Ad-RMS: Adaptive Regional Motion Statistics for Feature Matching Filtering

Bin Nan[1(✉)], Yinghui Wang[1(✉)], Yanxing Liang[1], Min Wu[2], Pengjiang Qian[1], and Gang Lin[3]

[1] School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, China
`6201613037@stu.jiangnan.edu.cn`, `wangyh@jiangnan.edu.cn`
[2] School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China
[3] Gastroenterology Department, Affiliated Hospital of Jiangnan University, Wuxi 214122, China

Abstract. The Grid-based Motion Statistics (GMS) is a popular feature matching filtering method, and can effectively support 3D reconstruction systems such as ORB-SLAM and has been used effectively in many fields. However, the GMS divides the image into a certain number of grids with a fixed size, which cannot better reflect the feature information of the region. So that when large affine changes occur in the images, the grids cannot delineate a reasonable consistent region. Such a region division leads to errors and even failures in the subsequent process using the grid to reject error feature matching. As a consequence, this paper proposes an adaptive regional motion statistics method based on adaptive region division for region detection to replace the fixed grid division, which enhances the affine invariance of the feature matching filtering algorithm, and verifies the effectiveness of this paper's method through the precision experiments of feature matching and homography matrix.

Keywords: Adaptive region · Regional motion statistics · Feature matching filtering · Affine invariance

1 Introduction

Feature matching is a fundamental part of computer vision research and is indispensable in applications such as image stitching [1], copy-move forgery detection [2], and 3D reconstruction. The main task of feature matching is to find correspondences between feature points in two images based on their descriptors, and it has always been a challenge to reject incorrect correspondences. The common methods of rejecting false matches are computationally complex and prone to global non-smoothness, which affects the accuracy of the rejected false matches [3].

The proposal of the GMS (Grid-based Motion Statistics) algorithm [4] effectively alleviates this problem to some extent, that is, GMS assumes that motion consistency will prevent random mismatches in a certain region from gathering in a certain region of another image. Through this assumption, the GMS algorithm is urged to achieve

more universal feature matching filtering without strict geometric constraints, and the matching precision is not lower than that of other algorithms in a shorter running time.

However, the GMS algorithm does not work as well with the more complex affine transform, which includes not only rotation and scale transformations but also shear transformation. which cause more types of deformations in the image and make it more difficult to filter feature matching. Liu [5] adding the LK optical flow constraint before using the GMS algorithm makes the algorithm more robust to rotation, illumination and blur changes, but does not solve the problem of affine transformation. Specifically, if the affine transformation occurs in the images, the GMS algorithm will have the problem of feature matching filtering as shown in Fig. 1. The reason for the problem is that the affine transformation makes the dissimilar local patterns similar, resulting in more feature matching in the more similar regions, and only less feature matching corresponds to the correct regions. When using the GMS algorithm to find a better corresponding relationship for feature matching, the adaptability to affine transformation first depends on the way of grids dividing, and then on the way of determining the correct correspondence of grids. Due to the fixed size grids of the original algorithm, there are several grids containing similar regional features in the image, and the correct grid correspondence and feature matching is filtered in the grid-based motion statistics judgment. The region features include grey scale, texture, geometry, etc. Thus, these problems encountered by the GMS algorithm are improved if the features in the regions divided by the same image are all distinguishable from other regions, that is, the regions are not similar.

From the above analysis, one of the reasons why the GMS algorithm cannot work well in the case of affine transformations of images is that the grids division does not have affine invariance. Therefore, in this paper, a new adaptive regional motion statistics method (Ad-RMS) is proposed based on the GMS algorithm. Specifically, we use the Maximally Stable Extremal Regions (MSER) algorithm [24], based on region detection by image scale to perform adaptive region division of the image instead of the uniform grid division in the GMS algorithm. And the constraints on the GMS are adjusted in detail to make them more suitable for the algorithms in this paper.

