

Research Article

A Novel Improved Probability-Guided RANSAC Algorithm for Robot 3D Map Building

Songmin Jia,^{1,2,3} Ke Wang,^{1,2,3} Xiuzhi Li,^{1,2,3} and Tao Xu^{1,2,3,4}

¹College of Electronic and Control Engineering, Beijing University of Technology, Beijing 100124, China

²Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing 100124, China

³Engineering Research Center of Digital Community, Ministry of Education, Beijing 100124, China

⁴School of Mechanical and Electrical Engineering, Henan Institute of Science and Technology, Xinxiang 453003, China

Correspondence should be addressed to Songmin Jia; jsm@bjut.edu.cn

Received 4 August 2015; Revised 19 February 2016; Accepted 16 March 2016

Academic Editor: Maan E. El Najjar

Copyright © 2016 Songmin Jia et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a novel improved RANSAC algorithm based on probability and DS evidence theory to deal with the robust pose estimation in robot 3D map building. In this proposed RANSAC algorithm, a parameter model is estimated by using a random sampling test set. Based on this estimated model, all points are tested to evaluate the fitness of current parameter model and their probabilities are updated by using a total probability formula during the iterations. The maximum size of inlier set containing the test point is taken into account to get a more reliable evaluation for test points by using DS evidence theory. Furthermore, the theories of forgetting are utilized to filter out the unstable inliers and improve the stability of the proposed algorithm. In order to boost a high performance, an inverse mapping sampling strategy is adopted based on the updated probabilities of points. Both the simulations and real experimental results demonstrate the feasibility and effectiveness of the proposed algorithm.

1. Introduction

RANSAC algorithm is one of the popular methods for sensor data registration and modeling. In some vision-based SLAM (Simultaneous Localization and Mapping) algorithms, RANSAC algorithm provides an efficient solution for image matching procedure and establishes the data association among different views [1–3]. There are two typical types of algorithm for image matching, the dense way [4, 5] and sparse way [6, 7]. In dense match approach, the whole image is used for parameter estimation [8, 9]. Although this method is quite robust, it may be inaccurate when occlusion regions exist in the matching images. Sometimes, the influence of occlusions is reduced by using a robust weighted cost function. In sparse match step, image features are detected by using SIFT, SURF, or any other feature detection algorithms. The features are matched by using the distance of feature descriptor, and the matching pairs are sometimes ambiguous. To efficiently achieve a correct matching result, some robust algorithms

were adopted to remove mismatching pairs, such as M-estimation [10], LMedS (Least Median of Squares) [11], or RANSAC (Random Sample Consensus) [12] algorithm. M-estimation established a new cost function with a robust weight. It worked well in some cases but was vulnerable to the noise. LMedS optimized the model by minimizing the median of errors. When the outlier rate was larger than 50%, M-estimation and LMedS might be no longer applicable. With the advantages of easy implementation and strong robustness, RANSAC algorithm was widely used in model parameter estimation problem. In standard RANSAC algorithm, a hypothesis set was randomly selected to estimate a parameter model. And an inlier set was detected by testing all input data with the estimated parameter model. A maximum size of inlier set was expected to be found within a predetermined iteration. However, the performance of this standard RANSAC was sometimes low. Even worse was the fact that the solution may not be reached when all iterations were finished.

To efficiently improve the performance of standard RANSAC algorithm, some methods have been proposed in recent decades. A hypothesis evaluation function and local optimization procedure were adopted to achieve a more accurate result. MSAC (M-estimation SAC) [10] evaluated the test point set with a bounded loss function to achieve a maximum likelihood consensus set. MLESAC (Maximum Likelihood SAC) [10] evaluated the hypothesis set by using the probability distribution of errors. The inlier error was modeled with an unbiased Gaussian distribution and outlier error used a uniform distribution. The maximum likelihood estimation was solved by minimizing a cost function. MAPSAC (Maximum A Posteriori Estimation SAC) [13] followed a Bayesian approach to solve the RANSAC problem with a robust MAP estimation. LO-RANSAC (Local Optimized RANSAC) [14] adopted an inner model reestimation procedure to improve the accuracy of the RANSAC algorithm.

Moreover, some heuristic mechanism sampling strategies and partial evaluation procedures were adopted to speed up the convergence of the algorithm. It seemed that a good sample strategy will reduce the time cost which was spent in finding the solution. A hypothesis set was selected based on the probabilities of test points in the Guided MLESAC [15]. In PROSAC (Progressive SAC) [16], the matching score was used as a prior knowledge for sorting the test data. A hypothesis set was selected among the data which was in the top-ranked matching score. It was also progressively tested on the less ranked data. In some extreme cases, the whole data would be tried in this algorithm. According to the assumption that an inlier tends to be closer to the inliers, NAPSAC (N Adjacent Points SAC) [17] sampled the data within a defined radius around a selected point. Based on the preliminary test of the hypothesis, Chum and Matas proposed R-RANSAC (Randomized RANSAC) [18] and R-RANSAC SPPR (Sequential Probability Ratio Test) [19]. These two methods performed a preliminary test based on $T_{d,d}$ and SPPR test after evaluating test points in every iteration. And a full test procedure was performed only when a hypothesis set passed the preliminary test. The preliminary test procedure effectively removed the obvious mistakes of hypothesis sets and improved the efficiency of RANSAC algorithm. Optimal RANSAC [20] adopted an inlier sample procedure to achieve a more accurate model estimation. When the size of the current inlier set was larger than a threshold, an inlier sample procedure would be performed to achieve a more reliable solution in the inlier set.

Furthermore, some intelligent algorithms such as Genetic Algorithm (GA) and multilayered feed-forward neural networks (NFNN) were also proposed in RANSAC algorithm. Rodehorst proposed a novel RANSAC algorithm based on GA [21]. In GASAC, the parents were generated by a standard RANSAC algorithm with a robust cost evaluation. Then, the best solution was achieved by using crossover and mutation operators on parents in GA procedure. Moumen presented a rather comprehensive study of robust supervised training of MFNN in a RANSAC framework from the standpoint of both accuracy and time [22]. In the iteration of RANSAC, the parameter model was estimated by using a small MFNN which was minimizing the mean squared error (MSE) with

a standard back propagation algorithm. All inlier points were used to reestimate a new parameter model by training a new MFNN. And a new hypothesis set was achieved by using this new MFNN model. The convergence solution was achieved until the inlier set did not change any more.

In this paper, we propose a novel improved probability-guided RANSAC (IPGSAC) algorithm for mobile robot 3D map building. Under the framework of standard RANSAC algorithm, two types of probabilities are evaluated for test points by using a total probability formula and the statistics of maximum size of inlier set. To achieve a more robust evaluation of test points, DS evidence theory [23] is adopted to synthesize the multisource evaluation of test points. Moreover, the theories of forgetting are employed to filter out the unstable inliers. Based on the probability of test points, an inverse mapping sampling strategy is utilized to improve the convergence rate of the proposed algorithm. Finally, this proposed IPGSAC algorithm is applied for the mobile robot 3D map building. All the experimental results show the feasibility and effectiveness of the proposed algorithm.

The rest of this paper is organized as follows: in Section 2, we summarize IPGSAC before explaining each in detail. The components of our robot map building procedure are detailed in Section 3. Our simulation and real experimental results are described in Section 4. Finally, we give our conclusions and future work in Section 5.

2. IPGSAC Algorithm

2.1. Methodology Overview. The proposed IPGSAC algorithm is illustrated in Figure 1. At the beginning of IPGSAC algorithm, the probabilities of test points are initialized with a hybrid distribution. Based on those probabilities, n points are selected for model estimation by employing an inverse mapping sampling strategy. The inlier and outlier sets are distinguished with a tolerance threshold r_0 . When the residual error r_i of point p_i is larger than r_0 , the p_i is identified as an outlier point. Then, two types of probabilities are evaluated for test points by using a total probability formula and the statistics of maximum size of inlier set. According to the average observations of inlier points, the theories of forgetting are employed to reduce the redundancy of unstable inliers. To achieve a more reliable evaluation, the probability evaluations are synthesized by using DS evidence theory. When the maximum iteration limit is arrived at or the stopping criterion is reached, the main loop of IPGSAC will be finished. Finally, we reestimate the model parameter by using all inliers to achieve a more reliable inlier set with 3~5 iterations. In the standard RANSAC, the minimum number of iterations k_{\max} ensures that a correct hypothesis set is achieved with a determined confidence level at least once and it can be estimated by

$$k_{\max} = \frac{\log(1-z)}{\log(1-w^n)}, \quad (1)$$

where z is the confidence level, w indicates the inlier rate, and n indicates the minimum number of test points for model estimation. In the proposed IPGSAC algorithm,

