the plane coordinate system (PCS) are not on the same point because of camera installation. Thus, translation T_C^P exists between CCS and PCS. In addition, the camera has two DOFs of pan and tilt relative to the vehicle. This scenario produces perspective projection transformation M_C^P between CCS and PCS. Having the same origin as the North–East–Up coordinate system (NCS), PCS has three DOFs of heading, pitch, and roll relative to NCS. The transformation from PCS to NCS can be represented by perspective projection transformation M_P^N. NCS and GCS are parallel to translation T_N^G. The three values of T_N^G are equal to the projection values of the origin of NCS on the three axes of GCS.

Accordingly, the transformation from image plane $F_I(x_I,y_I,z_I)$ in ICS to image plane $F_N(x_N,y_N,z_N)$in NCS can be expressed as Equation (8).

$$F_N = M_P^N \bullet M_C^P \bullet T_C^P \bullet T_I^C \bullet F_I \tag{8}$$

GCS employs the Gauss–Kruger coordinate on the XOY plane and altitude on the Z-axis. The origin of GCS is the intersection of the Greenwich Meridian and the Equator. Parallel to NCS, the

X-axis pointing to the north, the Y-axis pointing to the east, and the Z-axis pointing upward, a left-handed coordinate system is formed, as shown in Figure 2.

A central projection model needs to be established to solve collinear equations according to the central projection model hypothesis.

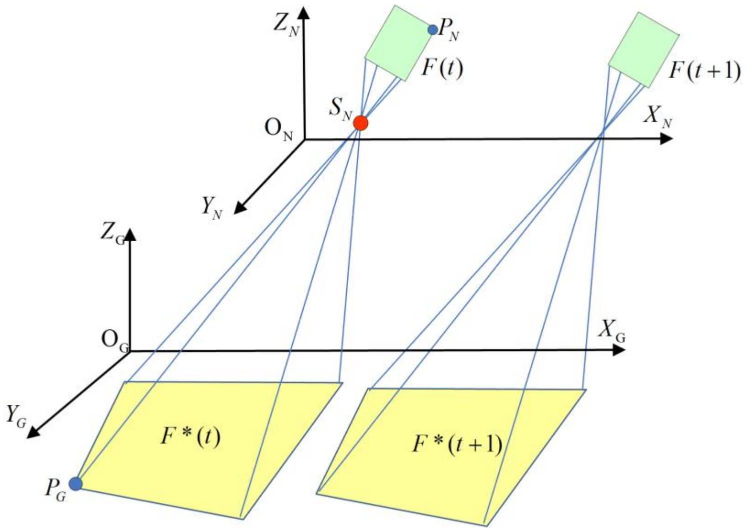

Figure 3. Collinear relationship between NCS and GSC.

As shown in Figure 3, during the transformation from the NCS to the GCS, the ground point P_G, the center of projection S_N and the image point P_N are in a same line. In the CCS, photography center is $S_I(0,0,-f)$, and f is the focal length of the camera. The photography center $S_N(x_N^S, y_N^S, z_N^S)$ in the NCS and $S_G(x_G^S, y_G^S, z_G^S)$ in the GCS can be obtained by Equation (9).

$$\begin{cases} S_N = M_P^N \bullet M_C^P \bullet T_C^P \bullet T_I^C \bullet S_I \\ S_G = T_N^G \bullet M_P^N \bullet M_C^P \bullet T_C^P \bullet T_I^C \bullet S_I \end{cases} \tag{9}$$

Equation (10) is obtained according to the theory of similar triangles. $F_N(x_N,y_N,z_N)$ can be calculated with Equation (8), and $F_G(x_G,y_G,z_G)$ can be obtained with Equation (11).

$$\begin{cases} x_G - x_G^s = \lambda \bullet (x_N - x_N^s) \\ y_G - y_G^s = \lambda \bullet (y_N - y_N^s) \\ z_G - z_G^s = \lambda \bullet (z_N - z_N^s) \end{cases} \tag{10}$$

$$\begin{cases} x_G = \dfrac{x_N - x_N^s}{z_N - z_N^s}(z_G - z_G^s) + x_G^s \\ y_G = \dfrac{y_N - y_N^s}{z_N - z_N^s}(z_G - z_G^s) + y_G^s \end{cases} \tag{11}$$

In Equation (11), z_G is the height of the object point. According to the field depth consistency hypothesis, the entire image is regarded as a plane. z_G is a known value.

T_I^C, T_C^P, M_C^P, M_P^N, and T_N^G can be expressed by 4 ×4 matrices. Pixel plane F_I is expressed as $(x_I,y_I,-f,1)^T$ with a homogeneous coordinate.

$$T_I^C = \begin{bmatrix} 1 & 0 & 0 & t_I^x \\ 0 & 1 & 0 & t_I^y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

$$T_C^P = \begin{bmatrix} 1 & 0 & 0 & t_C^x \\ 0 & 1 & 0 & t_C^y \\ 0 & 0 & 1 & t_C^z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13}$$

$$M_C^P = R_Z(pan)R_Y(tilt) = \begin{bmatrix} \cos(\text{pan}) & -\sin(\text{pan}) & 0 & 0 \\ \sin(\text{pan}) & \cos(\text{pan}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\text{tilt}) & 0 & \sin(\text{tilt}) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\text{tilt}) & 0 & \cos(\text{tilt}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{14}$$

$$M_P^N = R_Z(h)\bullet R_Y(p)\bullet R_X(r) = \begin{bmatrix} \cos(h) & -\sin(h) & 0 & 0 \\ \sin(h) & \cos(h) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(p) & 0 & \sin(p) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(p) & 0 & \cos(p) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(r) & -\sin(r) & 0 \\ 0 & \sin(r) & \cos(r) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{15}$$

$$T_N^G = \begin{bmatrix} 1 & 0 & 0 & t_N^x \\ 0 & 1 & 0 & t_N^y \\ 0 & 0 & 1 & t_N^z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{16}$$

The translation unit in Equations (12), (13), and (16) is meter, and the rotation unit in Equations (14) and (15) is degree. Each transformation is described below in Table 3.

The general transformation from image plane F_I in ICS to ground plane F_G in GCS is described above. However, owing to the different coordinate system forms in UAV systems, some adjustments (e.g., coordinate axis direction or rotation direction) are required in specific applications.

Table 3. Transformation description.

Transformation	Description
T_I^C	Translation from ICS to CCS; t_I^x and t_I^y are equal to the half of the physical width and height of the imaging plane in value
T_C^P	Translation from CCS to PCS; t_C^x, t_C^y, and t_C^z are the three translations on X-, Y-, and Z-axes
M_C^P	Projective projection from CCS to PCS; two DOFs: pan and tilt
M_P^N	Projective projection from PCS to NCS; *h*: heading, *p*: pitch, *r*: roll
T_N^G	Translation from NCS to GCS; t_N^x, t_N^y, and t_N^z are the three translations on X-, Y-, and Z-axes

According to this process, images $F(t)$ and $F(t+1)$ can be projected to the ground plane in GCS; then, two image planes $F^*(t)$ and $F^*(t+1)$ are achieved. The transformation from image plane F to ground plane F^* is established by $F^*(t)=H_1\,F(t)$ in Equation (2) and $F^*(t+1)=H_2\,F(t+1)$ in Equation (3). By substituting the four corners (x_t^n, y_t^n; n=1,2,3,4) of plane $F(t)$ and the four corners (x_t^{*n}, y_t^{*n}; n=1,2,3,4) of plane $F^*(t)$ into Equation (17), $H_1(h_0, h_1, h_2, h_3, h_4, h_5, h_6, h_7,)$ can be solved. By using the same method, H_2 can be obtained.

$$\begin{cases} x_t^{*n} = \dfrac{h_0 x_t^n + h_1 y_t^n + h_2}{h_6 x_t^n + h_7 y_t^n + 1} \\ y_t^{*n} = \dfrac{h_3 x_t^n + h_4 y_t^n + h_5}{h_6 x_t^n + h_7 y_t^n + 1} \end{cases} \tag{17}$$

2.3. Coarse GME

After coordinate transformation, the rotation or zoom scale between $F^*(t)$ and $F^*(t+1)$ becomes consistent in GCS. Without considering the precision of metadata and calculation, the transformation is absolutely accurate. Only translation T exists between $F^*(t)$ and $F^*(t+1)$. It can be estimated according to the relationship of two planes in GCS.