(a)   (b)

Fig. 1. The effect of the GMS algorithm on the affine transformation images. In Fig. 1(a), when the degree of affine transformation is small, the GMS filtered features are matched accurately, as shown in the grid area boxed in red; however, in Fig. 1(b), when the degree of affine transformation is large, the GMS algorithm assumes that the green grid is the correct counterpart, but the actual correct counterpart is the red grid. (Color figure online)

2   Related Works

In this paper, we focus on the improvement of feature matching precision by filtering out incorrect matching and retaining more correct feature point correspondences, and by constructing algorithms that can accommodate feature matching screening of affine transformed image pairs. At the same time, the constructed algorithm can adapt to the feature matching filtering of affine transformed image pairs. Typical feature extraction and description algorithms with some adaptability to the affine transform are SIFT [6], Harris-Affine, Hessian-Affine [7] and ASIFT [8]. The similarity comparison of the obtained feature descriptors in the above session is generally done using the nearest neighbor distance ratio, such as KNN (K-Nearest Neighbor) [9] and FLANN (Fast Library for Approximate Nearest Neighbors) [10], which in turn determine the distance from the measurement space to establish a preliminary correspondence of feature points.

The next step is the removal of incorrect matches, i.e. using local or global consistency constraints to filter the initial correspondence from the feature matching to get the correct correspondence. The most commonly used algorithm for the removal of false matches is based on resampling, represented by the classical method RANSAC [11]. It sets the correspondence algorithm of the feature matches as a parametric geometric relation such as using a fundamental matrix or a homography matrix. Later scholars made a series of improvements to RANSAC, such as LO-RANSAC [12], PROSAC [13], MAGSAC++ [14], and so on. However, this kind of method of rejecting incorrect feature matching is greatly affected by outliers. If the proportion of outliers is relatively large, it is very easy to reject the correct feature matching.

At the same time, some algorithms relax the geometric constraint and combine it with other constraints to achieve a good rejection of mis-matching in the face of affine transformations. For example, in recent years, there are the methods proposed by Jiang et al. [15] to make the corresponding algorithm for feature matching into spatial clustering with outliers. Lee et al. [16] define this type of problem as a Markov random field. Maier et al. [17] propose a guided matching algorithm based on statistical optical flow (GMbSOF). Lipman et al. [18] propose methods such as the Twisted Boundary Algorithm for solving feature point sets. However, the GMS algorithm has simpler constraints, lower computational cost, and better algorithm performance than the above algorithms.

The GMS algorithm divides the regions by a certain number of grids. This method is more general and the division is not affected by the image, but it lacks region information and consistency of the regions. Besides the grid-based region dividing methods, adaptive region dividing methods can also be used. The adaptive region method divides the image based on features such as grey scale, texture and geometry, and the resulting regions are non-intersecting and have distinctly different features. The advantage of the adaptive region division method is the consistency of the regions, i.e. when the images are transformed, the corresponding region can still be detected by this type of division. Traditional algorithms commonly used are the seeded region growing algorithm [19], region splitting and merging algorithm [20], and watershed algorithm [21]. The seeded region growing algorithm requires a homogeneous image pixel feature and a long computation time. The region splitting and merging algorithm is computationally large and region boundaries are easily lost. Compared with the two former algorithms,

the watershed algorithm can better preserve the boundaries of regions, and the MSER algorithm [22] implemented based on the watershed idea can extract stable regions in the image, even if the image undergoes various types of transformation, it has better adaptability.

In addition to the above three classical algorithmic concepts, other algorithms can achieve adaptive division of regions, such as detection algorithms based on feature space clustering [23]. There are various methods derived from clustering and numerous application scenarios, but the results of clustering algorithms are greatly influenced by parameters and more factors need to be considered, such as the number of clusters, initial parameters and operational complexity.