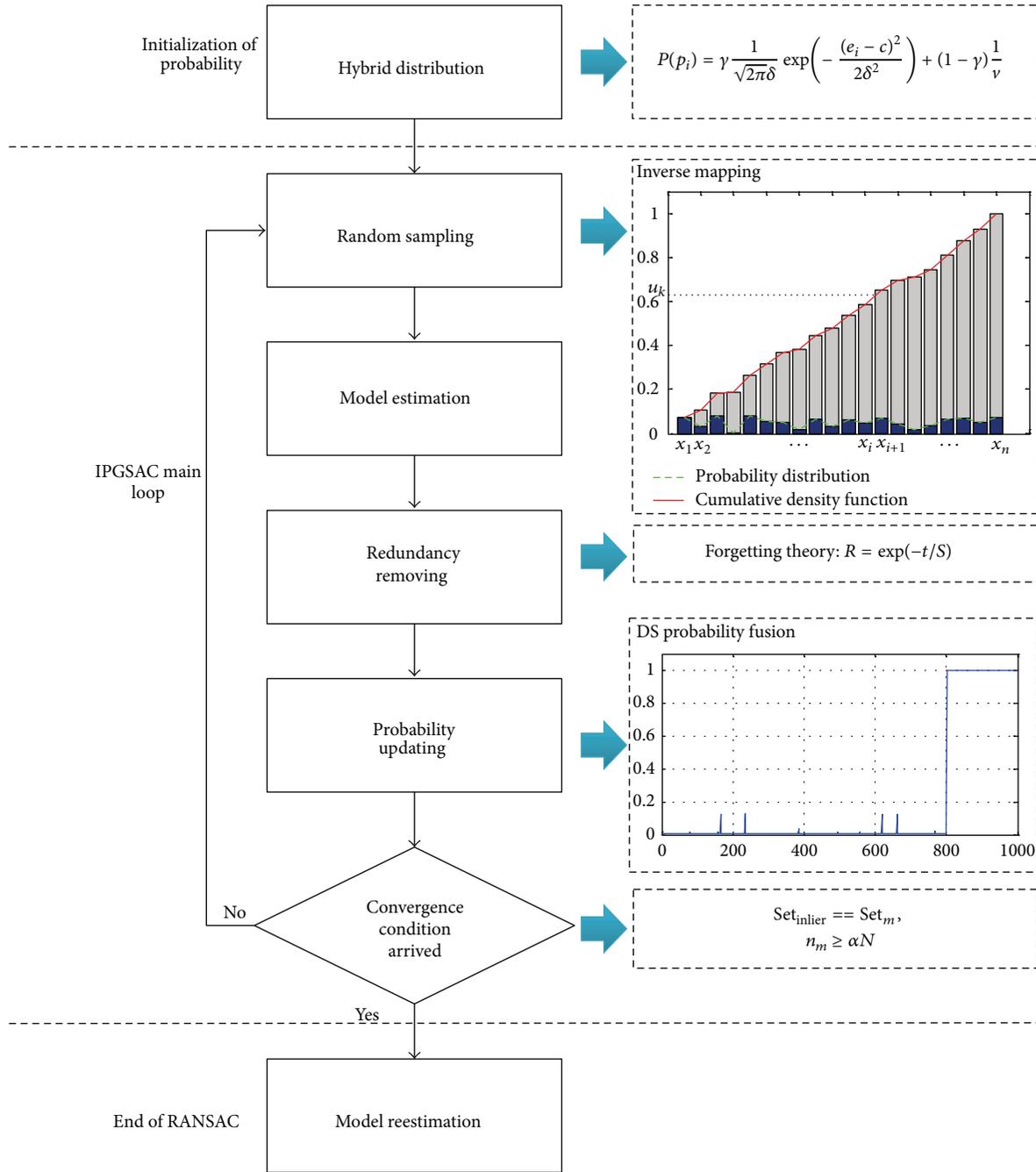


FIGURE 1: The illustration of the proposed IPGSAC algorithm.

the maximum number of iterations is limited by (1) with the confidence level 98%.

In this proposed algorithm, a maximum inlier set is expected to be found when the algorithm is converged. To avoid falling into local optimum, a pseudo inlier rate is used to limit the minimum size of inlier set. The stopping criterion is described as follows:

$$\begin{aligned} \text{Set}_{\text{inlier}} &== \text{Set}_m, \\ n_{\text{inlier}} &\geq \alpha N, \end{aligned} \tag{2}$$

where $\text{Set}_{\text{inlier}}$ is the inlier set which is acquired in the current iteration, n_{inlier} is the number of points in $\text{Set}_{\text{inlier}}$, Set_m indicates the maximum inlier set so far, and α is the pseudo minimum inlier rate. N is the total number of test points. Operator “==” means that two sets are very similar. The detailed description of each component of the proposed algorithm is given in the following sections.

2.2. Initialization of Probabilities. Inspired by MLESAC algorithm, we assume that the inlier data is satisfied to a normal

distribution and the outlier data is fitted to a uniform distribution. The hybrid distribution is described as follows:

$$P(p_i) = \gamma \frac{1}{\sqrt{2\pi}\delta} \exp\left(-\frac{(e_i - c)^2}{2\delta^2}\right) + (1 - \gamma) \frac{1}{\nu}, \quad (3)$$

where e_i is the residual error of p_i with the model estimated by using all data, c is the expectation of inlier errors, δ is the variance of e_i , ν determines the probability of outlier point, and γ indicates whether p_i is an inlier or not. When p_i is an inlier point, then $\gamma = 1$; otherwise $\gamma = 0$. To evaluate the distribution, the length of error interval L is calculated with the minimum and maximum of error. The mean and variance of error are used as an initial estimation of c and δ , respectively. By limiting the length of searching interval with the maximum value of gL and 2δ , c and δ are solved by using a standard normal distribution with several iterations. In this proposed algorithm, $\nu = N$ and $g = 0.1$.

2.3. Evaluation Inlier Probabilities. In standard RANSAC algorithm, the parameter model is estimated by using a random sample test set. According to this estimated model, the inlier and outlier set are determined in this iteration. While this procedure should be aimless, it needs a large number of iterations to find an optimal solution. This paper aims to provide an effective and reliable inlier evaluation method and speed up the convergence of RANSAC algorithm. In the BaySAC [24], Botterill et al. assumed that the probabilities of test points were independence in the same hypothesis set. Firstly, n points were selected to estimate a parameter model s . Then, all input points were tested to find an inlier set. The probabilities of inlier points were updated by using (4) and did not change the outlier's probabilities:

$$\begin{aligned} P(p_i \in M | s \notin M) &= \frac{P(p_i \in M) P(s \notin M | p_i \in M)}{P(s \notin M)} \\ &= \frac{P(p_i \in M) (1 - P(s \subseteq M | p_i \in M))}{1 - P(s \subseteq M)} \\ &= \frac{P(p_i \in M) (1 - P(s \subseteq M) / P(p_i \in M))}{1 - P(s \subseteq M)} \\ &= \frac{P(p_i \in M) - P(s \subseteq M)}{1 - P(s \subseteq M)}, \end{aligned} \quad (4)$$

where M is the consensus set that contains all inlier points and all correct models and p_i indicates a test point. $P(p_i \in M)$ evaluates the probability of a point being an inlier. $P(p_i \in M | s \notin M)$ describes the probability of a point being an inlier, under the condition that the estimated model s is not a subset of M . The BaySAC is poor when the probabilities of test points are the same. With the proposed initialization of probability, it is sometimes difficult to achieve the convergence of BaySAC. In order to improve the convergence of the proposed algorithm, a multisource evaluation strategy is employed to achieve a higher confident probability analysis of test points.

First of all, a full probability update procedure is used to achieve a more reliable and comprehensive evaluation of test points. The total probability update procedure is expressed as

$$\begin{aligned} P(p_i \in M) &= P(p_i \in M | s \notin M) P(s \notin M) \\ &\quad + P(p_i \in M | s \subseteq M) P(s \subseteq M), \end{aligned} \quad (5)$$

where $P(p_i \in M | s \notin M)$ is calculated by using (4) and $P(p_i \in M | s \subseteq M)$ indicates the probability of p_i being an inlier, under the condition that the estimated model s is a subset of M :

$$P(p_i \in M | s \subseteq M) = 1 - \left| \frac{r_i}{r_0} \right|, \quad (6)$$

where r_i is the residual error of point p_i with the current estimated model. $P(s \subseteq M)$ evaluates the probability of the estimated model being a correct model. As the number of the inliers illustrates the correctness of the estimated model, $P(s \subseteq M)$ is estimated by using the proportion of inlier in this iteration:

$$\begin{aligned} P(s \subseteq M) &= \frac{n_{\text{inlier}}}{N}, \\ P(s \notin M) &= \frac{n_{\text{outlier}}}{N}, \end{aligned} \quad (7)$$

where n_{inlier} and n_{outlier} are the number of inlier and outlier points, respectively. $N = n_{\text{inlier}} + n_{\text{outlier}}$ is the total number of test points. Moreover, n_{inlier} not only indicates the correctness of current model estimation, but also evaluates the possibility that the test point is an inlier. The statistics of maximum size of inlier set which contains the test points is employed to reevaluate the probabilities of test points in the iteration. The evaluation of inlier and outlier points are defined as

$$\begin{aligned} P_m(p_i \in M) &= \frac{n_i}{\tilde{n}}, \\ P_m(p_i \notin M) &= 1 - \frac{n_i}{\tilde{n}}, \end{aligned} \quad (8)$$

where n_i is the maximum size of inlier set which contains the test point p_i and $\tilde{n} = \sum_{i=1}^N P(p_i \in M)$ indicates the current predicted size of inlier set.

2.4. Probability Fusion with DS Evidence Theory. During the iteration of IPGSAC algorithm, the probabilities of test points are updated by using our evaluation criteria. As it is a type of uncertain information, the probabilities indicate the possibility that a point belongs to an inlier set. Generally, the information is characterized as inherently uncertain, imprecise, and incomplete. The representation and reasoning of uncertainty information are one of the important research issues in artificial intelligence (AI). Many researchers devote themselves to modeling the representation and reasoning of the uncertain information. According to the inherently uncertain characteristics of information, a more reasonable and accurate evaluation is achieved by synthesizing some different sources of information with some specified rules.

DS evidence theory, proposed by Dempster and Shafer, is one of the popular methods for the multisource information fusion. The evidence theory describes the uncertainty of information by using the discernment framework, the basic probability belief functions, and likelihood functions [23]. Suppose that there is a problem written as x and the discernment set Θ contains all the possible solutions of x . All the elements are mutually independent in the discernment set Θ . The solutions of x only take from Θ in any case. Θ is defined as a discernment framework and written as $\{\theta_1, \theta_2, \dots, \theta_i, \dots, \theta_n\}$. θ_i is a really defined focal element of discernment framework Θ ; n is the number of elements in Θ . The power set of a discernment framework 2^Θ contains all subsets of the discernment framework. There are 2^n elements in 2^Θ and each of the elements corresponds to a solution for x . The power set 2^Θ is composed of the following subsets:

- (1) It contains the empty set ϕ and all of the elements in Θ ;
- (2) If subset $A, B \in 2^\Theta$, then $A \cup B \subseteq 2^\Theta$;
- (3) Excepting (1) and (2), it no longer contains any other subsets.