As two image planes have already been corrected to GCS, the same content points of the two images in GCS could have the same coordinate. Leveraging this fact, translation T from $F^*(t)$ to $F^*(t+1)$ can be estimated by the position difference of the same point on two planes. As shown in Figure 4, the center of the overlapping area of $F^*(t)$ and $F^*(t+1)$ is point P, which is denoted by (x_t^{*p}, y_t^{*p}) in $F^*(t)$ and ($x_{t+1}^{*p}, y_{t+1}^{*p}$) in $F^*(t+1)$. Then, translation $T(dx^*, dy^*)$ from $F^*(t)$ to $F^*(t+1)$ can be expressed as Equation (18).

$$\begin{cases} dx^* = x_{t+1}^{*p} - x_t^{*p} \\ dy^* = y_{t+1}^{*p} - y_t^{*p} \end{cases} \tag{18}$$

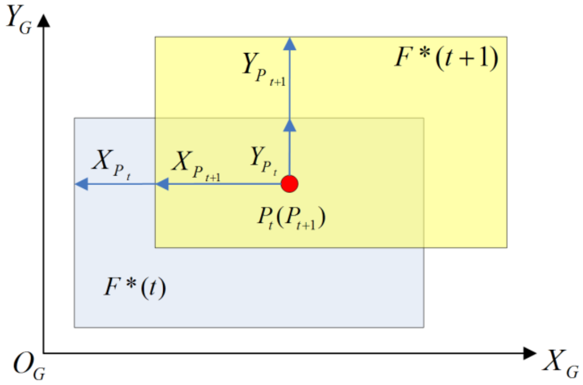

Figure 4. Translation from $F^*(t)$ to $F^*(t+1)$ in GCS.

Consequently, the transformation from $F(t)$ to $F(t+1)$ can be expressed as $F(t+1) = H_2^{-1} \cdot T \cdot H_1 \cdot F(t)$. $M_1 = H_2^{-1} \cdot T \cdot H_1$ represents the homography matrix between two images. The solution of M_1 is entirely based on the metadata and does not involve any image information. Accordingly, the computation is less than that in image-based methods. However, two issues need to be noted.

(1) Given that metadata precision is affected by equipment installation and the measured parameter, M_1 is not the real global motion between frames. The low precision of vehicle and camera parameters

leads to the poor capability of M_1 to represent global motion under the condition of large-scale rotation or zoom motion. The low precision of position parameters leads to the poor capability of M_1 to represent global motion under the condition of large-scale translation motion.

(2) Under a wide baseline condition, the initialization of matching windows position and reduction of searching computation mentioned in [38,39] cannot be achieved by M_1. When the behaviors of the dual platform and focal length change considerably, the same content in the two images would exhibit large distortion, which causes the block matching method to fail.

Owing to the high performance of INS, GPS, the camera, and other equipment mounted on the large medium-altitude UAV, M_1 can eliminate much of the distortion and translation between $F'(t)$ and $F(t+1)$. The matching of $F'(t)$ and $F(t+1)$ is under a narrow baseline condition. Utilizing $F'(t)$ as an auxiliary image, GME M between $F'(t)$ and $F(t+1)$ under a wide baseline condition can be converted to coarse GME M_1 between $F(t)$ and $F'(t)$; residual GME M_2 between $F'(t)$ and $F(t+1)$ is under a narrow baseline condition.

$$M = M_2 \bullet M_1 \tag{19}$$

2.4. Residual GME

2.4.1. Information and Contrast Feature

To maximize the use of reliable information in the image and ensure the precision of GME in the contents of interest, an information and contrast feature (I&C feature) for big-block selection was developed based on the content of the hypothesis of interest.

$$\begin{cases} I\&C \quad Feature = \kappa\lambda\sum_{\Omega}(I-\bar{I})^2 P_I \\ \lambda = \dfrac{1}{256}\sum_{I=0}^{255} Q(I) \end{cases} \tag{20}$$

where Ω is the image block window, I is the gray value of point (x, y) in Ω, $\bar{I}$ is the average of gray value of the image block, P_I is the probability of I in the image block, λ is the amount of gray information of the image block, and $Q(I)$ indicates the presence or absence of gray I in the image block. When I is present, $Q(I) = 1$; otherwise, $Q(I) = 0$. κ is the normalization factor. To facilitate the observation and calculations, all I&C features of blocks are normalized to (0, 255).

According to Equation (20), the region with rich information and high contrast has a high I&C feature value. The content of interest hypothesis indicates that people show more interest in regions with strong contrast and rich information than in open fields with consistent hue or brightness and minimal information; the image information in the contents of interest is reliable. Consequently, the selected image blocks should be located in the content of interest to ensure accurate registration of important image content and avoid mismatch in the region with minimal information and low contrast. Thus, the I&C feature can be utilized as an indicator in image block selection. Additionally, it can reduce the work required to eliminate many mismatching outpoints in [43,44]. As to the number of image blocks, only one big block can solve residual translation motion, and at least four big blocks would be sufficient for residual perspective projection motion estimation, unlike in conventional methods wherein each image block is matched.

2.4.2. Residual GME Based on Big-block Matching

The experiments indicate that the residual global motion is predominantly translation. In consideration of both computation and precision, M_2 can be solved by two models, namely, translation transformation and perspective projection transformation. Motion estimation between $F'(t)$ and $F(t+1)$ can use the conventional image block matching method. To maximize the use of the typical contents (house, tree) in the image, a big block would be appropriate.

As shown in Figure 5, an image was divided into 16 × 16 big blocks to maximize the use of several typical contents (house, tree). Through visual judgment, the yellow-marked region has low contrast and minimal information. The selection of image blocks should not be in this region.

The blocks whose I&C features are greater than a certain threshold value in auxiliary image $F'(t)$ were utilized to search best matching blocks in $F(t+1)$. Three-step search (TSS) [45] was employed as the search method. When the *MAD* between two matching blocks is less than a certain threshold or the number of matching exceeds the maximum value, the location is accepted as the best matching position.

$$MAD = \frac{1}{M \bullet N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} | C_{ij} - R_{ij} |, \tag{21}$$

where M and N are the column and row of the block and C_{ij} and R_{ij} are the pixels being compared in the current and reference blocks, respectively.

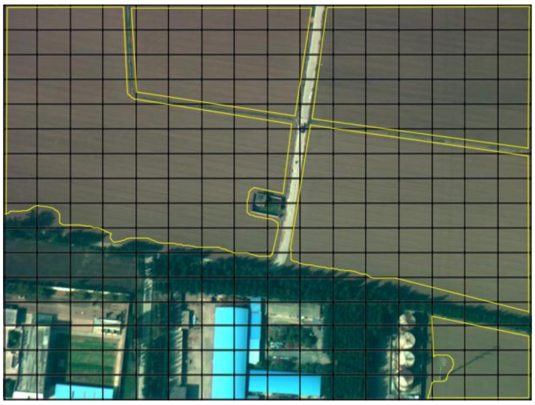

Figure 5. Image divided into big blocks. Several typical contents (house, tree) are in big blocks, which is useful in improving matching precision.

The center of several image blocks in auxiliary image $F'(t)$ is recorded as (x'_t, y'_t), and (x_{t+1}, y_{t+1}) is the corresponding point in $F(t+1)$. The motion vector calculated with block matching method is recorded as $V(d_x, d_y; t)$. The motion model from $F'(t)$ to $F(t+1)$ can be expressed as translation in Equation (22) or perspective projection transformation in Equation (23).

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x'_t \\ y'_t \\ 1 \end{bmatrix} \tag{22}$$

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ 1 \end{bmatrix} = \begin{bmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & 1 \end{bmatrix} \begin{bmatrix} x_t' \\ y_t' \\ 1 \end{bmatrix} \tag{23}$$

where t_x and t_y can be evaluated by the average translation of all image blocks calculated on X- and Y-axes. The translation can then be easily obtained. $m_0, m_1, ..., m_7$ are the parameters of the perspective projection matrix. At time t, the real perspective projection transformation from $F'(t)$ to $F(t+1)$ is denoted as M_t. The motion vector at (x,y) is denoted as $V^M(d_x^M, d_y^M; M_t)$.

$$\begin{cases} d_x^M = x_{t+1} - x_t' \\ d_y^M = y_{t+1} - y_t' \end{cases} \tag{24}$$

The best global motion can be described as the solution of M_t when the square distance between $V^M(d_x^M, d_y^M; M_t)$ and $V(d_x,d_y;t)$ is at the minimum.

$$M = \arg\min \sum_{\Omega} \left\| v^M(d_x^M, d_y^M; M_t) - v(d_x, d_y; t) \right\|^2 \tag{25}$$

where Ω is the center set of the image blocks involved in the calculation. The above equation is equivalent to Equation (26).

$$\begin{cases} E = \left(\sum_{\Omega} \left\| d_x^M - d_x \right\|^2 + \sum_{\Omega} \left\| d_y^M - d_y \right\|^2 \right) \\ (m_0, ..., m_7) = \arg\min E \end{cases} \tag{26}$$

However, this scenario leads to nonlinear optimization, which can be solved by using the Levenberg-Marquardt algorithm [46] or the Newton-Raphson algorithm [47] with a large amount of computation. To avoid nonlinear optimization, Farin *et al.* [25] replaced Euclidean error E with an algebraic error and then converted it to a linear least squares problem. Multiplying the Euclidean error with $(m_6 x_t' + m_7 y_t' + 1)^2$ results in an algebraic error, as shown in Equations (27) and (28).

$$E_a = \left(\sum_{\Omega} (x_{t+1} - x_t' - d_x)^2 + \sum_{\Omega} (y_{t+1} - y_t' - d_y)^2 \right) (m_6 x_t' + m_7 y_t' + 1)^2 \tag{27}$$

$$E_a = \sum_{\Omega} \left\{ \left[m_0 x_t' + m_1 y_t' + m_2 - (x_t' + d_x)(m_6 x_t' + m_7 y_t' + 1) \right]^2 + \left[m_3 x_t' + m_4 y_t' + m_5 - (y_t' + d_y)(m_6 x_t' + m_7 y_t' + 1) \right]^2 \right\} \tag{28}$$

Imposing the necessary condition $\partial E_a / \partial m_i = 0$ for a minimal error results in a linear equation system from which we can obtain $m_0, m_1, \ldots, m_7$.

