Received January 13, 2020, accepted February 10, 2020, date of publication February 13, 2020, date of current version February 25, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2973723

Image Feature Matching via Parallax Mapping for Close Range Photogrammetric Application

YANG DONG, DAZHAO FAN, QIUHE MA, AND SONG JI
Institute of Surveying and Mapping, Information Engineering University, Zhengzhou 450000, China

Corresponding author: Dazhao Fan (fdzcehui@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 41401534 and Grant 41971427.

ABSTRACT Image feature matching is an important step in close range photogrammetric applications, and implementing a fast, accurate and robust feature-based image matching technique is a challenging task. To solve the problems of traditional image feature matching algorithms, such as their low accuracy and long processing time, we present a parallax mapping-based matching (PMM) method that is able to improve the computation efficiency, accuracy and robustness of feature-based image matching for close range photogrammetric applications. First, the disparity of the initial corresponding points is calculated, and the mismatches are initially removed via local disparity clustering. Then, the local coordinates of the matching points are constrained by an image division grid. To ensure the correctness of the matching points, the fast polynomial transform is used for as a quadratic constraint. The method can extract high accuracy matching points from the original coarse matching points with low accuracy, and also preserve the true matching points to the greatest extent. Using a variety of experimental datasets and current mainstream feature extraction algorithms, we designed and compared commonly used feature-based image matching algorithms. The experimental results show that the proposed method is simple but effective, can meet the real-time calculation requirements, and outperforms the current state-of-the-art methods.

INDEX TERMS Image matching, rejecting mismatches, motion consistency, parallax mapping, polynomial.

I. INTRODUCTION

Image feature matching is one of the basic research directions in the fields of close range photogrammetry and computer vision. It has been widely used in camera calibration [1], 3D reconstruction [2], and pattern recognition [3]. The usual feature matching algorithm follows a two-step strategy [4], [5]. First, the initial corresponding points are calculated using feature similarity constraints, such as fast approximate nearest neighbors (FANN) [6] and brute force (BF) method. However, due to the inherent defects of the feature description algorithms, there are usually mismatches. Then, the secondary image matching or matching purification method is performed to reject mismatches [7]. The main issue at this step is to remove as many false matches as possible and keep the true matches.

The secondary image matching method mainly uses the corresponding points model to solve the problem by optimizing the model parameters and applying point constraints [8]. Therefore, the matching algorithm can be developed in two directions [9]: one is the establishment of the corresponding points model, and the other is the optimization of the algorithm that estimates the model parameters. Commonly used constraint models mainly include coplanar geometric homography matrix models, epipolar geometric fundamental matrix models, linear transformation models, affine transformation models, and projection transformation models [10], [11]. Commonly used model parameter solution algorithms include the M-estimation algorithm [12], the least median of squares (LMedS) algorithm [13], and the random sample consensus (RANSAC) algorithm [14].

Corresponding points models are mostly implemented using a priori constraints between corresponding image points. The constraint model can be divided into two categories [7], [8]: the strict geometric constraint model (parametric model) and the probabilistic constraint model (nonparametric model). Strict geometric models implement strict geometric relationship constraints between corresponding points, such as polar line geometric constraints and coplanar geometric constraints [15]. Probabilistic constraint models implement local probability constraints between corresponding points, such as motion consistency constraints and local composition consistency constraints [16].

Although the strict geometric model constraints can strictly perform the purification operation between initial corresponding points, they require solving higher precision

The associate editor coordinating the review of this manuscript and approving it for publication was Paolo Napoletano.

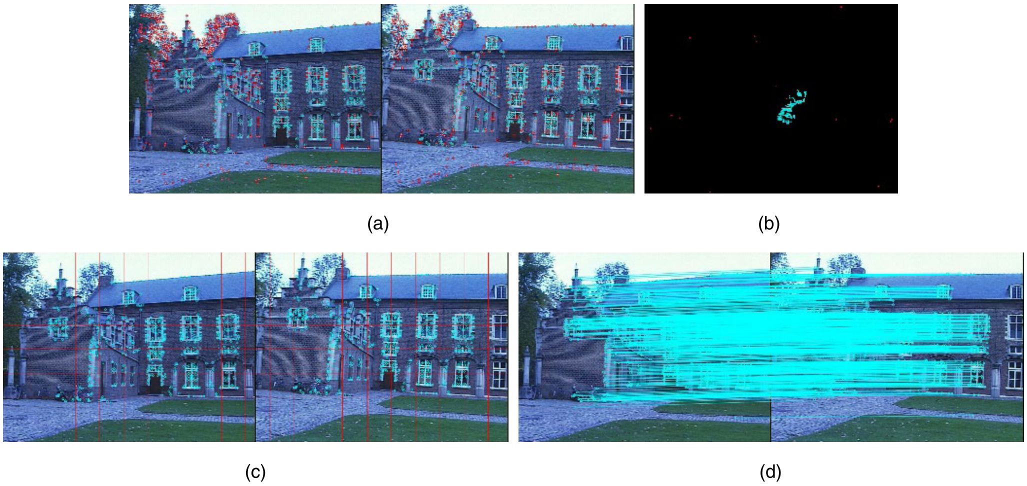
This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

model parameters, and the model parameters often have a certain locality and cannot be well adapted to matching points in complex scenes [9]. When using the homography matrix or the fundamental matrix iterative solution to match points, due to the existence of image distortion and the inconsistency of the scene depth, the solved homography matrix or the fundamental matrix cannot satisfy the threshold constraint for all correctly matched points. Thus, the calculation time is too long or correctly matched points are lost due to the parameter solution problem, as shown in Ref. [14]. The probabilistic constraint model uses a regular statistical method to match points. Generally, the statistical rules of the local matching points are used to constrain each other, and rich matching points can be obtained. However, the constraint model has too many parameters to solve, which can consume too much calculation time, as shown in Ref. [17].

Algorithms for estimating model parameters can generally be divided into two categories: robust regression estimation and random parameter estimation [18]. Robust regression estimation iteratively eliminates mismatches to obtain robust model parameters, but it only applies to cases with high original accuracy rates, such as the M-estimation [12]. The random parameter estimation uses a small number of randomly selected point sets to estimate the model parameters by continuously trying to obtain the optimal solution. This method can be applied to the case where the original accuracy rate is low, such as RANSAC [14]. Among them, the RANSAC algorithm can robustly find the inliers in a point set that has a large number of outliers and has been widely used. Several improved algorithms have been derived, such as the preemptive RANSAC (P-RANSAC) [19], randomized RANSAC (R-RANSAC) [20], and USAC [21]. However, the RANSAC algorithm needs to select a suitable initial set of inliers, and the matching result can easily fail into the local optimal solution, resulting in the omission of the correctly matching points. Furthermore, when the mismatch rate increases, the number of iterations also increases, resulting in a sharp increase in the computation time [9].

In recent years, some new probabilistic constraint models or nonparametric models have also been developed. The identifying point correspondences by correspondence function (ICF) [22] algorithm converts the mismatch removal problem into a correspondence function pair that maps the points in one image to their corresponding points in the other image. The vector field consensus (VFC) [23], [24] algorithm rejects mismatches by learning a smooth field to fit the potential inliers and it estimates a consensus inlier set. Similar to the VFC, the L2-minimizing estimate-based method (L2E) [25] also uses a nonparametric spatial transformation to deal with outliers. Ref. [26] describes a dual decomposition optimization technique for graph matching, in which they formulate the matching task as a complex cost function to remove mismatches. Ref. [27] scores candidate matches using the most promising neighbors, and uses them to update the neighbors for graph matching. Progressive graph matching (PGM) [28] alternately performs graph probabilistic progression and graph matching steps for graph matching. Ref. [29] formulates image matching as a clustering problem, and presents a linkage model and dissimilarity metric using the hierarchical agglomerative clustering (ACC). Ref. [16] propose an algorithm called Locality Preserving Matching (LPM) that uses the difference of the local neighborhood structures of inliers and outliers to identify mismatches from the given putative image feature correspondences. Ref. [30] casts feature matching as a spatial clustering problem with outliers, customizes the classic density based spatial clustering method of applications with noise (DBSCAN) to the context of feature matching, and proposes a robust feature matching algorithm based on spatial clustering, which is called RFM-SCAN. Ref. [31] proposes a robust point matching approach based on manifold regularization that is called robust point matching using manifold regularization (MR-RPM). The virtual line descriptor and semilocal matching (K-VLD) [32] method uses the local composition consistency of matching points, defines the virtual line descriptor, and designs a semiglobal matching strategy to match and purify points on the basis of an initial matching. The method obtains rich matching points and optimizes the accuracy and sufficiency, but the computational time is expensive. The coherence based decision boundaries (CODE) [17] algorithm uses motion consistency theory, combines it with the characteristics of affine-invariant scale-invariant feature transform (ASIFT) [33] descriptors, and solves the motion consistency constraint function to optimize the matching points based on the initial matching, which can obtain richer matching points than traditional method, however only ASIFT feature descriptors are considered and the computation time is too long. The grid-based motion statistics (GMS) [34] algorithm uses the local consistency of matching points and simple grid statistics for matching, can obtain rich matching points, is greatly optimized in terms of its efficiency and outperforms other real-time matching algorithms. As mentioned earlier, these methods do not use parametric geometric models as strict constraints; thus, relatively low-precision noisy matches may be difficult to separate from high-precision correct matches.

