



Technical Note

# A Novel and Effective Cooperative RANSAC Image Matching Method Using Geometry Histogram-Based Constructed Reduced Correspondence Set

Kuo-Liang Chung <sup>\*</sup>, Ya-Chi Tseng and Hsuan-Ying Chen

Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei 10672, Taiwan; m10915096@mail.ntust.edu.tw (Y.-C.T.); m11015095@mail.ntust.edu.tw (H.-Y.C.)

\* Correspondence: k.l.chung@mail.ntust.edu.tw

**Abstract:** The success of many computer vision and pattern recognition applications depends on matching local features on two or more images. Because the initial correspondence set—i.e., the set of the initial feature pairs—is often contaminated by mismatches, removing mismatches is a necessary task prior to image matching. In this paper, we first propose a fast geometry histogram-based (GH-based) mismatch removal strategy to construct a reduced correspondence set  $C^{reduced, GH}$  from the initial correspondence set  $C^{ini}$ . Next, we propose an effective cooperative random sample consensus (COOSAC) method for remote sensing image matching. COOSAC consists of a RANSAC, called  $RANSAC^{ini}$  working on  $C^{ini}$ , and a tiny RANSAC, called  $RANSAC^{tiny, GH}$  working on a randomly selected subset of  $C^{reduced, GH}$ . In  $RANSAC^{tiny, GH}$ , an iterative area constraint-based sampling strategy is proposed to estimate the model solution of  $C^{tiny, GH}$  until the specified confidence level is reached, and then  $RANSAC^{ini}$  utilizes the estimated model solution of  $C^{tiny, GH}$  to calculate the inlier rate of  $C^{ini}$ . COOSAC repeats the above cooperation between  $RANSAC^{tiny, GH}$  and  $RANSAC^{ini}$  until the specified confidence level is reached, reporting the resultant model solution of  $C^{ini}$ . For convenience, our image matching method is called the GH-COOSAC method. Based on several testing datasets, thorough experimental results demonstrate that the proposed GH-COOSAC method achieves lower computational cost and higher matching accuracy benefits when compared with the state-of-the-art image matching methods.

**Keywords:** confidence level; computational cost; hypothesize-and-verify; inlier rate; matching accuracy; model solution



**Citation:** Chung, K.-L.; Tseng, Y.-C.; Chen, H.-Y. A Novel and Effective Cooperative RANSAC Image Matching Method Using Geometry Histogram-Based Constructed Reduced Correspondence Set. *Remote Sens.* **2022**, *14*, 3256. <https://doi.org/10.3390/rs14143256>

Academic Editor: Liang-Jian Deng

Received: 3 June 2022

Accepted: 4 July 2022

Published: 6 July 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Remote sensing images have been applied in many applications, such as disaster assessment, construction site monitoring, building change detection, military applications, and so on. To extend the areas that remote sensing images can cover, image matching has received extensive attention. In the image matching field, establishing reliable feature correspondences for two or more images is a challenging issue, and it has many applications in stitching, registration, retrieval, object recognition, and 3D reconstruction. Prior to image matching, using the difference-of-Gaussian (DOG) technique, contrast checking strategy, and Hessian matrix-based testing rule, Lowe's scale-invariant feature transform (SIFT) [1] is used to extract the feature point sets of the source image  $I_s$  and that of the target image  $I_t$ , respectively. Furthermore, the initial correspondence set  $C^{ini}$  between the two feature point sets, i.e., the set of the initial feature pairs, is constructed such that for each correspondence  $c_i$  in  $C^{ini}$ ,  $1 \leq i \leq |C^{ini}|$ , the 128-dimensional feature vector of the left feature point of  $c_i$  is more similar to that of the right feature point of  $c_i$  than the other right feature points in  $C^{ini}$ . The initial correspondence set  $C^{ini}$  is usually contaminated by mismatches, i.e., outliers.

In what follows, given an initial correspondence set  $C^{ini}$ , we introduce the related works for removing mismatches and image matching.

### 1.1. Related Works

Given  $C^{ini}$ , Fischer and Bolles [2] proposed a pioneer image matching work, called the random sample consensus (RANSAC) method, to determine the homography transformation between two feature point sets in  $C^{ini}$ . Usually, the determined homography transformation is also called the model solution of  $C^{ini}$ . The detailed definition of the model solution of  $C^{ini}$  will be given in Section 3.1. The RANSAC method first randomly selects four correspondences, i.e., four feature-pairs, from  $C^{ini}$  to estimate the model solution as a hypothesis, and then using the estimated model solution, it counts the number of fitting correspondences and verifies the inlier rate of  $C^{ini}$ , denoted by  $IR^{ini,current}$ , where the inlier rate of  $C^{ini}$  is defined to be the number of fitting correspondences over the size of  $C^{ini}$ . Next, RANSAC repeats the above hypothesize-and-verify (HAV) process, and then it uses the newly estimated model solution to calculate the new inlier rate of  $C^{ini}$ , denoted by  $IR^{ini,new}$ . If  $IR^{ini,current} < IR^{ini,new}$ , the previous inlier rate  $IR^{ini,current}$  is replaced by the new inlier rate  $IR^{ini,new}$ ; otherwise, we repeat the above HAV process until the specified confidence level is reached. Here, the confidence level of RANSAC is defined to be a probability value  $p = 1 - (1 - (IR^{ini,current})^4)^k$ , where  $k$  denotes the number of iterative HAV processes required in RANSAC. The resultant model solution of  $C^{ini}$  thus serves as the established correspondence between two feature point sets in  $C^{ini}$ . Although RANSAC is robust, it usually requires huge computational cost under low inlier rate cases.

Capel [3] proposed a bail-out test approach to accelerate RANSAC. In his approach, first, a correspondence subset  $C^{sub}$  is constructed by randomly selecting a subset of  $C^{ini}$ . Using one model solution of  $C^{sub}$ , if the inlier rate of  $C^{sub}$  is significantly less than the current inlier rate of  $C^{ini}$ , the subset  $C^{sub}$  and the estimated model solution are discarded; otherwise, the inlier rate of  $C^{sub}$  is larger than the current inlier rate of  $C^{ini}$  and the estimated model solution of  $C^{sub}$  is used to calculate the inlier rate of  $C^{ini}$ . As a result, the bail-out test approach could reduce the average computational cost involved in the verification of each hypothesis, improving the computational cost performance of RANSAC. However, this approach might incorrectly reject good hypotheses; on the other hand, it might suffer from the accuracy degradation.

Instead of treating all correspondences equally in RANSAC, Chum and Matas [4] proposed a progressive sample consensus method to assign different sampling priorities of all correspondences in  $C^{ini}$  by using the ratio test strategy. Hence, their method is robust against a large number of correspondences with low inlier rate. The lower the correspondence's priority is, the later the correspondence participates the HAV process. For example, at the beginning of their method, the four correspondences with top four priorities are first selected as the hypothesis to calculate the inlier rate of  $C^{ini}$ . In the same argument, the four correspondences with slightly lower priorities are selected as the hypothesis in the next iteration. Under low inlier rate cases, their method achieved good robustness. However, due to the priority-based selection rule, the four correspondences with low priorities are often discarded after reaching the termination condition, leading to the matching accuracy degradation.

Matas and Chum [5] proposed a randomized RANSAC method to accelerate RANSAC. In their method, a small subset  $C^{small}$  is first constructed from  $C^{ini}$ . Once all correspondences in  $C^{small}$  are all inliers, the estimated model solution of  $C^{small}$  is used to calculate the inlier rate of the remaining correspondences in  $C^{ini} - C^{small}$  where the operator "-" denotes the set difference operator. Their method achieved better computation performance relative to RANSAC and the bail-out test approach [3]. However, there is room for improvement on the matching accuracy performance.

Due to the simplicity and robustness of RANSAC, the above-mentioned related works [3–5] are inspired by RANSAC. In [6], Raguram et al. provided a comprehensive overview of the RANSAC-like methods and introduced a universal RANSAC framework

to extend the HAV process of RANSAC by considering more practical and computational issues, such as prefiltering unreliable correspondences, performing sampling check, etc. Consequently, it brings out more aspects in the image matching issue.