2.5. Image Motion Monitor

An image motion monitor was created to select an appropriate processing strategy (whether to use coarse GME or not) by monitoring the image motion scales of translation, rotation, and zoom, as shown in Figure 6. When the motion scale is large, image matching is under a wide baseline condition; coarse and residual GME are both performed. Otherwise, only residual GME based on big-block matching is required.

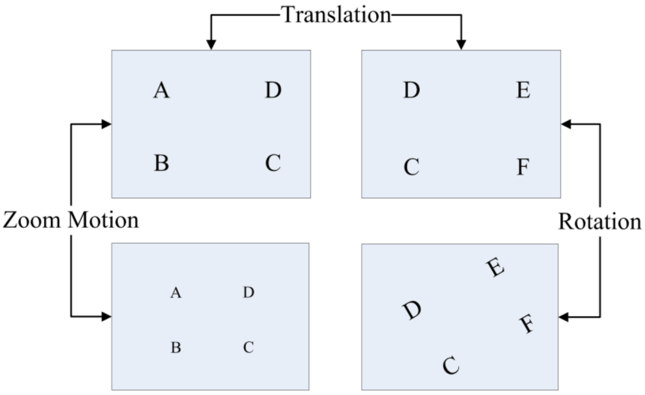

Figure 6. Three basic motions between frames of a UAV video.

How to use existing information to represent the three basic motion scales is a key issue. Coordinate transformation theory and spatial geometry were adopted to solve this problem. The scale of image translation can be manifested as the sum of image center shift on X- and Y-axes. Given that the value cannot be calculated directly, the center shift in GCS can be acquired first and then multiplied by image proportion. Image zoom motion is relevant to focal length and imaging distance. Image rotation is determined by the angle between the north and the projection of the camera optical axis on the ground plane in GCS.

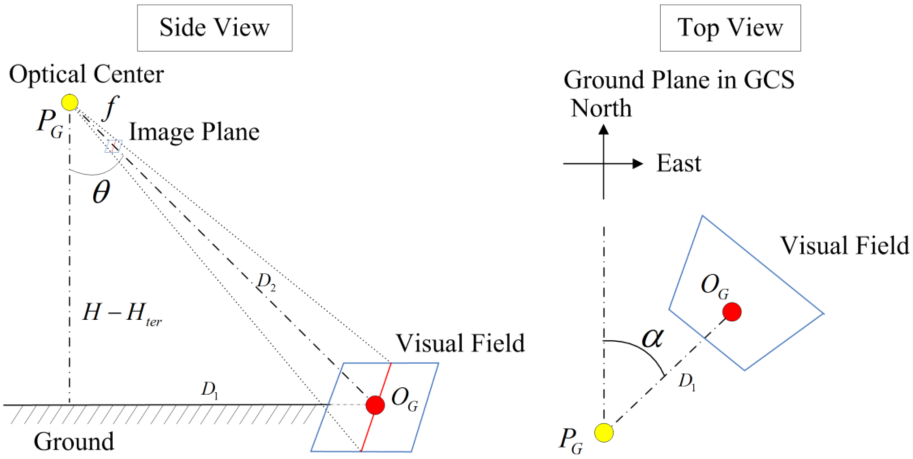

Figure 7. Sketch of UAV imaging.

To derive three basic motion scales, several notions need to be defined. As shown in Figure 7, D_1 stands for the projection distance between the projection of the optical center and visual field center. Imaging distance D_2 is the spatial distance between the optical center and visual field center. Projection point $O_G(x_G^O, y_G^O)$ in GCS transformed from image center O_I in ICS can be obtained through coordinate transformation. For a simplified calculation, the lens center can be replaced by the UAV position denoted as $P_G(x_G^P, y_G^P, z_G^P)$. UAV height is denoted as H, and terrain height is denoted as H_{ter}. According to spatial geometry, D_1 and D_2 can be calculated with Equation (29). Scale S represents the image proportion of the pixel distance and the actual distance; it is relevant to focal length f, sensor pixel size u, and imaging distance D_2. The angle between the north and the projection of the optical axis of the camera on the ground plane at the imaging time is denoted as α and can be solved by triangle calculation on the ground plane of GCS.

$$\begin{cases} D_1 = \sqrt{(x_G^O - x_G^P)^2 + (y_G^O - y_G^P)^2} \\ D_2 = \sqrt{(H - H_{ter})^2 + D_1^2} \\ S = f/(u \bullet D_2) \end{cases} \tag{29}$$

The monitor scales for translation, rotation, and zoom motion denoted as δ_T, δ_R, and δ_Z can be calculated with Equation (30). To maintain the consistency of the image proportion, S is set to a constant value S_0 during the transformation from ICS to GCS.

$$\begin{cases} \delta_T = (\Delta x_G + \Delta y_G) \bullet S_0 \\ \delta_R = D_1 \bullet \tan(\Delta\alpha) \bullet S \\ \delta_Z = \Delta(Col/S) \bullet S \end{cases} \tag{30}$$

After defining $f(x) = \begin{cases} 1, x > 0 \\ 0, \text{other} \end{cases}$, the image motion monitor can be expressed as Equation (31).

$$g_M(t) = f(\delta_T{}^t - \kappa_T) + f(\delta_R{}^t - \kappa_R) + f(\delta_Z{}^t - \kappa_Z) \tag{31}$$

When at least one scale exceeds its limited threshold, the monitor outputs 1; otherwise, it outputs 0. Marked as κ_T, κ_R, and κ_Z, the limited thresholds of translation, rotation, and zoom motion are different, and their specific values are recommended through experiments. When $g_M(t)>0$, a large-scale motion occurs between frames, and global motion needs to be solved by both coarse and residual GME; otherwise, only residual GME based on big-block matching is required.

3. Results and Discussion

3.1. Study Area and Dataset

In the experiments, a medium-altitude UAV that can cruise at an altitude of 3000 m to 5000 m was employed. GPS, INS, a radio altimeter, a barometric altimeter, and a CCD camera were mounted on the UAV. The camera has two DOFs relative to the vehicle. The images underwent camera calibration by the camera itself before GME. The study area is located in the east plain part of China. The main types of landforms include city, village, and open field. The maximum height of the terrain fluctuations and man-made buildings is below 100 m. The area and flight path are shown in Figure 8.

After several flights, a database that includes approximately 100 hours of video and original metadata was established. We assumed that translation, rotation, and zoom motion represent the three basic motions in a UAV video and that the actual motion is composed of these three basic motions. Thus, three types of image and metadata with large-scale motions were selected as the experimental dataset together with 100 groups of images and corresponding metadata for each type of motion (300 groups included). Several image examples of the three types are shown in Figure 9.

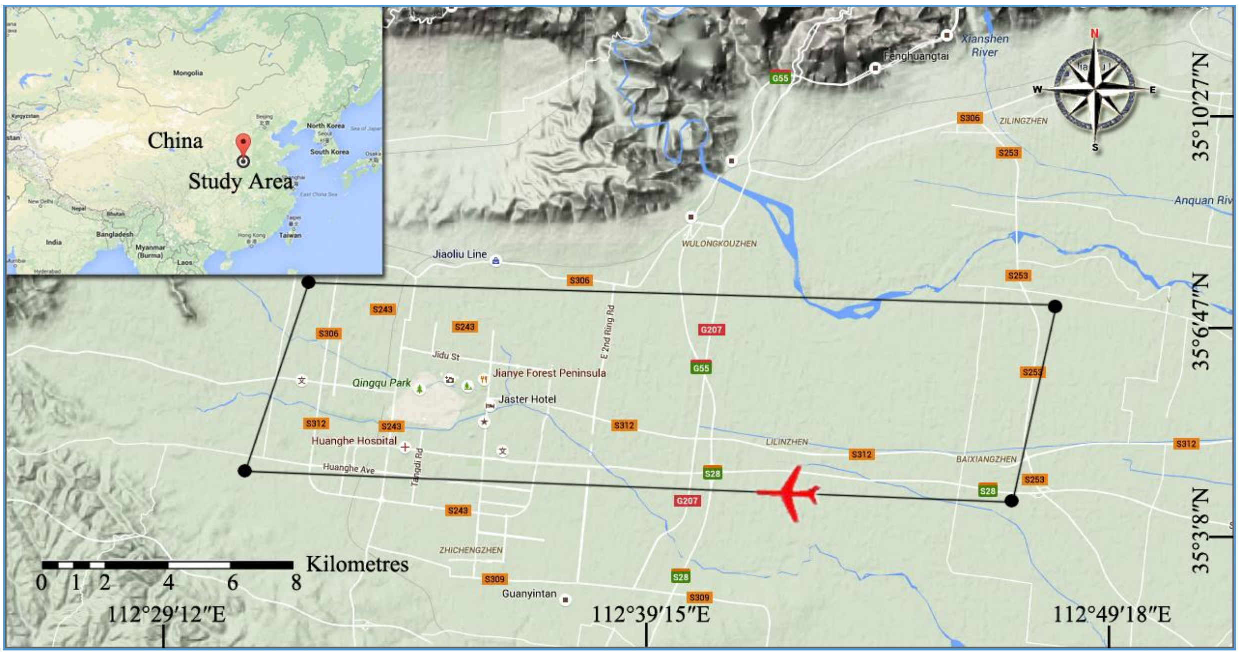

Figure 8. Study area and flight path.