Although some matching methods work well, all of them have their own advantages and application scopes but are difficult to robustly and accurately integrate to achieve real time performance. To solve these problems, this paper utilizes motion consistency theory, parallax mapping and image calibration theory, and presents a parallax mapping-based matching (PMM) method for close range photogrammetric application in the rigid case. The parallax between the initial corresponding point is calculated, and the local motion consistency is reflected by the cumulative connected domain calculation between the parallax coordinates. Then, the local coordinates of the matching points are constrained by an image division grid. To ensure the correctness of the matching points, the fast polynomial transform is used as the quadratic constraint. Our method is effective and robust, can meet the requirements of real-time calculations, and outperforms the



(a)

(b)

(c)

(d)

FIGURE 1. PMM implementation details. (a) Initial feature matching. The cyan dots represent inliers, and the red dots represent outliers. (b) Calculate the parallax map, and remove the mismatches via local disparity clustering. The cyan dots form large connected domains, and the red dots form obvious isolated points. (c) Compute the image grid, and the local coordinates of the matching points are constrained by the image division grid. (d) Polynomial iterative calculation for final matching.

state-of-the-art methods in terms of its accuracy. The process is shown in Fig. 1.

In summary, our contributions are as follows:

- The local motion consistency is transformed into two consistency discriminant criteria. We show that these constraints can robustly achieve feature matching;
- An effective image feature matching method based on parallax mapping is designed, which can be used in real-time feature matching applications;
- A complete experiment is designed, and the performances of current mainstream feature extraction and matching algorithms are compared and analyzed;
- PMM is demonstrated to be significantly more accurate than RATIO (second nearest neighbor distance ratio test) [35], RANSAC [14], K-VLD [32] and the recently proposed GMS 34], LPM [16], and RFM-SCAN [30] methods.

The structure of the rest of this paper is as follows: a detailed description of the method we proposed is given in Section II, the details of our experiment are given in Section III, the discussion of the results is given in Section IV, and a summary and our expected future work are finally given in Section V.

II. METHOD

In this section, the PMM feature matching method is introduced. This method takes close range photogrammetric images (rigid case) as the main matching objective. First, we construct the initial feature matching set, which is obtained using the BF or FANN algorithm. Then, we consider the local motion consistency, calculate the parallax map, and perform local disparity clustering, which effectively removes the outliers from a large number of initial matches. Then, the image is divided into grids, and the local coordinates of the matching points are constrained by the image division grid. Finally, using the polynomial constraint model, complete image feature matching.

A. MOTION CONSISTENCY THEORY

The theory of motion consistency refers to the consistency of the local-pixel motion of an image and describes the basic law of the image content changes in a scene [36]. The image motion field characterizes the variation in the local-pixel motion of an image. The motion-smoothing function describes the consistency of the local-motion field, and it is introduced below.

1) MOTION FIELD

At a certain moment, a point p_0 on the surface of an object corresponds to a point p_i on an image, and the geometric relationship between the two points is determined by the projection equation. For pinhole camera photography, photographic light follows the perspective projection equation, that is, the object point, the image point and the projection center should be on the same beam of photographic light. Suppose the object point p_0 moves at a speed v_0 relative to the camera. The motion of object point p_0 causes the corresponding image point p_i to move at a speed v_i on the image plane. Suppose that during the time period δt, the point on the surface of the object moves by $\delta r_0 = v_0 \delta t$, and then the point p_i on the corresponding image moves by $\delta r_i = v_i \delta t$. Therefore, the speed of motion of the points is defined as

$$v_0 = dr_0 \big/ dt v_i = dr_i \big/ dt \tag{1}$$

where the relationship between r_0 and r_i can be determined by the perspective projection equation

$$\frac{1}{f'} r_i = \frac{1}{r_0^T \hat{z}} r_0 \tag{2}$$

Eq. (2) is used to guide t, and we can obtain

$$\frac{1}{f'}v_i = \frac{(r_0^T\hat{z})v_0 - (v_0^T\hat{z})r_0}{(r_0^T\hat{z})^2} = \frac{(r_0 \times v_0)\times\hat{z}}{(r_0^T\hat{z})^2} \tag{3}$$

Thus, through Eq. (3), a velocity vector can be assigned to the image point p_i, all of which constitutes the motion field.

Locally adjacent points on an object have similar velocities, and so the following inferences can be obtained:

Inference 1. In most areas of an image, the motion field obtained by Eq. (3) is continuous;

Inference 2. Exceptions occur only in the image area corresponding to the contour of the object, that is, the motion field is discontinuous in these areas.

2) MOTION FIELD SMOOTHING FUNCTION

The motion field in most areas of an image is continuous, and so every point in the motion field $\{p_i, v_i\}$ can be described using the function $f(p)$. The function $f(p)$ can be linearly combined by a series of smoothing functions $\{f_k(p)\}_{k=1,2,\ldots,K}$ [17], [37], [38]

$$v_i = f(p) = \sum_{k=1}^{K} a_k(p)f_k(p) \tag{4}$$

where $a_k(p)$ is the corresponding weight function.

The observation data $\hat{v}_i$ corresponding to the image point p_i contain noise, and so Eq. (4) can be corrected to

$$\hat{v}_i = f(p_i) + n_i = \sum_{k=1}^{K} a_k(p_i)f_k(p_i) + n_i \tag{5}$$

where n_i is the noise. According to the research in Ref. [28], the series smoothing function $f_k(.)$ can be composed of the following two items:

$$f_k(p) = H_k + \phi_k(p) \tag{6}$$

where H_k is the offset and $\phi_k(p)$ is the smoothing function under the motion consistency constraint. For the entire motion field function $f(p)$, the following energy minimization conditions should be met:

$$\begin{aligned} E(f(p), v) &= \sum_i \Phi(\hat{v}_i - f(p_i)) + \lambda \sum_{k=1}^{K} \Psi_k \\ &= \sum_i \Phi(\hat{v}_i - \sum_{k=1}^{K} a_k(p_i)f_k(p_i)) + \lambda \sum_{k=1}^{K} \Psi_k \end{aligned} \tag{7}$$

where λ is the corresponding weight value and $\Phi(.)$ is the Huber function

$$\Phi(z) = \begin{cases} z^2 & \|z\| \le \varepsilon \\ 2\varepsilon \|z\|_1 - \varepsilon^2 & \|z\| > \varepsilon \end{cases} \tag{8}$$

$\Psi_k(.)$ is the motion consistency constraint imposed on the function $\phi_k(.)$ to suppress the noise n in the observation data $\hat{v}$

$$\Psi_k = \int \frac{\left|\bar{\phi}_k(w)\right|^2}{\bar{g}(w)} dw \tag{9}$$

where $\bar{\phi}_k(.)$ is the Fourier transform of the function $\phi_k(.)$, and $\bar{g}(.)$ is the Fourier transform of the Gaussian function with the standard deviation γ.

The motion field smoothing function describes the consistency of the local-pixel motion of the image. Therefore, the motion field smoothing function can be used to apply motion constraints to image points. The partial image points are used to solve the motion field smoothing function, and then the solution function is used to constrain all image points so that the mismatches can be rejected.

B. OUR APPROACH

The motion field smoothing function mentioned earlier is complex. If we directly use it to perform point matching, the costs of solving the model parameters are expensive. Therefore, further consideration needs to be given to how to simplify the overall solution process and obtain a practical solution. In this section, the parallax mapping method is used to directly skip the model-solving step and refine the matching points.

1) CORRESPONDING POINT PARALLAX

Consider a simplified stereo camera imaging model and assume that the optical axes of the two cameras are parallel to each other and perpendicular to the photographic baseline. The origin of the coordinate system is selected as the midpoint of the photographic baseline. For the object points whose coordinates are $(x, y, z)^T$ in the scene, the corresponding image points are (x_l', y_l') and (x_r', y_r'), respectively. Then, the following can be obtained from the perspective projection equation:

$$\frac{x_l'}{f'} = \frac{x + b/2}{z}\frac{x_r'}{f'} = \frac{\text{x-b/2}}{z} \tag{10}$$

and,

$$\frac{y_l'}{f'} = \frac{y_r'}{f'} = \frac{y}{z} \tag{11}$$

where f' is the focal length. The following can be obtained by combining Eq. (10) and Eq. (11).

$$x = b\frac{x_l' + x_r'}{2(x_l' - x_r')} \quad y = b\frac{y_l' + y_r'}{2(x_l' - x_r')} \quad z = b\frac{f'}{x_l' - x_r'} \tag{12}$$

The transformation relation between object point coordinates and image point coordinates in the scene is described in Eq. (12), where the shift $\Delta x' = x_l' - x_r'$ is called the parallax. More generally, for ordinary stereo images, the coordinate difference between two image points $\Delta x' = x_l' - x_r'$ is the left-right parallax and $\Delta y' = y_l' - y_r'$ is the up-down parallax. The image parallax is related to the object coordinates when the photographic baseline b and the photographic focal length f' remain unchanged. The smaller the parallax is, the farther the object points are from the camera; and the larger the parallax is, the closer the object points are to the camera. The change of the parallax reflects the change of the object points and the change of the scene motion field.

Therefore, the parallax of the image can be used to represent the overall motion field of the scene.

2) PARALLAX MAPPING-BASED REJECTING MISMATCHES

The essence of matching purification between images is to use known constraints to eliminate mismatches. The parallax $(\Delta x', \Delta y')$ between the corresponding points (p_l, p_r) reflects the motion of the object relative to the camera and characterizes the overall motion field v of the image. There is a consistency constraint between the motion fields of the image, as described by the motion-smoothing function $f(p)$. Therefore, the motion-smoothing function $f(p)$ can be used to refine the matching points. However, the motion-smoothing function is complex and is difficult to solve. Ref. [17] simplifies the motion-smoothing function and designs an algorithm to solve the motion-smoothing function, but the time consumption cannot be ignored. The purpose of solving the motion-smoothing function is to constrain the matching points and achieve matching purification. That is, the purified matching points can realize the optimal solution of the motion-smoothing function. Therefore, we can find the matching points that can satisfy the conditions of the motion-smoothing function, skip the function-solving step, and directly purify the matching points. Finding sufficient conditions or approximate equivalence conditions for the solution of the motion-smoothing function can achieve the matching purification of matching points. Combining the motion-smoothing function $f(p)$ and the image parallax, the motion consistency constraint of the local pixel in the image can be converted into the following two conditions.