In DS evidence theory, a probability belief function is adopted to describe the uncertainty of evidence. This probability belief function is defined as $m : 2^\Theta \rightarrow [0, 1]$, such that

$$\begin{aligned} \sum_{A \in \Theta} m(A) &= 1, \\ m(\phi) &= 0, \end{aligned} \quad (9)$$

where ϕ is an empty set, m is a probability belief function in 2^Θ , and $m(A)$ describes the confidence probability of solution A .

Generally, the probability belief function is used to evaluate the probability of a reasoning that endows the supposing of the solutions. The final judgment is generally influenced by many factors. According to the DS evidence theory, a synthesis method is illustrated to fuse two or more basic probability belief functions by using a sum of orthogonal operations. Assume that m_1 and m_2 are the corresponding probability belief functions which are mutually independent in the same discernment set Θ . For the focal elements A_1, A_2 and B_1, B_2 , the synthesis operation is described as

$$\begin{aligned} K &= \sum_{A_i \cap B_j = \phi} m_1(A_i) m_2(B_j) \\ m(A) &= \begin{cases} \frac{\sum_{A_i \cap B_j = A} m_1(A_i) m_2(B_j)}{1 - K} & \forall A \subset \Theta, A \neq \phi \\ 0 & A = \phi, \end{cases} \end{aligned} \quad (10)$$

where K is conflict coefficient and $1/(1 - K)$ is regularization factor.

In this proposed IPGSAC algorithm, the status of test samples is assigned as inlier(A) or outlier(B) and the discernment framework Θ is $\{A, B\}$. The probability brief function

is written as $[m(A_i), m(B_i)]$ for each test point. According to Section 2.3, the probability brief function is defined as

$$\begin{aligned} m_1(A) &= P(p_i \in M), \\ m_1(B) &= P(p_i \notin M), \\ m_2(A) &= P_m(p_i \in M), \\ m_2(B) &= P_m(p_i \notin M). \end{aligned} \quad (11)$$

2.5. Probability Filtering with Theories of Forgetting. For some low constrained model estimation problem, there exist a few of outlier points which easily satisfy the current estimated model but are not stable. During the DS fusion procedure, this incorrect judgment will have a great influence on the convergence of the proposed algorithm. Even worse is the fact that the sample probability of inlier is finally much lower than the outlier and the algorithm is hardly converged. To further improve the proposed algorithm, the theories of forgetting are used to filter out the unstable inliers and achieve a more stable solution.

According to theories of forgetting, the forgetting curve can be roughly described as

$$R = \exp\left(-\frac{t}{S}\right), \quad (12)$$

where R is memory retention, S is the relative strength of memory, and t describes the time. As our proposed algorithm is an online learning procedure, we use (13) to describe the forgetting and reviewing procedure:

$$R = \exp(o - 1), \quad (13)$$

where o indicates the average observations when the inlier set is similar to current maximum set. And the maximum inlier size of p_i is refreshed by $n_i = Rn_i$.

Figure 2 shows the estimated parameters for the fundamental matrix estimation when the outlier rate is 80%. Figure 2(a) demonstrates a typical online learning procedure with 50 iterations. During the iteration, the probabilities of test points are updated by using a total probability formula. After 30 iterations, a convergent solution is achieved with a stable sample probability of inlier and outlier. In Figure 2(b), the maximum inlier size of point is adopted to reinforce the probabilities of test points by using proposed DS fusion procedure. Due to the incorrect judgment of test points, the probability of outlier is waved between 0 and 0.8. And the sample probability of inlier is decreased after several iterations. Although a correct inlier set is achieved in the 3rd iteration, it becomes more and more difficult to reach the convergence of the algorithm. Figure 2(c) illustrates the result of our proposed algorithm which is guided by proposed probability update procedure. After a correct hypothesis set is found, the probabilities of test points obviously distinguished that the probability of inlier is near to 1 and outlier is near to 0. The influence of the incorrect judgment is filtered by using the theories of forgetting. After a few iterations, the curve of the predicted inlier size is much smoother than illustrated in

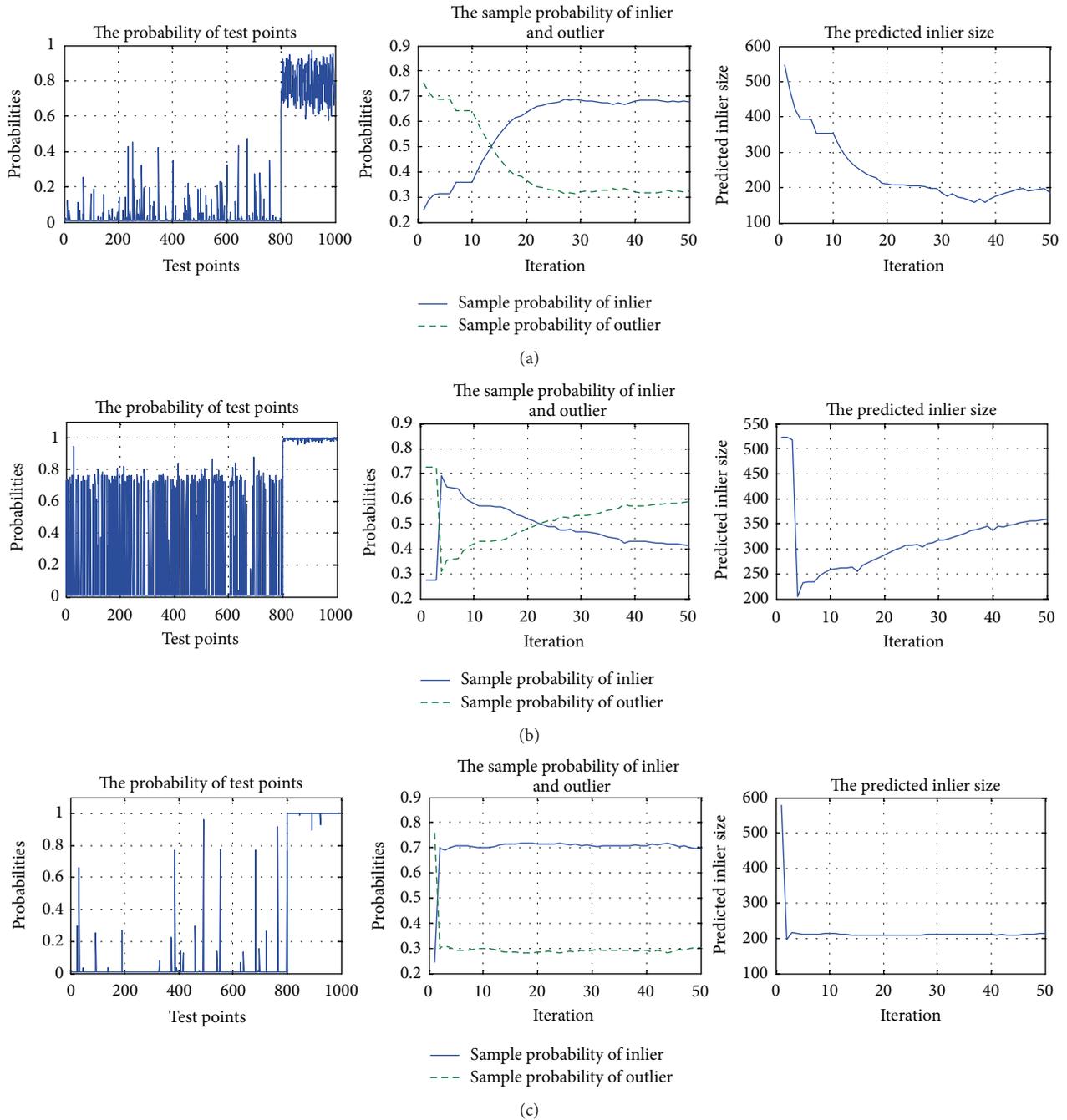


FIGURE 2: The estimated parameters during the fundamental estimation simulation. (a) shows the results by using the total probability formula, (b) illustrates the result with DS evidence theory, and (c) demonstrates the result of the procedure of the proposed algorithm.

Figures 2(a) and 2(b) and it is highly consistent with the presetting of the simulation. The results demonstrate the high performance and reliability of the proposed algorithm.

2.6. Inverse Mapping Sampling Strategy. During the procedure of RANSAC, a good sampling strategy is an important guarantee of convergence of algorithm. In IPGSAC, the probabilities of test points are evaluated and merged by using DS evidence theory. To speed up the convergence rate of the proposed algorithm, an inverse mapping sampling strategy

[25, 26] is utilized for the hypothesis set selection according to the updated probability of test points. The principle of this sampling procedure is as follows: assume that the probability distribution function of a continuous random variable X is written as $F(x)$. $F^{-1}(u)$ is an inverse function of $F(x)$ where $u \in [0, 1]$. $F^{-1}(u)$ and $F(x)$ have the same probability distribution function and the probability distribution is expressed as

$$P(X \leq x) = P(F^{-1}(u) \leq x) = P(U \leq F(x)). \quad (14)$$

For a discrete random variable X , a cumulative probability distribution function is expressed as

$$P(X \leq x) = \sum_{x_i \leq x} P(x_i). \quad (15)$$

In the inverse mapping sampling strategy, $[0, 1]$ is divided into n subsections which is corresponding to sample points. The k th subsection is expressed as $[\sum_{i=1}^{k-1} P(x_i), \sum_{i=1}^k P(x_i)]$. When a uniform sampled variable $u_k \in [0, 1]$ is achieved in the i th subsection, the corresponding data of this subsection is the expected output. In our proposed algorithm, the probability of test point p_i is normalized by $w_i = P(p_i) / \sum_{i=1}^n P(p_i)$. And a sampling cumulative density function (CDF) F is defined by $F(p_i) = \sum_{k=1}^i w_k$. The expected output is p_i when the random variable u_k is in the subsection $[F(p_i), F(p_{i+1}))]$.