Ma et al. [7] proposed an effective vector field consensus (VFC) method to establish a reduced correspondence set  $C^{reduced}$  from  $C^{ini}$ . In VFC, the reduced correspondence set is determined based on the coherence of the underlying motion fields rather than the geometric constraints. The inlier rate of  $C^{ini}$  is expressed as the rate of the size of  $C^{reduced}$ , denoted by  $|C^{reduced}|$ , over  $|C^{ini}|$ , accelerating the matching performance and preserving the accuracy performance. Using the classical density-based spatial clustering method of applications with noise [8], Jiang et al. proposed a robust feature matching (RFM) method [9] to detect groups of matches with similar motion. In RFM, the correspondence subset with inconsistent motion features is discarded. Setting the inlier rate of  $C^{ini}$  to the rate of  $|C^{reduced}|$  over  $|C^{ini}|$ , RFM can also accelerate the matching performance and preserve the accuracy performance.

Ma et al. [10] proposed a locality preserving (LP) matching method to prune correspondences in  $C^{ini}$  if they violate the spatial neighborhood relationship. Using Ma et al.'s method [10] as a preprocessing step to create a reduced correspondence set  $C^{reduced}$ , Wang et al. [11] randomly sampled four correspondences from  $C^{reduced}$ , and then calculated the estimated model solution of  $C^{reduced}$ . Further, using the estimated model solution, the HAV process is performed on  $C^{reduced}$  to obtain the inlier rate of  $C^{reduced}$ . They repeated the above model solution estimation process for  $C^{reduced}$  until the specified confidence level is reached. Finally, the resultant model solution of  $C^{reduced}$  is used to calculate the inlier rate of  $C^{ini}$ , achieving execution time and accuracy improvements. For convenience, Wang et al.'s method is called the LP-RANSAC method.

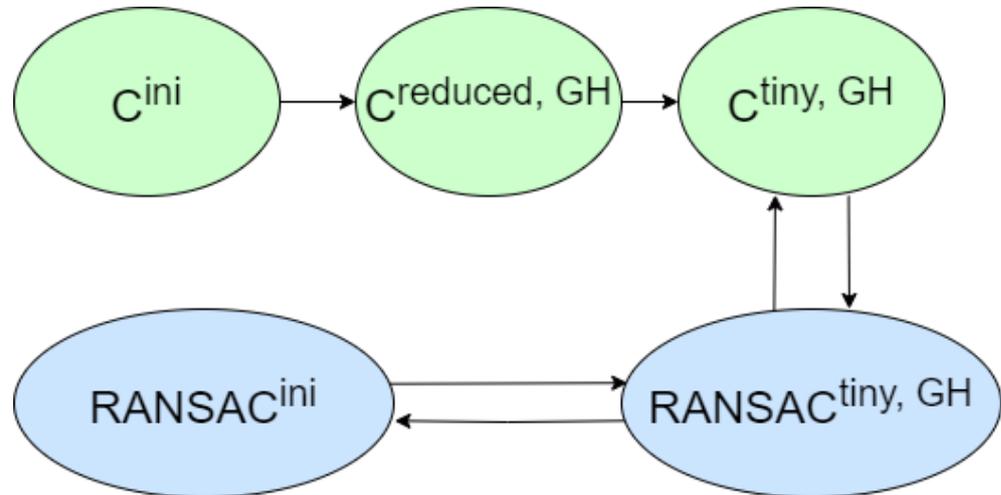
Inspired by the image segmentation and registration methods in [12–14], Liu et al. [15] proposed a motion consistency- and correspondence growing-based method to reduce the correspondence set by using a coarse-to-fine strategy. Wang and Chen [16] proposed a guided local outlier factor (GLOF) method to filter out abnormal correspondences. In GLOF, a tentative correspondence set is first constructed by matching multi-features. Next, mismatches are identified and removed by using the anomaly detection technique and a user-defined threshold, creating a robust reduced correspondence set  $C^{reduced}$ . Expressing the inlier rate of  $C^{ini}$  as the rate of  $|C^{reduced}|$  over  $|C^{ini}|$ , GLOF can enhance the matching accuracy performance.

## 1.2. Contributions

In this paper, for remote sensing matching, we propose a new cooperative RANSAC (COOSAC) method using a geometry histogram-based (GH-based) constructed reduced correspondence set. For convenience, the proposed method is called the GH-COOSAC method, as depicted in Figure 1, and its contributions are clarified below.

1. Differing from the previous correspondence set reduction methods, we propose a fast GH-based mismatch removal strategy to construct a reduced correspondence set  $C^{reduced,GH}$  from  $C^{ini}$  such that for all correspondences in  $C^{reduced,GH}$ , the orientation- and length-consistency can be preserved.
2. Differing from the related image matching works introduced in Section 1.1, we propose a GH-COOSAC method which consists of a RANSAC, called  $RANSAC^{ini}$  working on  $C^{ini}$ , and a tiny RANSAC, called  $RANSAC^{tiny,GH}$  working on a tiny correspondence set  $C^{tiny,GH}$  that is randomly constructed from a small subset of  $C^{reduced,GH}$ . In addition, an iterative area constraint-based sampling strategy is deployed into  $RANSAC^{tiny}$  to estimate the model solution of  $C^{tiny,GH}$  until the specified confidence level is reached. Then,  $RANSAC^{ini}$  utilizes the estimated model solution  $H$  to calculate the inlier rate of  $C^{ini}$ . We repeat the above cooperative HAV processes between  $RANSAC^{tiny,GH}$  and  $RANSAC^{ini}$  until the specified confidence level is reached, reporting the best model solution of  $C^{ini}$ . The advantages of the proposed method are

- low computational cost, in particular, under the inlier rate interval [0.5, 0.9], and high matching accuracy benefits when compared with the other methods [2,7,9,11,16].
- Based on several testing datasets with different inlier rates, thorough experimental results justify that the proposed GH-COOSAC method achieves low computational cost and high matching accuracy benefits relative to RANSAC [2] and the four state-of-the-art methods, namely VFC [7], RFM [9], LP-RANSAC [11], and GLOF [16].



**Figure 1.** The sketched flow diagram of the proposed GH-COOSAC method.

The remainder of this paper is organized as follows. In Section 2, the proposed GH-based mismatch removal strategy is presented to construct a reduced correspondence set  $C^{reduced, GH}$ . In Section 3, the proposed GH-COOSAC method is presented. In Section 4, thorough experiments are carried out to demonstrate low computational cost and high matching accuracy benefits of the proposed method. In Section 5, some concluding remarks are addressed.

## 2. The Proposed Geometry Histogram-Based (GH-Based) Mismatch Removal Strategy to Construct a Reduced Correspondence Set

Based on the initial correspondence set  $C^{ini}$ , this section first presents the proposed fast GH-based mismatch removal strategy to construct a reduced correspondence set  $C^{reduced, GH}$ . Next, the computational complexity of the proposed GH-based strategy is analyzed. Here, each image-pair is assumed to have mainly positional transformation with negligible rotational differences.

### 2.1. Reconstructing the Reduced Correspondence Set $C^{reduced, GH}$

Considering every correspondence  $c_i = [(x_i, y_i), (x'_i, y'_i)]$  in  $C^{ini}$ ,  $1 \leq i \leq |C^{ini}|$ , the angle of  $c_i$  is expressed as

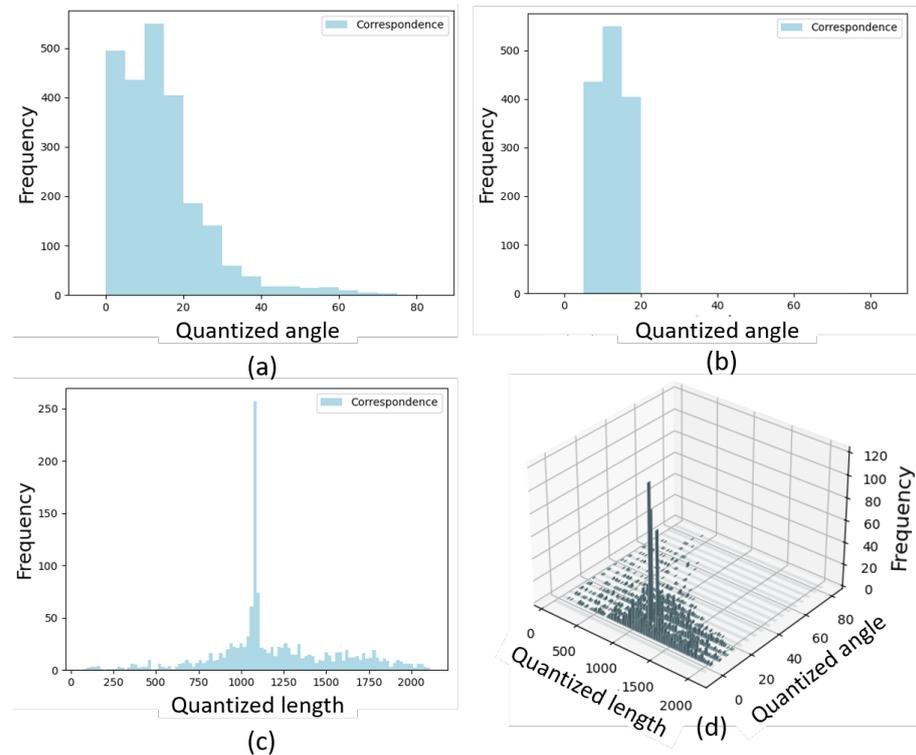
$$Angle(c_i) = \tan^{-1} \frac{y'_i - y_i}{x'_i - x_i} \quad (1)$$