Condition 1. The parallax of the matching points should have a certain continuity.

For two adjacent correct matching points (x_1, y_1) and (x_2, y_2), the corresponding matching coordinates are (x_1', y_1') and (x_2', y_2'), and the corresponding parallax coordinates are$(\Delta x_1 = x_1 - x_1', \Delta y_1 = y_1 - y_1')$ and $(\Delta x_2 = x_2 - x_2', \Delta y_2 = y_2 - y_2')$.

Condition 1 can be expressed as follows:

$$\sqrt{(\Delta x_1 - \Delta x_2)^2 + (\Delta y_1 - \Delta y_2)^2} < \gamma \tag{13}$$

This condition characterizes the continuity between the motion fields and describes the consistency of the motion between images. This condition is characterized by the corresponding parallax map having a certain continuity or local-clustering characteristics. The parallax map of the image is formed by mapping the left-right parallax values of the matching point to the abscissae and the up-down parallax values to the ordinates.

Condition 2. The coordinates of the matching points should have a certain continuity.

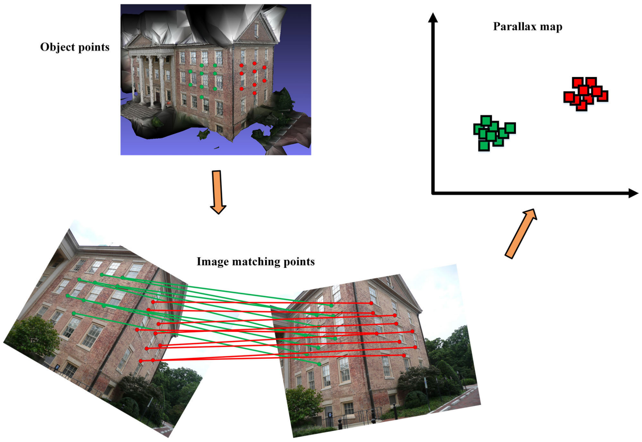


FIGURE 2. Parallax mapping.

For the above two adjacent correct matching points, condition 2 can be expressed as follows:

$$\begin{cases} \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} < \lambda \\ \sqrt{(x_1' - x_2')^2 + (y_1' - y_2')^2} < \lambda \end{cases} \tag{14}$$

This condition characterizes the continuity between the object points, confining the image points within the local area. This condition is expressed as the corresponding pixel coordinates having a certain continuity or local clustering, that is, there should be no abnormal outliers.

In Fig. 2, the red dots and the green dots represent feature points. For the object points with local consistency, the image point coordinates have a certain continuity, and the corresponding parallax map also has a certain continuity.

The above two conditions approximate the motion consistency and can satisfy the solution of the motion-smoothing function. By using these two conditions as constraints, the purification between matching points can be achieved. Next, we will focus on how to integrate these two conditions into the image matching process.

The existing feature extraction and description operators can be used to obtain rich original matching points, and most of the points have good continuity with the coordinates. Therefore, it is possible to apply the constraint from Condition 1 and then the constraint from Condition 2. To improve the motion consistency constraint of these matching points, first, the continuity value of the parallax value can be judged, and then the point coordinate continuity is judged, as follows.

1) After matching the points, calculate the parallax value of the initial matching points of one image with respect to the other image by using one image as a reference.
2) Calculate the parallax values. The left-right parallax values are the abscissae, and the up-down parallax values are the ordinates. Then, project these values to the Cartesian coordinate system to obtain the parallax map.
3) Identify the connected domain in the parallax map and find the connected domain in the parallax map that is

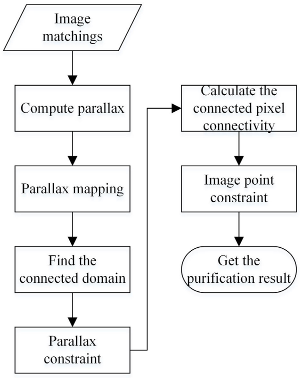


Image matchings
Compute parallax
Parallax mapping
Find the connected domain
Parallax constraint
Calculate the connected pixel connectivity
Image point constraint
Get the purification result

FIGURE 3. Parallax mapping-based mismatches rejection flow chart.

greater than the set threshold α as the rough purification result.

4) Determine whether the coordinates of the matching points in the selected cumulative connected domain have a certain continuity and eliminate the outliers to obtain the final purification result.

In the above steps, the connected domain in step 3) characterizes the continuity of the parallax value, and satisfies Condition 1. The connected domain statistics should consider the gray values of the parallax map, where the gray value is the cumulative number of mapping parallax points. In step 4), the judgment of the coordinate continuity of the matching points satisfies Condition 2, and when the continuity judgment is performed, since the values of the matching points are discrete, the actual operation may be performed according to whether the point is a clustering point. The nonclustering points are eliminated as outliers to complete the constraint purification of the matching points. The overall process is shown in Fig. 3.

3) OPTIMIZATION OF THE MISMATCHES REJECTION VIA GENERALIZED POLYNOMIAL MAPPING

Parallax mapping can be used to achieve fast feature matching. To achieve higher accuracy, further geometric constraints can be used to perform secondary matching purification between image points. Commonly used geometric constraints include the translational transformation, the affine transformation, the perspective transformation, epipolar constraints, and homography constraints. The essence of geometric constraints is to establish a suitable geometric model to constrain matching points. However, the existing geometric constraint relationships are mostly designed for a particular situation and cannot be adapted to a wider range of situations. For example, in the case of epipolar constraints, an epipolar model between images is needed, but there is no strict epipolar constraint relationship between the images acquired by the line sensor. The homography constraint is used to constrain the image in the same plane, but it is not applicable to the case where the scene depth change is complicated. Therefore, this section attempts to design a more generalized polynomial mapping model to achieve perfect matchings for in multiple types of images.

Consider the motion consistency constraint. For the corresponding points (x'_l, y'_l) and (x'_r, y'_r), there is a locally continuous parallax $(\Delta x', \Delta y')$, resulting in

$$\begin{cases} x'_l = \Delta x' + x'_r \\ y'_l = \Delta y' + y'_r \end{cases} \tag{15}$$

According to Weierstrass's first approximation theorem [39], there is a polynomial approximation for the continuous function on a closed interval. A polynomial approximation can be used to describe the relationship between a local point and the parallax. Then, we can describe the local continuous parallax $(\Delta x', \Delta y')$ as a cubic function of the image point (x'_r, y'_r) as

$$\begin{cases} \Delta x' = a_0 + a_1 x'_r + a_2 y'_r + a_3 x'_r y'_r + a_4 x'^2_r + a_5 y'^2_r \\ \qquad + a_6 x'^2_r y'{}_r + a_7 x'_r y'^2_r + a_8 x'^3_r + a_9 y'^3_r \\ \Delta y' = b_0 + b_1 x'_r + b_2 y'_r + b_3 x'_r y'_r + b_4 x'^2_r + b_5 y'^2_r \\ \qquad + b_6 x'^2_r y'_r + b_7 x'_r y'^2_r + b_8 x'^3_r + b_9 y'^3_r \end{cases} \tag{16}$$

Bringing Eq. (16) into Eq. (15), the constraint relationship between the local corresponding points (x'_l, y'_l) and (x'_r, y'_r) can be obtained. Thus, the generalized geometric constraint model can be defined by motion consistency theory as follows:

$$\begin{cases} x'_l = \bar{a}_0 + \bar{a}_1 x'_r + \bar{a}_2 y'_r + \bar{a}_3 x'{}_r y'{}_r + \bar{a}_4 x'^2_r + \bar{a}_5 y'^2_r \\ \qquad + \bar{a}_6 x'^2_r y'_r + \bar{a}_7 x'_r y'^2_r + \bar{a}_8 x'^3_r + \bar{a}_9 y'^3_r \\ y'_l = \bar{b}_0 + \bar{b}_1 x'_r + \bar{b}_2 y'_r + \bar{b}_3 x'_r y'_r + \bar{b}_4 x'^2_r + \bar{b}_5 y'^2_r \\ \qquad + \bar{b}_6 x'^2_r y'_r + \bar{b}_7 x'_r y'^2_r + \bar{b}_8 x'^3_r + \bar{b}_9 y'^3_r \end{cases} \tag{17}$$

Eq. (17) defines the motion consistency relationship between local corresponding points. The essence of this relationship is to simplify the motion-smoothing function and describe it as a relatively simple local polynomial, which can allow for more stringent purification optimization of the matching points. Since the matching points have been preliminarily purified, it is faster and easier to determine the polynomial optimization solution. The overall process is shown in Fig. 4.

Therefore, the overall PMM process is as follows. First, the initial feature matching is performed for the input image pairs. Then, the parallax map is calculated, and the fast connected domain analysis is performed by using the two-pass algorithm. Since there are matching points corresponding to the connected domains whose accumulated numbers are greater than the selected threshold, the Condition 1 constraint is implemented. Then, the image is divided into grids, and the image points falling in each grid are counted. If the accumulated numbers in the grid are greater than the threshold, it is considered that the image points in the grid have clustering characteristics, and so the Condition 2 constraint is realized. Finally, using the polynomial constraint model, the matching points in each connected domain are constrained, the error matrix is computed, the polynomial parameters are solved

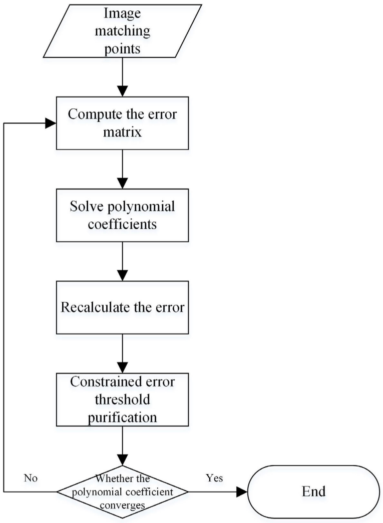


Image matching points
Compute the error matrix
Solve polynomial coefficients
Recalculate the error
Constrained error threshold purification
Whether the polynomial coefficient converges
No
Yes
End

FIGURE 4. Polynomial purification.