3   Method

In this paper, we propose a feature matching filtering algorithm with adaptive regional motion statistics. With the core idea of local support matching of the GMS algorithm, we combine the method of dividing adaptive regions with regional motion statistics to constitute the Ad-RMS algorithm, and the specific algorithm framework diagram is shown in Fig. 2.
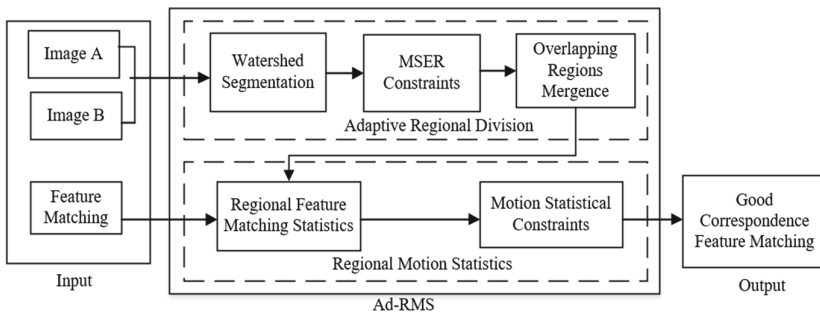

Fig. 2. Framework of our method.

The Fig. 2 shows the basic framework of the Ad-RMS algorithm in this paper. The algorithm is mainly divided into two modules: adaptive region division and regional motion statistics. The grey-scale maps of image A and image B are input to the adaptive regional division module individually. The images are divided into connected regions under different thresholds by the watershed algorithm. And the stable adaptive regions are obtained by the maximum stable extreme value region constraint in the MSER algorithm, and the overlapping parts of the adaptive region are merged to obtain the region division result of the two images. The above division results and feature point matches are input to the regional motion statistics module, where the corresponding feature matches for each region are counted and filtered by the motion statistics constraints to produce the corresponding region and the filtered feature point matches.

3.1   Adaptive Regional Division

This algorithm uses an adaptive regional division method for the input image, using Nister David's modified watershed method to implement the MSER algorithm [24]. The core idea of the watershed approach is to fill the current basin with water at any place and then spread it around until the whole image is submerged, obtaining a connected area at each level as the water level rises.

This is achieved by starting from a point in the image and using the 4-neighborhood lookup to create a set of pixel points related to the grey level threshold of the current point. And during the lookup process, the set of points is manipulated according to the change in the threshold of the lookup point to obtain connected regions with different grey level thresholds. The core process of the watershed algorithm is shown in Algorithm 1.

When Algorithm 1 is finished, the regions under various thresholds can be judged to be maximum stable extreme value regions using Eq. (1), q(i) is the rate of change of region Q_i at threshold i. When it is less than the set maximum rate of change, the connected region is considered to be a maximum stable extreme value region.

$$q(i) = \frac{|Q_i - Q_{i-\text{ delta}}|}{|Q_{i-\text{ delta}}|} \tag{1}$$

Based on the algorithm's idea and process, it is known that the region of a certain threshold is gradually expanded by smaller regions within the region than its threshold, so the list of regions obtained can be labeled in reverse order, and if a pixel has been labeled, the region containing that pixel has been labeled. The Fig. 3 shows the results of using adaptive region delineation.

Algorithm 1: Watershed division

Input: Gray scale image matrix M;
Initialize: region pixel stack C; border pixel heap H; regions R; current = (0,0);

1:   C.push(new block())
2:   **for** $c_i \in$ fourNeighbor(current) **do**
3:     **if** c_i is accessible **then**
4:       H.push(c_i)
5:       **if** c_i.grayValue < current.grayValue **then**
6:         current← c_i; **GOTO** label 2
7:       **end if**
8:     **end if**
9:   **end for**
10: CT ← C. getTopStack(); CT.push(current)
11: h ← H.pop()
12: **if** h is NULL **then**
13:    **exit function**
14: **end if**
15: **if** h.grayValue == CT.grayValue **then**
16:    current ← h; **GOTO** label 3
17: **else**
18:    CT2 ← C.getSecondTopStack();
19:    **if** h.grayValue < CT2.grayValue **then**
20:      CT.grayValue ← h.grayValue;
21:      **GOTO** label 3
22:    **else**
23:      R.push(CT); merge(CT,CT2);
24:      CT ← C.getTopStack(); **GOTO** label 20
25:    **end if**
26: **end if**