respectively, where  $0 \leq Angle(c_i) \leq 180^\circ$ . Here, the term “angle” and the term “orientation” denote the same thing. In addition, the length of  $c_i$  is expressed as

$$Length(c_i) = \sqrt{(x'_i - x_i)^2 + (y'_i - y_i)^2} \quad (2)$$

Based on the orientations of all correspondences in  $C^{ini}$  which are obtained from one image-pair in the dataset “Airport” (<https://www.sensefly.com/education/datasets/?dataset=1419>, accessed on 6 April 2022), Figure 2a illustrates the quantized orientation

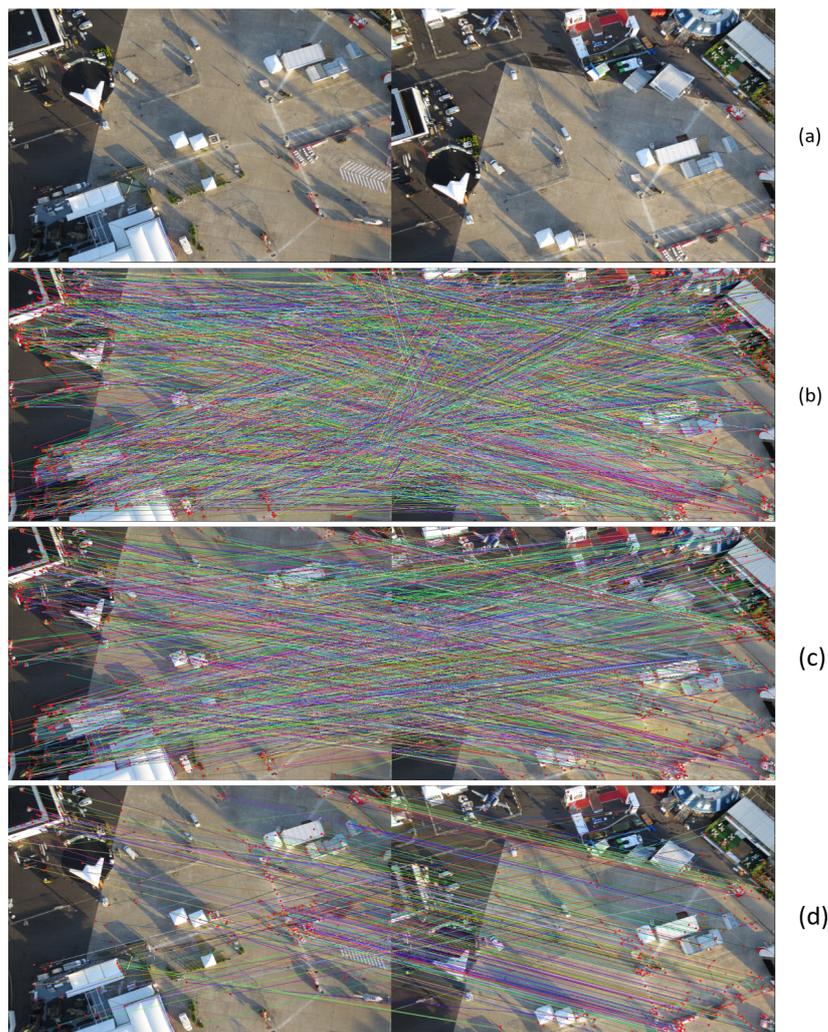
histogram of  $C^{ini}$ , namely  $H(\theta)$ , where the  $x$ -axis denotes the quantized orientation  $\theta$ , where the size of one bin occupies five degrees, and the  $y$ -axis denotes the frequency. In  $H(\theta)$ , we select the correspondence subset in the highest peak and its left and right  $k_1$  neighboring peaks as a tentative reduced correspondence set  $C^{\theta,k_1} (\subset C^{ini})$ . Figure 2b demonstrates the reduced quantized orientation histogram  $H(\theta, k_1)$  for  $k_1 = 1$ . Consequently, we have filtered out the orientation-inconsistency correspondences in  $C^{ini}$ , but retain the relatively orientation-consistent correspondences in  $C^{ini}$ .



**Figure 2.** The construction of a reduced correspondence set  $C^{reduced,GH}$  for one image-pair in the “Airport” dataset. (a) The quantized orientation histogram of  $C^{ini}$ . (b) The quantized orientation histogram of the tentative reduced correspondence set  $C^{\theta,k_1}$ . (c) The quantized length histogram of (b). (d) The histogram of the reduced correspondence set  $C^{reduced,GH}$ .

Based on the lengths of all correspondences in Figure 2b, Figure 2c illustrates the corresponding quantized length histogram of Figure 2b, where the length of one bin in the histogram occupies 20 pixels. Further, we select the correspondence subset in the highest peak and its left and right neighboring peaks of Figure 2c, namely  $C^{\theta,k_1:L,k_2}$  where “L” denotes the quantized length, as the reduced correspondence set  $C^{reduced,GH}$ . Figure 2d demonstrates the histogram of the reduced correspondence set  $C^{reduced,GH}$  for  $k_2 = 1$ . As a result, we collect all correspondences from Figure 2d as the reduced correspondence set  $C^{reduced,GH}$  in which all correspondences are relatively orientation- and length-consistent.

Figure 3a depicts the testing image-pair used in Figure 2. In Figure 3b, the initial correspondence set  $C^{ini}$  of the testing image-pair is shown by those 2643 lines. Based on the quantized orientation histogram in Figure 2b, Figure 3c illustrates the 1390 lines of the reduced correspondence set  $C^{\theta,k_1}$  for  $k_1 = 1$ . Figure 3d illustrates the reduced correspondence set  $C^{reduced,GH}$  with  $|C^{reduced,GH}| = 392$  by using the proposed GH-based mismatch removal strategy.



**Figure 3.** The construction of one reduced correspondence set by using the proposed GH-based mismatch removal strategy. (a) The input image-pair. (b) The initial correspondence set  $C^{ini}$  shown by those 2643 lines. (c) The reduced correspondence set  $C^{\theta,k_1}$  shown by those 1390 lines. (d) The reduced correspondence set  $C^{reduced,GH}$  shown by those 392 lines.

## 2.2. Computational Complexity Analysis

From the description of the proposed GH-based mismatch removal strategy to construct the reduced correspondence set  $C^{reduced,GH}$ , by Equation (1), it takes  $O(1)$  time to calculate the orientation of every  $c_i \in C^{ini}$ . Therefore, it takes  $O(|C^{ini}|)$  time to construct the orientation histogram of the initial correspondence set  $C^{ini}$ .

By Equation (2), the length of every  $c_i \in C^{\theta,1}$  can be calculated in  $O(1)$  time. Therefore, for all correspondences in  $C^{\theta,1}$ , their lengths can be calculated in  $O(|C^{\theta,1}|)$  time. Furthermore, it takes  $O(|C^{\theta,1;length,1}|)$  time to construct the reduced correspondence set  $C^{reduced,GH}$ . Because  $|C^{\theta,1;length,1}| \ll |C^{\theta,1}| \ll |C^{ini}|$ , totally, it takes  $O(|C^{ini}|)$  time to construct the reduced correspondence set  $C^{reduced,GH}$  from  $C^{ini}$ .

**Proposition 1.** Given an initial correspondence set  $C^{ini}$ , the reduced correspondence set  $C^{reduced,GH}$  can be constructed in  $O(|C^{ini}|)$  time, i.e., linear-time, where  $|C^{reduced,GH}| \ll |C^{ini}|$ , by using the proposed GH-based Mismatch Removal Strategy.