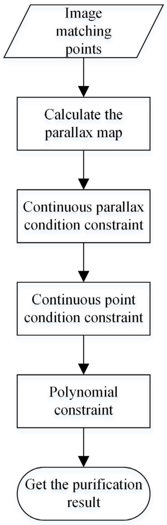


Image matching points
Calculate the parallax map
Continuous parallax condition constraint
Continuous point condition constraint
Polynomial constraint
Get the purification result

FIGURE 5. Overall PMM flow chart.

via singular value decomposition, and the fast purification of points in each connected domain is achieved, thereby completing feature matching. The overall process is shown in Fig. 5.

C. IMPORTANT DESIGN CONSIDERATIONS AND PROPERTIES

Parallax map clustering. In practical applications, a large number of initial matching points should be obtained so that relatively continuous local feature points or parallax maps can be formed. However, in the case where the partially extracted matching points are relatively discrete, a good mapping result may not be formed on the parallax map. Therefore, for this case, in the parallax mapping, the mapping can be performed by means of region expansion. For example, if the point (x', y') has a parallax mapping coordinate $(\Delta x', \Delta y')$, the point $(\Delta x', \Delta y')$ in the parallax map is accumulated when mapping, but to extend the parallax continuity, the neighborhood around the point $(\Delta x', \Delta y')$ can be accumulated, as shown in Fig. 6. This kind of regional extension mapping, which is similar to the expansion operation in image morphology, can better extend the local continuity of the parallax map. Similarly, when the Condition 2 constraint is imposed, the extracted feature point coordinates are not continuous. In this case, the point clustering method can be used to eliminate the outliers and improve the overall operational robustness, as shown in Fig. 7.

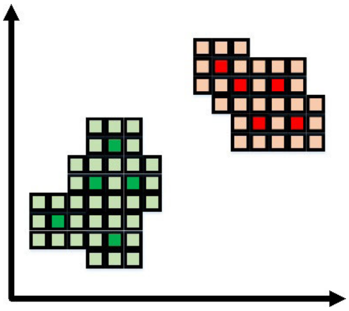


FIGURE 6. Parallax mapping extension. The dark-colored dots represent the original mapping. The light-colored dots represent the extended mapping.

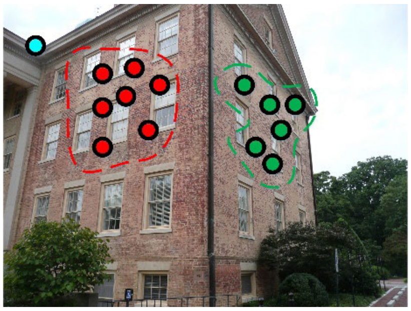


FIGURE 7. Point clustering. The cyan dots represent outliers, and the red and green dots represent inliers.

Quadratic purification of local generalized polynomial constraints. The matching based on parallax mapping, which has been able to perform well, can meet the requirements of most applications. However, in some applications where the accuracy requirement is more stringent, such as camera calibration, polynomial quadratic optimization can be performed to improve the robustness of the matching results. The local generalized polynomial is essentially the simplification of the motion-smoothing function in the local image region. In practice, generalized polynomial constraints can be used for different connected domains in the parallax map to reflect the locality. In addition, the matching and purification operations have been performed via Condition 1 and Condition 2, respectively, and the remaining matching points have a high accuracy rate. When the generalized polynomial is solved, the solution process is also very fast and consumes relatively small computing resources. In addition, in this paper, the cubic polynomial is used as the polynomial function. We use this particular polynomial because when the degree of the polynomial is too high, the Longge phenomenon easily occurs; and when the degree of the polynomial is too low, it cannot adequately describe the constraints between matching points.

D. IMPLEMENTATION DETAILS

Condition 1 constraint implementation details. To achieve the Condition 1 constraint and to speed up the connectivity of the parallax map after the mapping, we use the two-pass method [40] to conduct fast analysis of the connected domain.

Condition 2 constraint implementation details. When the Condition 2 constraint is implemented, we use the fast coordinate-grid judgment method to determine whether the points are continuous due to the discreteness of the point coordinates. The local consistency judgment is performed on the image points after the Condition 1 constraint is implemented. Because of the discreteness of the matching points, the image is divided into a grid. If the number of image points falling within the grid is greater than the threshold, then the local image points falling within the grid are considered to have continuity; otherwise, the rejection is performed.

Polynomial constraint implementation details. When applying a polynomial fitting constraint, the number of correctly matched points is the main component. Here, the singular value decomposition method is used for rapid iterative optimization. The specific algorithm comes from our previous research [9].

III. EXPERIMENTAL RESULTS

Image matching is mainly used to analyze the correspondence between image pairs in the same scene. Image matching has three main functions [41] in close range photogrammetric applications: similarity measurement, which is mainly used for image retrieval, target retrieval, target recognition and geolocation; geometry estimation, which is mainly used for camera calibration and other aspects; and data association, which is mainly used for 3D reconstruction, simultaneous localization and mapping (SLAM), image stitching, and visual tracking. Based on these functions, to discover the practical application requirements of the image matching algorithm, we used a variety of datasets and multiple indicators for the evaluation and analysis.

A. DATASETS

Four datasets were used for the evaluation, including ordered video images and unordered images. Ordered video images include ordered indoor video images and ordered outdoor video images; and the unordered images include unordered single-source images and Internet images. The ordered indoor video images, ordered outdoor video images, and unordered single-source image data are provided in Ref. [41], and the unordered Internet image data are downloaded from Flickr. The specific dataset information is shown in Table 1 and Fig. 8.

TABLE 1. Dataset information.

Serial number	Name	Number of images	Number of image pairs	Description	Sources
1	Office	1000	933	Indoor video data, rich scenes and strong texture, 480x640	TUM
2	Street	1000	800	Outdoor video data, car camera shooting, street view perspective, 376x1241	KITTI
3	Castle	30	435	Urban photography, repeated textures, wide baselines, 3072x2048	Strecha
4	Church	30	435	Downloaded image data from Flickr, source is complicated, size is inconsistent	Flickr

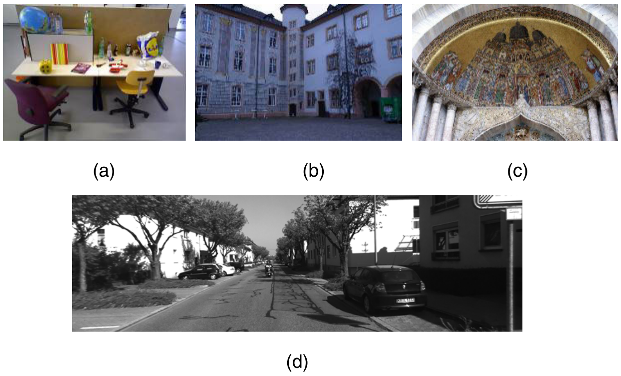


FIGURE 8. Datasets. (a) Office. (b) Castle. (c) Church. (d) Street.

Among them, the Office data are taken from the TUM dataset [42], which are indoor order video image data; the Street data are taken from the KITTI benchmark [43] which are outdoor video ordered image data; the Castle data are taken from the Strecha dataset [44], which are single source unordered image data; and the Church data are downloaded from Flickr, and the source is complex and unordered. In the image matching experiments, the image pair was constructed using the method from Ref. [41]. For the ordered Office data and the Street data, the original video sequences were divided into n nonoverlapping fragments, and each fragment contained k frames. The first image of each fragment is set as the reference image, which is matched to the next $k - 1$ frames. Among them, the Office data original video frame rate is 30 fps, and k is set to 15. The Street data original video frame rate is 10 fps, and k is set to 5. For the unordered Castle data and Church data, a pairwise image traversal matching method is used to construct image pairs for the matching experiments. These four datasets can effectively cover multiple close range photogrammetric image matching applications, and the experiment can be used to comprehensively characterize the performance of the image matching algorithm in practical applications.

B. EVALUATION METRICS

Regarding the evaluation metrics, considering the accuracy, sufficiency, and efficiency requirements of the algorithm, the matching algorithm used in the overall image matching process is evaluated by three metrics: the correct rate, the loss rate and the time consumption.

The correct rate is the percentage of correctly matched points out of all matching points. The loss rate is the percentage of points that are ultimately lost out of all the original correctly points. The time consumption is the total time required for the overall matching process. From the definition, the higher the correct rate is, the lower the loss rate is and the shorter the time consumption is, the better the performance of the algorithm. The key to evaluating the performance is being able to distinguish the correctly matched points. Since the number of matching points in the experiments is large, it is difficult to carry out traditional manual identification. Here, a triangulated back-projection error constraint is used to determine whether the matching points are correct or not.

Knowing the camera intrinsic parameters, rotation vector and translation vector of the two images, the matching points (x_l', y_l') and (x_r', y_r') can be triangulated to calculate the object point coordinate $(\bar{x}, \bar{y}, \bar{z})$. Then, back-project the calculated object points to the image to obtain the back-projection image points $(\bar{x}_l', \bar{y}_l')$ and $(\bar{x}_r', \bar{y}_r')$. Next, calculate the difference between the coordinates of the back-projection image points and the original image points, $(\delta x_l' = \bar{x}_l' - x_l', \delta y_l' = \bar{y}_l' - y_l')$ and $(\delta x_r' = \bar{x}_r' - x_r', \delta y_r' = \bar{y}_r' - y_r')$, respectively. Thus, we can calculate the triangulated back-projection error of the matching points as

$$\delta r = \max(\sqrt{(\delta x_l')^2 + (\delta y_l')^2}, \sqrt{(\delta x_r')^2 + (\delta y_r')^2}) \tag{18}$$

For ideal conditions, the back-projection error of the matching points should be zero, but due to the existence of matching errors, parameter errors and calculation errors, the real back-projection error should be small. Therefore, it is possible to determine whether the matching points are correct or not by setting different triangulated back-projection error thresholds ε. In the experiment, the Office, Street, and Castle datasets have real camera intrinsic parameters, rotation vectors, and translation vectors that can be directly used. The Church dataset consists of Internet images, and its intrinsic camera parameters, rotation vector and translation vector are obtained by the COLMAP [45], [46] adjustment.

