The Visual Computer
https://doi.org/10.1007/s00371-020-01928-z

ORIGINAL ARTICLE

Accurate estimation of feature points based on individual projective plane in video sequence

Huajun Liu[1] · Shiran Tang[1] · Dian Lei[1] · Qing Zhu[2] · Haigang Sui[3] · Gaojian Zhang[1] · Chao Li[1]

Abstract
The stability and quantity of feature matching in video sequence is one of the key issues for feature tracking and some relevant applications. The existing matching methods are based on feature detection, which is usually affected by illumination conditions, noise or occlusions, and this will directly influence matching results. In this paper, we propose an accurate prediction method for interest point estimation in video sequence by extracting the stable mapping for each undetected point in its suitable projective plane, which is based on coplanar feature points that have already been detected in adjacent frames. The proposed prediction method breaks the limitation of the previous approaches that largely rely on feature detection. Our experiments show that our method not only predicts features accurately, but also enriches the correspondences, which prolongs the track length of features.

Keywords Feature point prediction · Homography · Projective plane · Video sequence

1 Introduction

Stable feature detection and correct matching are the premise of obtaining a long feature tracking in video sequence, which directly affects the precision of relevant applications, such as bundle adjustment [30], image registration [37], panorama production [33], and 3D reconstruction [11,15].

Feature detection is the precondition of the existing matching methods. However, the projective points of the same 3D point may not be detected stably in adjacent frames due to illumination change and other factors, which will result in fewer correspondences. Therefore, it is one of the key issues to find more stable feature points for consecutive matching in video frames.

The feature matching approaches that are widely used are based on feature detection, such as Kanade–Lucas–Tomasi tracking [17] based on Harris [12] corner detector, matching with descriptors based on scale-invariant feature transform (SIFT) [23], oriented FAST or rotated BRIEF (ORB) [28]. However, when there exists illumination change, noise or repeated texture, traditional matching methods which only use a single type of descriptors by a strict threshold cannot acquire enough correct correspondences. In order to enrich correct matches, Zhang et al. presented an improved two-pass matching strategy which utilizes geometry constraints and color constancy [36] to achieve good results. But their method still adopts a single type of descriptors, with the matches not sufficient as we expected. Hu and Lin attempted to perform feature matching by integrating adaptive descriptor selection and progressive candidate enrichment into image matching to

✉ Huajun Liu
huajunliu@whu.edu.cn

✉ Shiran Tang
Lightang@whu.edu.cn

✉ Dian Lei
dian_lei@whu.edu.cn

Qing Zhu
zhuq66@263.net

Haigang Sui
haigang_sui@263.net

Gaojian Zhang
2019282110172@whu.edu.cn

Chao Li
ldl1118@whu.edu.cn

[1] School of Computer Science, Wuhan University, Wuhan, China

[2] Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu, China

[3] State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, Wuhan, China

Published online: 09 August 2020

(a) (b) (c) (d)

Fig. 1 Comparisons between the brute-force SIFT matching and our method. **a**, **b** The results of dense point cloud and reconstruction with texture by brute-force SIFT matching. **c**, **d** The results of dense point cloud and reconstruction with texture by our method

obtain more feature correspondences [14], which breaks the restriction of merely using one type of descriptors and offers a new way to tackle the problem. However, all of the above methods perform matching in the detected interest points. Could potential and undetected interest points be found out? If their positions can be accurately estimated, the correspondences will be greatly increased.

Aiming at the previous limitation, we introduce a novel approach for the prediction of interest point to enrich correspondences, which fully utilizes both intra-frame feature locations and their shift relations in inter-frame. The key idea is to automatically establish 2D projective transformations for undetected interest points in adjacent frames, which is achieved by searching for coplanar feature points. Utilizing the distribution of features, our approach can accurately predict most of the undetected interest points, which will increase feature correspondences in video sequence and enhance the stability of feature tracking. Compared to the method of using feature tracks computed by brute-force SIFT matching, the 3D construction results can be more precise and complete by our method. The comparisons of recovered 3D point clouds and reconstructed 3D models with texture are shown in Fig. 1, which reveals that our method can greatly improve the integrity of 3D reconstruction models.

2 Related work

Since the work in this paper involves feature matching, feature point recovering and homography estimation, the relevant work will be reviewed in detail.

2.1 Feature matching

Variant kinds of matching methods are based on feature point detection approaches, such as SIFT [23], speeded-up robust features (SURF) [3], ORB [28], binary robust independent elementary features [5] and binary robust invariant scalable keypoints [19]. What's more, SuperPoint [8] and Local Feature Net [27], deep learning-based methods of keypoint detection and description, are prevalent as well. However, matching methods using the above features may have difficulties in keeping the balance between matching precision and quantity due to strict thresholds. Besides, due to the effects such as noise, repetitive texture and unknown distortion, using simple distance approaches tends to result in mismatches. In order to get more correct correspondences, various methods were proposed. Cho et al. proposed a hierarchical agglomerative correspondence clustering (ACC) approach by adopting both photometric similarity and pairwise geometric constraints [6], but it heavily depends on the affine-covariant feature detectors, which results in relatively narrow applications. Ma et al. proposed a vector field consensus (VFC) method [24] and a locally linear transforming (LLT) method [25] to get correct inliers, both of which use regularized kernel methods to successfully resist quite a large number of outliers. Li et al. proposed a support-line descriptor for mismatches elimination, which is based on multiple adaptive binning gradient histograms and affine-invariant ratios between line structures [21], but the affine model cannot effectively deals with significant geometric distortions. They also designed a region descriptor, termed 4FP-Structure [20], by using four feature points for outlier removal and match expansion. Nevertheless, these matching methods only focus on enriching the correspondences in those interest points that have been detected with a single type of descriptors, with the final matches still limited.

An alternative approach is to utilize graph matching, which constructs feature matching according to the similarity of feature descriptors and their spatial arrangement changes in different views [4]. Based on the above method, progressive matching approaches that combine graph matching with Bayesian probabilities [7] or density sampling [32] were proposed individually. Both of them can effectively increase matches. However, it tends to introduce a number of mismatches because the current graph matching results might be noisy.

In addition, there is another popular way that uses multi-view geometry constraints for feature matching. For example, Yan et al. proposed to use the epipolar geometry for clustered points to increase matches for an unordered image set [35]. At the same period, Zhang et al. proposed to use

a planar motion segmentation strategy based on already matched points, which combines with the color constancy and polar geometry to find more correspondences [36]. The above matching methods can only obtain correspondences by a single type of feature descriptors. Considering that different descriptors can be used for feature points description, Hu and Lin proposed to leverage multiple descriptors to enrich the matches [14]. Since the methods above just acquire correspondences among detected interest points, they can only match in a narrow range. However, feature points that cannot be detected in a specific feature detection mode exist objectively. Could the potential undetected interest points be directly located? If this issue can be solved, more matches will be obtained. This problem motivates us to study on accurate prediction of the undetected feature points, aiming to break the limitation of current detecting methods.