Output: regions R

Algorithm 2: Regional Motion Statistics

Input: the region of two images R_a, R_b; the original feature matching m
Initialize: feature matching to each region between the two images M_{ij}; the number of M_{ij} is N_{ij}

1:   **for** $i = 0$ to R_a **do**
2:     $j = 0$, flag=1
3:     **for** $k = 0$ to R_b **do**
4:       **if** $N_{ik} > N_{ij}$ **then**
5:         $j = k$
6:       **end if**
7:     **end for**
8:     **for** $t = 0$ to R_a **do**
9:       **if** $N_{tj} > N_{ij}$ **then**
10:        flag = 0
11:        **break**
12:       **end if**
13:     **end for**
14:     **if** $N_{ij} > \tau$ and flag == 1**then**
15:       *goodMatches*.push(M_{ij})
16:     **end if**
17:   **end for**

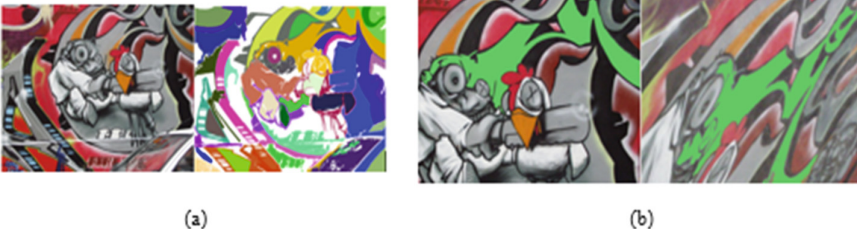**Output**: *goodMatches*

(a)   (b)

Fig. 3. Results of using adaptive regional delineation. Figure 3(a) shows that the region delimitation boundaries are obvious, and one color represents a region and the presence of white undetected regions is identified as an unstable region. Figure 3(b) shows that this method can still delineate the corresponding regions when the image is transformed.

3.2 Regional Motion Statistics

After the adaptive region division in Subsect. 3.1, the filtering of feature matches is carried out using the region motion statistical algorithm after the image has been divided into different regions. The algorithm uses the core idea in the GMS algorithm: the

consistency of motion will make other matches.in the same region have similar motion if they are correctly matched. The motion consistency can be represented by the statistical algorithm, where feature points within a region of the image correspond to another image, and will cluster together if the correspondence is correct, and not vice versa.

When filtering feature matches by the motion statistics method, it can be simply assumed that the idea is more reliable when the total number of matches is higher. It is also possible to derive this idea from the following reduction.

Definition: the number of feature points in the left image I_l is L and the number of feature points in the right image I_r is R. The regions to be matched in the two images are P_l and P_r, and the number of feature points contained in both are l and r. Assume that the matching algorithm is accurate, using as $f_l^t = t$.

From this, we deduce that when regions P_l, P_r are the corresponding regions, the probability that the feature points of region P_l corresponding to the nearest neighbor match in region P_r is denoted as p_t, as shown in Eq. (2). And $p_f=(1 - t)r/R$.

$$p_t = p(f_l^r|T_l^r) = t + (1 - t)\frac{r}{R} \tag{2}$$

The matching of each feature point is independent and the probability of the number of feature points γ_i in the region corresponding to a common region with a feature point can be approximated by a binomial distribution.

$$\gamma_i \sim \begin{cases} B(l, p_t), & \text{the correct region corresponds} \\ B(l, p_f), & \text{the wrong region corresponds} \end{cases} \tag{3}$$