C. EVALUATION SETUP

Our evaluation used 15 feature algorithms, including 15 feature extraction and description algorithms and 2 initial matching algorithms. The feature extraction and description algorithms include the SIFT [35], SURF [47], ORB [48], AKAZE [49], KAZE [50], BRISK [51], DLCO [52], LATCH [53], FREAK [54], DAISY [55], BinBoost [56], MSD [57], STAR [58], FAST [59], and AGAST [60]. The initial matching algorithms include the FANN and BF. The specific comparison methods are shown in Table 2.

TABLE 2. List of the experimental initial matching algorithms.

Serial number	Extractor	Descriptor	Initial Matcher
1	SIFT	SIFT	FANN
2	SURF	SURF	FANN
3	ORB	ORB	BF (HAMMING)
4	AKAZE	AKAZE	BF (HAMMING)
5	KAZE	KAZE	FANN
6	BRISK	BRISK	BF (HAMMING)
7	SIFT	DLCO	FANN
8	SIFT	LATCH	BF (HAMMING)
9	SIFT	FREAK	BF (HAMMING)
10	SIFT	DAISY	FANN
11	SIFT	BinBoost	BF (HAMMING)
12	MSD	ORB	BF (HAMMING)
13	STAR	SIFT	FANN
14	FAST	SIFT	FANN
15	AGAST	SIFT	FANN

In the experiment, RATIO, GMS, K-VLD, RANSAC, LPM, RFM-SCAN and our algorithm were used to purify the initial matching points. Our algorithm is divided into four types: PMM1, in which only the Condition 1 consistency constraint is used for matching; PMM2, in which all consistency constraint conditions are used for matching; PMM3, in which the Condition 1 consistency constraint and the polynomial optimization are used to reject mismatches; and PMM4, in which all consistency constraint conditions and the polynomial optimization are used to reject mismatches.

We set the cumulative numbers threshold in the parallax map connected domain to $\alpha = 10$. Points are considered as continuous clustering points if the number of local clusters is greater than 2. The polynomial error threshold is set to 3, the triangulated back-projection error threshold is set to $\varepsilon = 3$, and the RATIO threshold is set to 0.8. RANSAC uses the fundamental matrix model, and the error threshold is set to 3. GMS, K-VLD, LPM and RFM-SCAN use the default parameters. For all feature extraction and matching algorithms, the triangulated back-projection error calculations are implemented by using OpenCV functions with their default thresholds. Among the ORB feature extraction parameters, we set the maximum number of features detected. We set the number to 100,000 to remove this limitation to a fair comparison. The experimental platform is configured with an i7-CPU and 16 GB of memory, and our algorithm is implemented in C++ and Matlab. Due to the memory limitations of the experimental platform, the K-VLD algorithm is not used for the experiment when the number of original matching points is greater than 20,000 in the fourth dataset.

D. EXPERIMENTAL RESULTS

The experiments were performed using the four datasets. The experimental results are shown in Figs. 9 to 10 and Tables 3 to 6. Fig. 9 shows the matching results of the 10 different feature matching algorithms and 15 feature algorithms for the four datasets. To facilitate the visual statistical analysis, the matching results of all image pairs constructed in each dataset are averaged and then visualized. Fig. 9(a) shows the experimental results of dataset 1, (b) shows the experimental results of dataset 2, (c) shows the experimental results of dataset 3, and (d) shows the experimental results of dataset 4. In the experimental results of each dataset in Fig. 9,

TABLE 3. Time consumption statistics of the experiment with dataset 1 (millisecond).

Algorithm	Number of initial matching points	Initial correct rate	GMS	K-VLD	RANSAC	LPM	RFM-SCAN	PMM1	PMM2	PMM3	PMM4
SIFT	642.15	0.63	0.68	296.75	3.44	25.67	54.88	0.84	0.89	16.54	16.49
SURF	1088.38	0.79	0.82	547.96	0.97	29.86	124.61	0.66	0.70	18.24	16.10
ORB	2560.27	0.85	0.88	1085.84	1.36	40.95	523.68	0.97	1.13	16.39	16.22
AKAZE	621.53	0.89	0.69	362.80	0.30	27.40	58.94	0.48	0.62	16.30	15.96
KAZE	835.55	0.84	0.75	456.18	0.52	26.83	88.92	0.53	0.62	18.55	16.94
BRISK	778.49	0.74	0.68	417.13	1.77	26.33	76.27	0.72	0.77	16.27	14.85
DLCO	640.91	0.63	0.72	301.96	3.56	25.61	54.27	0.88	1.01	16.68	16.25
LATCH	597.52	0.76	0.66	324.65	1.23	24.64	52.81	0.71	0.87	15.22	14.68
FREAK	581.57	0.68	0.68	283.41	1.87	26.36	49.93	0.76	0.91	16.00	15.64
DAISY	654.25	0.82	0.71	368.10	0.73	24.99	60.68	0.62	0.76	15.83	14.64
BinBoost	636.06	0.62	0.69	296.41	3.93	21.40	50.55	0.95	1.04	15.85	15.62
MSD	477.57	0.89	0.58	308.35	0.28	22.53	37.99	0.48	0.62	13.75	14.10
STAR	203.04	0.86	0.61	161.44	0.27	20.19	15.68	0.23	0.37	13.48	13.18
FAST	1522.74	0.94	0.83	751.18	0.47	31.78	219.33	0.56	0.64	15.46	14.71
AGAST	1357.29	0.94	0.81	717.14	0.41	30.32	180.48	0.50	0.60	15.22	14.54

TABLE 4. Consumption statistics of the experiment with dataset 2 (millisecond).

Algorithm	Number of initial matching points	Initial correct rate	GMS	K-VLD	RANSAC	LPM	RFM-SCAN	PMM1	PMM2	PMM3	PMM4
SIFT	1170.69	0.48	0.81	463.26	18.49	27.72	117.62	1.42	1.43	30.69	29.22
SURF	1642.10	0.53	0.92	712.61	21.25	31.97	205.98	1.68	1.75	37.21	35.29
ORB	5706.50	0.61	1.22	2641.68	48.11	58.26	2073.55	4.95	5.20	72.26	72.75
AKAZE	950.39	0.63	0.73	562.45	9.04	25.88	94.89	1.11	1.16	31.47	29.73
KAZE	1283.08	0.61	0.82	784.08	12.43	27.83	148.17	1.23	1.29	41.11	38.21
BRISK	2058.51	0.56	0.83	912.67	22.51	32.95	307.55	1.96	2.01	42.03	41.53
DLCO	1162.09	0.46	0.80	462.46	19.42	26.01	114.70	1.48	1.48	29.44	27.82
LATCH	1001.82	0.54	0.75	491.57	13.94	25.12	96.44	1.29	1.30	29.05	27.25
FREAK	994.37	0.49	0.73	423.55	15.01	25.24	92.82	1.30	1.32	28.11	26.71
DAISY	1148.05	0.67	0.77	706.86	9.29	27.01	126.72	1.29	1.27	38.69	35.81
BinBoost	1148.53	0.50	0.79	492.11	16.50	26.52	114.86	1.36	1.38	31.49	29.94
MSD	744.18	0.70	0.65	492.06	5.55	24.54	68.51	1.12	1.22	28.47	27.26
STAR	345.22	0.71	0.64	327.54	3.09	21.96	22.28	0.70	0.81	15.92	15.73
FAST	4043.16	0.77	1.12	2663.53	17.54	49.78	1155.12	3.06	3.32	91.40	90.88
AGAST	3821.02	0.76	1.07	2500.93	18.32	47.92	1037.51	3.07	3.22	88.80	88.17

the figure on the left shows the statistical results of the final matching points. The horizontal axis is the 15 feature algorithms, and the vertical axis is the average number of final matching points for each feature algorithm. The middle figure shows the correct rate results of the final matching points. The horizontal axis is the 15 feature algorithms, and the vertical axis is the average correct rate for each feature algorithm. The figure on the right shows the loss rate of the final matching points. The horizontal axis is the 15 feature algorithms, and the vertical axis is the average loss rate for each feature algorithm. The result in Fig. 9 is the average of the experimental results of all the constructed image pairs in each dataset using the different feature matching methods with different feature algorithms. Fig. 10 shows the visual results of the four datasets using PMM1 and the SIFT feature algorithm.

TABLE 5. Consumption statistics of the experiment with dataset 3 (millisecond).