2.2 Feature points recovering

In frame sequences, KLT [17] is a typical method of feature tracking in a whole sequence, but it should be used in a strict condition. Besides, it may be easily distracted by occlusions, repeated structures and image noise, which may cause the loss of some interest points. Targeting at this problem, effective ways of predicting and recovering lost interest points must be applied to prolong feature tracks. For example, the factorization methods are used in measurement matrix to estimate unknown parts of trajectories [1,16,29]. In essence, these methods depend on the projection of 3D shape, but the inaccuracy of the 3D shape deduced from measurement matrix leads to the inexact points recovery. Instead of presupposing the resolution of 3D points, we predict undetected points directly by 2D transformation. In our method, the constraints of intra-frame feature locations and stable projective transformation in adjacent frames are introduced to improve the prediction accuracy.

2.3 Homography estimation

In consecutive video frames, homography transformation is a suitable way to build the inter-frame mappings. In homography estimation, RANSAC [10] is a classic way to remove outliers for high-confidence matches. With the development of deep neural networks, DeTone and Erraphy estimated the homography transformations by deep learning approaches [8,9]. Their methods are more suitable for image pairs which are composed by a single plane and the methods require a large amount of training datasets. However, one image pair usually consists of multiple homographies, and a single global homography cannot ensure accurate transformation for each point. Different from the methods that conduct a single transformation, Zhang et al. [36] and Jin et al. [18], respectively, solved this problem with the multi-layer homography algorithm, which utilizes iterations of RANSAC to segment an image composed of multiple projective planes. However, the process roughly obtains several homographies, which is not exact enough. Barath and Hajder [2] proposed to estimate the homography from one affine correspondence that consists of a point pair and the related local affine transformation mapping the pixels infinitely close to the point locations from the first to the second image. Nevertheless, it cannot make a homography estimation for the undetected points. To solve this problem, we introduce an accurate homography estimation method for undetected points prediction, which is calculated by coplanar points in adjacent frames.

3 Our method

Based on the stable projective transformation for coplanar points among consecutive frames, our proposed method can accurately predict the undetected interest points by establishing their own exact projective transformations.

Stable keypoints and Lost keypoints

In frame sequence $t-m,\ldots,t-1,t$, keypoints which can be detected in all the frames are treated as ***stable keypoints***. The ones that can be detected in frames $t-m,\ldots,t-1$, but cannot be detected in frame t are noted as ***lost keypoints*** in frame t. In our paper, we select the stable keypoints to calculate a homography $\boldsymbol{H}$ for prediction of each lost keypoint.

For a pair of images, there exists a 2D homography $\boldsymbol{H}$ between the correspondences X and X', both of which are from the same 3D point P. The 2D homography maps $X=[x,Y,1]^{\mathrm{T}}$ to $X'=\left[x',Y',1\right]^{\mathrm{T}}$ in homogeneous coordinates:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{1}$$

Since a homography has 8 degrees of freedom, by setting $h_{33}=1$, four pairs of matched points are usually used to determine a unique $\boldsymbol{H}$. If more than four correspondences are given, the set of equations $\boldsymbol{Ah=0}$ may be over-determined. Therefore, there is an exact solution of $\boldsymbol{A}$ which has rank 8 because no noisy data exists. In conclusion, it is a critical issue to find four suitable point correspondences for accurate calculation of $\boldsymbol{H}$.

Since a homography reflects an exact mapping and an image is usually composed of multiple planes, each point

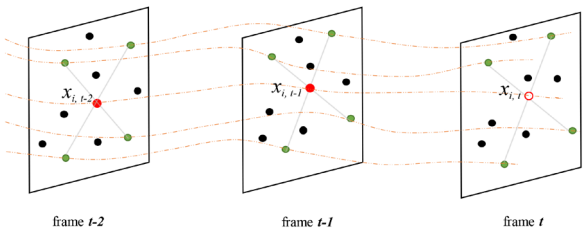


Fig. 2 The process of searching for coplanar points. The dotted lines represent trajectories of each point in consecutive frames. The red hollow point $x_{i,t}$ represents the lost keypoint in frame t, and the red filled points $x_{i,t-1}$ and $x_{i,t-2}$ represent its corresponding points in frame $t-1$ and $t-2$. The green points represent the four point-pairs that we searched from the black stable keypoints, and these four point-pairs can be used to predict $x_{i,t}$

correspondence should be mapped by its own homography for accurate matching. Therefore, the key issue of accurate prediction is to find an exact projective plane that is suitable for each lost keypoint. The two principles for determining the projective plane can be defined as follows:

Principle (I): The four point-pairs used for homography estimation, in which no three points are collinear, need to be exactly coplanar in each view.

Principle (II): The lost keypoint and these four points should be precisely on the same projective plane in each view.

For Principle (I), in order to ensure $\boldsymbol{A}$ has rank 8 and avoid degenerating configurations, the situation that three of the four points are collinear must be prevented. In this way, a unique homography can be obtained. With the Principle (I) satisfied, Principle (II) means that if the additional pair of feature points is exactly coplanar with the four point-pairs in either view, $\boldsymbol{A}$ for the multi-point configuration (more than four point-pairs) still has rank 8, and the mapping of two projective planes will not be affected.

Considering that the points have the similar shift relations in adjacent frames, we choose the coplanar *stable keypoints* in previous frames to calculate $\boldsymbol{H}$ for prediction. If there exists a lost interest point $x_{i,t}$ in frame t, where we cannot find its own coplanar points from those detected points, appropriate four points can be determined in frame $t-1$ and $t-2$ by projection. And then their correspondences in frame $t-1$ and t will be used to estimate the homography for the lost keypoint prediction, as shown in Fig. 2.

Since the linearity of straight line is an important property which can be preserved by homography [13], the intersection point of two straight lines will be exactly transformed by the homography. This inspires us to construct the lost keypoint as the intersection point of two lines connected by four coplanar points, and the lost keypoint will be coplanar with them. In addition, the process of searching for suitable stable keypoints could also be highly efficient, as shown in Fig. 3.