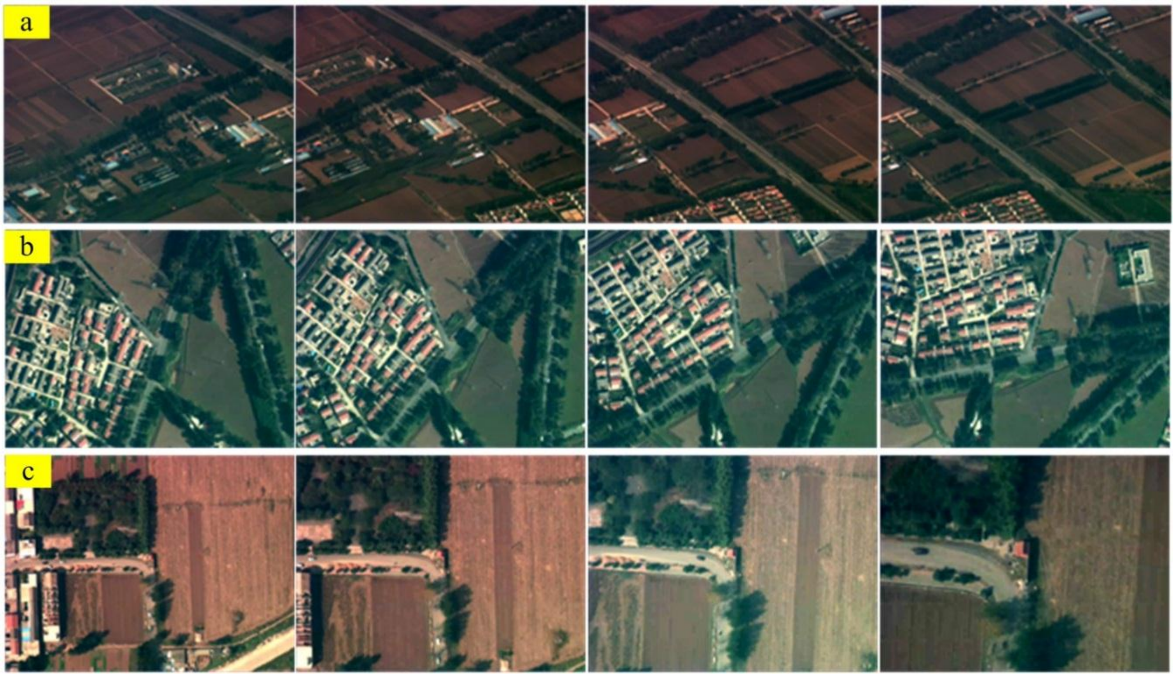

Figure 9. Image examples of the dataset: (**a**) images of translation, (**b**) images of rotation, and (**c**) images of zoom motion.

3.2. Coarse GME

The experimental process of coarse GME is illustrated by a group of data with two images (images A and B) and metadata. According to the workflow of MaGME shown in Figure 10, image A is set as $F(t)$, and image B is set as $F(t+1)$.

Large-scale rotation exists between two images. At this point, GME is a wide baseline registration problem. With the method proposed in Section 2.2, the transformation matrix from ICS to GCS (H_1, H_2)

can be computed by the metadata. After this transformation, two images of $F^{*}(t)$ and $F^{*}(t+1)$ in GCS can be obtained (shown in Figure 11).

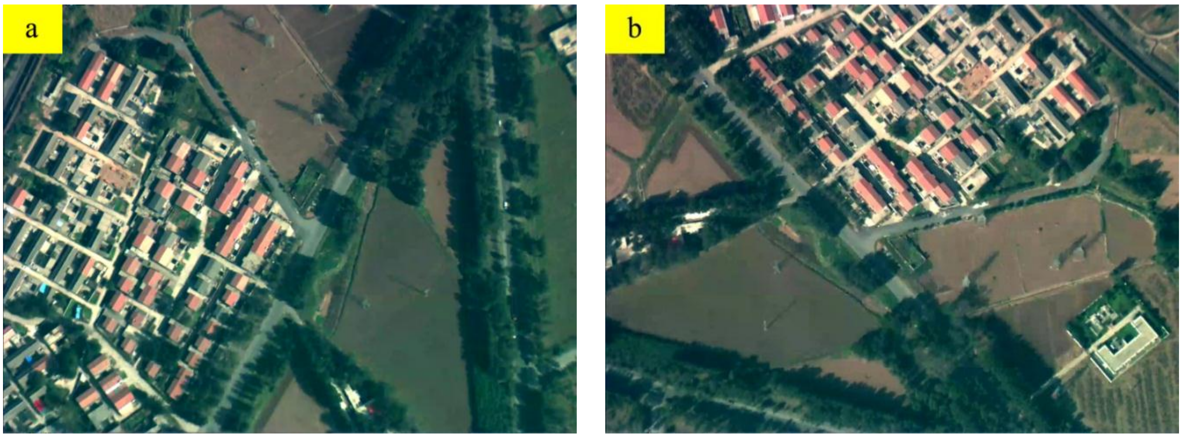

Figure 10. Images in ICS: (**a**) image A-$F(t)$ and (**b**) image B-$F(t+1)$.

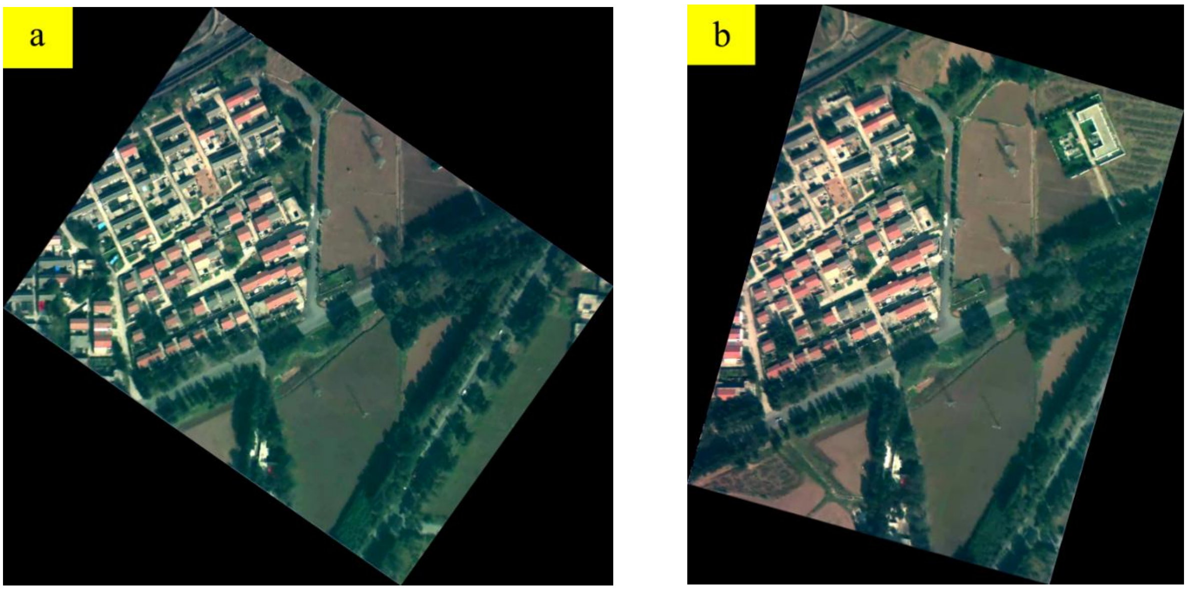

Figure 11. Images after transformation in GCS: (**a**) image A*-$F^{*}(t)$ and (**b**) image B*-$F^{*}(t+1)$.