Algorithm	Number of initial matching points	Initial correct rate	GMS	K-VLD	RANSAC	LPM	RFM-SCAN	PMM1	PMM2	PMM3	PMM4
SIFT	2417.68	0.26	0.74	2941.72	45.16	33.49	369.19	3.84	3.77	56.06	54.10
SURF	8658.74	0.24	1.64	7435.46	144.24	85.16	4426.56	14.77	15.01	150.15	147.26
ORB	10549.07	0.20	1.68	9119.63	189.39	102.36	6267.24	15.59	15.98	174.13	178.29
AKAZE	3528.42	0.24	0.90	4139.48	60.81	39.94	704.00	5.18	5.16	88.64	88.50
KAZE	3096.64	0.20	1.10	4745.01	57.33	36.89	554.93	4.36	4.62	77.45	78.08
BRISK	2870.17	0.18	0.92	4060.10	56.89	35.38	479.20	4.32	4.37	51.31	50.29
DLCO	2379.45	0.24	1.11	3014.35	45.01	33.28	362.00	3.52	3.38	51.31	49.52
LATCH	2398.77	0.23	1.00	3345.80	44.12	33.76	361.13	3.56	3.31	60.63	59.19
FREAK	2259.57	0.20	0.78	3179.52	45.43	33.88	319.61	3.55	3.25	45.36	43.85
DAISY	2099.51	0.29	0.56	2969.41	36.21	33.40	321.22	2.90	3.48	65.28	62.03
BinBoost	2378.85	0.25	0.95	2980.03	44.81	34.30	361.13	4.51	3.71	59.13	55.78
MSD	1173.45	0.21	0.48	3223.88	27.42	24.79	113.01	1.76	2.40	26.72	26.00
STAR	577.52	0.29	0.75	2228.10	16.00	23.33	41.37	1.47	1.88	25.14	23.27
FAST	4277.05	0.30	1.20	5012.51	67.14	45.99	1318.73	6.57	6.90	128.55	127.17
AGAST	4910.43	0.32	1.36	5519.48	71.44	52.83	1765.54	6.90	7.49	155.91	155.95

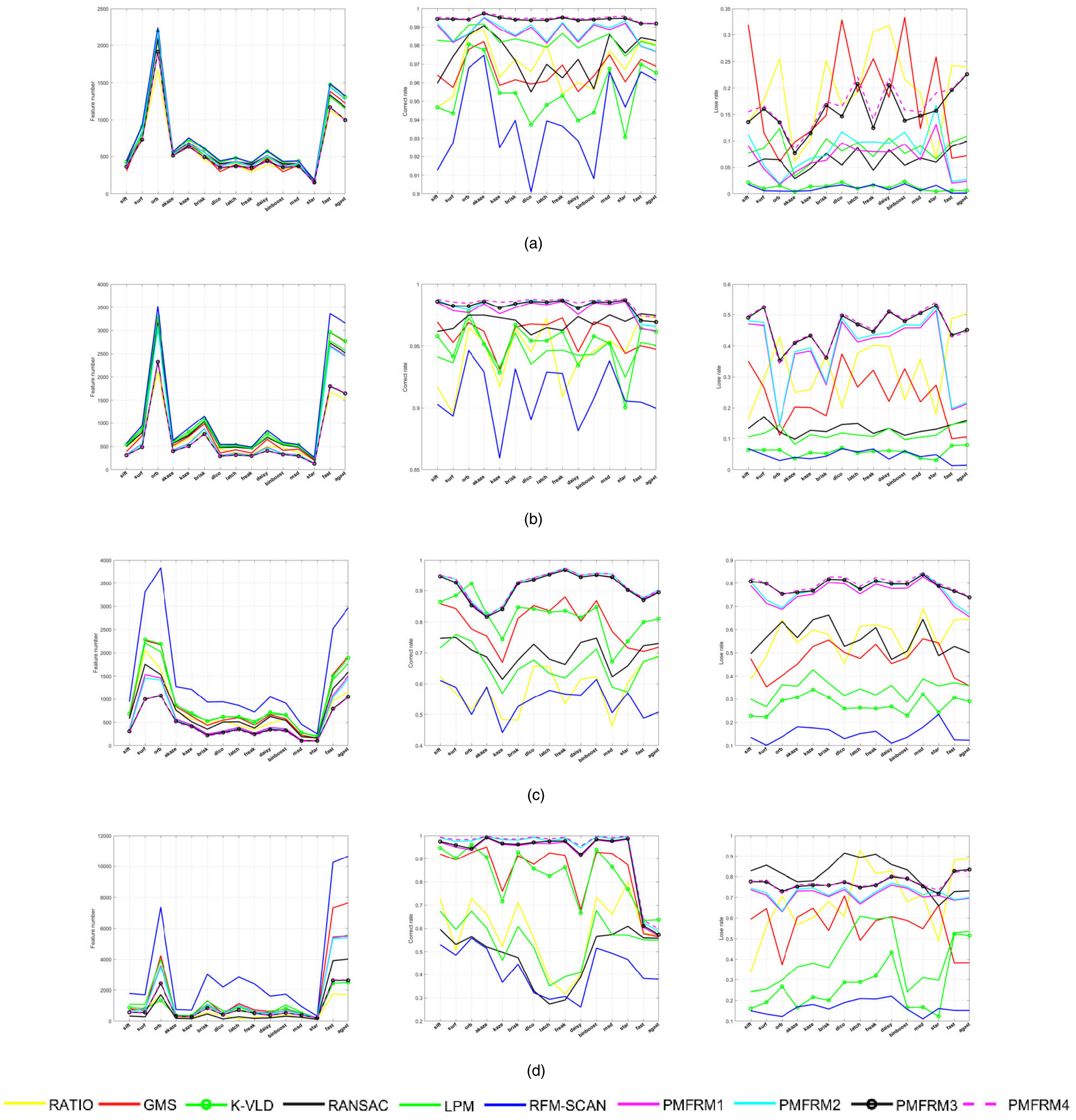
TABLE 6. Consumption statistics of the experiment with dataset 4 (millisecond).

Algorithm	Number of initial matching points	Initial correct rate	GMS	K-VLD	RANSAC	LPM	RFM-SCAN	PMM1	PMM2	PMM3	PMM4
SIFT	7639.32	0.11	1.40	3048.94	144.71	68.02	3209.68	12.82	12.97	99.90	96.87
SURF	7903.14	0.10	1.68	3106.10	148.76	69.35	3360.41	14.52	14.89	94.15	87.20
ORB	28055.58	0.13	5.06	8690.18	507.82	326.25	65720.22	80.61	63.10	426.86	450.75
AKAZE	2567.77	0.13	0.81	1744.95	54.71	32.45	425.31	4.86	2.51	55.37	54.43
KAZE	2202.08	0.13	1.24	1454.11	47.38	33.42	322.36	4.08	4.23	57.81	54.04
BRISK	10130.78	0.10	1.91	6382.80	195.28	94.21	6690.82	21.03	19.73	147.73	148.80
DLCO	7449.32	0.06	1.31	5179.79	139.43	59.60	3109.70	13.97	13.15	81.63	73.90
LATCH	6859.72	0.07	1.48	6529.24	125.84	52.45	2763.98	10.50	11.70	100.80	98.20
FREAK	6717.76	0.06	1.17	6925.03	126.65	51.71	2617.18	11.36	11.94	92.99	89.86
DAISY	5043.16	0.09	1.11	4679.49	99.23	49.27	1558.41	9.60	9.96	91.47	87.29
BinBoost	7156.01	0.11	1.71	2844.39	131.21	64.84	2801.99	11.40	11.20	97.60	96.20
MSD	2843.08	0.15	1.20	1568.58	57.00	34.64	521.59	4.49	5.61	65.12	61.19
STAR	824.09	0.20	1.22	731.38	21.15	25.00	73.72	1.78	1.19	45.63	39.70
FAST	21663.93	0.23	3.02	18655.65	344.26	262.64	43416.69	44.85	40.14	455.63	445.53
AGAST	22703.22	0.23	3.23	18144.14	352.26	268.09	42745.44	42.54	39.72	418.21	410.58

Tables 3 to 6 show the time consumption statistics of the GMS, K-VLD, RANSAC, LPM, RFM-SCAN and PMM feature matching algorithms using different feature algorithms on the four datasets. The first column in each table is the names of the different feature algorithms, the second column is the average number of initial matching points out of all the image pairs in the corresponding dataset, and the third column is the average initial correct rate out of all the image pairs in the corresponding dataset. The fourth through the tenth columns are the average time consumption for each of the different feature matching algorithms.

IV. DISCUSSION

In Fig. 9 and Tables 3 through 6, the results are the averages of the experimental results for all the image pairs constructed under the dataset, which increases the robustness of



(a)

(b)

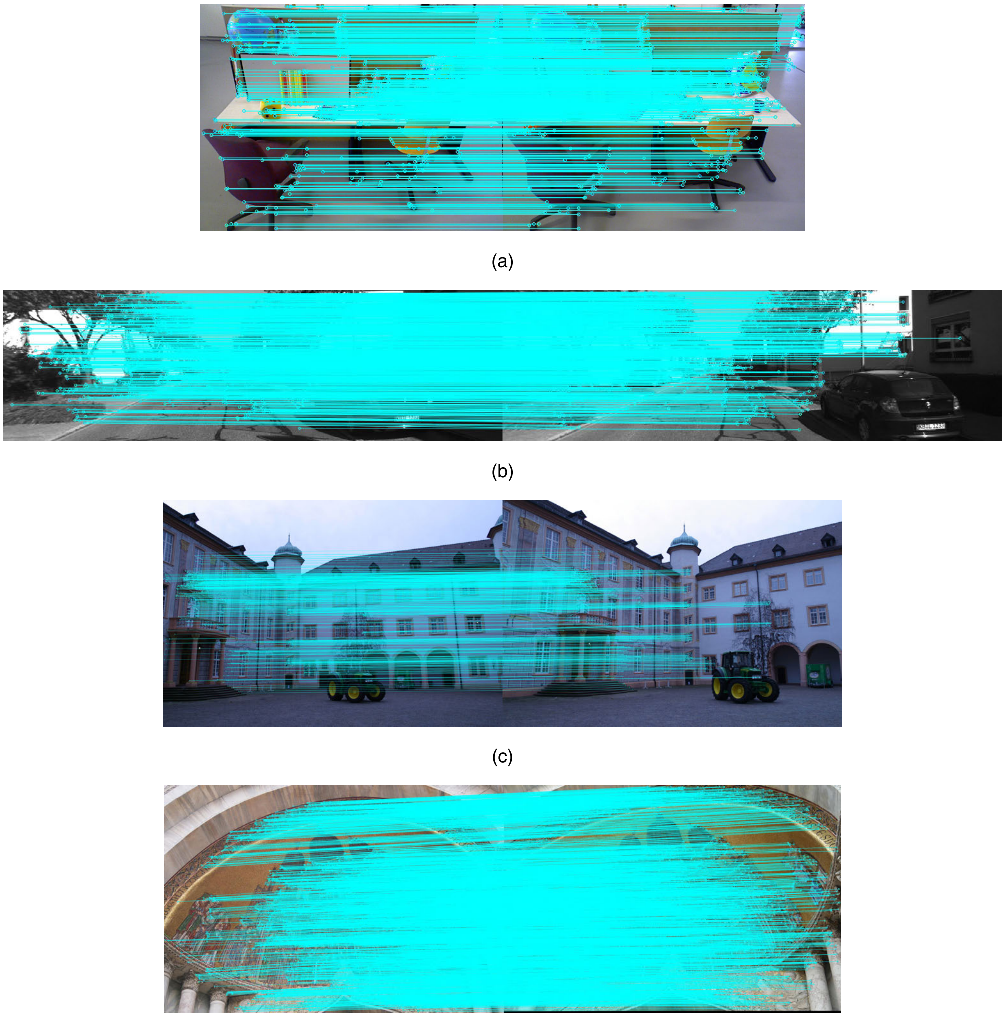
(c)