## 3. The Proposed GH-COOSAC Method for Image Matching

Given an initial correspondence set  $C^{ini}$ , using the GH-based mismatch removal Strategy mentioned in Section 2, suppose the reduced correspondence set  $C^{reduced,GH}$  has

been constructed. In this section, the proposed GH-COOSAC method for image matching is presented. Initially, a tiny correspondence set  $C^{tiny,GH}$  is constructed by randomly selecting a subset of the reduced correspondence set  $C^{reduced,GH}$ , where  $|C^{tiny,GH}| \ll |C^{reduced,GH}|$  and in our experiment, we set  $|C^{tiny,GH}| = 0.2 * |C^{reduced,GH}|$ .

To provide a more readable introduction, we sketch the logic behind the main steps of the proposed GH-COOSAC method. In the GH-COOSAC method, COOSAC consists of a RANSAC, called  $RANSAC^{ini}$  which works on  $C^{ini}$ , and a tiny RANSAC, called  $RANSAC^{tiny,GH}$  which works on  $C^{tiny,GH}$ . In  $RANSAC^{tiny,GH}$ , an iterative area constraint-based sampling strategy is proposed to estimate the model solution of  $C^{tiny,GH}$  until the specified confidence level, say 99.5%, is reached. Let the estimated model solution of  $C^{tiny,GH}$  be denoted by H. Then,  $RANSAC^{ini}$  utilizes the estimated model solution H to calculate the inlier rate of  $C^{ini}$ . COOSAC repeats the above cooperation between  $RANSAC^{tiny,GH}$  and  $RANSAC^{ini}$  until the specified confidence level of  $RANSAC^{ini}$ , say 99.5%, is reached. Finally, the resultant model solution of  $C^{ini}$  is reported.

### 3.1. The HAV Process in $RANSAC^{tiny,GH}$

As depicted in the hypothesis stage of  $RANSAC^{tiny,GH}$  in Figure 4, the proposed area constraint-based sampling strategy first randomly selects four correspondences from  $C^{tiny,GH}$ , and then it examines whether the area of every two selected correspondences is larger than the specified threshold  $T_{area}$ . Empirically,  $T_{area}$  is set to 1000 pixels. In Figure 5, the area surrounded by two selected correspondences is depicted. If the calculated area of every two selected correspondences is larger than  $T_{area}$ ,  $RANSAC^{tiny,GH}$  proceeds to solve the model solution of  $C^{tiny,GH}$  by using the four selected correspondences; otherwise, the selected four correspondences are dropped, and we select another four correspondences from  $C^{tiny,GH}$  again.

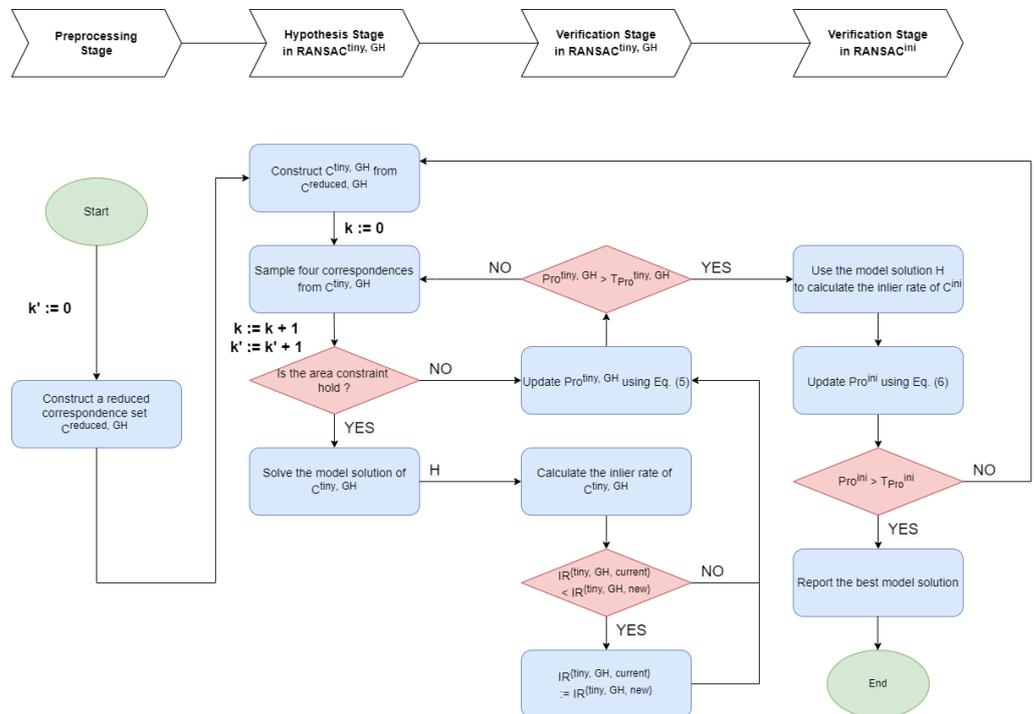
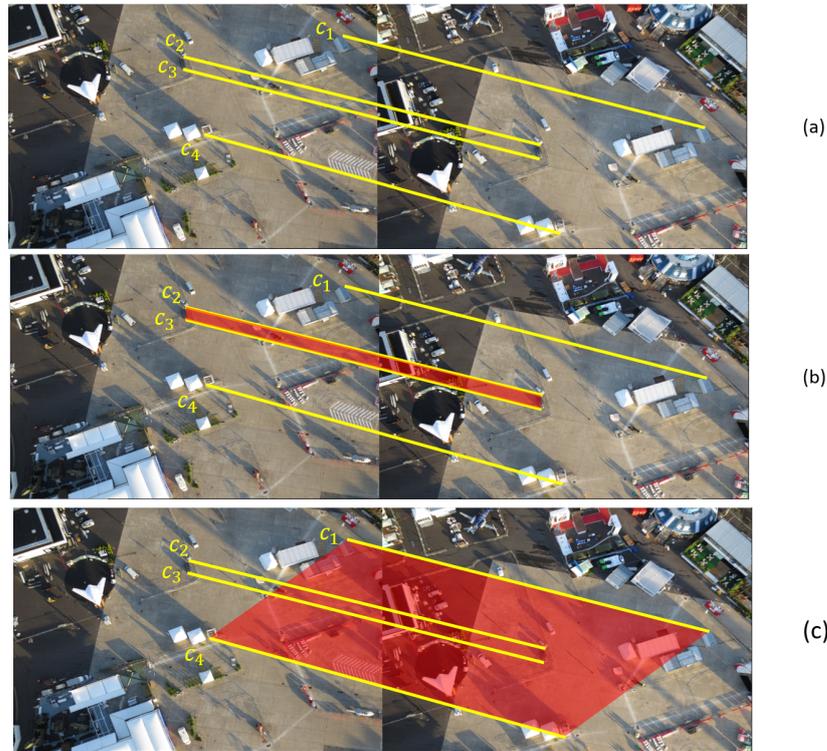


Figure 4. The detailed flowchart of the proposed GH-COOSAC method.

Suppose the four selected correspondences are denoted by  $c_1 = ((x_1, y_1), (x'_1, y'_1))$ ,  $c_2 = ((x_2, y_2), (x'_2, y'_2))$ ,  $c_3 = ((x_3, y_3), (x'_3, y'_3))$ , and  $c_4 = ((x_4, y_4), (x'_4, y'_4))$ . Among the four selected correspondences, selecting any two correspondences as one combination,

there are six combinations in total. For each combination, denoted by  $(c_i, c_j)$  for  $i \neq j$ , the area of the quadrilateral surrounded by  $c_i$  and  $c_j$  can be calculated by

$$Area(c_i, c_j) = \left| \frac{x_1 * y_2 + x_2 * y'_2 + x'_2 * y'_1 + x'_1 * y_1 - y_1 * x_2 - y_2 * x'_2 - y'_2 * x'_1 - y'_1 * x_1}{2} \right| \quad (3)$$



**Figure 5.** The depiction of the area surrounded by two selected correspondences used in the area constraint-based sampling strategy. (a) Four selected correspondences,  $c_1$ ,  $c_2$ ,  $c_3$ , and  $c_4$ , marked by four yellow lines. (b) The area (in red) surrounded by the two correspondences,  $c_2$  and  $c_3$ , being less than the specified threshold  $T_{area}$ . (c) The area (in red) surrounded by the two correspondences,  $c_1$  and  $c_4$ , being larger than the specified threshold  $T_{area}$ .