In our proposed IPGSAC algorithm, a hypothesis set is randomly selected by using the inverse mapping sampling strategy with the probabilities of test points. When the probabilities of test points are equal, the sampling method obeys a uniform sampling approach and it is consistent with the standard RANSAC algorithm. When those probabilities are different, the procedure will improve the sample probability of the point which has a high probability. When a correct hypothesis set is achieved, especially, the probabilities of inlier will be greatly improved with the proposed evaluation procedure. Then, the sampling probability of these inliers will be increased rapidly. Therefore, it will effectively speed up the convergence rate of the proposed algorithm.

2.7. Implementation of IPGSAC Algorithm. A general framework of the proposed IPGSAC algorithm is shown in Algorithm 1. The proposed algorithm can be divided into the initialization of probability, the main loop, and the final reestimation. At the beginning of this algorithm, some parameters should be defined before the procedure of each part. In the initialization step, the test points' probabilities are evaluated by the hybrid distribution. Then, the main loop of the proposed algorithm starts to find the solution of model estimation. In this procedure, a hypothesis set Set_{test} containing n test points is randomly selected for model estimation by using the inverse mapping sampling strategy. When the model estimation error is larger than the tolerance threshold r_0 , the model estimation fails and the probabilities of test points are updated by using (4) with $P(s < M) = 0$. When the model estimation succeeds, the probabilities of test points are evaluated by using the total probability formula. And the maximum inlier size of test point in $\text{Set}_{\text{inlier}}$ is refreshed by current inlier size n_{inlier} . To reduce the influence of incorrect judgment, a forgetting factor R is computed with the average of observation when an inlier set is similar to current maximum set. Other evaluations of test points are estimated with the maximum inlier size n_i and forgetting factor. A more reliable probability evaluation of test points is achieved by employing the DS evidence theory. Finally, the convergent constraint should be checked in every iteration. When the convergent constraint arrives, the reestimation step is performed in several times to achieve a more reliable inlier set and model estimation.

3. Robot 3D Map Building Using IGPSAC

3.1. Preliminaries Definition. Kinect is a low cost depth sensor which is widely used in 3D map building system. Assume that the image and depth captured by Kinect are precalibrated by using the method presented in [27] and the depth is prewarped into the image frame. The intrinsic parameter matrix K of RGB camera is written as

$$K = \begin{bmatrix} f_u & 0 & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix}, \quad (16)$$

where (f_u, f_v) and (c_u, c_v) are the focal and image center of RGB cameras, respectively. According to these intrinsic parameters, the projection from a 3D point to image pixel is described as

$$u = K\pi(p), \quad (17)$$

where $p = (x, y, z)$ is a 3D point in current camera frame and $\pi(p) = (x/z, y/z)$.

During the map building procedure, Kinect is fixed on a mobile robot with a consistent translational speed. We assume that the pose of robot is the same as the Kinect and the world coordinate is set at the location where robot starts to run. At the time i , robot's pose is described as

$$T_i = \begin{bmatrix} R_i & t_i \\ 0 & 1 \end{bmatrix}, \quad (18)$$

where $T_i \in \text{SE}(3)$; $\text{SE}(3)$ is a special rigid transformation group. R_i is the rotation matrix; t_i is the translation vector. According to the *Lie* algebra, T_i is expressed as

$$T_i = \exp(\tilde{\zeta}), \quad (19)$$

where $\zeta = (\alpha, \beta, \gamma, t_x, t_y, t_z)$, and $\tilde{\zeta}$ is define as

$$\tilde{\zeta} = \begin{bmatrix} 0 & \gamma & -\beta & t_x \\ -\gamma & 0 & \alpha & t_y \\ \beta & -\alpha & 0 & t_z \\ 0 & 0 & 0 & 0 \end{bmatrix}. \quad (20)$$

3.2. Pose Estimation Based on IPGSAC. In this paper, FAST corner algorithm [28, 29] is employed to detect the features and the BRIEF [30] descriptor is utilized for feature matching. With the image matching pairs, a derivative ICP [31] algorithm is adopted based on the projection errors. Generally, there exist some invalid depth measurement regions in the captured depth images. Suppose that all matching pairs mentioned in this section have the valid depth measurement. The pose estimation model is described as

$$E = \arg \min_{T_k} \frac{1}{N} \sum_{\Omega} \|r_i(\xi)\|, \quad (21)$$

```

(1) Parameter setting: input data  $Data$ ,  $\alpha = 0.05$ ,
    minimum model estimation data size  $n$ , tolerance threshold  $r_0$ ,
    setup Model function  $modelFunc$ ,  $modelInlier$ ,
    reset maximum inlier size  $p_i$ ,  $n_i = n$ ;
(2) Estimate the maximum iterations  $k_{max}$ ;
(3) Initialize the probabilities of test points by using a hybrid distribution;
(4) for  $i = 1$  to  $k_m$  do
(5)    $Set_{test} = inverse\_sampling(Data, n)$ ;
(6)    $(error, s) = modelFunc(Set_{test})$ ;
(7)   if  $error > r_0$ 
(8)     Update probabilities of points in  $Set_{test}$ ,  $P(p_i \in M) = P(p_i \in M | s \notin M)$ ;
(9)   else
(10)     $(Set_{inlier}, r, n_{inlier}) = modelInlier(Data, s)$ ;
(11)    Update the probabilities of all test points  $P(p_i \in M)$  by using (5);
(12)    Update the maximum inlier size. For  $p_i$  in  $Set_{inlier}$ ,  $n_i = \max(n_i, n_{inlier})$ ;
(13)  end if
(14)  Compute the forgetting factor  $R$ ;
(15)  Compute  $P_m(p_i \in M) = R(p_i)n_i/\bar{n}$ ;
(16)  Update the probability  $P(p_i \in M)$  using DS evidence theory;
      
$$K = P(p_i \in M)P_m(p_i \notin M) + P(p_i \notin M)P_m(p_i \in M)$$

      
$$P(p_i \in M) = \frac{P(p_i \in M)P_m(p_i \in M)}{1 - K}$$

(17)  if  $n_{inlier} > \alpha N$  and  $Set_m == Set_{inlier}$ 
(18)    break;
(19)  end if
(20)  if  $n_{inlier} > maxInlierSize$ 
(21)    Update  $Set_m$  with  $Set_{inlier}$ ;
(22)     $maxInlierSize = n_{inlier}$ ;
(23)  end if
(24) end for
(25) for  $i = 1$  to  $k$  do
(26)   $(error, s) = modelFunc(Set_m)$ ;
(27)   $(Set_m, r, n_i) = modelInlier(Data, s)$ ;
(28) end for

```

ALGORITHM 1: IPG SAC algorithm.

where r_i is the projection error and defined as $r_i(\xi) = K\pi(T(\xi)p_i) - x_i$; p_i is a point in the reference point cloud which is corresponding to the new captured feature x_i . Ω indicates a set of image matching pairs; N is the size of Ω .

To achieve the optimization of (21), the projection error r_i is approximated by first-order Taylor expansion

$$r_i = r_i(0) + J\xi, \quad (22)$$

where $J = (\partial r_i / \partial \xi)|_{\xi=0}$.

According to Euler-Lagrange formula, the optimal solution of (21) is solved by using a Newton iteration step with

$$\begin{aligned} \frac{\partial E}{\partial \xi} &= \frac{1}{N} \sum_{\Omega} J^T (r_i(0) + J\xi), \\ \xi &= -\sum_{\Omega} (J^T J)^{-1} \sum_{\Omega} J r_i(0). \end{aligned} \quad (23)$$

Due to the ambiguity of image matching procedure, the proposed IPGSAC algorithm is utilized for removing the outliers (mismatching pairs) and achieving an accurate pose estimation. The detail of this pose estimation procedure is similar to Algorithm 1.

3.3. Keyframe Selection with Maximum Matching Distance.

With the accurate estimated poses, a 3D Map is built with the captured depth and color data. Considering that there are about 300,000 points in every frame, it is necessary to create a more concise representation of the 3D map. A keyframe selection mechanism is employed with maximum matching distance in the algorithm. The map building procedure will start only when a new keyframe is captured. However, the accuracy of pose estimation is largely dependent on the overlap of the input data. Therefore, a distance threshold d_0 and an angle threshold θ_0 are employed for the keyframes selection. In the procedure of keyframe selection, if the pose estimation between current frame I_{i+1} and keyframe I_{kf}^{k-1} fails or the distance d or the angle θ between them is larger than the thresholds, the frame I_i will be chosen as a new keyframe. The principle of keyframe selection is described as follows:

$$I_{kf}^k = \{I_i \mid icp(I_{kf}^{k-1}, I_i) > 0, icp(I_{kf}^{k-1}, I_{i+1}) \leq 0\}, \quad (24)$$

where $icp(I_1, I_2)$ evaluates the procedure of pose estimation and $icp = 0$ when pose estimation fails or $d \geq d_0$ or $\theta \geq \theta_0$; for others, $icp = 1$.