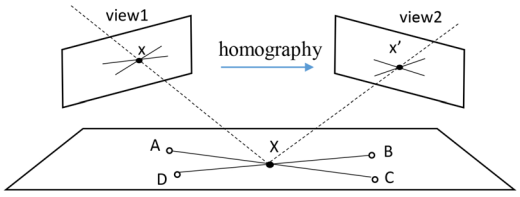


Fig. 3 Projections of collinear points in the plane are still collinear in two views. Point X is located at the intersection of two lines, guided by four points A, B, C and D in each view

Line selecting and sorting

For the corresponding point $x_{i,t-1}$ in frame $t-1$ of one lost keypoint $x_{i,t}$ in frame t, we need to select its four coplanar points from *stable keypoints*, as shown in Fig. 2. By connecting each two keypoints in frame $t-1$, we can get straight line sets $\{l_j\}_{j=1}^{n}$. And then according to the distances between the lost keypoint $x_{i,t-1}$ and each straight line l_j, eligible lines are retained when distance $d\left(x_{i,t-1}, l_j\right)$ in Eq. (2) is below the threshold δ, which can be denoted as $L_{t-1}\,(l_1, \ldots l_m)$ $(m < n)$.

$$d\left(x_{i,t-1}, l_j\right) = \frac{\left|l_j^T \cdot x_{i,t-1}\right|}{\sqrt{a_j^2 + b_j^2}} \tag{2}$$

where $x_{i,t-1} = [u, v, 1]^{\mathrm{T}}$ and $l_j = \left[a_j, b_j, c_j\right]^{\mathrm{T}}$.

According to the keypoint $x_{i,t-1}$ and line sets $L_{t-1}\,(l_1, \ldots l_m)$ in frame $t-1$, we can obtain their corresponding point $x_{i,t-2}$ and line sets $L_{t-2}\left(l'_1, \ldots l'_m\right)$ in frame $t-2$. If the distance $d(x_{i,t-2}, l'_j)$ between the point $x_{i,t-2}$ and line l'_j is also below the threshold δ, the corresponding line l_j in L_{t-1} is considered to be one of the candidates to determine four coplanar points which can be used for prediction.

We sort the candidate lines from smallest to largest according to the mean value D_j of two distances,

$$D_j = (d(x_{i,t-1}, l_j) + d(x_{i,t-2}, l'_j))/2, \tag{3}$$

noted as $\left\{L_{k,t-1} |\, k = 1, 2, 3, \ldots, s\right\}$ $(s \leq m)$, and we choose two lines each time according to the line sequence for determining four most suitable points to estimate an accurate homography for the lost keypoint $x_{i,t}$.

However, ***degenerate configurations*** need to be prevented. In other words, any three of the four points cannot be collinear. This can be estimated by calculating the angle θ of the two lines in each frame. If the conditions that $\theta_{i,t-1} > \varphi$ and $\theta_{i,t-2} > \varphi$ are both satisfied, degenerate configurations can be prevented. We can get the angle of two lines by

$$\theta = \arctan \left| \frac{k_1 - k_2}{1 + k_1 k_2} \right| \tag{4}$$

where k_1, k_2 are slopes of two lines.

The ***projective error e*** for points $x_{i,t-1}$ and $x_{i,t-2}$ can be defined as follows:

$$\boldsymbol{e} = \left\| \pi \left(H_{i,t-1} x_{i,t-1} \right) - x_{i,t-2} \right\| \tag{5}$$

where $H_{i,t-1}$ is the homography obtained by the four point-pairs in frame $t-1$ and $t-2$ and π represents the normalization of the homogeneous coordinates.

The four point-pairs can be retained only if the projective error $\boldsymbol{e}$ is below the threshold ε, which is infinitely close to zero when the lost keypoint and four selected points are exactly coplanar. However, it is unstable to determine the four points only based on one pair of adjacent frames, so a predictive window including multi-pair of adjacent frames should be used for testing. In the predictive window, according to the line sequence $\left\{ L_{k,t-1} | \; k = 1, 2, 3, \ldots, s \right\}$, the first set of four point-pairs that has no degenerate configurations and satisfies the constraints of projective error in consecutive frames are utilized for lost keypoint $x_{i,t}$ prediction. Otherwise, $x_{i,t}$ cannot be precisely predicted. The process of lost keypoint prediction is described in Algorithm 1.

Algorithm 1 Procedure for lost keypoint prediction

Input: consecutive frames $t-m, \ldots, t-1, t$.

1 Find lost keypoint $x_{i,t}$ for frame t and stable keypoints according to general detection and feature matching approaches.

2 Obtain the line sequence according to ***line selecting and sorting*** in frame $t-1$ and $t-2$.

3 Choose the most suitable two lines in the line sequences $\left\{ L_{k,t-1} | \; k = 1, 2, 3, \ldots, s \right\}$.

for i=0, .., s-1, **do**
  for j=i, .., s, **do**
    Find and evaluate the four point-pairs by preventing ***degenerate configurations***, and evaluating ***projective error e*** for each adjacent frames in the prediction window.
    if conditions satisfied, **then**
      goto Step 4.
    end if
  end for
end for
if no suitable two lines are found, **then**
  $x_{i,t}$ cannot be precisely predicted.
end if
4 Calculate $\boldsymbol{H_{i,t}}$ by the determined four most suitable point-pairs in the frame $t-1$ and t to predict the corresponding point $x_{i,t}$ in frame t.

Output: estimated position of point $x_{i,t}$.

4 Experiments and analysis

In this section, we conducted comprehensive experiments to analyze our method both qualitatively and quantitatively in the following aspects. (1) Different window sizes in our method are tested for predictive precision. (2) The proposed method is compared with six state-of-the-art feature matching methods, such as ACC, VFC, LLT, 4FP and two matching methods based on deep learning, i.e., SuperPoint and LF-Net. The parameters are set according to their literature suggestion and fixed throughout all experiments, and the implementations of these algorithms are obtained from the authors' websites, as shown in Table 1. Especially, as for the SuperPoint matching, we pre-train an interest point detector, called MagicPoint on synthetic data for 200,000 iterations. We apply a Homographic Adaptation procedure to generate pseudo-ground truth labels using the MSCOCO 2014 [22] training dataset split which has 80,000 images and the MagicPoint base detector. The generated labels are used to train a network that jointly extracts interest points and descriptors from an image. In this experiment, the network is trained on 240×320 grayscale COCO datasets for 200,000 iterations and evaluated using aerial image datasets. The same training dataset and iterations are used to LF-Net training. (3) Based on the feature tracks computed by SIFT matching and 4FP matching methods, track lengths are compared for our method and theirs on aerial image datasets. (4) The average time consumption of each point's prediction is analyzed based on different quantities of points, which are used to search for four point-pairs. (5) Based on the default parameters, we analyze the sensitivity by using different variations of these parameters. (6) For the close-range sequences, we compare the completeness and precision of 3D reconstruction, using the feature tracks computed by SIFT matching and our method based on SIFT. For the training and evaluating of the two deep learning methods, i.e., superpoint and LF-Net, we use the NVIDIA GeForce 1080Ti GPU, and the other tests are evaluated on CPU using Intel Core i5-6400 @ 2.7 GHz.