To verify the effect of distortion elimination between images A and B, we conducted image registration and fusion based on geographic information, as shown in Figure 12. The approach of three-channel fusion is shown in Equation (32), where C represents the channel of R, G, and B.

$$C = C_A \bullet C_B / 255 \tag{32}$$

The two images maintain their consistency in shape and size in GCS. This result is proven by the clarity and lack of aliasing in the overlap pixels. The translation between images A and B in GCS can be represented by translation from $X_AO_AY_A$ to $X_BO_BY_B$. By using the method proposed in Section 2.3, translation T can be obtained as (−248, 36) pixels. Finally, coarse GME M_1 can be obtained with Equation (5).

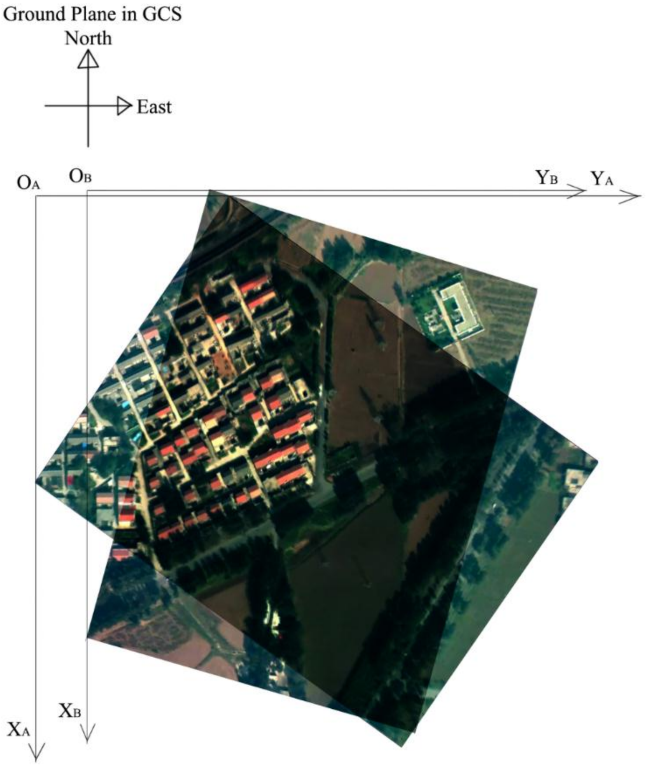

Figure 12. Registration and fusion of two images in GCS.

3.3. Residual GME

Through coarse GME, auxiliary image A' can be obtained as $F'(t)$ with Equation (6); this image is represented as a gray image in Figure 13. Auxiliary image A' has several valid pixels that cannot be compensated by image A contents because the corresponding contents do not exist in image A.

To determine if M_1 can accurately represent the global motion between images A and B, an image fusion experiment was designed with Equation (32), where C only represents the image gray value. The image on the left in Figure 14 is the direct fusion result of images A' and B, and the image on the right is the fusion result after a certain translation. The edges of houses and roads in the image on the left exhibit aliasing, which indicates that the fusion did not reach pixel-level registration precision. However, in the image on the right, the overlapping region has sharp and clear edges, which indicates higher matching accuracy. These results indicate that M_1 cannot represent the global motion accurately. Translation plays the main role in the residual motion between images A' and B; a small amount of distortion also provides a little contribution.

With Equation (20), we can calculate the I&C feature of each block in image A'. By using the I&C feature map, image block selection for residual GME becomes easy, quick, and reliable.

As shown in Figure 15, the higher the I&C feature value on the left is, the brighter the blocks on the right are. These blocks have a high probability of being selected in residual GME. Hence, accurate estimation of the contents of interest can be ensured, and the amount of image blocks can be reduced at the same time.

The motion vector field with all image blocks involved in matching is shown in Figure 16a. The fusion result of the motion vector field and I&C feature map is displayed in Figure 16b. The blocks with a high I&C feature value obtain motion estimation approximately the same in size and orientation, whereas in the blocks with a low I&C feature value, the motion vectors are erroneous. Accordingly, using I&C feature as an indicator to select image blocks for residual GME is reasonable.

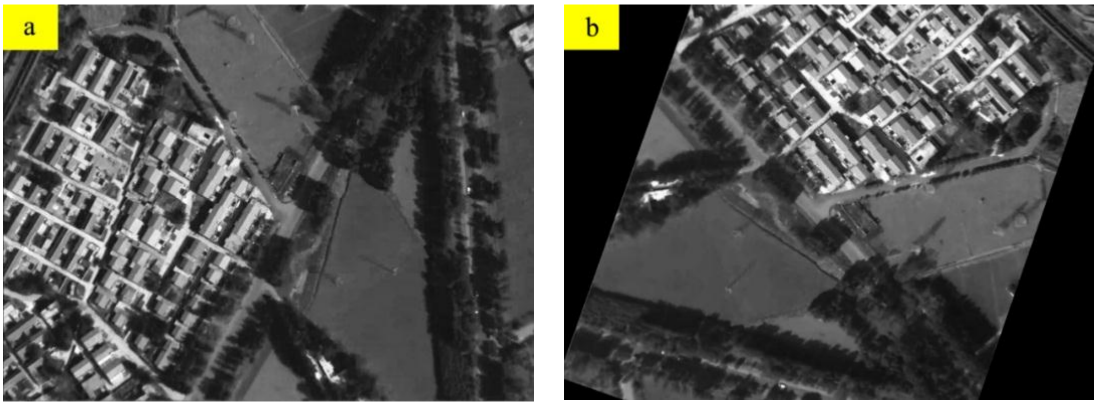

Figure 13. Transformation from image A to image A': (**a**) image A-*F*(*t*) and (**b**) auxiliary image A'-*F'*(*t*).

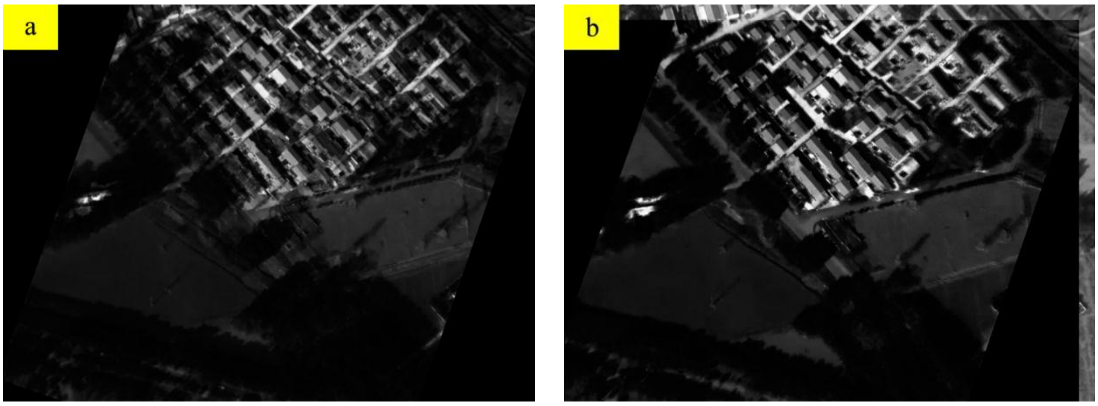

Figure 14. Fusion of image A'-*F'*(*t*) and image B-*F*(*t+1*): (**a**) fusion of image A' and image B without translation. (**b**) Fusion of image A' and image B with some translation.

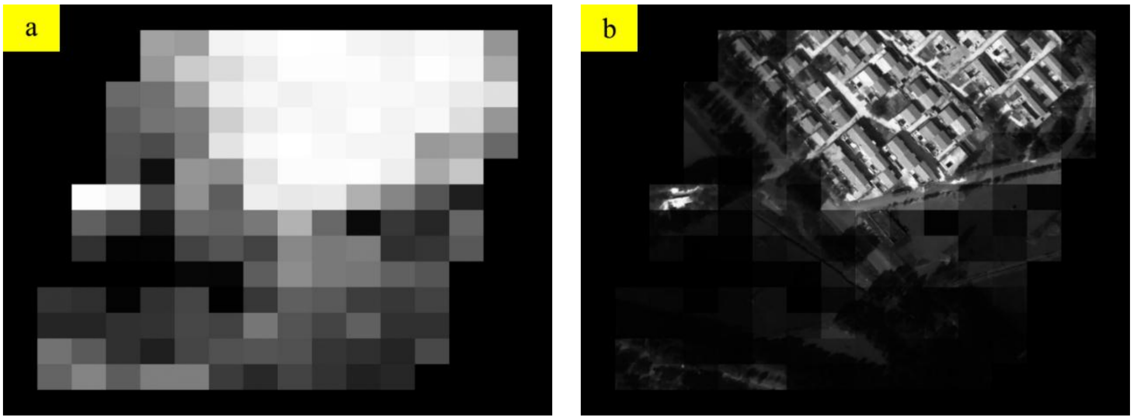

Figure 15. I&C feature value map and fusion result: (**a**) I&C feature map of the image and (**b**) fusion of the I&C feature map and image.