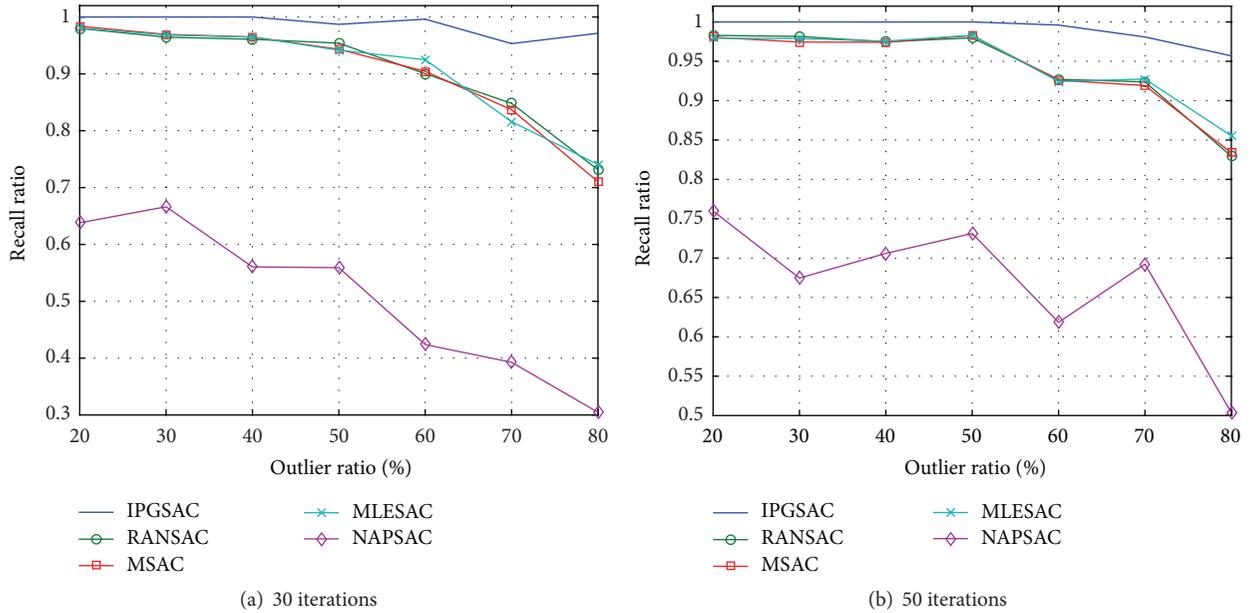


FIGURE 3: The recall ratios with 30 and 50 iterations in line fitting simulation.

3.4. Loop Detection and Optimization. During the map building, the quality of global map is largely dependent on the accuracy of the pose estimation. However, a drift arises with the accumulation of the pose estimation error during the 3D map building experiment. To effectively achieve the global optimization of the 3D map, a vision-based loop detecting method is used to check the loop closure when a new keyframe is captured. During the loop detecting procedure, the relative poses between current keyframe and previous nonadjacent keyframes are estimated by using the pose estimation algorithm. A successful estimation of relative pose means that a loop closure is detected and the optimization algorithm will be performed immediately. In this paper, TORO (tree-based network optimizer) [32, 33] algorithm is employed to achieve a global optimization of the generated map. In TORO algorithm, a tree structure is constructed by using the robot's poses and relative poses among keyframes. The error function which evaluates the observed and calculated relative pose is minimized to optimize all the estimated poses and achieve a global consistent 3D map. The optimization function is described as

$$T_{kf} = \arg \min_{T_{kf}} F(T_{kf}),$$

$$F(T_{kf}) = \sum_{(i,j) \in G} e(T_{kf}^i, T_{kf}^j, T_{ij})^T \Lambda e(T_{kf}^i, T_{kf}^j, T_{ij}), \quad (25)$$

where $T_{kf} = \{T_{kf}^1, \dots, T_{kf}^i, \dots, T_{kf}^n\}$ is the vector descriptor of robot's poses, T_{ij} is an observed relative pose between i th keyframe and j th keyframe, Λ represents, respectively, the mean and the information matrix of a constraint relating the poses, and G is a relative tree. $e(T_{kf}^i, T_{kf}^j, T_{ij})$ is an error function that measures how well the poses satisfy the observed constraint T_{ij} .

4. Experimental Results

4.1. Simulation Results

4.1.1. Line Fitting Simulation. The principle of line fitting is as follows: assume that there exists a point set $\{(x_1, y_1), (x_2, y_2), \dots, (x_{n-1}, y_{n-1}), (x_n, y_n)\}$, the fitting line is expressed as $ax + by + c = 0$, and a, b, c satisfy

$$\sum_{i=0}^n \begin{bmatrix} x_i^2 & x_i y_i & x_i \\ x_i y_i & y_i^2 & y_i \\ x_i & y_i & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = 0. \quad (26)$$

We generated a random simulation data set which fitted with a given line. The standard deviation of inlier points was set to 1, and the outlier points obeying a uniform distribution in the minimum rectangular region contained the all inliers. The size of test points was set to 500. The outlier rates were set to 20%, 30%, 40%, 50%, 60%, 70%, and 80%, respectively. During the simulation, 2 points were used to estimate the line function. And α, r_0 were set to 0.05 and 2, respectively. We performed the RANSAC algorithm with fixed iterations and recorded the recall ratio in different outlier rates.

Figure 3 shows the average recall ratios of different RANSAC algorithms by recording 50 repeat experiments of the line fitting. In Figure 3, the recall ratio of the proposed IPGSAC algorithm is the largest in those five algorithms and it is nearly 100% when the outlier rate is lower than 60%. With the increasing of outlier rate, the recall ratios of other four RANSAC algorithms are rapidly decreased and the proposed algorithm still keeps the highest recall ratio with a small reduction. Table 1 illustrates the statistics of iterations when IPGSAC is convergent in 50 repeat experiments. The result illustrates that the average iteration of the proposed algorithm

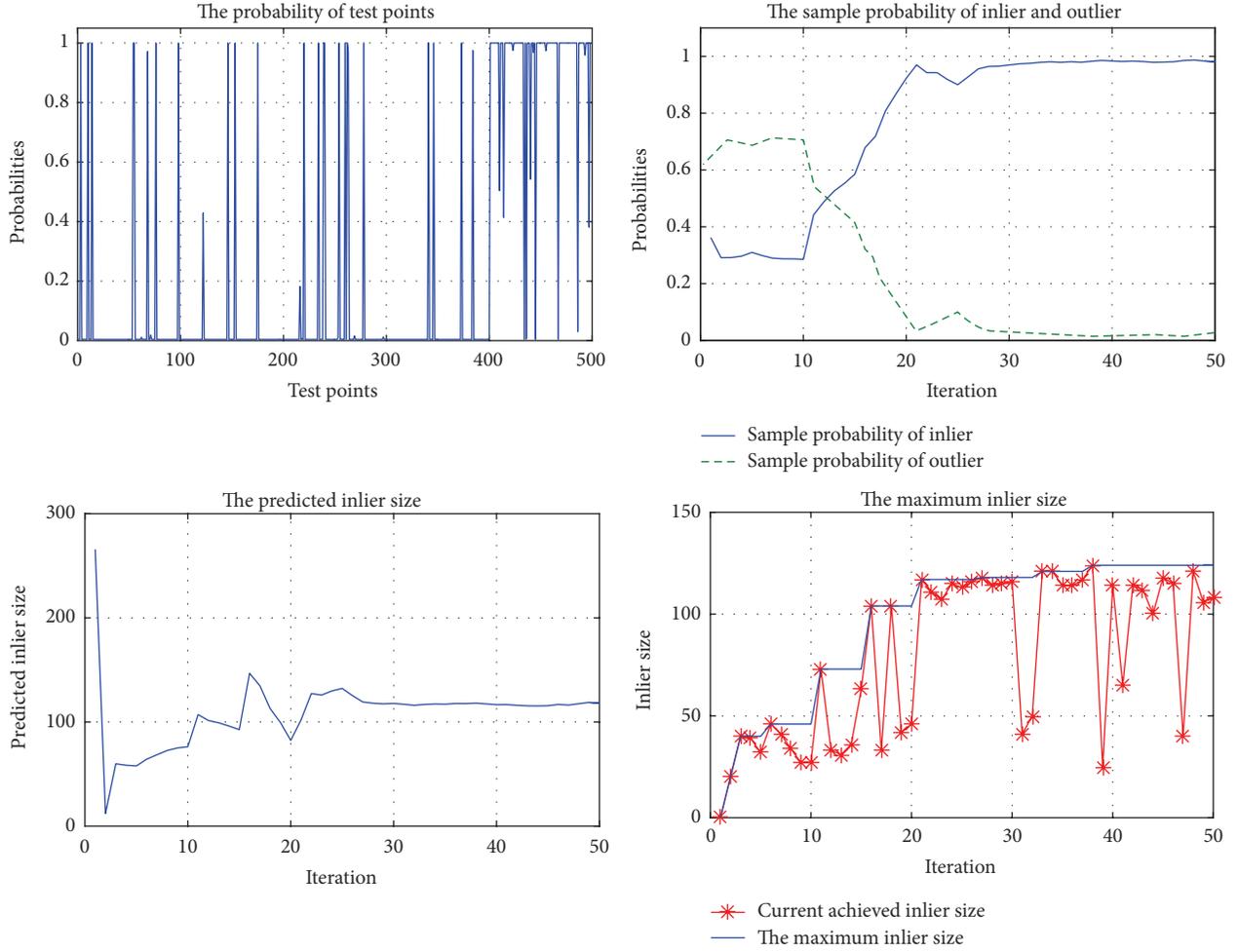


FIGURE 4: The simulation result of line fitting when the outlier rate is 80%.

TABLE 1: The statistics of iterations when IPGSAC algorithm is converged in line fitting.