The probability mass function of this binomial distribution can be drawn based on the above equation, as shown in Fig. 4. The values are set to $t = 0.6$ and $r/R = 0.1$. Only when the number of features in the two corresponding regions reaches the threshold, the correct event for the region will occur. By comparing the two plots in Fig. 4, it is confirmed that the greater the distinguishability of the incorrect and correct regional correspondence when there is more feature matching. In turn, this reflects the more reliable constraint on the regional motion statistics at this moment.
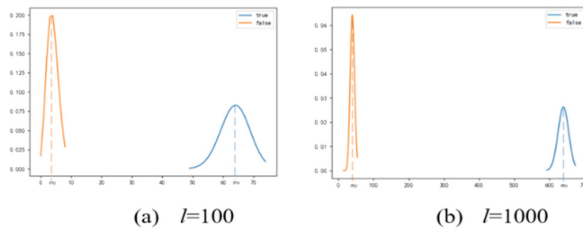

(a) l=100   (b) l=1000

Fig. 4. Corresponding probability mass functions for correct/incorrect regions. Fig. (a) sets $l = 100$ and Fig. (b) sets $l = 1000$. The orange curves are correspondence of the wrong region and the blue curves are correspondence of the correct region. (Color figure online)

The GMS algorithm evaluates the distinction between correct and incorrect region correspondence in the same probability distribution described by P as expressed in

Eq. (4), considering that the larger P is the more significant the distinction. Since P increases as l increases, Eq. (4) is approximated $P \propto \sqrt{l}$. It can be that the more matching points a region contains, the more distinguishable the correct and incorrect matches are.

$$P = \frac{m_t - m_f}{s_t + s_f} = \frac{lp_t - lp_f}{\sqrt{lp_t(1 - p_t)} + \sqrt{lp_f(1 - p_f)}} \propto \sqrt{l} \tag{4}$$

In implementing the algorithm, the method of region division makes regions covering the same grey scale feature information, so that the correspondence of such regions in an image pair can only be one-to-one. This is different from the GMS algorithm where grid pairs should allow many-to-one. The core algorithm for regional motion statistics is shown in Algorithm 2.

The GMS algorithm is to judge the grid as corresponds to the threshold value of $\tau = \alpha \sqrt{l}$, where the α parameter is empirical and is generally set to 6. However, the regional motion statistics method proposed in this paper does not divide the image uniformly into a certain number. Set the threshold $\tau + \alpha l$, where l is the number of feature matching in the region and α takes values from 0 to 1. The algorithm in this paper sets the empirical value α according to the number of feature matching contained in a region. It is necessary to adopt a suitable threshold design, set the threshold as shown in Eq. (5), where the average number of feature points contained in the grid is called *average_count*. The threshold setting of Eq. (5) is based on the previous conclusion: the more number of feature matches in a region, the more obvious distinction between correct and incorrect regions. In this case, a more relaxed threshold can be set to ensure both the correct rate and feature matching of the large region is not easily filtered. On the contrary, setting strict thresholds for small regions only ensure precision.

$$\tau = \begin{cases} 0.98l \, l \in (0, average\_count/4) \\ 0.85l \, l \in (average\_count/4, 4 * average\_count) \\ 0.6l \, l \in (a * average\_count, +\infty) \end{cases} \tag{5}$$

4 Experiment

4.1 Evaluation

The algorithm evaluation metric used in this experiment is based on Mikolajczyk [25] using a precision rate evaluation algorithm. Define image A is transformed into image B by the homography matrix H_1, and feature point a in image A transformed by H_1 and its corresponding coordinates of feature point b in image B. If the distance is less than the threshold ε, the correspondence is considered correct, as shown in Eq. (6). In this paper, the threshold value is set to 1.

$$\varepsilon < dist(H_1 a, b) \tag{6}$$

Building on the idea that evaluation metrics mentioned by Jin [26] should focus on algorithm performance in the downstream task. We also use the precision of calculating the homography matrix after feature matching, which is measured by the precision ratio of the feature matching after the homography matrix transformation.