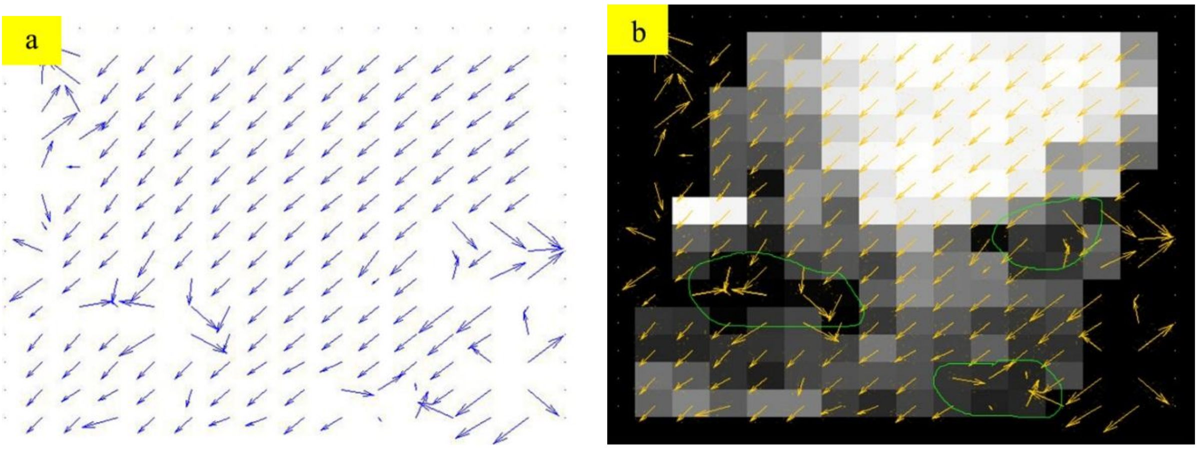

Figure 16. Analysis of the motion vector. (**a**) Motion vector field of all blocks. (**b**) Fusion of the motion vector field and I&C feature map.

3.4. Performance of the Entire Algorithm

In the performance test experiment, we selected 300 groups of images and corresponding metadata with three typical types of motions (translation, rotation, and zoom). To simulate the condition of wide baseline registration, large-scale motions widely exist between frames. Aside from the proposed MaGME(T) (MaGME with residual translation motion estimation) and MaGME(P) (MaGME with residual perspective projection motion estimation), BM-GME (GME based on block matching) and SIFT-GME (SIFT based GME) were used for comparison.

For the block matching algorithm of BM-GME, one may refer to [45]. For the model solution, one may refer to [25].

The homography matrix of SIFT-GME was computed with the SIFT matching method in [48,49]. SIFT features are invariant to image scale and rotation and are known to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. After scale-space extrema detection, keypoint localization, orientation assignment, and keypoint description, SIFT features are generated as vectors with 128 dimensions. SIFT-GME involves three major stages: feature detection, feature matching, and global motion solution. In feature detection, the initial Gaussian smoothing parameter (σ_0) is 1.6, and the number of sampled intervals per octave is 3. In feature matching, a modification of the k-d tree algorithm called the best-bin-first search method [50] is applied to identify the nearest neighbors with high probability using only a limited amount of computation. In global motion solution, the global motion is represented by the perspective projection model in Equation (1). The RANSAC algorithm is utilized to select the matching features and calculate the perspective projection matrix.

Under each motion condition, the motion scales between frames were calculated with Equation (30). After GME compensation, the PSNR values of the four methods were computed with Equation (33), where *MSE* is the mean squared error. The results are shown in Figures 17 to 19 and analyzed in Table 4.

$$\mathrm{PSNR}=10\log_{10}(255^2 / MSE) \tag{33}$$

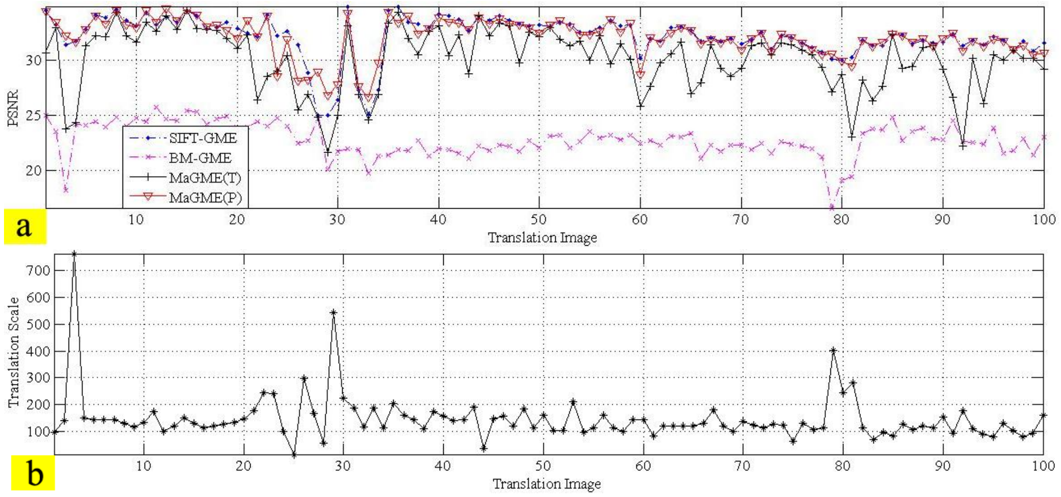

Figure 17. Performance analysis under a large-scale translation condition: (**a**) PSNR of the four methods and (**b**) translation scale of the images.

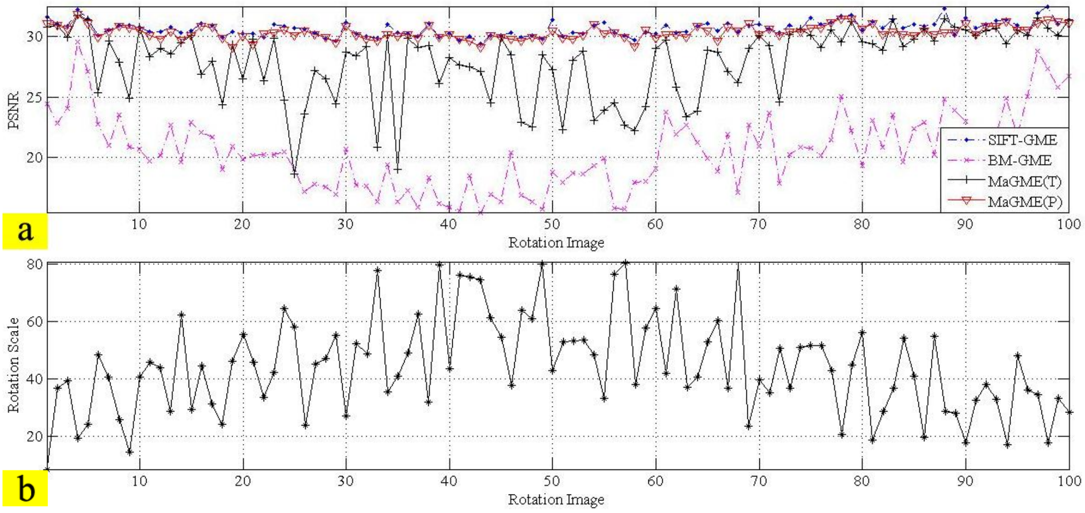

Figure 18. Performance analysis under a large-scale rotation condition: (**a**) PSNR of the four methods and (**b**) rotation scale of the images.

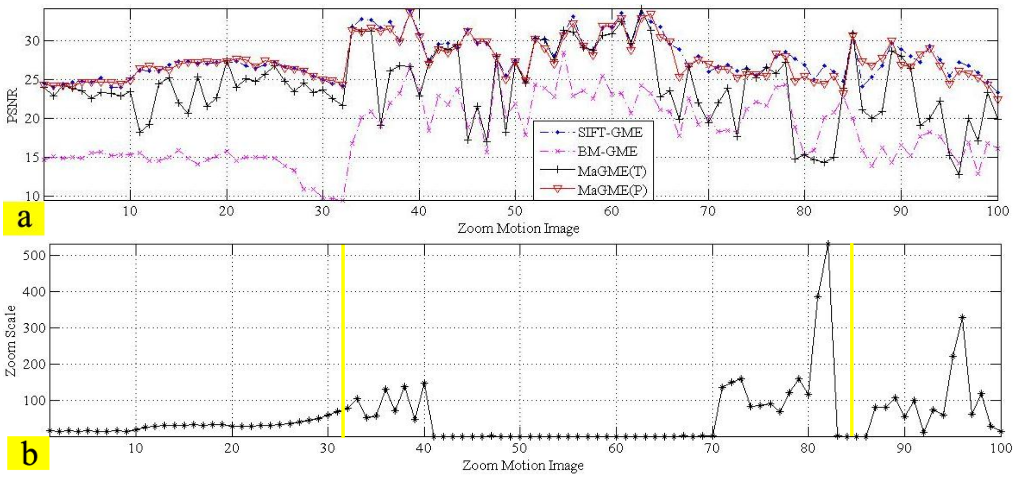

Figure 19. Performance analysis under a large-scale zoom motion condition: (**a**) PSNR of the four methods and (**b**) zoom scale of the images.