Outlier rate	20%	30%	40%	50%	60%	70%	80%
Min iterations	3	3	3	4	3	7	12
Max iterations	19	17	17	46	35	76	92
Average iterations	7.95	8.65	8.60	13.05	16.80	27.05	41.55

is much smaller than the maximum number of iterations estimated by (1).

Figure 4 shows the estimated parameters when the outlier rate is 80% within 50 iterations. At the beginning of iteration, the sample probability of outlier points was much larger than the inliers. During the iteration, the probabilities of test points were updated by using proposed evaluation rule. After 20 iterations, the probabilities of inlier points were near to 1 and outlier was near to 0. The sample probability of inlier points was increased rapidly with the increasing of the probability of inlier points. Based on the principle of inverse mapping sampling strategy, the selection of test sample set trended to

the points with high probability in our proposed algorithm. It effectively improved the convergence rate of our algorithm. In fact, the algorithm reached the stop criterion in the 21st iteration. After 26 iterations, the estimated number of inlier set was converged to determined values, and it was high consistent with the actual acquired number of inlier points. Figure 5 illustrates the line fitting results when the outliers are 70% and 80%. The line fitting simulation results show that the proposed IPGSAC algorithm can rapidly localize inlier points. The probability updating algorithm can effectively improve the selection probability of inlier point and speed up the convergence rate of the proposed algorithm.

4.1.2. Homography Estimation Simulation. Assume that $u_{\text{curr}} = \{u_0, u_1, \dots, u_n\}$ and $u_{\text{ref}} = \{u_0^{\text{ref}}, u_1^{\text{ref}}, \dots, u_n^{\text{ref}}\}$ are two correspondent image feature point sets. The homography relationship is defined as

$$au_i = Hu_i^{\text{ref}}, \quad (27)$$

where a is a scaling factor and H is a homography matrix. According to *Lie* algebra, homography matrix H can be

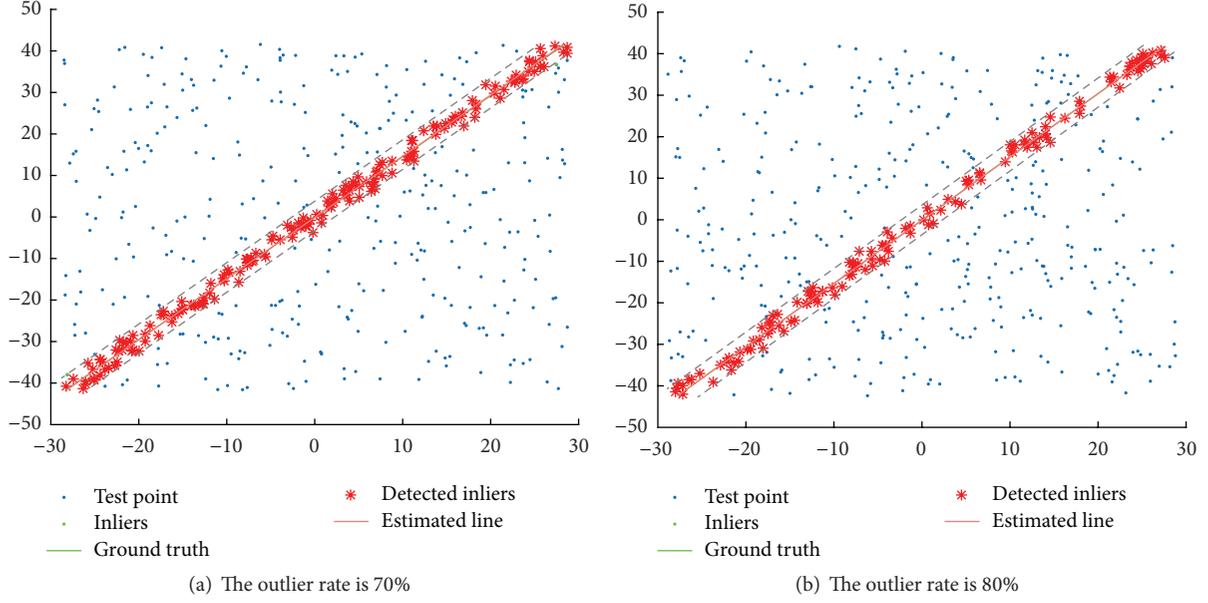


FIGURE 5: The line fitting results of the proposed algorithm when the outlier rates are 70% and 80%.

expressed as an exponent mapping by using an 8-variable $x = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$:

$$H = \exp(A(x)),$$

$$A = \begin{bmatrix} x_5 & x_3 & x_1 \\ x_4 & -x_5 - x_6 & x_2 \\ x_7 & x_8 & x_6 \end{bmatrix}. \quad (28)$$

In order to estimate the homography matrix H , an energy function is established by using the projection errors:

$$E = \sum_{i=0}^n \|r_i\|, \quad r_i = \pi(Hu_i^{\text{ref}}) - u_i, \quad (29)$$

where m ($m \geq 4$) is the number of matching pairs which are used for the homography estimation. To optimize this object function, we follow the linear expansion and Newton iteration method. The minimum of energy function (29) is achieved by solving (30) iteratively:

$$\sum_{i=1}^n \left(\frac{\partial r_i}{\partial x} \right)^T \frac{\partial r_i}{\partial x} \xi = - \sum_{i=1}^n \left(\frac{\partial r_i}{\partial x} \right)^T r_i. \quad (30)$$

A random simulation data set was generated which fitted with a given homography relationship. To enhance reliability and authenticity of our simulation, the standard deviation of inlier points was set to 1 pixel and the outlier points obeyed the uniform distribution within the image. The size of test points was set to 1,000, and the outlier rates were set to different values which were the same as in the line fitting simulation. During the simulation, we selected four points to estimate the homography. α, r_0 were set as 0.05 and 2, respectively. The simulation was repeated 50 times to achieve a more reliable result in each outlier rate.

TABLE 2: The statistics of iterations when IPGSAC is converged in homography estimation simulation.

Outlier rate	20%	30%	40%	50%	60%	70%	80%
Min iterations	3	3	3	3	4	4	3
Max iterations	10	15	14	13	21	42	99
Average iterations	4.66	5.42	5.46	5.82	8.98	14.60	26.74

Figure 6 shows the compared results of the simulation. In the simulation, the recall ratio of the proposed algorithm is much higher than other four RANSAC algorithms in any tested outlier rates. Even when the outlier is 80%, this proposed algorithm is still available with the recall ratio 0.85962 after 20 iterations. Table 2 demonstrates the statistics of iterations when IPGSAC is converged in the homography estimation simulation. These results illustrate that the proposed IPGSAC algorithm is more effective than other four RANSAC algorithms and demonstrate the high performance of the proposed algorithm.

4.1.3. Fundamental Matrix Estimation Simulation. In this paper, 8-point method is used for fundamental matrix estimation. Assume that $m = (u, v, 1)$ and $m' = (u', v', 1)$ are an image matching pair; the fundamental matrix is limited with following equation:

$$m' F m = 0, \quad (31)$$

where F is a 3×3 matrix. We follow the method in [6] to estimate F , and the error function is defined as

$$r = \frac{(m' F m)^2}{\|F^T m'\| + \|F m\|}. \quad (32)$$

The simulation data was generated similarly as in homography simulation which fitted with a given fundamental

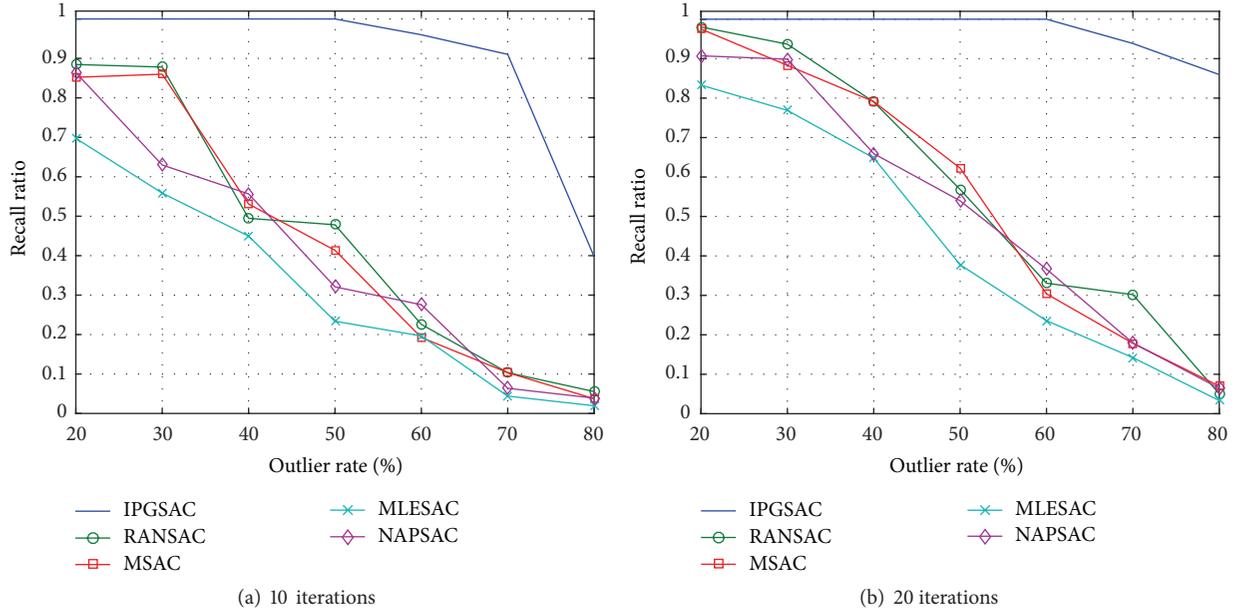


FIGURE 6: The recall ratios with 10 and 20 iterations in homography estimation simulation.