In the six combinations, if there exists one combination  $(c_i, c_j)$  whose area is less than the specified threshold  $T_{area}$ , it indicates that the four selected correspondences,  $c_1$ ,  $c_2$ ,  $c_3$ , and  $c_4$ , are invalid and should be discarded. Figure 5 illustrates one invalid case because the area surrounded by  $c_2$  and  $c_3$  is less than the threshold, so the four selected correspondences are discarded. Otherwise, if the area of every combination is larger than or equal to  $T_{area}$ , the four selected correspondences are valid and can be used to estimate the model solution of  $C^{tiny, GH}$ .

Then, we put the coordinates of the four valid correspondences,  $c_1$ ,  $c_2$ ,  $c_3$ , and  $c_4$ , into the following homography equation:

$$\begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \quad (4)$$

where  $((x_i, y_i), (x'_i, y'_i)) = c_i$  for  $1 \leq i \leq 4$ . By using the SVD (singular value decomposition) method [17] to solve Equation (4), the eight homography parameters,  $h_{00}$ ,  $h_{01}$ ,  $h_{02}$ ,  $h_{10}$ ,  $h_{11}$ ,  $h_{12}$ ,  $h_{20}$ , and  $h_{21}$ , can be solved, and the solution is used as the estimated model solution of  $C^{tiny, GH}$ .

As depicted in the verification stage of Figure 4, we utilize the estimated model solution of  $C^{tiny, GH}$  to calculate the inlier rate of  $C^{tiny, GH}$ , denoted by  $IR^{tiny, GH, new}$ . Let  $IR^{tiny, GH, current}$

denote the highest inlier rate of  $C^{tiny,GH}$  so far. If  $IR^{tiny,GH,current} < IR^{tiny,GH,new}$ ,  $IR^{tiny,GH,current}$  is replaced by  $IR^{tiny,GH,new}$ ; otherwise, the value of  $IR^{tiny,GH,current}$  is retained.

The above HAV process of  $RANSAC^{tiny,GH}$  is repeated until the confidence level of  $RANSAC^{tiny,GH}$ , denoted by  $Pro^{tiny,GH}$ , is larger than the specified confidence level. Here,  $Pro^{tiny,GH}$  is defined by

$$Pro^{tiny,GH} = 1 - (1 - (IR^{tiny,GH,current})^4)^k \tag{5}$$

where the parameter  $k$  denotes the number of iterations required by  $RANSAC^{tiny,GH}$ . For example, suppose  $IR^{tiny,GH,current} = 0.6$  and the iteration number spent is  $k = 32$ , the value of  $Pro^{tiny,GH}$  can achieve at least 99% ( $= 1 - (1 - (0.6)^4)^{32}$ ). In our implementation, as demonstrated in Section 4, the specified confidence level  $T_{Pro^{tiny,GH}}$  is set to 99.5%.

### 3.2. The Verification Process in $RANSAC^{ini}$

Suppose in the first round, the number of the HAV iterations used in  $RANSAC^{tiny,GH}$  is  $k_1$  and the confidence level,  $Pro^{tiny,GH}$ , reaches the specified threshold  $T_{Pro^{tiny,GH}}$ . In the proposed COOSAC mechanism, once  $RANSAC^{ini}$  receives the estimated model solution of  $RANSAC^{tiny,GH}$ , namely H, it views H as the hypothesis of  $C^{ini}$  directly. Then, in the verification stage of  $RANSAC^{ini}$ ,  $RANSAC^{ini}$  utilizes the hypothesis H to calculate the first inlier rate of the initial correspondence set  $C^{ini}$ , denoted by  $IR^{ini,current}$ . The confidence level of  $RANSAC^{ini}$  can be expressed as

$$Pro^{ini} = 1 - (1 - (IR^{ini,current})^4)^{k_1} \tag{6}$$

However, after performing  $k_1$  HAV iterations on  $C^{tiny,GH}$ , the resultant model solution of  $RANSAC^{tiny,GH}$ , might be trapped into a local optimization. In order to jump out of the trap, in the second round, we create a new tiny correspondence set from the reduced correspondence set  $C^{reduced,GH}$  again. By the same argument, the HAV process of  $RANSAC^{tiny,GH}$  is repeated  $k_2$  times until its confidence level reaches the specified confidence level. Then, the newly resultant model solution of  $RANSAC^{tiny,GH}$  is utilized by  $RANSAC^{ini}$  to calculate the new inlier rate of  $C^{ini}$ , denoted by  $IR^{ini,new}$ . If  $IR^{ini,current} < IR^{ini,new}$ , the value of  $IR^{ini,current}$  is replaced by that of  $IR^{ini,new}$ ; otherwise, the value of  $IR^{ini,current}$  is retained. After completing the second round, the confidence level of  $RANSAC^{ini}$  can be expressed as

$$Pro^{ini} = 1 - (1 - (IR^{ini,current})^4)^{k_1+k_2} \tag{7}$$

Following the same argument, suppose the cooperative HAV process between  $RANSAC^{tiny,GH}$  and  $RANSAC^{ini}$  is repeated  $m$  rounds. That is,  $RANSAC^{tiny,GH}$  is constructed  $m$  times and totally,  $k_1, k_2, \dots,$  and  $k_m$  iterations are required to obtain the  $m$  estimated model solutions of the randomly constructed  $m$  tiny correspondence sets.  $RANSAC^{ini}$  utilizes the received  $m$  estimated model solutions to obtain  $m$  inlier rates of  $C^{ini}$ , and finally, it selects the model solution with the highest inlier rate, namely  $IR^{ini,current}$ , as the resultant model solution. Consequently, the final confidence level of  $RANSAC^{ini}$  can be expressed as

$$Pro^{ini} = 1 - (1 - (IR^{ini,current})^4)^{k_1+k_2+\dots+k_m} \tag{8}$$

At this time, the value of  $Pro^{ini}$  in Equation (8) achieves the specified confidence level, e.g., 99.5%.

### 3.3. Computational Complexity Analysis and Benefit

From the above discussion, in the proposed GH-COOSAC method, we know that overall,  $RANSAC^{ini}$  receives  $m$  different model solutions from  $RANSAC^{tiny,GH}$ , and  $RANSAC^{ini}$  takes  $O(m * |C^{ini}|)$  time to complete all verification processes on  $C^{ini}$ . In addition,  $RANSAC^{tiny,GH}$  takes  $O(m * |C^{tiny,GH}|)$  time to construct  $m$  tiny correspondence sets from

the reduced correspondence set  $C^{reduced,GH}$ , and it takes  $O((k_1 + k_2 + \dots + k_m) * |C^{tiny,GH}|)$  time to obtain these  $m$  model solutions to be used by  $RANSAC^{ini}$ . Totally, the proposed GH-COOSAC method takes  $O(m * |C^{ini}|) + O(m * |C^{tiny,GH}|) + O((k_1 + k_2 + \dots + k_m) * |C^{tiny,GH}|)$  time to complete the whole image matching task. Because  $|C^{tiny,GH}| \ll |C^{ini}|$ ,  $O(m * |C^{ini}|) + O(m * |C^{tiny,GH}|)$  can be simplified to  $O(m * |C^{ini}|)$ . Consequently, we have the following result.

**Proposition 2.** *The proposed GH-COOSAC method takes  $O(m * |C^{ini}|) + O((k_1 + k_2 + \dots + k_m) * |C^{tiny,GH}|)$  time to complete the image matching task.*

Under the confidence level, i.e., 99.5%, Proposition 2 indicates that the proposed GH-COOSAC method takes  $O(m * |C^{ini}|) + O((k_1 + k_2 + \dots + k_m) * |C^{tiny,GH}|)$  time for the image matching task. Because the inlier rate of  $C^{ini}$  is less than the inlier rate of  $C^{tiny,GH}$ , the traditional RANSAC method  $RANSAC^{ini}$  need take at least  $O((k_1 + k_2 + \dots + k_m) * |C^{ini}|)$  time to reach the same confidence level, 99.5%, for the image matching task. Because practically, the two conditions, namely  $(m \ll k_1 + k_2 + \dots + k_m)$  and  $(|C^{tiny,GH}| \ll |C^{ini}|)$ , hold, it yields

$$O(m * |C^{ini}|) + O((k_1 + k_2 + \dots + k_m) * |C^{tiny,GH}|) < O((k_1 + k_2 + \dots + k_m) * |C^{ini}|) \quad (9)$$

Equation (9) implies the following result.