4.2 Dataset and Input Data

Our algorithm responds to situations where the scene is affine transformed images, mainly using the Graffiti image set from the VGG dataset [25]. This part of image set from the low to a high degree of affine transformation.

In tests of the affine transform images, data from both ASIFT and Harris-Affine feature detection and KNN matching are used as input data for the filtering algorithm. The Ad-RMS algorithm is compared without the use of the false feature matching removal algorithm and the GMS algorithm. The data obtained without using of the false feature matching removal algorithm represents the original matching data that has not been filtered.

Also to measure the performance of the Ad-RMS algorithm in this paper on other datasets, i.e. bikes, boats, leuven. The bikes dataset is blur variation. The boats dataset is rotation with scale variation. The leuven dataset is illumination variation. In this part of the dataset, the common feature extraction algorithms, i.e. SIFT, SURF and ORB, were used as the input data for the algorithm after matching. Using the same three algorithms as above.

4.3 Parameters

By analysing the algorithm processes, the main parameters of this experiment are the *delta* of Eq. (1), the threshold of q(i) (the maximum rate of change of the allowed regions) and the threshold of the percentage of regional features matching set in Eq. (5). Through the above parameter adjustment, the main parameters that have an impact on the results of the algorithm are delta, and α for case 2 in Eq. (5). The results of the influence of the two parameters are shown in Fig. 5.

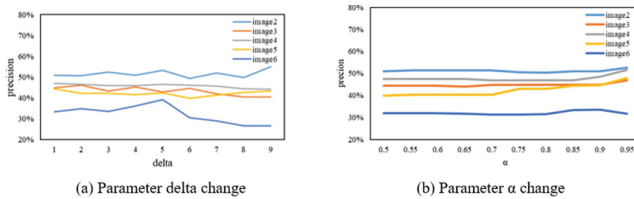(a) Parameter delta change   (b) Parameter α change

Fig. 5. The main parameters influence

The Fig. 5(a) shows how the algorithm precision varies for the parameter *delta* in the range [1, 9]. The average of precision over the same pair of images is calculated to be close to the accuracy when *delta* is 1. And it can be inferred from the formula that when *delta* is 1, the constraint on the region is smaller and the image region can be preserved as much as possible. Therefore, the parameter of *delta* is generally set to 1.

The Fig. 5(b) shows the variation of the algorithm precision for the threshold α adjustment range of [0.5, 0.95] for the condition 2 in Eq. (5). The higher the threshold value, the higher the precision. But the number of filtered feature matches decreases sharply. And the threshold value of 0.85 is considered to be an appropriate value.

4.4   Comparative Analysis of Results

The Fig. 6 show the results of the initial matching data obtained by the ASIFT and Harris-Affine algorithms under the Graffiti dataset when input to different filtering algorithms. The feature matching results of ASIFT are better input data for the affine transform. The filtering performed on this basis better reflects the performance of the algorithm. In terms of the precision of the algorithm, the Ad-RMS algorithm has improved over the GMS algorithm. And the homography precision of the Ad-RMS algorithm is significantly higher than that of the GMS algorithm in the case of large affine transformations. The Harris-Affine algorithm shows that the GMS and Ad-RMS algorithm can still improve the matching precision of feature matching in the case of poor input data. And the Ad-RMS algorithm in this paper is more effective.
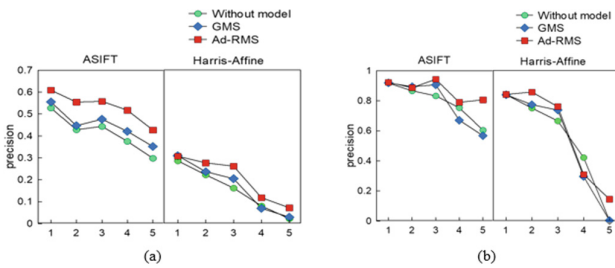

Fig. 6. Experimental comparison of the Graffiti dataset. Where each image contains the variation of the precision rate in Fig. (a) and the variation of the homography precision in Fig. (b).