Datasets

Two categories of consecutive sequences are applied to examine the performance of our method. (1) Five aerial video sequences from UAV123 Dataset [26], including *building3, building4, car2, car3* and *car13*, which have ordinary texture, as shown in Fig. 4. (2) Two close-range sequences captured by a handheld camera, including *car* and *dustbin*. In the first dataset category, *building3* and *building4* are captured with 30 fps, and *car2, car3* and *car13* are captured with 96 fps. In the second dataset category, both of the two sequences are captured with a handheld camera of 30 fps, with strong light and reflected glare in part of the views.

Table 1 Parameter settings for compared methods

Methods	Parameter setting	Source code
ACC [6]	$K_{AP} = 10$; $r_{AP} = 0.05$; $\delta_D = 25$; $\tau_a = 1$; $\tau_m = 1$	http://cv.snu.ac.kr/research/~acc/
VFC [24]	$\beta = 0.1$; $\lambda = 3$; $\tau = 0.75$; $\gamma = 0.9$; $a = 10$	https://sites.google.com/site/jiayima2013/
LLT [25]	$K = 15$; $\lambda = 1000$; $\tau = 0.5$; $\gamma = 0.9$; $\beta = 0.1$; $M = 15$; $a = 10$	https://sites.google.com/site/jiayima2013/
4FP [20]	$\sigma = 8.5$; $n = 3$; $M = \{5, 8, 10\}$; $K = \{8, 6, 4\}$ $t = 8$	http://www.escience.cn/people/lijiayuan/dmysjj.html
Super Point [8]	$D = 256$; $\lambda_d = 250$; $m_p = 1$; $m_n = 0.2$; $\lambda = 0.0001$; $lr = 0.001$; $\beta = (0.9, 0.999)$	https://github.com/MagicLeapResearch/SuperPointPretrainedNetwork
LF-Net [27]	$\lambda_{\text{pair}} = 0.01$; $\lambda_{\text{ori}} = \lambda_{\text{scale}} = 0.1$	https://github.com/vcg-uvic/lf-net-release

(a) building4 (b) building3 (c) car13 (d) car2 (e) car3

Fig. 4 Five scenes taken from UAV123 dataset for testing

Table 2 Threshold values of our method

Parameter	Notation	Default value
Distance between lost keypoint and lines	δ	0.5 pixel
Angle of selected two lines	φ	5°
Projective error	ε	1.0 pixel

Parameter settings

The parameters in our paper are set as shown in Table 2. A large threshold of the distance between the lost keypoint and the line results in large number of candidates, which will make the process of searching for four point-pairs time-consuming, so we set δ to 0.5 pixel. In order to prevent degenerate configurations, the parameter φ is set to 5°. If the angle of selected two lines $\theta > \varphi$, the situation that three points are collinear can be prevented. To ensure the precision of prediction, the threshold ε of projective error for each pair of adjacent frames is set to 1.0 pixel.

Note, the SIFT detection and matching in this paper are performed by Vlfeats [31], with the VLSIFT peak threshold and edge threshold set as 0 and 10, respectively. In addition, the parameters of other feature matching method are set according to their literature suggestion and are fixed throughout all experiments. The implementations of these algorithms are obtained from the authors' websites.

4.1 Different window size for prediction

Different size of prediction window will determine variant four point-pairs, which can affect the predictive precision and predicted number of undetected interest points. By using five image datasets taken by UAV, as shown in Fig. 4, frame 1 to frame 450 are selected for each sequence. In order to test the predictive effects on different window sizes with short-baseline and wide-baseline, we select one frame for every 3 and 15 frames separately for each sequence to constitute new sequences. By setting the window size from 2 to 4, we test the predicted number and predictive precision for lost keypoints. The prediction window is illustrated as Fig. 5.

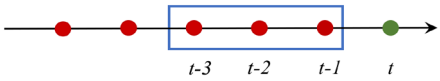


Fig. 5 Prediction window. The green point represents the frame in which lost keypoints exist and the red points represent the previous frames. The blue rectangular box represents the prediction window. In this figure, three previous frames are used for prediction, so the window size is 3

We take the stable points obtained by Vlfeats as the groundtruth, and use the Leave-one-out evaluation to test the number of predicted points and the mean of projective errors in five datasets. During the prediction, each one of the stable keypoints is chosen as the lost point and the others are regarded as candidates for four points selection. For each lost keypoint, if we can find its corresponding coplanar four points which exist stably in adjacent frames, it can be regarded as one predictable point. By setting different window size and sliding the window across the whole sequences, the comparison results are shown as Fig. 6.

In Fig. 6a, b, we can find that compared to wide-baseline, short-baseline can obtain more accurate predicted results. Meanwhile, all the prediction errors of both kinds of baselines are below 1 pixel. In general, as the window size grows, the prediction precision increases accordingly. To be detailed, the errors increase significantly with the window size switching from 2 to 3. And they increase slightly when the size grows from 3 to 4. This is because double estimation of coplanarity in window size 3 enhances the prediction validation, and more times of coplanarity estimation may not be needed.

In Fig. 6c, d, we can find that in general, compared with the wide-baseline, predicting with short-baseline can acquire more predicted points and the predicted number decreases more slowly as the window size changes from 2 to 3. The reason is that fewer changes will occur between adjacent frames in the short-baseline sequences than wide-baseline. And more coplanar points which can be used for prediction are retained.

As shown in Fig. 6, with the growth of window size, the prediction precision increases accordingly, while the predicted number decreases. Therefore, to balance the predictive accuracy and the predicted number, the window size of 3 is adopted in our following experiments.

4.2 Comparisons with other methods

We compare the predictive precision of our method with several state-of-the-art feature matching methods, such as ACC, VFC, LLT, 4FP and two matching methods based on deep learning, i.e., SuperPoint and LF-Net. These methods aim to find more correspondences among detected feature points. Different from them, our method can improve the number of correspondences by predicting undetected feature points. Considering these six methods focus on matching, we choose wide-baseline sequences for testing. For each of these five aerial datasets, we match correspondences in every pair of adjacent frames in the whole sequence by Vlfeats and take them as benchmarks. In addition, by estimating the homography based on them, the new correspondences, which are obtained by our approach and the six methods, are used to test mean error and RMSE (root-mean-square error) individually, both of which represent the precision of matching or prediction. As shown in Fig. 7, our method achieves more accurate results than theirs and keeps the lowest fluctuations on five aerial datasets. The reason is that our method is based on exactly coplanar points, which provides a more accurate homography for prediction.