**Proposition 3.** *Under the same confidence level, the computational cost of the proposed GH-COOSAC method is lower than the RANSAC method.*

In the next section, thorough experimental data will be used to justify the low computational cost and high matching accuracy benefits of the proposed GH-COOSAC method relative to RANSAC [2]. In Section 4.3, a discussion will be provided to explain why under the inlier rate interval [0.5, 0.9], the computational cost of the proposed method is lower than the state-of-the-art methods in [7,9,11,16].

#### 4. Experimental Results

Three public remote sensing correspondence datasets, namely “Airport”, “Small Village”, and “University Campus” which are downloaded from the website: <https://www.sensefly.com/education/datasets/> (accessed on 6 April 2022), used to evaluate the execution time and accuracy performance of the proposed GH-COOSAC method, RANSAC, and the four state-of-the-art methods, namely VFC [7], RFM [9], LP-RANSAC [11], and GLOF [16]. To measure the average accuracy performance of each considered image matching method, three accuracy metrics, namely precision, recall, and F1-score, are used.

For fairness, all comparative methods and the proposed method have been implemented on a computer with an Intel Core i7-4790 CPU 3.6 GHz and 24 GB RAM. The operating system is the Microsoft Windows 10 64-bit operating system. The program development environment is Visual Studio 2019. The execution time (in seconds) required in each considered method is measured by running the execution code of that method. The C++ source code of RANSAC [2] can be accessed from OpenCV; the C++ source code of VFC [7] can be accessed from the website <https://github.com/jiayi-ma/VFC> (accessed on 6 April 2022). The Matlab source codes of the RFM method [9] (see website <https://github.com/StaRain/RFM-SCAN>) (accessed on 6 April 2022) and the GLOF method [16] (see website <https://github.com/gwang-cv/GLOF> (accessed on 6 April 2022)) are transformed to the C++ source codes. In the LP-RANSAC method [11], the Python source code of the LP method [10] accessed from the website [https://github.com/jiayi-ma/LPM\\_Python](https://github.com/jiayi-ma/LPM_Python) (accessed on 6 April 2022) has been transformed to the C++ source code. The C++ source code of our GH-COOSAC method can be accessed from the website <https://github.com/YCccTseng/GH-COOSAC> (accessed on 6 April 2022). In the next subsection, the configuration of the proposed method is described.

#### 4.1. The Parameter Setting of the Proposed GH-COOSAC Method

For each dataset, five testing image-pairs are selected to compare the performance for the considered methods. Overall, fifteen testing image-pairs as well as their initial correspondence sets are used and they can be accessed from the website: <https://github.com/YCccTseng/GH-COOSAC> (accessed on 6 April 2022). Each initial correspondence set  $C^{ini}$ , in the next paragraph, we explain how to create the nine different correspondence sets, each set containing a ground truth inlier sets, for the nine inlier rates, 0.1, 0.2, 0.3, ..., and 0.9, respectively. For example, for the inlier rate 0.1, the rate of the created ground truth inlier set over the created correspondence set equals 0.1.

In one initial correspondence set  $C^{ini}$ , we first randomly sample four correspondences  $10^8$  times, and then  $10^8$  inlier sets are produced by using the traditional RANSAC method. In the  $10^8$  inlier sets, we select the inlier set with the maximal inlier rate, say  $R_{inlier}$ , as the initial ground truth inlier set. Based on the specified inlier rate  $r_{inlier} \in \{0.1, 0.2, \dots, 0.9\}$ , if  $r_{inlier} > R_{inlier}$ , we discard the outliers from the initial ground truth inlier set gradually until the specified inlier rate is reached; otherwise, if  $r_{inlier} < R_{inlier}$ , we discard the inliers from the initial ground truth inlier set gradually until the specified inlier rate is reached, producing the ground truth inlier set with the specified inlier rate.

In our implementation, the size of the tiny correspondence set  $C^{tiny, GH}$  follows the constraint:  $|C^{tiny, GH}| = 0.2 * |C^{reduced, GH}|$ . The specified confidence level is set to 99.5% to both  $RANSAC^{tiny, GH}$  and  $RANSAC^{ini}$ . As described in Section 3.2, according to the specified confidence level, the value of  $m$  and the values of  $k_1, k_2, \dots, k_m$  are automatically determined by the proposed GH-COOSAC method. For simplicity, the proposed GH-COOSAC method is also called “Ours”.

#### 4.2. Statistical Accuracy Comparison

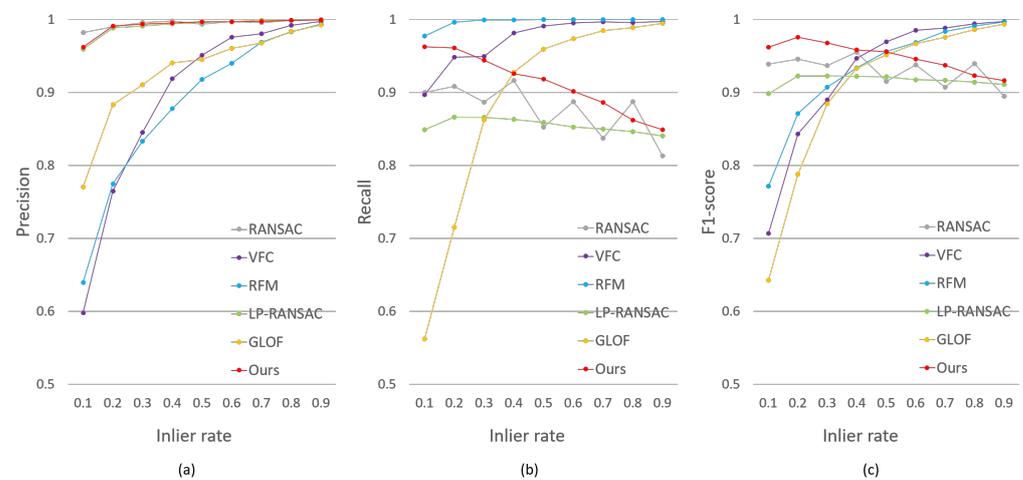
To compare the accuracy performance among the considered methods, the above-mentioned three metrics are used. Let true positive (TP), true negative (TN), false positive (FP), and false negative (FN) be defined by the number of correspondences that are correctly verified as inliers, that are correctly verified as outliers, that are erroneously verified as inliers, and that are erroneously verified as outliers, respectively. The three accuracy metrics, namely precision, recall, and F1-score are defined by

$$\begin{aligned} precision &= \frac{TP}{TP + FP} \\ recall &= \frac{TP}{TP + FN} \\ F1 - score &= \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \end{aligned} \quad (10)$$

In Equation (10), the value of “precision” indicates the rate of the number of correspondences that are correctly verified as inliers over the sum of the number of correspondences that are correctly verified as inliers and the number of correspondences that are erroneously verified as inliers. The value of “recall” indicates the rate of the number of correspondences that are correctly verified as inliers over the sum of the number of correspondences that are correctly verified as inliers and the number of correspondences that are erroneously verified as outliers. F1-score indicates the precision-recall tradeoff, and the value of “F1-score” can be viewed as an overall matching accuracy performance.

Based on the “Airport” dataset, Figure 6a–c illustrate the precision, recall, and F1-score performance comparison of the six considered methods, respectively. From Figure 6a, we observe that the proposed GH-COOSAC method, RANSAC, and LP-RANSAC have clearly better precision performance relative to RFM, VFC, and GLOF. From Figure 6b,c, under low inlier rate cases, the proposed method has good recall and F1-score performance, but under middle inlier cases, the proposed method just achieves fair recall and F1-score performance.

Figure 6b,c indicate that under high inlier cases, VFC, RFM, and GLOF have clearly better recall and F1-score performance.



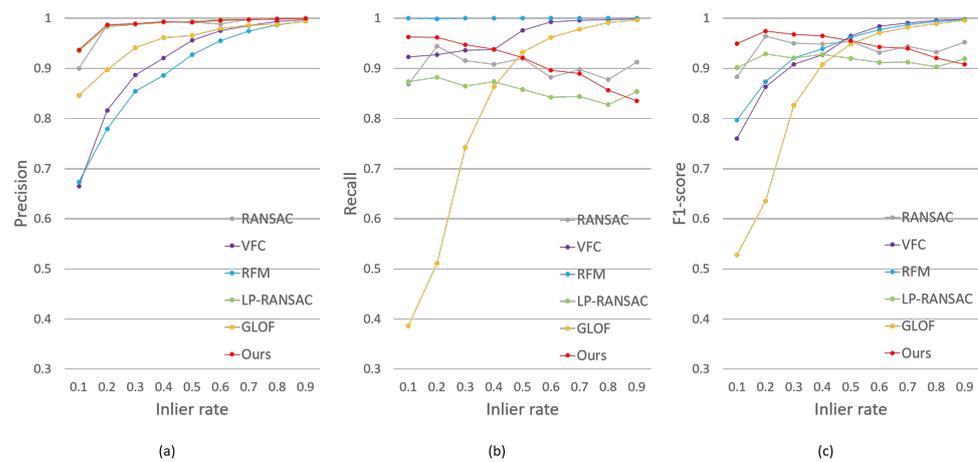
**Figure 6.** Matching accuracy performance comparison of the considered six methods for the “Airport” dataset. (a) For precision. (b) For recall. (c) For F1-score.