In the three experiments, the images and metadata in translation and rotation were continuous, whereas zoom motion data were collected from three scenes separated by vertical bars.

As shown in Figures 17 to 19, the PSNR curves of SIFT-GME, MaGME(T), and MaGME(P) are higher than the PSNR curve of BM-GME in most cases. The PSNR curve of MaGME(P) is almost similar to that of SIFT-GME. When the curves of motion scale increase, the four PSNR curves decline. However, the applicability of SIFT-GME, MaGME(T), and MaGME(P) to large-scale motions is better than that of BM-GME.

The average values of motion scales were computed under three conditions, namely, translation scale = 147.6, rotation scale = 44.4, and zoom scale = 52.8. The average PSNR values are shown in Table 4.

Table 4. Average PSNR.

Test Sequence (Frame Number)	**SIFT-GME Average PSNR (dB)**	**BM-GME Average PSNR (dB)**	**MaGME(T) Average PSNR (dB)**	**MaGME(P) Average PSNR (dB)**
Translation (100)	32.18	22.77	30.10	32.09
Rotation (100)	30.70	20.49	27.94	30.36
Zoom motion (100)	27.67	18.22	24.01	27.37
Average PSNR	30.18	20.49	27.35	29.94

The experiments were conducted on a standard Intel Core i5-based PC (2.3 GHz). The results show that the performance of MaGME(T) and MaGME(P) improved by 33% and 46%, respectively, compared with that of the conventional BM-GME method. The performance of MaGME(P) is close to that of SIFT-GME (only 1% lower). However, we found that SIFT feature matching fails when the local features change because of large-scale zoom motion. This condition causes SIFT-GME to crash. We did not use these data in the experiments. However, the failure of SIFT feature matching does occur often when the landforms are water, desert, or other types with few local features in images or under the condition of local features changing considerably. According to the experiments, the proposed MaGME(T) and MaGME(P) exhibit better performance than conventional BM-GME under large-scale motion conditions. Without depending on image local features, they can be utilized for more special landforms.

In the image motion monitor, the thresholds of translation, rotation, and zoom motion are related to several attributes of system structure, imaging parameters, landforms, and other factors. The best values are recommended by the experiments. However, we can assign three initial values according to the size of the big block and searching window used in residual GME. κ_T can be considered the average size of the big block in value; $\kappa_T = (Row/16+Col/16)/2$. κ_R denotes the translation of the image center because of rotation in value. The maximum of κ_R could be half of κ_T. However, the effect of image-block distortion is considered, and $\kappa_R=1/4\cdot\kappa_T$. Zoom motion is a factor sensitive to image local features and image-block distortion. κ_Z can be strictly confined to $\kappa_Z = 1/8\cdot\kappa_T$.

With regard to time consumption and according to [21–25], in the computation of the three typical GME methods, the pixel based method is larger than the feature based one, and the feature based method is larger the vector based one. SIFT-GME is a feature-based GME method, and BM-GME is a vector-based method. The computation of SIFT-GME is mainly required in SIFT feature detection and matching. In general, the existence of hundreds of features in one image can make the algorithm difficult to implement in real time. Remote images acquired by UAVs usually have rich textures. Thousands or tens of thousands of SIFT features need to be calculated, which would seriously affect real-time performance. By contrast, the calculation speed of BM-GME is high. In several digital video compression standards, such as H.264 and MPEG-4, block matching-based methods are applied to real-time GME.

MaGME is a selective big-block matching-based GME method. The complexity of MaGME is approximately equal to the sum of the three parts of metadata calculation, I&C feature detection, and selective block matching. The overall computation of MaGME is related to the number of blocks involved in block matching. Experiments show that if block matching employs less than 50% of the image blocks to calculate the global motion matrix, the computation amount in MaGME is less than that in BM-GME. In the actual process, the number of required image blocks is much smaller than this number. MaGME only requires minimal computation in coarse GME and a small number of image blocks to solve the residual GME. For the residual GME in particular, MaGME(T) requires at least one image block to calculate translation, whereas MaGME(P) requires at least four image blocks to compute the eight parameters of perspective projection transformation. Therefore, the amount of computation in MaGME is less than that in BM-GME; consequently, it is also less than that in SIFT-GME.

4. Conclusions

GME is a key step in many video applications for UAVRS. Given that conventional image-based GME methods do not perform well when a UAV’s motion and behavior change significantly or when image information is not rich, a method of metadata-assisted GME called MaGME was developed in this study for medium-altitude UAVs.

The main contributions of this study are threefold. First, GME was divided into coarse and residual GME. Coarse GME was solved according to the theory of coordinate transformation. With the assistance of an auxiliary image, the large-scale motion effect on image matching was eliminated, and the wide baseline condition was converted to a narrow baseline one. Second, to maximize the use of reliable information in the image and ensure high-precision motion estimation of the contents of interest, an I&C feature detection algorithm was designed to describe the information content and contrast simultaneously. Based on the I&C feature, a big-block matching method was developed to complete residual GME. Third, an image motion monitor was designed to determine the scale of video motion and select the appropriate processing strategy.

A medium-altitude UAV was employed to collect experimental data. Three typical groups of datasets, including translation, rotation, and zoom, were set up to test four GME methods. These four methods are the proposed MaGME(T) (MaGME with residual translation motion estimation), proposed MaGME(P) (MaGME with residual perspective projection motion estimation), GME based on block matching, and SIFT-based GME. The PSNR and motion scale values of the three datasets were computed and analyzed (300 images and metadata samples in all). The results show that the proposed MaGME(T) and MaGME(P) exhibit encouraging performance when the motion scale is large. The two methods can be applied to images with a few local features in several special landforms. The results of this research can be applied to other medium- or high-altitude UAVs with a similar system structure.

However, some future work, including image motion monitor optimization and embedded implementation of the entire method, need to be done through further analysis and experiments.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 61450008).

Author Contributions

Hongguang Li wrote the program and the manuscript. Xinjun Li conceived and designed the scheme of the method. Wenrui Ding organized the experiments and provided some valuable suggestions. Yuqing Huang revised the manuscript.

Conflicts of Interest

The authors declare no conflict of interest.

Reference

1. Pajares, G. Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs). *J. Photogramm. Eng. Remote Sens*. **2015**, *81*, 281–329.
2. Vachtsevanos, G.J.; Valavanis, K.P. Military and civilian unmanned aircraft. In *Handbook of Unmanned Aerial Vehicles*; Valavanis, K.P., Vachtsevanos, G.J., Eds.; Springer Netherlands: Berlin, Germany, 2015; pp. 93–103.
3. Toma, A. Use of Unmanned Aerial Systems in Civil Applications. Ph.D. Thesis, Politecnico di Torino, Torino, Italy, 2015.
4. D'Oleire-Oltmanns, S.; Marzolff, I.; Peter, K.D.; Ries, J.B. Unmanned aerial vehicle (UAV) for monitoring soil erosion in Morocco. *Remote Sens*. **2012**, *4*, 3390–3416.
5. Tong, X.H.; Liu, X.F.; Chen, P.; Liu, S.J.; Luan, K.F.; Li, L.Y.; Liu, S.; Liu, X.L.; Xie, H.; Jin, Y.M.; *et al*. Integration of UAV-based photogrammetry and terrestrial Laser scanning for the three-dimensional mapping and monitoring of open-pit mine areas. *Remote Sens*. **2015**, *7*, 6635–6662.
6. Bendig, J.; Bolten, A.; Bennertz, S.; Broscheit, J.; Eichfuss, S.; Bareth, G. Estimating biomass of barley using crop surface models (CSMs) derived from UAV-based RGB imaging. *Remote Sens*. **2014**, *6*, 10395–10412.
7. Getzin, S.; Nuske, R.; Wiegand, K. Using unmanned aerial vehicles (UAV) to quantify spatial gap patterns in forests. *Remote Sens*. **2014**, *6*, 6988–7004.
8. Towler, J.; Krawiec, B.; Kochersberger, K. Radiation mapping in post-disaster environments using an autonomous helicopter. *Remote Sens*. **2012**, *4*, 1995–2015.
9. Skoglar, P.; Orguner, U.; Törnqvist, D; Gustafsson, F. Road target search and tracking with gimbaled vision sensor on an unmanned aerial vehicle. *Remote Sens*. **2012**, *4*, 2076–2111.
10. Watts, A.; Ambrosia, V.; Hinkley, E. Unmanned aircraft systems in remote sensing and scientific research: Classification and considerations of use. *Remote Sens*. **2012**, *4*, 1671–1692.