The Fig. 7 show the comparison results of the different filtering algorithms under the three datasets bikes, boats and leuven. In most experiments, our algorithm is better than the GMS algorithm. However, only when the illuminated scenes, our algorithm is affected by the limitations of the made region division which causes its results to be poor under the SIFT input data. For other input data in this scene, our algorithm is still better than the GMS algorithm.

5   Summary

The advantage of grid-based motion statistics is to correlate the motion consistency of feature matching with its statistical distribution. This allows for faster and more stable rejection of incorrect feature matching. But in the case of large image affine transformations, dividing the image into equal rectangular regions is not good enough for filtering false feature matching. Therefore, this paper proposes the Ad-RMS algorithm based on the GMS algorithm. Our algorithm adaptively divides regions based on image grey-scale values, then uses regions that do not intersect each other and have consistent image information for regional motion statistics, improving the adaptability and robustness of filtering feature matching in the case of affine transformation of the images. In a normal scene dataset, the feature matching data obtained by different feature extraction algorithms are filtered, and our algorithm has an advantage over the GMS algorithm in more
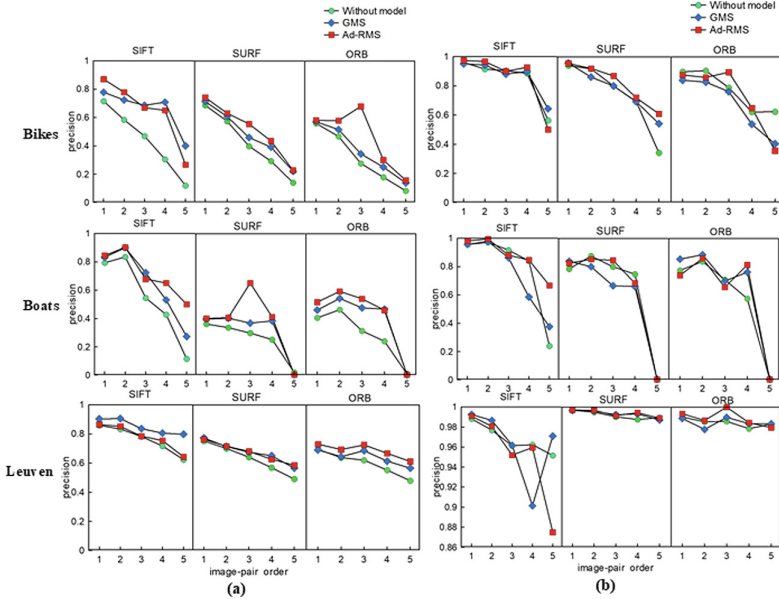
Fig. 7. Test results for other image set. Each image contains the variation of the precision in Fig. (a) and the homography precision in Fig. (b).

scenes. It is also verified that our algorithm provides good support for the application of various methods for multi-view 3D reconstruction such as SLAM, SFM, etc.

References

1. Liu, Q., Xi, X., Zhang, W., Yang, L., Hanajima, N.: Improved image matching algorithm based on LK optical flow and grid motion statistics. Int. J. Comput. Appl. T. **68**(1), 49–57 (2022)
2. Shi, Z., Wang, P., Cao, Q., Ding, C., Luo, T.: Misalignment-eliminated warping image stitching method with grid-based motion statistics matching. Multimed. Tools Appl. **81**(8), 10723–10742 (2022)
3. Lin, WY.D., Cheng, M., Lu, J., Yang, H., Do, M.N., Torr, P.: Bilateral functions for global motion modeling. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol. 8692, pp. 341–356. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10593-2_23
4. Bian, J.W., Lin, W.Y., Matsushita, Y., Yeung, S.K., Nguyen, T.D., Cheng, M.M.: GMS: grid-based motion statistics for fast, ultra-robust feature correspondence. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, pp. 4181–4190. IEEE, Honolulu (2017)