As mentioned in Equation (10), the metric “F1-score” is used to measure the precision-recall tradeoff of each considered image matching method. To provide the average quantitative F1-score performance of each method, three inlier rate intervals, namely [0.1, 0.5], [0.5, 0.9], and [0.1, 0.9], are considered. Based on the “Airport” dataset, Table 1 illustrates the average F1-score performance comparison for the three inlier rate intervals. From Table 1, we observe that in the interval [0.1, 0.5], the proposed method has the best F1-score performance, namely 0.964 in boldface. In the interval [0.5, 0.9], VFC has the best F1-score performance, namely 0.987 in boldface, but RFM and GLOF are quite competitive with VFC. In the complete interval [0.1, 0.9], the proposed method has the best F1-score performance, namely 0.949 in boldface.

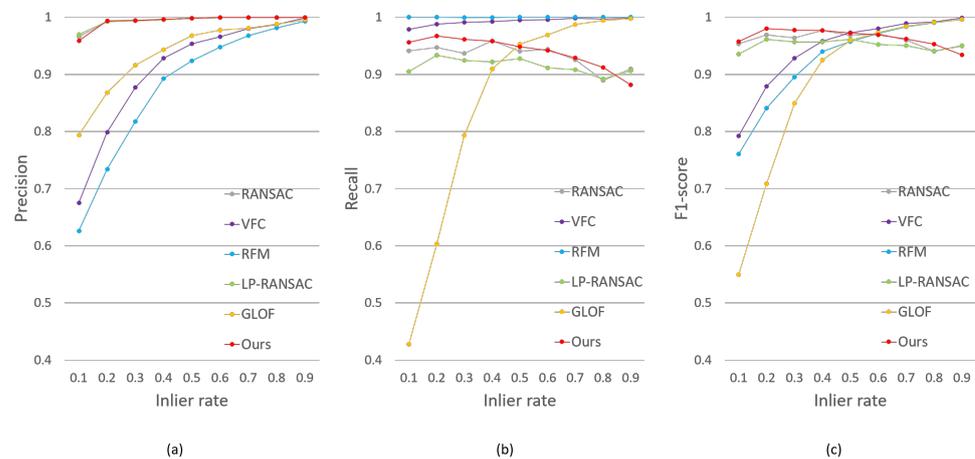
**Table 1.** Average F1-score comparison for the “Airport” dataset.

Inlier Rate Interval	RANSAC [2]	VFC [7]	RFM [9]	LP-RANSAC [11]	GLOF [16]	Ours
[0.1, 0.5]	0.939	0.871	0.888	0.918	0.840	<b>0.964</b>
[0.5, 0.9]	0.919	<b>0.987</b>	0.979	0.917	0.975	0.936
[0.1, 0.9]	0.930	0.925	0.931	0.917	0.903	<b>0.949</b>

Based on the “Small Village” and “University Campus” datasets, Figures 7a–c and 8a–c demonstrate the precision, recall, and F1-score performance comparison of the six considered methods, respectively. Tables 2 and 3 tabulate the average quantitative F1-score performance of each method for the three inlier rate intervals, namely [0.1, 0.5], [0.5, 0.9], and [0.1, 0.9] for the “Small Village” dataset and the “University Campus” dataset, respectively. It comes to the same conclusion that in the interval [0.1, 0.5], the proposed method has the best F1-score performance in boldface. In the interval [0.5, 0.9], VFC has the best F1-score performance in boldface, but RFM and GLOF are quite competitive with VFC. In the complete interval [0.1, 0.9], the proposed method has the best F1-score performance in boldface.



**Figure 7.** Matching accuracy performance comparison of the considered six methods for the “Small Village” dataset. (a) For precision. (b) For recall. (c) For F1-score.



**Figure 8.** Matching accuracy performance comparison of the considered six methods for the “University Campus” dataset. (a) For precision. (b) For recall. (c) For F1-score.

**Table 2.** Average F1-score comparison for the “Small Village” dataset.

Inlier Rate Interval	RANSAC [2]	VFC [7]	RFM [9]	LP-RANSAC [11]	GLOF [16]	Ours
[0.1, 0.5]	0.940	0.885	0.898	0.920	0.769	<b>0.962</b>
[0.5, 0.9]	0.943	<b>0.987</b>	0.983	0.913	0.977	0.933
[0.1, 0.9]	0.940	0.932	0.939	0.916	0.865	<b>0.947</b>

**Table 3.** Average F1-score comparison for the “University Campus” dataset.

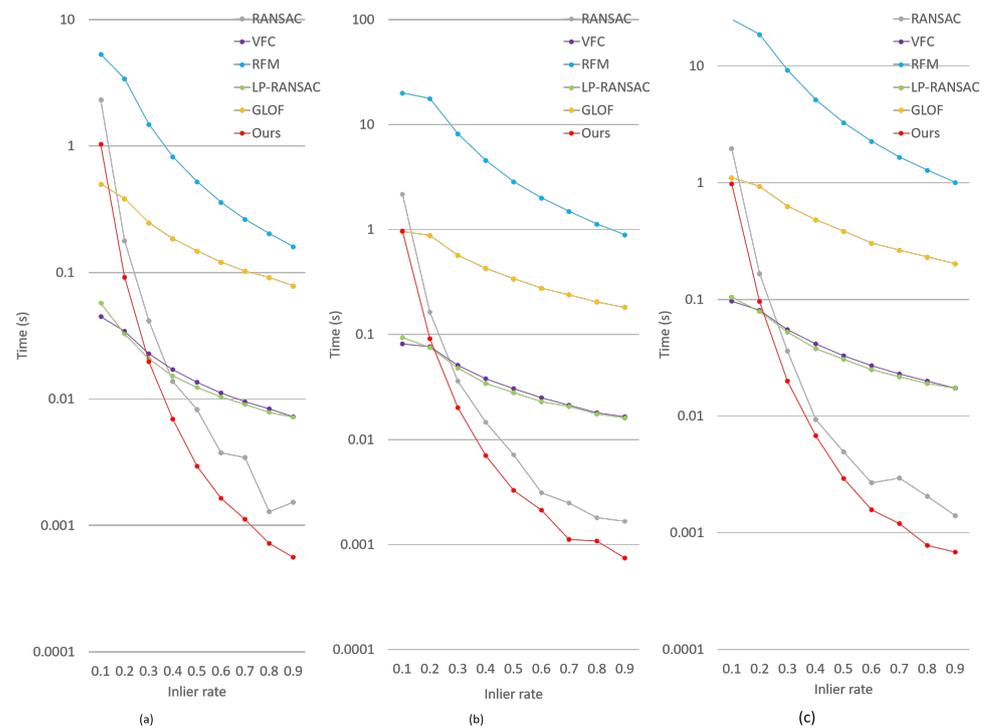
Inlier Rate Interval	RANSAC [2]	VFC [7]	RFM [9]	LP-RANSAC [11]	GLOF [16]	Ours
[0.1, 0.5]	0.966	0.906	0.879	0.954	0.798	<b>0.973</b>
[0.5, 0.9]	0.958	<b>0.986</b>	0.980	0.951	0.981	0.958
[0.1, 0.9]	0.961	0.943	0.926	0.952	0.882	<b>0.965</b>

#### 4.3. Execution Time Comparison Furthermore, Discussion

Based on the three testing datasets, the average execution time (in seconds) required in each considered method is illustrated in Figure 9, where the  $x$ -axis denotes the ground truth inlier rate and the  $y$ -axis denotes the required average execution time.