4.3 Track length

Track length is another critical indicator of prediction. Since our approach can be used to enrich correspondences based on existing detecting and matching methods, track length is compared for our approach based on both classic brute-force SIFT matching and state-of-the-art 4FP matching. Considering different baselines between consecutive frames have different predictive results, in this experiment, we adopt two aerial image datasets, *building4* and *car2*, which have different frame frequencies. By testing consecutive 30 frames, which are individually selected from the two datasets above, Fig. 8 shows the track length comparisons between our method and the other two consecutive feature matching approaches. With the prediction window size set as 3, the lost keypoints in frame 1–3 cannot be predicted. Therefore, the same tracked points are obtained in the first three frames by ours and the corresponding approach. But we can observe an obvious improvement by our method compared to two consecutive matching methods in the following frames. Besides, the chart illustrates that the number equals to zero when tracking more than about 20 frames by brute-force SIFT matching, but a large number of points can still be found by our method. Compared to brute-force SIFT matching, 4FP matching can obtain more initial correspondences. However, by using our approach based on 4FP, the tracked number in each frame has been further improved. In addition, Figs. 9 and 10 are the matching results of tracked points by tracking 30 frames, and we can obviously find that more tracked points are obtained by our method than the corresponding consecutive matching approach.

4.4 Computational cost

As we stated above, since our method works as a further prediction step based on other matching methods, it seems not fair to compare the running time with them. Considering this, we evaluate the prediction time of our approach on the

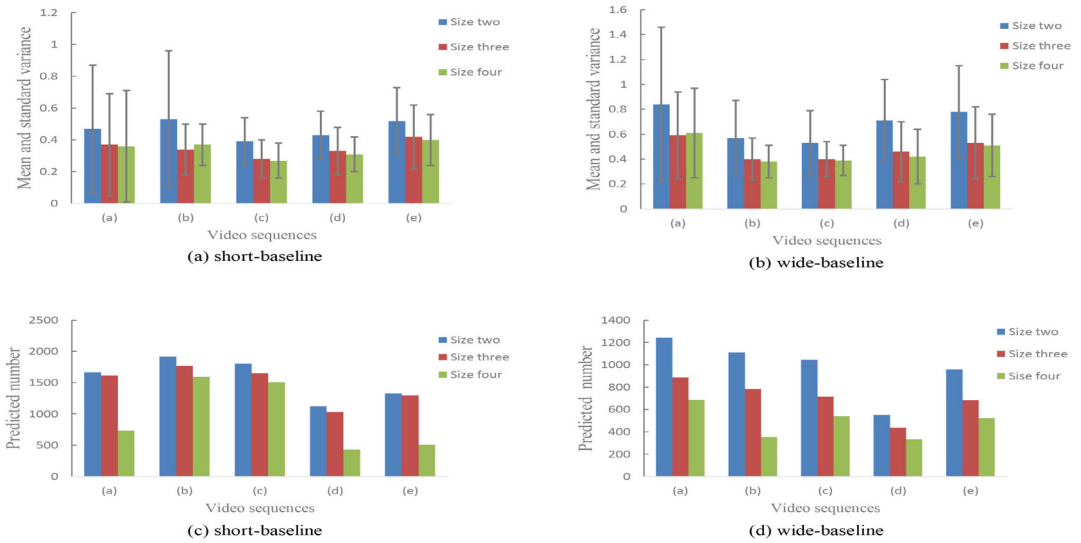


Fig. 6 Mean error, standard variance and predicted number for different window size on two kinds of baselines aerial datasets. **a**, **b** Represent mean error and standard variance. **c**, **d** Represent predicted number

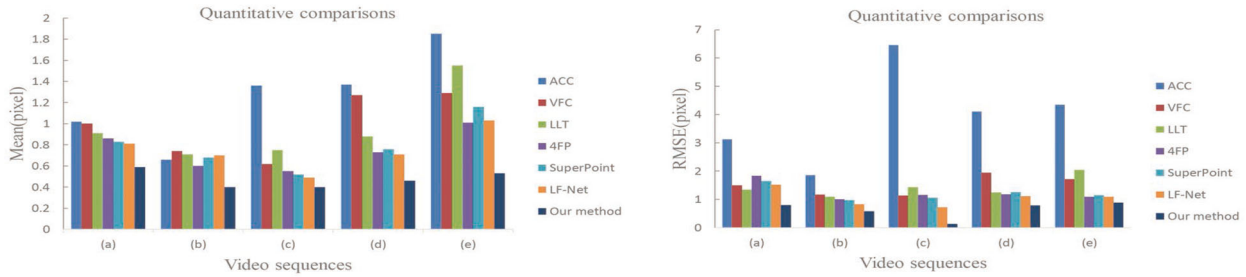


Fig. 7 Comparisons of mean error and RMSE with other state-of-the-art methods

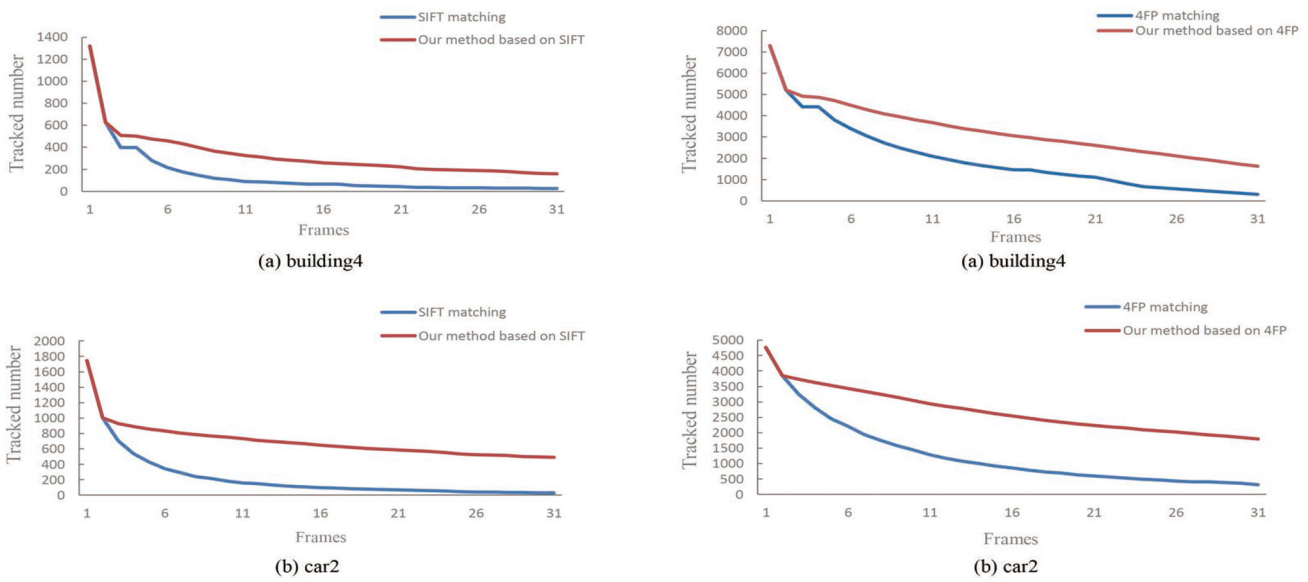


Fig. 8 Comparisons for tracked number in consecutive frames. The charts are the comparisons for SIFT matching, 4FP matching and our method based on them