(d)

FIGURE 9. Statistical results of the experiments. (a) Ordered video image matching results on dataset 1 (TUM). (b) Ordered video image matching results on dataset 2 (KITTI). (c) Unordered image matching results on dataset 3 (Strecha). (d) Unordered image matching results on dataset 4 (Flickr).

the analysis and makes the experimental results have more practical referential significance. Fig. 9 shows that for the different datasets, the different feature matching algorithms have the same trend in the number of matching points obtained under different feature algorithms. Among the ordered video images (dataset 1 and dataset 2), the ORB, FAST, and AGAST feature algorithms in the experiment have more final matching points, and the STAR feature algorithm obtains the fewest matching points. In general, RFM-SCAN, LPM, K-VLD and PMM1 have many matching points, PMM3 and PMM4 have relatively small numbers of matching points. In the unordered image-matching experiments (dataset 3 and dataset 4), the SURF, ORB, and AGAST feature algorithms obtained more final matching points. In general, RFM-SCAN, LPM, GMS, PMM1 and PMM2 have many final matching points, RANSAC and RATIO have relatively small numbers of final matching points.

In terms of the correct rate of the matching points, Fig. 9 shows that, for the ordered video image dataset, the PMM series algorithms are better than the other algorithms,



(a)

(b)

(c)

(d)

FIGURE 10. Visualization of the experimental results. (a) Ordered video image matching results on dataset 1 (TUM). (b) Ordered video image matching results on dataset 2 (KITTI). (c) Unordered image matching results on dataset 3 (Strecha). (d) Unordered image matching results on dataset 4 (Flickr).

and the PMM3 and PMM4 algorithms have the highest correct rates. The RATIO and RFM-SCAN algorithms have relatively low correct rates. For the unordered image dataset, the correct rate of the PMM series algorithms are obviously better than those of other algorithms, and the RATIO and RFM-SCAN algorithms have the lowest correct rates. Fig. 10 shows that after using the PMM1 algorithm, the image matching result has a high correct rate and no obvious mismatches.

In terms of the loss rate of the matching points, Fig. 9 shows that for the first ordered video image dataset (dataset 1), the K-VLD and RFM-SCAN algorithms have significantly lower loss rates, and the differences in the remaining matching algorithms loss rates are not obvious for the multiple feature algorithms. In dataset 2, the loss rates of the PMM3 and PMM4 algorithms are significantly higher. For the unordered image dataset, the RFM-SCAN algorithm has a relatively low loss rate, and the RANSAC, PMM3, and PMM4 algorithms have relatively high loss rates.

In terms of the overall time consumption of the matching, it can be seen from Tables 3 through 6 that the GMS, PMM1 and PMM2 algorithms always maintain relatively lower time consumption in all experiments, and the K-VLD and RFM-SCAN algorithms have much higher time consumption than other algorithms.

It can be seen from Table 3 through 6 that there are fewer initial matching points for the ordered video image dataset than for the unordered image dataset, but the original correct rate is much higher than that of the unordered image dataset. This result occurred because the unordered image dataset has a higher image resolution and a richer texture, and so more feature points can be extracted. The content difference between the image pairs of the unordered image dataset is relatively large, and the content difference between the image pairs constructed by the ordered image dataset is relatively small; therefore, the initial matching correct rate of the ordered image dataset is relatively high. This point is also shown in Fig. 8. The number of initial matching points of dataset 1 through dataset 4 sequentially increases. For dataset 1, the time consumption of the PMM1 algorithm is lower than that of the GMS algorithm. As the number of initial matching points increases, the time consumption of GMS increases relatively slowly, and the time consumption of PMM1 increases relatively fast; therefore, the GMS time consumption is lower than the PMM1 time consumption for dataset 4. However, it is worth noting that the overall calculation time for PMM1 is still better than 25 Hz in dataset 4, which can meet real-time calculation requirements. Furthermore, the GMS algorithm and the PMM series algorithms have little correlation with the initial correct rate, which is determined by the characteristics of the algorithm. The RANSAC algorithm is obviously related to the number and the correct rate of the initial matching points, and is greatly affected by its fluctuation, which is determined by the random sampling characteristics of the algorithm. The time consumption of the LPM algorithm also increases as the matching points increases. The overall time consumption in the experiment is at a medium level. The time consumption of the RFM-SCAN algorithm increases sharply as the number of matching points increases. When the number of matching points exceeds 20,000 in dataset 4, the time consumption reaches tens of seconds. The K-VLD algorithm has obvious time consumption, which is determined by the complexity of its calculation process. Therefore, in the experiment with dataset 4, when the number of initial matching points is greater than 20,000, memory overflow occurs when the K-VLD algorithm is used for matching.

Overall, the PMM algorithm proposed in this paper has a relatively high matching accuracy and low time consumption. The performances of PMM1 and PMM2 are similar, but PMM2 is slightly better than PMM1 in terms of the correct rate, and it is slightly worse than PMM1 in terms of the loss rate and the calculation time consumption. The performances of PMM3 and PMM4 are similar, but PMM4 is slightly better than PMM3 in terms of the correct rate, and it is slightly worse than PMM3 in terms of the loss rate and the calculation time consumption. The overall correct rates of PMM3 and PMM4 are better than those of PMM1 and PMM2. PMM3 and PMM4 worse than PMM1 and PMM2 in terms of the loss rate and the calculation time consumption. We can achieve good matching performance by using Condition 1 only (PMM1), and the polynomial constraint calculation significantly increases the time consumption while improving the correct rate; furthermore, the “price ratio” is relatively low. Therefore, it is recommended that the PMM1 algorithm be used in general applications to achieve good performance. In the applications where the accuracy needs to be high, such as camera calibration, the PMM4 algorithm is recommended.

It can be seen from Figs. 9(a) and (b) that the PMM series algorithms have significantly lower loss rates when using the ORB algorithm, which is caused by the principle of the cumulative statistics of the parallax map in the algorithm. When the extracted feature points are dense, the performance of the PMM algorithm can be exerted. When the content of the image pairs have large differences, the extracted matching points are relatively discrete. At this point, if parallax mapping is being performed, it is difficult for the local correct matching points to form an effective connected domain, which ultimately leads to the motion consistency constraint, Condition 1, failing. This phenomenon can also be seen in the unordered image matching loss rates of Figs. 9(c) and (d). In practical applications, this deficiency can be compensated by a parallax mapping region expansion, as mentioned earlier. In general, the PMM algorithm proposed in this paper relies on the calculation results of the initial feature extraction and matching algorithm. For practical applications with some challenging conditions, such as changing lighting conditions and low contrast, the current mainstream algorithms, such as ORB, can obtain a large number of initial matches, and the algorithm presented in this paper can achieve good matching purification results. For scenes with very sparse structures or ‘featureless’ regions, it is still one of the current difficult challenges. The initial matching results of the current mainstream algorithms are very sparse, and our algorithm will fail, which is also the next research direction. Therefore, in terms of the correct rate, the loss rate and the time consumption, ORB-PMM1 is better than existing algorithms in ordered video image matching. For real-time calculations, the proposed algorithm outperforms the RATIO, GMS, K-VLD, RANSAC, LPM and RFM-SCAN algorithms in terms of the accuracy, achieving state-of-the-art results.

V. CONCLUSION

Starting with the principle of local motion consistency and combining it with parallax mapping, the motion consistency constraint of matching points is divided into two constraint conditions. On the basis of considering strict motion consistency constraints, the quadratic constraint is implemented by using local polynomials. Using a variety of datasets and current mainstream feature matching algorithms, rich feature matching experiments are designed. The experimental results effectively verify the feasibility of our algorithm. For real-time calculations, the proposed algorithm outperforms the RATIO, GMS, K-VLD, RANSAC, LPM and RFM-SCAN algorithms in terms of its accuracy. In the practical ordered video image matching application, it is recommended to use the algorithm combination of ORB-PMM1 to achieve good performance. Of course, our algorithm also has certain deficiencies. In the matching of unordered image datasets, and in the case of extracting feature points that are relatively discrete, there may be a relatively high loss rate. Solving this problem is the future direction of our research.

REFERENCES

[1] L. Yin, X. Wang, Y. Ni, K. Zhou, and J. Zhang, "Extrinsic parameters calibration method of cameras with non-overlapping fields of view in airborne remote sensing," *Remote Sens.*, vol. 10, no. 8, p. 1298, Aug. 2018.

[2] R. Almadhoun, A. Abduldayem, T. Taha, L. Seneviratne, and Y. Zweiri, "Guided next best view for 3D reconstruction of large complex structures," *Remote Sens.*, vol. 11, no. 20, p. 2440, Oct. 2019.