### 4.3.1. Execution Time Comparison

Tables 4–6 tabulate the average execution time performance for the “Airport” dataset, the “Small Village” dataset, and the “University Campus” dataset, respectively. From Tables 4–6, we observe that under the two inlier rate intervals, namely [0.1, 0.5] and [0.1, 0.9], VFC and LP-RANSAC require the least execution time, and the proposed GH-COOSAC method is ranked third. Under the inlier rate interval [0.5, 0.9], the proposed method requires the least execution time and is ranked first.



**Figure 9.** Execution time performance comparison of the considered six methods. (a) For “Airport”. (b) For “Small Village”. (c) For “University Campus”.

### 4.3.2. Discussion

Proposition 3 indicates that the proposed GH-COOSAC takes less computational cost relative to RANSAC [2]. No matter what the inlier rate interval is, Tables 4–6 always justified the low computational cost merit of the proposed GH-COOSAC method relative to RANSAC.

Equation (5) implies that to reach the specified confidence level, 99.5 %, for high inlier rate cases, the probability of the event, namely  $IR^{tiny,GH,current} < IR^{tiny,GH,new}$ , is often higher than that for low inlier rate cases in each round. On the other hand, under higher inlier rate cases, the proposed GH-COOSAC method takes less iterations in each round than that under low inlier rate cases. This can explain why in Tables 4–6, under the inlier rate interval [0.5, 0.9], the proposed method achieves a clearly low computational cost benefit.

In summary, Tables 1–6 indicate that the proposed GH-COOSAC method achieves low computational cost and high matching accuracy benefits when compared with RANSAC [2] and the four state-of-the-art methods [7,9,11,16].

**Table 4.** Average execution time (in seconds) comparison for the “Airport” dataset.

Inlier Rate Interval	RANSAC [2]	VFC [7]	RFM [9]	LP-RANSAC [11]	GLOF [16]	Ours
[0.1, 0.5]	0.515	<b>0.027</b>	2.320	0.028	0.294	0.232
[0.5, 0.9]	0.004	0.010	0.303	0.009	0.108	<b>0.001</b>
[0.1, 0.9]	0.287	<b>0.019</b>	1.398	<b>0.019</b>	0.207	0.129

**Table 5.** Average execution time (in seconds) comparison for the “Small Village” dataset.

Inlier Rate Interval	RANSAC [2]	VFC [7]	RFM [9]	LP-RANSAC [11]	GLOF [16]	Ours
[0.1, 0.5]	0.484	<b>0.056</b>	10.689	<b>0.056</b>	0.636	0.220
[0.5, 0.9]	0.003	0.022	1.681	0.021	0.248	<b>0.002</b>
[0.1, 0.9]	0.270	<b>0.040</b>	6.554	<b>0.040</b>	0.454	0.123

**Table 6.** Average execution time (in seconds) comparison for the “University Campus” dataset.

Inlier Rate Interval	RANSAC [2]	VFC [7]	RFM [9]	LP-RANSAC [11]	GLOF [16]	Ours
[0.1, 0.5]	0.438	<b>0.061</b>	12.339	<b>0.061</b>	0.708	0.222
[0.5, 0.9]	0.003	0.024	1.902	0.023	0.278	<b>0.001</b>
[0.1, 0.9]	0.244	0.044	7.548	<b>0.043</b>	0.505	0.124

## 5. Conclusions

The proposed GH-COOSAC method for remote sensing image matching has been presented. In GH-COOSAC, the computational complexity analysis of the GH-based mismatch removal strategy has been provided to show the low computational cost benefit of the proposed strategy. In addition, a detailed computational complexity comparison between the proposed GH-COOSAC method and the traditional RANSAC method is given to show the computational cost superiority of the proposed method. No matter what the inlier rate interval is, experimental results always justify the low computational cost merit of the proposed method relative to RANSAC. In Section 4.3.2, the detailed discussion is provided to explain why under the inlier rate interval [0.5, 0.9], the proposed method achieves a clearly low computational cost benefit relative to all comparative methods. Generally, thorough experimental results demonstrate that the proposed GH-COOSAC method achieves low computational cost and high matching accuracy benefits relative to RANSAC [2] and the four state-of-the-art image matching methods [7,9,11,16].

**Author Contributions:** Conceptualization, K.-L.C. and Y.-C.T.; methodology, K.-L.C. and Y.-C.T.; software, Y.-C.T. and H.-Y.C.; validation, K.-L.C. and Y.-C.T.; formal analysis, K.-L.C. and Y.-C.T.; investigation, K.-L.C., Y.-C.T., and H.-Y.C.; resources, Y.-C.T. and H.-Y.C.; data curation, Y.-C.T. and H.-Y.C.; writing—original draft preparation, K.-L.C.; writing—review and editing, K.-L.C. and Y.-C.T.; visualization, Y.-C.T.; supervision, K.-L.C.; project administration, K.-L.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the contract MOST-111-2221-E-011-126-MY3 of the Ministry of Science and Technology, Taiwan.

**Data Availability Statement:** Airport data were obtained from the senseFly platform (<https://www.sensefly.com/education/datasets/?dataset=1419>, accessed on 6 April 2022), Small village data were obtained from the senseFly platform (<https://www.sensefly.com/education/datasets/?dataset=1419>, accessed on 23 March 2022), University Campus data were obtained from the senseFly (<https://www.sensefly.com/education/datasets/?dataset=1419>, accessed on 6 April 2022).

**Acknowledgments:** The authors appreciate the proofreading help of C. Harrington to improve the manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
2. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 381–395. [[CrossRef](#)]
3. Capel, D. An effective bail-out test for RANSAC consensus scoring. In Proceedings of the British Machine Vision Conference 2005, Oxford, UK, 5–8 September 2005; pp. 629–638.
4. Chum, O.; Matas, J. Matching with PROSAC—Progressive sample consensus. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 220–226. [[CrossRef](#)]
5. Chum, O.; Matas, J. Optimal Randomized RANSAC. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 1472–1482. [[CrossRef](#)] [[PubMed](#)]
6. Raguram, R.; Chum, O.; Pollefeys, M.; Matas, J.; Frahm, J.M. USAC: A Universal Framework for Random Sample Consensus. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2022–2038. [[CrossRef](#)] [[PubMed](#)]
7. Ma, J.; Zhao, J.; Tian, J.; Yuille, A.L.; Tu, Z. Robust Point Matching via Vector Field Consensus. *IEEE Trans. Image Process.* **2014**, *23*, 1706–1721. [[CrossRef](#)] [[PubMed](#)]
8. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96), Portland, OR, USA, 2–4 August 1996; pp. 226–231.
9. Jiang, X.; Ma, J.; Jiang, J.; Guo, X. Robust Feature Matching Using Spatial Clustering with Heavy Outliers. *IEEE Trans. Image Process.* **2020**, *29*, 736–746. [[CrossRef](#)] [[PubMed](#)]
10. Ma, J.; Zhao, J.; Jiang, J.; Zhou, H.; Guo, X. Locality Preserving Matching. *Int. J. Comput. Vis.* **2019**, *127*, 512–531. [[CrossRef](#)]
11. Wang, G.; Sun, X.; Shang, Y.; Wang, Z.; Shi, Z.; Yu, Q. Two-View Geometry Estimation Using RANSAC With Locality Preserving Constraint. *IEEE Access* **2020**, *8*, 7267–7279. [[CrossRef](#)]
12. Lin, W.Y.; Wang, F.; Cheng, M.M.; Yeung, S.K.; Torr, P.H.; Do, M.N.; Lu, J. CODE: Coherence Based Decision Boundaries for Feature Correspondence. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 34–47. [[CrossRef](#)] [[PubMed](#)]
13. Ma, J.; Jiang, J.; Zhou, H.; Zhao, J.; Guo, X. Guided Locality Preserving Feature Matching for Remote Sensing Image Registration. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 4435–4447. [[CrossRef](#)]
14. Yang, C.; Zhang, M.; Zhang, Z.; Wei, L.; Chen, R.; Zhou, H. Non-rigid point set registration via global and local constraints. *Multimed. Tools Appl.* **2018**, *77*, 1–19. [[CrossRef](#)]
15. Liu, Y.; Li, Y.; Dai, L.; Lai, T.; Yang, C.; Wei, L.; Chen, R. Motion Consistency-Based Correspondence Growing for Remote Sensing Image Matching. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [[CrossRef](#)]
16. Wang, G.; Chen, Y. Robust feature matching using guided local outlier factor. *Pattern Recognit.* **2021**, *117*, 107986. [[CrossRef](#)]
17. Datta, B.N. *Numerical Linear Algebra and Applications*; Brooks/Cole: Seattle, WA, USA, 1995; pp. 315–324.