Fig. 9 Matching comparisons for tracking 30 frames by brute-force SIFT matching and our method based on it. The left column pictures are the results for SIFT matching. The right column pictures are the results for our method based on SIFT. The upper two comparative pictures are taken from *building4*, and the lower two comparative pictures are taken from *car2*

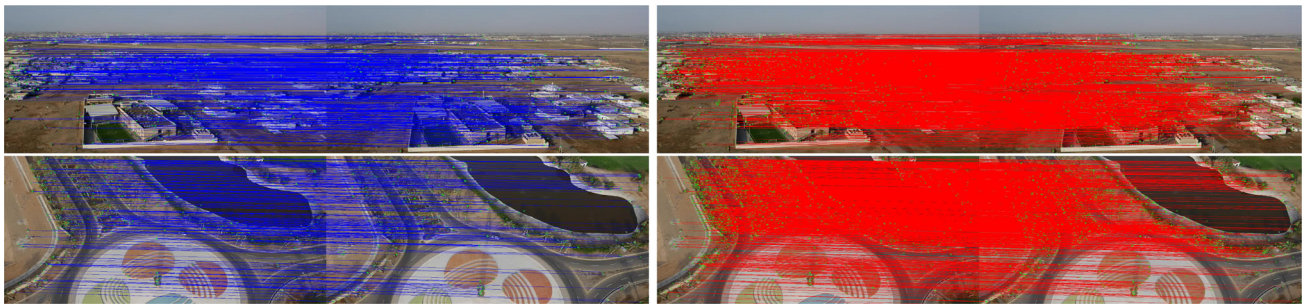


Fig. 10 Matching comparisons for tracking 30 frames by 4FP matching and our method based on it. The left column pictures are the results for 4FP matching. The right column pictures are the results for our method based on 4FP. The upper two comparative pictures are taken from *building4*, and the lower two comparative pictures are taken from *car2*

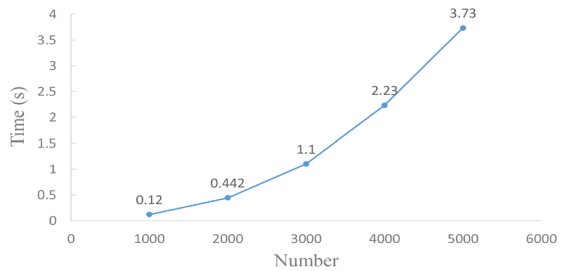


Fig. 11 Average prediction time for one lost keypoint based on different quantities of stable keypoints

matches, which have been processed by other matching methods. We show the result based on 4FP detection method as an example. After the detected feature points are randomly captured, the average efficiency of prediction for one lost keypoint is shown in Fig. 11, with the number of matched feature points, which are used for four point-pairs searching, ranging from 1000 to 5000. We can observe that the prediction time increases with the growth of detected points. The reason is that more stable keypoints will add more burdens on *line selecting and sorting*, whose purpose is to find the most suitable points for prediction.

4.5 Parameter configuration

Default parameters are set in previous experiments as most of them can balance the predicted number, accuracy and the consumed time. And then we have tested the sensitivity by using different variations of these parameters. When we change one of these parameters, the others are kept as the default values. In this experiment, the initial feature points detection and matching steps are performed with SIFT. We randomly select 10 frames from each dataset above and the total number is defined as the whole lost keypoints need to be predicted. Accordingly, the predicted number is defined as the whole lost keypoints predicted by our method. The predicted ratio is defined as *predicted number/total number* and the time cost is defined as the average efficiency of prediction for one lost keypoint. (1) The first parameter is δ, which controls the distance between the lost keypoint and the line. As shown in Table 3, compared with 0.5, a smaller $\delta = 0.1$ pixel could result in a lot lost keypoints that cannot find their coplanar four point-pairs, though the time cost is lower. In contrast, a larger $\delta = 1.5$ pixel could result in more candidates, which takes too much time, and the predicted number does not change, since we only need four coplanar point-pairs without degenerating. (2) The second parameter

Table 3 Parameter configuration

Parameter	Setting	Ratio (%)	Predicted error (pix)	Time cost (s)
δ	0.1	67.4	0.5	0.182
	0.5	71.9	0.512	0.192
	1.5	71.9	0.516	0.346
φ	2	72.6	0.835	0.214
	5	71.9	0.512	0.192
	10	67	0.513	0.191
ε	0.5	40	0.497	0.192
	1	71.9	0.512	0.192
	1.5	83.8	0.729	0.192

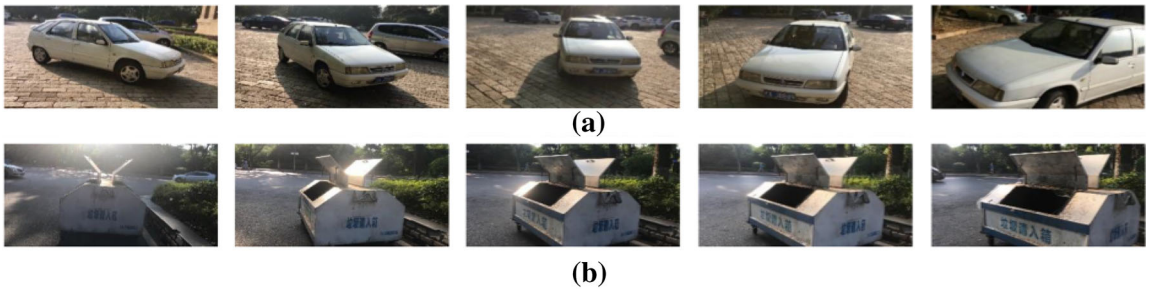


Fig. 12 Five key frames of "car" and "dustbin" example

is φ, which controls the angle of selected two lines. As shown in Table 3, a smaller $\varphi = 2°$ could predict a bit more lost keypoints. But in some cases, it might produce degenerate configurations and make the prediction results not accurate. However, we can see it clearly that a larger $\varphi = 10°$ does not work well on predicted ratio, because a overlarge φ will filter some suitable four point-pairs. (3) The third parameter is ε, which controls the projective error that is used to evaluate the stability of chosen four point-pairs. As is shown in Table 3, $\varepsilon = 0.5$ pixel could result in a small predicted number, in spite of the slight accuracy improvement. Conversely, we can see that the predicted mean error increases significantly when ε changes from 1 to 1.5 pixel, even though the predicted number increases correspondingly. The reason may be that a large ε could filter less poor four point-pairs, which could acquire more predicted lost keypoints, but the predicted error cannot be ensured. Therefore, we adopt the default parameters as is shown in Table 2.