11. Kendoul, F. Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. *J. Field Robot*. **2012**, *29*, 315–378.
12. Mai, Y.; Zhao, H.; Guo, S. The analysis of image stabilization technology based on small-UAV airborne video. In Proceedings of International Conference on Computer Science & Electronics Engineering, Hangzhou, China, 23–25 March 2012; pp. 586–589.
13. Brockers, R.; Humenberger, M.; Kuwata, Y.; Matthies, L.; Weiss, S. Computer vision for micro air vehicles. In *Advances in Computer Vision & Pattern Recognition*; Kisačanin, B., Gelautz, M., Eds.; Springer International Publishing: Berlin, Germany, 2014; pp. 73–107.
14. Kanade, T.; Amidi, O.; Ke, Q. Real-Time and 3D Vision for Autonomous Small and Micro Air Vehicles. Available online: http://www.cs.cmu.edu/~ke/publications/ke_CDC_04_AUV.pdf (accessed on 5 July 2015).
15. Wang, Y.; Hou, Z.; Leman, K.; Chang, R. Real-Time Video Stabilization for Unmanned Aerial Vehicles. Available online: http://www1.i2r.a-star.edu.sg/~ywang/papers/MVA_2011_Real-Time%20Video%20Stabilization%20for%20Unmanned%20Aerial%20Vehicles.pdf (accessed on 5 July 2015).
16. Bhaskaranand, M.; Gibson, J.D. Low-complexity video encoding for UAV reconnaissance and surveillance. In Proceedings of Military Communications IEEE Conference, Baltimore, MD, USA, 7–10 November 2011; pp. 1633–1638.
17. Rodr ǵuez-Canosa, G.R.; Thomas, S.; del Cerro, J.; Barrientos, A.; MacDonald, B. A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera. *Remote Sens*. **2012**, *4*, 1090–1111.
18. Hsieh, J.; Yu, S.; Chen, Y. Motion-based video retrieval by trajectory matching. *IEEE Trans. Circuits Syst. Video Technol*. **2006**, *16*, 396–409.
19. Zhang, H.; Yang, Z.; Zhang, L.; Shen, H. Super-resolution reconstruction for multi-angle remote sensing images considering resolution differences. *Remote Sens*. **2014**, *6*, 637–657.
20. Turner, D.; Lucieer, A.; Watson, C. An automated technique for generating georectified mosaics from ultra-high resolution unmanned aerial vehicle (UAV) imagery, based on structure from motion (SfM) point clouds. *Remote Sens*. **2012**, *4*, 1392–1410.
21. Huang, Y.R. A fast recursive algorithm for gradient-based global motion estimation in sparsely sampled field. In Proceedings of the Eighth International Conference on Intelligent Systems Design and Applications, Washington, DC, USA, 26–28 November 2008.
22. Tok, M.; Glantz, A.; Krutz, A.; Sikora, T. Feature-based global motion estimation using the Helmholtz principle. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011.
23. Chen, K.; Zhou, Z.; Wu, W. Progressive motion vector clustering for motion estimation and auxiliary tracking. *ACM Trans. Multimed. Comput. Commun. Appl*. **2015**, *11*, 1–23.
24. Haller, M.; Krutz, A.; Sikora, T. Evaluation of pixel- and motion vector-based global motion estimation for camera motion characterization. In Proceedings of the Image Analysis for Multimedia Interactive Services, London, UK, 6–8 May 2009.
25. Farin, D.; de With, P.H.N. Evaluation of a feature-based global-motion estimation system. *Proc. SPIE* **2005**, doi:10.1117/12.632680.

26. Okade, M.; Biswas, P.K. Fast camera motion estimation using discrete wavelet transform on block motion vectors. In Proceedings of the Picture Coding Symposium, Krakow, Poland, 7–9 May 2012.
27. Yoo, D.; Kang, S.; Kim, Y.H. Direction-select motion estimation for motion-compensated frame rate up-conversion. *J. Disp. Technol*. **2013**, *9*, 840–850.
28. Amirpour, H.; Mousavinia, A. Motion estimation based on region prediction for fixed pattern algorithms. In Proceedings of the International Conference on Electronics, Computer & Computation, Ankara, Turkey, 7–9 November 2013.
29. Sung, C.; Chung, M.J. Multi-scale descriptor for robust and fast camera motion estimation. *IEEE Signal Process. Lett*. **2013**, *20*, 725–728.
30. Krutz, A.; Glantz, A.; Tok, M.; Esche, M.; Sikora, T. Adaptive global motion temporal filtering for high efficiency video coding. *IEEE Trans. Circuits Syst. Video Technol*. **2012**, *22*, 1802–1812.
31. Areekath, L.; Palavalasa, K.K. Sensor assisted motion estimation. In Proceedings of the Conference on Engineering & Systems, Uttar Pradesh, India, 12–14 April 2013.
32. Chen, X.; Zhao, Z.; Rahmati, A.; Wang, Y.; Zhong, L. Sensor-assisted video encoding for mobile devices in real-world environments. *IEEE Trans*. *Circuits Syst. Video Technol.* **2011**, *21*, 335–349.
33. Wang, G.; Ma, H.; Seo, B.; Zimmermann, R. Sensor-assisted camera motion analysis and motion estimation improvement for H.264/AVC video encoding. In Proceedings of the International Workshop on Network & Operating System Support for Digital Audio & Video, ACM, Toronto, ON, Canada, 7–8 June 2012.
34. Strelow, D.; Singh, S. Optimal motion estimation from visual and inertial measurements. In Proceedings of the IEEE Workshop on Applications of Computer Vision, Orlando, FL, USA, 3–4 December 2002.
35. Rodr ǵuez, A.F.; Ready, B.B.; Taylor, C.N. Using telemetry data for video compression on unmanned air vehicles. In *Collection of Technical Papers-AIAA Guidance*, Navigation, and Control Conference, Keystone, CO, USA, 21–24 August 2006.
36. Gong, J.; Zheng, C.; Tian, J.; Wu, D. An image-sequence compressing algorithm based on homography transformation for unmanned aerial vehicle. In Proceedings of the International Symposium on Intelligence Information Processing & Trusted Computing, IEEE, Huanggang, China, 28–29 October2010.
37. Bhaskaranand, M.; Gibson, J.D. Global motion assisted low complexity video encoding for UAV applications. *IEEE J. Sel. Top. Signal Process*. **2015**, *9*, 139–150.
38. Angelino, C.V.; Cicala, L.; Persechino, G. A Sensor aided H.264 encoder tested on aerial imagery for SFM. In Proceedings of the International Conference on Image Processing (ICIP), Paris, France, 27–30 October, 2014.
39. Angelino, C.V.; Cicala, L.; Cicala, L.; de Mizio, M.; Leoncini, P.; Baccaglini, E.; Gavelli, M.; Raimondo, N.; Scopigno, R. Sensor aided H.264 Video Encoder for UAV applications. In Proceedings of IEEE Picture Coding Symposium, San JoSe, CA, USA, 8–13 December 2013.
40. Bhaskaranand, M.; Gibson, J.D. Low-complexity video encoding for UAV reconnaissance and surveillance. In Proceedings of the IEEE Military Communications Conference, Baltimore, MD, USA, 7–10 November 2011.

41. Gariepy, R. UAV Motion estimation using low quality image features. In Proceedings of the Collection of Technical Papers-AIAA Guidance, Navigation, and Control Conference, Toronto, ON, Canada, 2–5 August 2010.
42. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*; Cambridge University Press: Cambridge, UK, 2001; pp. 1865–1872.
43. Dinh, T.N.; Lee, G. Efficient motion vector outlier removal for global motion estimation. In Proceedings of the IEEE International Conference on Multimedia & Expo (ICME), Barcelona, Spain, 11–15 July 2011.
44. Chen, Y.; Bajic, I. Motion vector outlier rejection cascade for global motion estimation. *IEEE. Signal Process. Lett.* **2010**, *17*, 197–200.
45. Choudhury, H.A.; Saikia, M. Survey on block matching algorithms for motion estimation. In Proceeding of 2014 International Conference on Communications and Signal Processing (ICCSP), Melmaruvathur, India, 3–5 April 2014.
46. Jinxin, Z.; Junping, D. Automatic image parameter optimization based on Levenberg-Marquardt algorithm. In Proceedings of the IEEE International Symposium on Industrial Electronics, Seoul, Korea, 5–8 July 2009.
47. Su, Y.; Sun, M.; Hsu, V. Global motion estimation from coarsely sampled motion vector field and the applications. In Proceedings of IEEE Transactions on Circuits & Systems for Video Technology, Bangkok, Thailand, 25–28 May 2003.
48. Lowe, D.G. Object recognition from local scale invariant feature. In Proceedings of the IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999.
49. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*. **2004**, *60*, 91–110.
50. Jeffrey, S.B.; David, G.L. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In Proceedings of the Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, 17–19 June 1997.