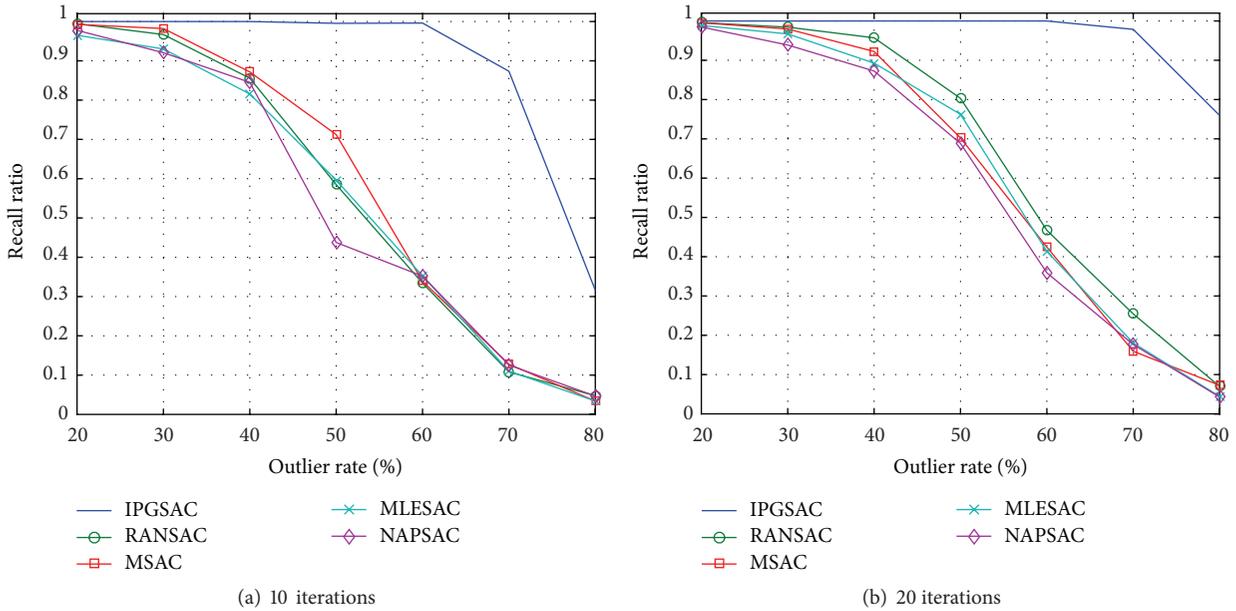


FIGURE 7: The recall ratios with 10 and 20 iterations in fundamental matrix estimation simulation.

relationship. The standard deviation of inlier points was set to 1 pixel and the outlier points obeyed the uniform distribution within the image. During the experiment, α and r_0 were set to 0.05 and 1, respectively. We also recorded the average recall ratios in 50 repeat simulations when the outlier rates were 20%, 30%, 40%, 50%, 60%, 70%, and 80%, respectively.

Figure 7 shows the recall ratios of different RANSAC algorithms in 10 and 20 iterations, respectively. Due to the low probability of inlier test set selection, the recall ratios of the other four RANSAC algorithms begin to decrease rapidly when the outlier rate is larger than 30%. As the fundamental

matrix estimation is solved with several linear equations, the constraint condition is much weaker than the homography relationship. The average recall ratios of fundamental matrix estimation are a little better than the homography estimation. Table 3 illustrates the statistics of iterations when IPGSAC is converged with 50 repeat experiments. It shows that it only needs 26.5 iterations in average to deal with the outlier rate 80%, and it is much more efficient than other four algorithms.

4.1.4. The Time Cost and Accuracy of IPGSAC. In the previous section, some results are given to illustrate the high

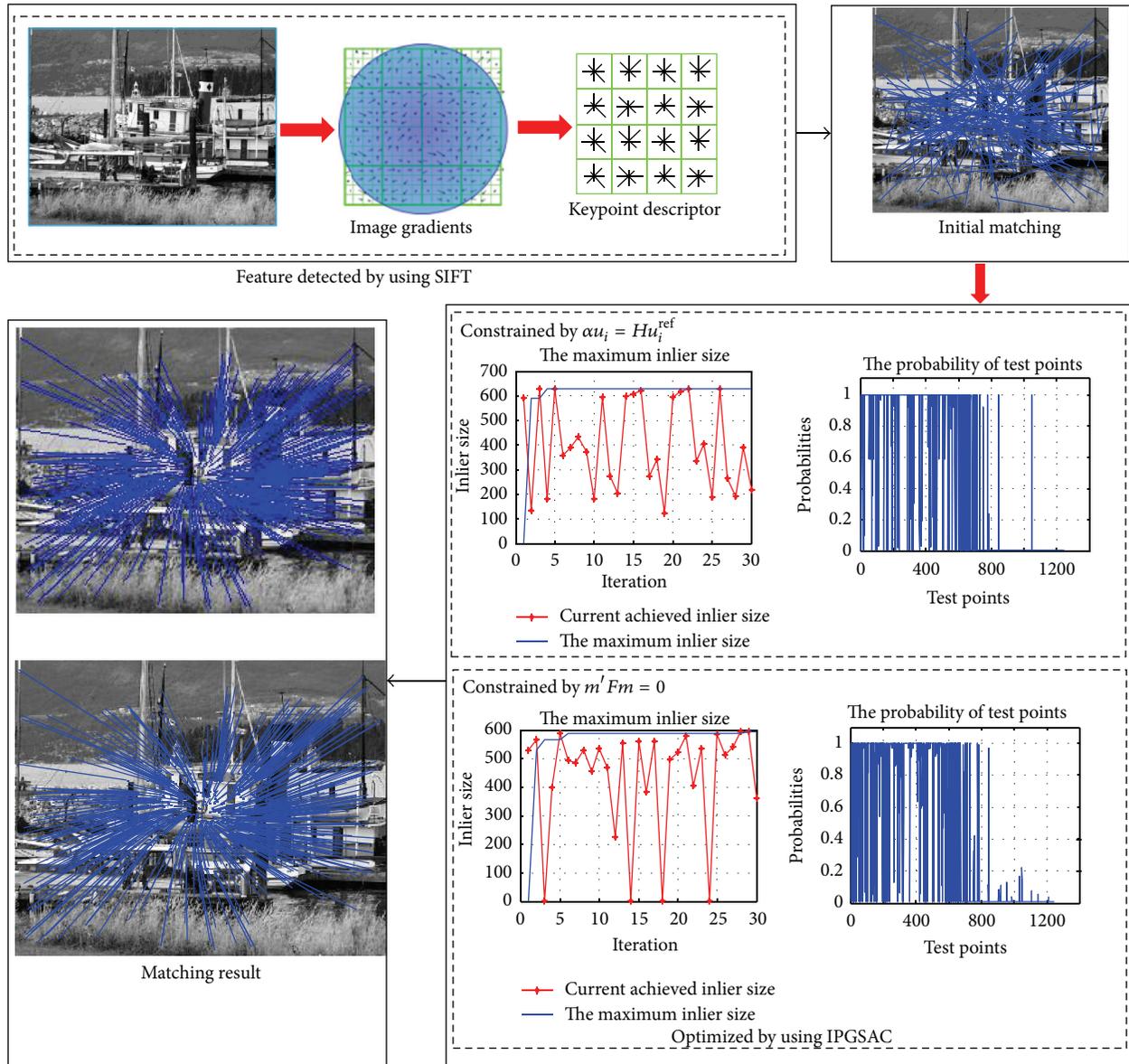


FIGURE 8: The image matching experimental results.

performance of the proposed algorithm. In some application of RANSAC, the time cost and the accuracy of the proposed algorithm are important issue for the real time requirement. As a typical application, the fundamental matrix estimation simulation was used for evaluating the time cost and the accuracy of the proposed algorithm. To ensure the convergence of each RANSAC algorithm, the maximum number of iterations was set to 100 and the other parameters were set the same as those in Section 4.1.3. All timing results were obtained on a PC with Intel i5 760 CPU (2.80 GHz) and 4 GB RAM.

The time cost, mean absolute error, and recall ratio of five RANSAC algorithms are illustrated in Tables 4–6. As shown in the tables, the recall ratios of those five RANSAC algorithms are very similar to each other when the outlier rate is lower than 40%. Generally, a lot of iterations are needed to ensure that an inlier set is found in the standard RANSAC,

MSAC, MLESAC, and NAPSAC. Therefore, the time cost of those algorithms is much higher than the proposed algorithm. The recall ratios of those algorithms are decreased rapidly with the outlier rate increasing. The time cost of the proposed algorithm is the lowest with the smallest mean absolute error and highest recall ratio during the simulation. And the time cost has a small increase with the raising of the outlier rate. The results further demonstrate the low time cost and the high accuracy of the proposed algorithm.

4.2. Real Experimental Results

4.2.1. Image Matching Experiments. During the experiment, two images taken from different perspective in a real scene [34] were used for image matching. Firstly, the SIFT algorithm was used for feature detection and matching, as shown in Figure 8. There were 1244 feature pairs which were matched

TABLE 3: The statistics of iterations when IPGSAC is converged in fundamental estimation simulation.

Outlier rate	20%	30%	40%	50%	60%	70%	80%
Min iterations	3	3	3	3	3	3	3
Max iterations	7	6	9	8	39	54	99
Average iterations	3.60	3.94	4.02	4.22	10.88	10.88	26.5

TABLE 4: The time cost of different RANSAC algorithms.