4.6 3D reconstruction

We compare 3D reconstruction results through two close-range videos with low texture, strong sunlight and reflected glare, which are captured by a moving camera for validating the effects of predicted points. Five key frames for each scene are shown in Fig. 12. Since our method can prolong the track length of features, we can obtain more accurate 3D reconstruction results. Therefore, to validate the distribution of predicted points by our method, we compare the 3D reconstruction results of brute-force SIFT matching and our method. We use them to perform 3D reconstruction using visual SFM [34] which will discard inaccurate 3D points restricted by the reprojection error, and Fig. 13 is the comparison results of two methods. We can find that our method can obtain better recovered 3D point clouds and better reconstructed 3D models with texture than the brute-force SIFT matching approach. Besides, compared to the above method, our method completes some of the holes produced by strong light reflection or low texture.

In order to verify the precision of our predicted points in the process of reconstruction, we compared the recovered camera trajectories for two scenes. By superimposing the recovered camera trajectories of two methods, Fig. 13e, f shows the high accuracy of the results as all trajectories are not drifted, in which the red camera trajectory is obtained by using the feature tracks computed by brute-force SIFT matching method and the green camera trajectory is obtained by our method.

5 Limitation and discussion

Although we have tested the performance of our method on a range of diverse datasets and 3D reconstruction application, it also suffers from the following limitations. First, our method requires consecutive coplanes to predict inter-

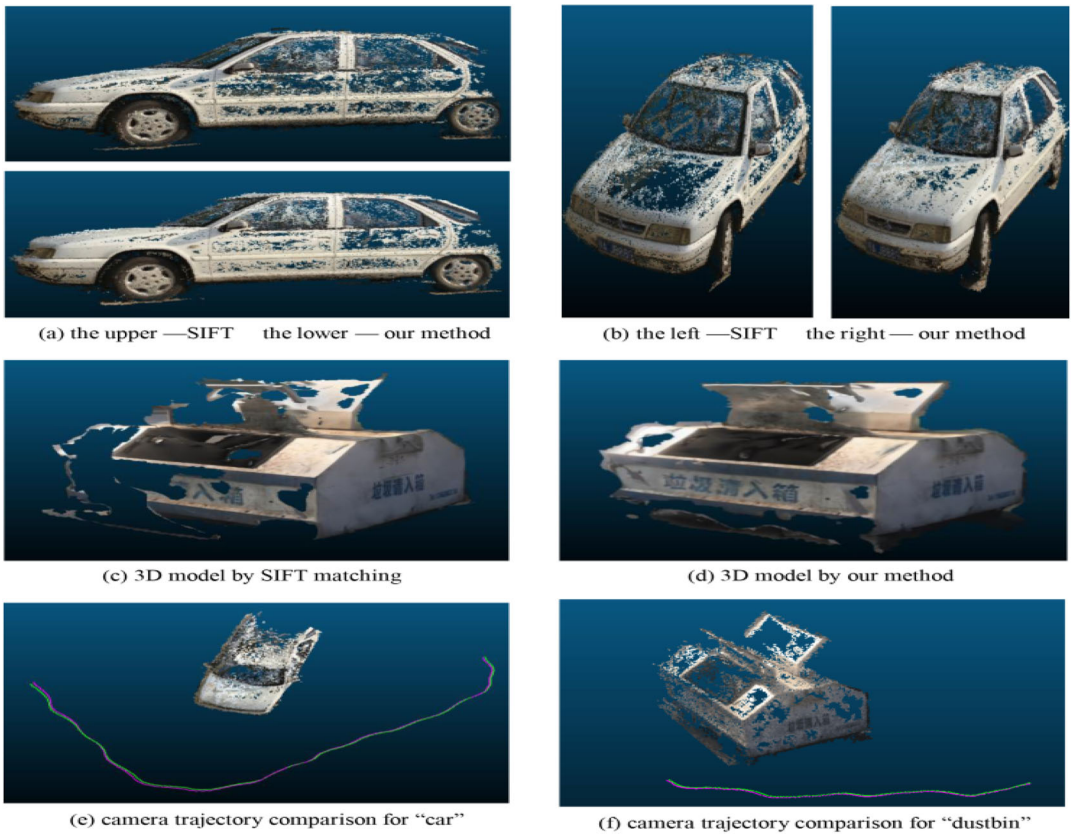


Fig. 13 Results for 3D point cloud, 3D model with texture and camera trajectory recovery. **a**, **b** Comparisons of recovered 3D points for "car" by two methods in different views. **c**, **d** Comparisons of reconstructed 3D model with texture for "dustbin" by two methods. **e**, **f** Camera trajectory comparisons for two examples by two methods

est points in consecutive frames. In such case, homography may not be accurately estimated for the lost keypoints of moving objects, which might result in inadequate prediction number. However, the prediction of static scenes would not be affected by some small moving objects, as they do not bother the process of acquiring coplanar four point-pairs from static scenes. Second, our method could acquire relative instead of absolute coplanar point-pairs for lost keypoints in most scenes. But in some extremely complicated scenes where the detected feature points belong to too many different planes, with complicated changing of camera relative pose, not pure translation or rotation, it can be really difficult to obtain relative coplanes, since our method has strict demand for accuracy. Thus, the predicted number of lost keypoints is limited to some extent, yet the precision accuracy can be ensured.

6 Conclusions

In this paper, we introduced a novel approach to predict lost keypoints in video frames. Different from state-of-the-art matching methods, our approach can automatically select four coplanar points, which are stable and also coplanar with the lost keypoint in adjacent frames, to establish an accurate homography for the lost keypoint prediction. Our method breaks the limitation of the previous approaches that largely rely on detected points for feature matching. Compared to other matching methods, our approach can obtain more correspondences with higher precision. In addition, our method not only can prolong the track length, but also can reconstruct 3D models that are more complete. In the future work, we will improve our algorithm in selecting lines to keep a balance between precision and computational cost.