5. Gan, Y., Zhong, J., Vong, C., Gan, Y., Zhong, J., Vong, C.: A novel copy-move forgery detection algorithm via feature label matching and hierarchical segmentation filtering. Inf. Process. Manag. **59**(1), 102783 (2022)
6. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**, 91–110 (2004)
7. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. Int. J. Comput. Vis. **60**, 63–86 (2004)
8. Yu, G., Morel, J.M.: ASIFT: an algorithm for fully affine anvariant comparison. Image Process. Line **1**, 11–38 (2011)
9. Altman, N.S.: An introduction to kernel and nearest-neighbor nonparametric regression. Am. Stat. **46**(3), 175–185 (1992)
10. Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. In: Proceedings of the Fourth International Conference on Computer Vision Theory and Applications, VISAPP 2009, pp. 331–340. SciTePress, Lisboa (2009)
11. 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)
12. Chum, O., Matas, J., Kittler, J.: Locally optimized RANSAC. In: Michaelis, B., Krell, G. (eds.) Pattern Recognition. DAGM 2003. Lecture Notes in Computer Science, vol. 2781, pp. 236–243. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-45243-0_31
13. Chum, O., Matas, J.: Matching with PROSAC-progressive sample consensus. In: Proceedings of Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2005, pp. 220–226. IEEE, Washington (2005)
14. Barath, D., Noskova, J., Ivashechkin, M., Matas, J.: MAGSAC++, a fast, reliable and accurate robust estimator. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR 2020, pp. 1304–1312. IEEE, Seattle (2020)
15. Jiang, X., Ma, J., Jiang, J., Guo, X.: Robust feature matching using spatial clustering with heavy outliers. IEEE Trans. Image Process. **29**, 736–746 (2019)
16. Lee, S., Lim, J., Suh, I.H.: Progressive feature matching: incremental graph construction and optimization. IEEE Trans. Image Process. **29**, 6992–7005 (2020)
17. Maier, J., Humenberger, M., Murschitz, M., Zendel, O., Vincze, M.: Guided matching based on statistical optical flow for fast and robust correspondence analysis. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science, vol. 9911, pp. 101–117. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_7
18. Lipman, Y., Yagev, S., Poranne, R., Jacobs, D.W., Basri, R.: Feature matching with bounded distortion. ACM Trans. Graph **33**(3), 1–14 (2014)
19. Adams, R., Bischof, L.: Seeded region growing. IEEE Trans. Pattern Anal. Mach. Intell. **16**(6), 641–647 (1994)
20. Li, Y., Li, L.: A split and merge EM algorithm for color image segmentation. In: Proceedings of International Conference on Intelligent Computing and Intelligent Systems, ICIS 2009, pp. 395–399. IEEE, Shanghai (2009)
21. Vincent, L., Soille, P.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Trans. Pattern Anal. Mach. Intell. **13**(6), 583–598 (1991)
22. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image Vis. Comput. **22**(10), 761–767 (2004)
23. Carevic, D., Caelli, T.: Region-based coding of color images using Karhunen-Loeve transform. Graph. Models Image Process. **59**(1), 27–38 (1997)
24. Nistér, D., Stewénius, H.: Linear time maximally stable extremal regions. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) Computer Vision – ECCV 2008. ECCV 2008. Lecture Notes in

Computer Science, vol. 5303, pp. 183–196. Springer, Berlin, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88688-4_14

25. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**(10), 1615–1630 (2005)

26. Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K.M., Trulls, E.: Image matching across wide baselines: from paper to practice. Int. J. Comput. Vis. **129**(2), 517–547 (2021)