[3] D. Tapete and F. Cigna, "Detection of archaeological looting from space: Methods, achievements and challenges," *Remote Sens.*, vol. 11, no. 20, p. 2389, Oct. 2019.

[4] J. Ma, X. Jiang, J. Jiang, J. Zhao, and X. Guo, "LMR: Learning a two-class classifier for mismatch removal," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 4045–4059, Aug. 2019.

[5] J. Ma, H. Zhou, J. Zhao, Y. Gao, J. Jiang, and J. Tian, "Robust feature matching for remote sensing image registration via locally linear transforming," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6469–6481, Dec. 2015.

[6] M. Muja, and D. G. Lowe. (2017). *Fast Library for Approximate Nearest Neighbors*. [Online]. Available: http://www.cs.ubc.ca/research/flann/

[7] Y. Li, Q. Huang, Y. Liu, Y. Huang, and X. Sun, "Efficient properties-based learning for mismatch removal," *IEEE Access*, vol. 7, pp. 149612–149622, 2019.

[8] J. Chen, J. Ma, C. Yang, and J. Tian, "Mismatch removal via coherent spatial relations," *J. Electron. Imag.*, vol. 23, no. 4, Jul. 2014, Art. no. 043012.

[9] Y. Dong, D. Fan, S. Ji, and R. Lei, "The purification method of matching points based on principal component analysis," *Acta Geodaetica Cartographica Sinica*, vol. 46, no. 2, pp. 228–236, 2017.

[10] J. Zhong, Q. Du, and Y. Liu, "Robust method of skew correction based on feature points matching," *J. Image Graph.*, vol. 18, no. 7, pp. 738–745, 2013.

[11] L. Xie, S. Chen, J. Zhang, and J. Tian, "Purifying algorithm for rough matched pairs using Hough transform," *Int. J. Image Graph.*, vol. 20, no. 8, pp. 1017–1025, 2015.

[12] W. Li and J. J. Swetits, "The linear l1 estimator and the Huber M-estimator," *SIAM J. Optim.*, vol. 8, no. 2, pp. 457–475, May 1998.

[13] D. L. Massart, L. Kaufman, P. J. Rousseeuw, and A. Leroy, "Least median of squares: A robust method for outlier and model error detection in regression and calibration," *Anal. Chim. Acta*, vol. 187, pp. 171–179, 1986.

[14] O. Chum and J. Matas, "Optimal randomized RANSAC," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1472–1482, Aug. 2008.

[15] D. Fan, Y. Dong, and Y. Zhang, "Satellite image matching method based on deep convolution neural network," *Acta Geodaeticaet Cartographica Sinica*, vol. 47, no. 6, pp. 844–853, 2018.

[16] J. Ma, J. Zhao, J. Jiang, H. Zhou, and X. Guo, "Locality preserving matching," *Int. J. Comput. Vis.*, vol. 127, no. 5, pp. 512–531, May 2019.

[17] W.-Y. Lin, F. Wang, M.-M. Cheng, S.-K. Yeung, P. H. S. Torr, M. N. Do, and J. Lu, "CODE: Coherence based decision boundaries for feature correspondence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 34–47, Jan. 2018.

[18] P. J. Huber, "Robust statistics," in *International Encyclopedia of Statistical Science*. Berlin, Germany: Springer, 2011.

[19] D. Nistér, "Preemptive RANSAC for live structure and motion estimation," *Mach. Vis. Appl.*, vol. 16, no. 5, pp. 321–329, Dec. 2005.

[20] J. Matas and O. Chum, "Randomized RANSAC," Center Mach. Perception, Czech Tech. Univ., Prague, Czech Republic, 2001.

[21] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm, "USAC: A universal framework for random sample consensus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 2022–2038, Aug. 2013.

[22] X. Li and Z. Hu, "Rejecting mismatches by correspondence function," *Int. J. Comput. Vis.*, vol. 89, no. 1, pp. 1–17, Aug. 2010.

[23] J. Ma, J. Zhao, J. Tian, X. Bai, and Z. Tu, "Regularized vector field learning with sparse approximation for mismatch removal," *Pattern Recognit.*, vol. 46, no. 12, pp. 3519–3532, Dec. 2013.

[24] J. Ma, J. Zhao, J. Tian, A. L. Yuille, and Z. Tu, "Robust point matching via vector field consensus," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1706–1721, Apr. 2014.

[25] J. Ma, J. Zhao, J. Tian, Z. Tu, and A. L. Yuille, "Robust estimation of nonrigid transformation for point set registration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2147–2154.

[26] L. Torresani, V. Kolmogorov, and C. Rother, "Feature correspondence via graph matching: Models and global optimization," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 596–609.

[27] M. Cho, J. Sun, O. Duchenne, and J. Ponce, "Finding matches in a haystack: A max-pooling strategy for graph matching in the presence of outliers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2083–2090.

[28] M. Cho and K. Mu Lee, "Progressive graph matching: Making a move of graphs via probabilistic voting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 398–405.

[29] M. Cho, J. Lee, and K. Mu Lee, "Feature correspondence and deformable object matching via agglomerative correspondence clustering," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 1280–1287.

[30] X. Jiang, J. Ma, J. Jiang, and X. Guo, "Robust feature matching using spatial clustering with heavy outliers," *IEEE Trans. Image Process.*, vol. 29, pp. 736–746, 2020.

[31] J. Ma, J. Wu, J. Zhao, J. Jiang, H. Zhou, and Q. Z. Sheng, "Nonrigid point set registration with robust transformation learning under manifold regularization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 12, pp. 3584–3597, Dec. 2019.

[32] L. Zhe and R. Marlet, "Virtual line descriptor and semi-local matching method for reliable feature correspondence," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 16.1–16.11.

[33] J.-M. Morel and G. Yu, "ASIFT: A new framework for fully affine invariant image comparison," *SIAM J. Imag. Sci.*, vol. 2, no. 2, pp. 438–469, Jan. 2009.

[34] J. Bian, W.-Y. Lin, Y. Matsushita, S.-K. Yeung, T.-D. Nguyen, and M.-M. Cheng, "GMS: Grid-based motion statistics for fast, ultra-robust feature correspondence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2828–2837.

[35] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[36] B. K. P. Horn, *Robot Vision*. Cambridge, MA, USA: MIT Press, 1986.

[37] A. L. Yuille and N. M. Grzywacz, "The motion coherence theory," in *Proc. 2nd Int. Conf. Comput. Vis.*, Aug. 2005, pp. 344–353.

[38] A. Myronenko, X. B. Song, and A. Miguel, "Non-rigid point set registration: Coherent point drift," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 1009–1016.

[39] N. I. Achieser, *Theory of Approximation*. Chelmsford, MA, USA: Courier Corporation, 2013.

[40] L. Shapiro, and G. Stockman, *Computer Vision*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[41] J. Bian, L. Zhang, Y. Liu, W.-Y. Lin, M.-M. Cheng, and I. D. Reid, "Image matching: An application-oriented benchmark," 2017, *arXiv:1709.03917*. [Online]. Available: http://arxiv.org/abs/1709.03917

[42] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 573–580.

[43] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[44] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen, "On benchmarking camera calibration and multi-view stereo for high resolution imagery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[45] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4104–4113.

[46] J. L. Schonberger, E. Zheng, J. M. Frahm, and M. Pollefeys, "Pixelwise view selection for unstructured multi-view stereo," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 501–518.

[47] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[48] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.

[49] P. Alcantarilla, J. Nuevo, and A. Bartoli, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," in *Proc. Brit. Mach. Vis. Conf.*, 2013, pp. 1–12.

[50] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "KAZE features," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 214–227.

[51] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2548–2555.

[52] K. Simonyan, A. Vedaldi, and A. Zisserman, "Learning local feature descriptors using convex optimisation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1573–1585, Aug. 2014.

[53] G. Levi and T. Hassner, "LATCH: Learned arrangements of three patch codes," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–9.

[54] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 510–517.

[55] E. Tola, V. Lepetit, and P. Fua, "DAISY: An efficient dense descriptor applied to wide-baseline stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 815–830, May 2010.

[56] T. Trzcinski, M. Christoudias, and V. Lepetit, "Learning image descriptors with boosting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 597–610, Mar. 2015.

[57] F. Rombari and L. D. Stefano, "Interest points via maximal self-dissimilarities," in *Proc. Asian Conf. Comput. Vis.*, vol. 2014, pp. 586–600.

[58] A. Motilal, K. Kurt, and R. B. Morten, "Censure: Center surround extremas for realtime feature detection and matching," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 102–115.

[59] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 105–119, Jan. 2010.

[60] M. Elmar, D. H. Gregory, B. Darius, S. Michael, and H. Gerhard, "Adaptive and generic corner detection based on the accelerated segment test," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 183–196.

YANG DONG received the B.S. and M.S. degrees in photogrammetry and remote sensing from Information Engineering University, in 2014 and 2017, respectively, where he is currently pursuing the Ph.D. degree in surveying and mapping, under the supervision of Prof. Q. Ma and Prof. D. Fan. His current research interests include image matching and 3D reconstruction.

DAZHAO FAN received the Ph.D. degree in photogrammetry and remote sensing from Information Engineering University, in 2007. He is currently a Professor with Information Engineering University. His current research interest includes digital photogrammetry and its applications.

QIUHE MA received the B.S. and M.S. degrees in photogrammetry and remote sensing from the Institute of Surveying and Mapping, in 1983 and 1993, respectively. She is currently a Professor with Information Engineering University. Her current research interest includes image mapping and its applications.

SONG JI received the Ph.D. degree in photogrammetry and remote sensing from Information Engineering University, in 2012. He is currently an Associate Professor with Information Engineering University. His current research interests include image processing and 3D reconstruction.