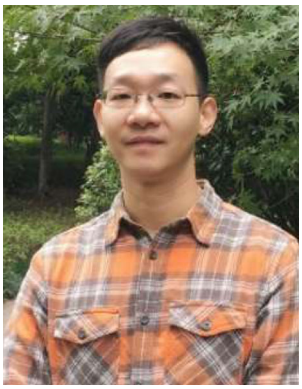
Acknowledgements This work was funded by the National Natural Science Foundation of China (NSFC) (41771427 and 41631174).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Ackermann, H., Rosenhahn, B.: Trajectory reconstruction for affine structure-from-motion by global and local constraints. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2890–2897. IEEE (2009)
2. Barath, D., Hajder, L.: A theory of point-wise homography estimation. Pattern Recognit. Lett. **94**, 7–14 (2017)
3. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: speeded up robust features. In: European Conference on Computer Vision, pp. 404–417. Springer (2006)
4. Berg, A.C., Berg, T.L., Malik, J.: Shape matching and object recognition using low distortion correspondences. In: CVPR (1), pp. 26–33. Citeseer (2005)
5. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: Brief: binary robust independent elementary features. In: European Conference on Computer Vision, pp. 778–792. Springer (2010)
6. Cho, M., Lee, J., Lee, K.M.: Feature correspondence and deformable object matching via agglomerative correspondence clustering. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 1280–1287. IEEE (2009)
7. Cho, M., Lee, K.M.: Progressive graph matching: making a move of graphs via probabilistic voting. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 398–405. IEEE (2012)
8. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: self-supervised interest point detection and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 224–236 (2018)
9. Erlik Nowruzi, F., Laganiere, R., Japkowicz, N.: Homography estimation from image pairs with hierarchical convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 913–920 (2017)
10. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)
11. Geneva, P., Maley, J., Huang, G.: An efficient Schmidt-EKF for 3d visual-inertial slam. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
12. Harris, C.G., Stephens, M., et al.: A combined corner and edge detector. In: Alvey Vision Conference, vol. 15, pp. 10–5244. Citeseer (1988)
13. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2003)
14. Hu, Y.T., Lin, Y.Y.: Progressive feature matching with alternate descriptor selection and correspondence enrichment. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 346–354 (2016)
15. Ilg, E., Saikia, T., Keuper, M., Brox, T.: Occlusions, motion and depth boundaries with a generic network for disparity, optical flow or scene flow estimation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 614–630 (2018)
16. Jacobs, D.W.: Linear fitting with missing data for structure-from-motion. Comput. Vis. Image Underst. **82**(1), 57–81 (2001)
17. Jianbo, S., Tomasi, C.: Good features to track. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 593–600 (1994)
18. Jin, Y., Tao, L., Di, H., Rao, N.I., Xu, G.: Background modeling from a free-moving camera by multi-layer homography algorithm. In: 2008 15th IEEE International Conference on Image Processing, pp. 1572–1575. IEEE (2008)
19. Leutenegger, S., Chli, M., Siegwart, R.: Brisk: Binary robust invariant scalable keypoints. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2548–2555. IEEE (2011)
20. Li, J., Hu, Q., Ai, M.: 4fp-structure: a robust local region feature descriptor. Photogramm. Eng. Remote Sens. **83**(12), 813–826 (2017)
21. Li, J., Hu, Q., Ai, M., Zhong, R.: Robust feature matching via support-line voting and affine-invariant ratios. ISPRS J. Photogramm. Remote Sens. **132**, 61–76 (2017)
22. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: common objects in context. In: European Conference on Computer Vision, pp. 740–755. Springer (2014)
23. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
24. Ma, J., Zhao, J., Tian, J., Yuille, A.L., Tu, Z.: Robust point matching via vector field consensus. IEEE Trans. Image Process. **23**(4), 1706–1721 (2014)
25. Ma, J., Zhou, H., Zhao, J., Gao, Y., Jiang, J., Tian, J.: Robust feature matching for remote sensing image registration via locally linear transforming. IEEE Trans. Geosci. Remote Sens. **53**(12), 6469–6481 (2015)
26. Mueller, M., Smith, N., Ghanem, B.: A benchmark and simulator for UAV tracking. In: European Conference on Computer Vision, pp. 445–461. Springer (2016)
27. Ono, Y., Trulls, E., Fua, P., Yi, K.M.: Lf-net: learning local features from images. In: Advances in Neural Information Processing Systems, pp. 6234–6244 (2018)
28. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: Orb: an efficient alternative to sift or surf. In: ICCV, vol. 11, p. 2. Citeseer (2011)
29. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. Int. J. Comput. Vis. **9**(2), 137–154 (1992)
30. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment a modern synthesis. In: International Workshop on Vision Algorithms, pp. 298–372. Springer (1999)
31. Vedaldi, A., Fulkerson, B.: Vlfeat: an open and portable library of computer vision algorithms. In: Proceedings of the 18th ACM International Conference on Multimedia, pp. 1469–1472. ACM (2010)
32. Wang, C., Wang, L., Liu, L.: Progressive mode-seeking on graphs for sparse feature matching. In: European Conference on Computer Vision, pp. 788–802. Springer (2014)
33. Weinmann, M., Weinmann, M., Hinz, S., Jutzi, B.: Fast and automatic image-based registration of TLS data. ISPRS J. Photogramm. Remote Sens. **66**(6), S62–S70 (2011)
34. Wu, C., et al.: Visualsfm: a visual structure from motion system (2011)
35. Yan, Q., Yang, L., Liang, C., Liu, H., Hu, R., Xiao, C.: Geometrically based linear iterative clustering for quantitative feature correspondence. In: Computer Graphics Forum, vol. 35, pp. 1–10. Wiley Online Library (2016)
36. Zhang, G., Liu, H., Dong, Z., Jia, J., Wong, T.T., Bao, H.: Efficient non-consecutive feature tracking for robust structure-from-motion. IEEE Trans. Image Process. **25**(12), 5957–5970 (2016)
37. Zitova, B., Flusser, J.: Image registration methods: a survey. Image Vis. Comput. **21**(11), 977–1000 (2003)



Dr. Huajun Liu received Ph.D. degree from School of Computer at Wuhan University. Now he is an associate professor in School of Computer Science, Wuhan University, P.R.China. In recent years, Prof. Liu's research interests include virtual geographic environments, computer vision, computational photography and deep learning.

Shiran Tang received B.S. degree from Mianyang Teachers' college and M.S. degree from School of Computer Science, Wuhan University, P.R.China. His research interests include deep learning, image processing and SLAM.

Dian Lei received B.S. degree from Nanjing University of Aeronautics and Astronautics. Now she is pursuing M.S. degree in School of Computer Science, Wuhan University, P.R.China. Her research interests include computer vision, deep learning and image processing.

Dr. Qing Zhu is Chang Jiang Scholars professor in photogrammetry and GIS, Professor Committee Director, Faculty of Geosciences and Environmental Engineering of Southwest Jiaotong University, P.R.China. In recent years, Prof. Zhu 's research interests include digital terrain modeling (DEM), three-dimensional geographic information system (3D GIS) and virtual geographic environments (VGE).

Dr. Haigang Sui received the Ph.D. degree from Wuhan University, China, in 2002. He is currently a Professor in remote sensing with Wuhan University, Wuhan, China. He has authored or coauthored more than 50 scientific articles in journals, books, and conference proceedings. His work focuses on remote sensing.

Gaojian Zhang received B.S. degree from Wuhan University of Science and Technology. Now he is pursuing M.S. degree in School of Computer Science, Wuhan University, P.R.China. His research interests include computer vision and deep learning.

Chao Li received B.S. degree from Lanzhou University. Now he is pursuing M.S. degree in School of Computer Science, Wuhan University, P.R.China. His research interests include computer vision, deep learning and video translation.