Outlier rate	20%	30%	40%	50%	60%	70%	80%
IPGSAC	0.0479	0.0493	0.0519	0.0509	0.0608	0.0628	0.0959
RANSAC	0.4378	0.4376	0.4379	0.4439	0.4695	0.4415	0.429
MSAC	0.4299	0.4295	0.4296	0.4371	0.464	0.4332	0.4285
MLESAC	0.4505	0.4488	0.4472	0.4541	0.4786	0.4504	0.4455
NAPSAC	0.4374	0.4367	0.4364	0.444	0.4766	0.4421	0.4365

TABLE 5: The mean absolute error of different RANSAC algorithms.

Outlier rate	20%	30%	40%	50%	60%	70%	80%
IPGSAC	0.2159	0.2347	0.2348	0.2672	0.2396	0.2533	0.2926
RANSAC	0.3099	0.3506	0.3685	0.4256	0.5444	0.7716	0.8717
MSAC	0.256	0.29	0.327	0.4541	0.5019	0.7787	0.7886
MLESAC	0.2581	0.2824	0.3101	0.376	0.4236	0.5553	0.5569
NAPSAC	0.2821	0.3367	0.4036	0.4955	0.5591	0.776	0.7858

TABLE 6: The recall ratio of different RANSAC algorithms.

Outlier rate	20%	30%	40%	50%	60%	70%	80%
IPGSAC	0.9988	0.9973	0.9951	0.9901	0.9763	0.9773	0.9494
RANSAC	0.9988	0.9974	0.9941	0.9813	0.904	0.5588	0.1517
MSAC	0.9988	0.9973	0.9933	0.9884	0.9293	0.6061	0.1472
MLESAC	0.9983	0.9961	0.9884	0.9704	0.877	0.5245	0.1954
NAPSAC	0.9987	0.9967	0.992	0.9689	0.8657	0.5206	0.1525

by using the distance of feature descriptor. We used 8-point method and 4-point method for the fundamental matrix and homography matrix estimation, respectively. We also replicated all the matching experiment 50 times; the average outlier rate was about 52% which was achieved by using the proposed algorithm.

During the homography estimation experiment, α and r_0 were set to 0.05 and 4, respectively. The image matching result is shown in Figure 8. The probabilities of test points have obvious boundaries and the probability of inlier points is near to 1. Table 7 illustrates the number of inlier points achieved by different RANSAC algorithms with different iterations. The experimental result is highly consistent with the simulations. This proposed IPGSAC algorithm needs no more than 20 iterations to get convergence in average.

In fundamental matrix estimation, α and r_0 were set to 0.05 and 1, respectively. The probabilities of test points are quite similar with homography for foundation estimation. Table 8 shows the number of inlier points achieved by different RANSAC algorithms with different iterations. This experimental result is also similar with the simulation of fundamental matrix estimation. This proposed IPGSAC

TABLE 7: The inlier size with different iterations in homography estimation.

Iterations	IPGSAC	RANSAC	MSAC	MLESAC	NAPSAC
10	593.46	101.7	129.48	71.40	138.26
20	595.46	187.94	242.42	118.80	261.42
30	596.46	301.68	290.10	194.66	299.56

TABLE 8: The inlier size with different iterations in fundament estimation.

Iterations	IPGSAC	RANSAC	MSAC	MLESAC	NAPSAC
10	595.36	311.16	311.38	356.28	288.34
20	610.80	386.48	389.62	462.12	401.40
30	611.30	446.10	427.58	511.04	445.52

algorithm also needs no more than 20 iterations to achieve a convergent solution in average. While the homography matrix is a plane constraint of features and the fundamental matrix is a common constraint in image matching, the result has some tiny difference with homography estimation. All

TABLE 9: The evaluation of absolute translational error (m).

		ICP-based method	Dense image-based method	Our method
Freiburg1/desk	min	0.092303	0.016672	0.007411
	max	1.078730	0.178265	0.246963
	RMSE	0.5715	0.1134	0.093726
Freiburg1/desk2	min	0.016459	0.021073	0.003560
	max	1.024154	0.427586	0.317763
	RMSE	0.368354	0.119614	0.096128



(a) Mobile robot platform



(b) Experimental environment

FIGURE 9: The platform and experimental environment.

those model estimation results demonstrate the feasibility and effectiveness of the proposed IPGSAC algorithm.

4.2.2. The 3D Map Building Experiment. The 3D mapping experiment was taken in an indoor environment shown in Figure 9. The experimental platform consisted of a Pioneer3-DX mobile robot and a Microsoft Kinect. The P3-DX mobile robot is an intelligent mobile platform equipped with an embedded PC. The Kinect provides the depth and color data captured in the experimental scene. The maximum field of view of Kinect is 52° and the image resolution is 640×480 with the refresh frequency 30 Hz. In the experiment, the Kinect was fixed on the mobile robot and captured the depth and color data in the scene. The distance threshold d_0 and angle threshold θ_0 were set to 0.2 and 0.3, respectively. The tolerance threshold r_0 was equal to 1. The pseudo minimum inlier rate α was 0.05. To localize the robot, four points were used in the IPGSAC-based pose estimation procedure.

To evaluate the proposed algorithm, we compare our method with standard ICP-based tracking algorithm and a dense image-based tracking algorithm [9] using the RGBD SLAM benchmark provided by Technical University of Munich [35]. Table 9 shows the evaluations of the absolute translational error by using those three algorithms with two typical indoor data sets. In the experiment, ICP-based tracking method almost failed with the largest evaluation of the absolute translational error. Due to the blur and occlusion regions of the input data, the dense image-based method

easily falls into local optimum. In our proposed method, the data association was established by using the feature matching procedure and it sometimes noised by the mismatched features. And the noise would be removed in our IGPSAC-based pose estimation procedure. Although the IGPSAC-based pose estimation procedure is limited by the accuracy of feature detection, the smallest RMSE (Root Mean Square Error) of the absolute translational error is achieved by using our proposed method.

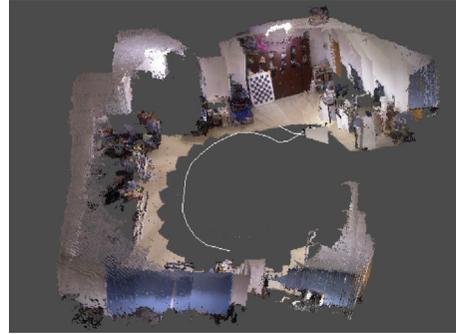
In the indoor experiment, the robot was instructed to traverse around in the environment. And the translational speed of mobile robot was about 0.1 m/s. The origin of coordinate of robot was set at the location where robot started to move. While the mobile robot was moving, the depth and color image were captured by the Kinect and the robot's pose was estimated by using the algorithm detailed in Section 3. When a keyframe was captured, the global map was updated with the new coming keyframe and the loop detection procedure was performed to estimate the relation pose between this new coming keyframe and previous nonadjacent keyframes. When a loop closure was detected, the TORO optimization algorithm would be performed to achieve a global optimization of generated map. Table 10 illustrates the number of iterations when the proposed IPGSAC is converged during the mapping experiment. According to (1), the minimum iteration is in the range of 60~7725 when the confidence level is 98%. The iteration when the proposed IPGSAC-based algorithm is converged is much littler than the estimated

TABLE 10: The statistics of iterations when IPGSAC is converged in 3D map building.

Outlier rate	0.25–0.35	0.35–0.45	0.45–0.55	0.55–0.65	0.65–0.75	075–0.85
Count	19	31	247	354	112	19
Min iterations	2	2	2	2	2	7
Max iterations	9	25	46	92	61	87
Average iterations	5.21	7.70	9.06	13.66	15.47	20.29



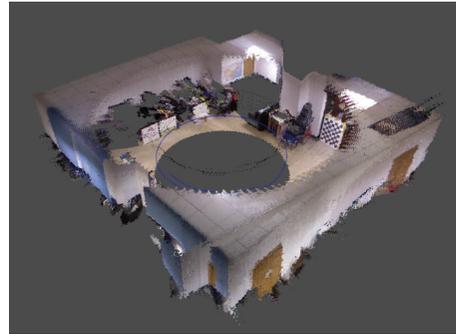
(a) ICP-based method



(b) Dense image-based method



(c) RANSAC-based method



(d) The proposed IPGSAC-based method

FIGURE 10: The generated mapping before the loop optimization.

minimum iterations. It is highly consistent with our previous simulations and verifies the high efficiency of the proposed algorithm.

Moreover, the ICP-based pose estimation, the dense image-based pose estimation, and the standard RANSAC-based pose estimation were also employed for map building. Figure 10 shows the generated mapping before the loop optimization by using those methods. During the map building, the ICP-based tracking algorithm failed due to the similar spatial structure of the input point clouds. Although the dense image-based method tracked the mobile robot well, the generated map sank seriously. In the RANSAC-based method, it was sometimes easy to converge to local optima and tracking procedure would fail as indicated by the red circle in Figure 10(c). In the IPGSAC-based pose estimation method, the pose estimation procedure worked well and the generated map had a very slightly subsidence during the experiment. With those estimated poses, the loop closure was easily detected and it was much easier to perform loop optimization procedure and achieve a global consistent map. The final mapping experimental result of our method is shown in

Figure 11. All the experimental results demonstrate the feasibility and effectiveness of this proposed algorithm.

5. Conclusions

In this paper, we present a novel RANSAC algorithm for robot 3D map building. Firstly, two types of the probabilities are updated and evaluated for test points by using a total probability formula and the maximum number of inlier points. Then, a more reliable evaluation of test points is achieved by using the DS evidence theory. To reduce the influence of incorrect judgment, the theories of forgetting are utilized to filter out the unstable inlier points. Finally, an inverse mapping sample algorithm is adopted to sample the points based on the updated probabilities of points. All simulation and real experimental results show that the proposed algorithm can quickly and efficiently achieve a stable convergence in a few iterations and demonstrate the feasibility and effectiveness of the proposed algorithm.

As is illustrated in the experiment results, the proposed algorithm is much more efficient than some other RANSAC

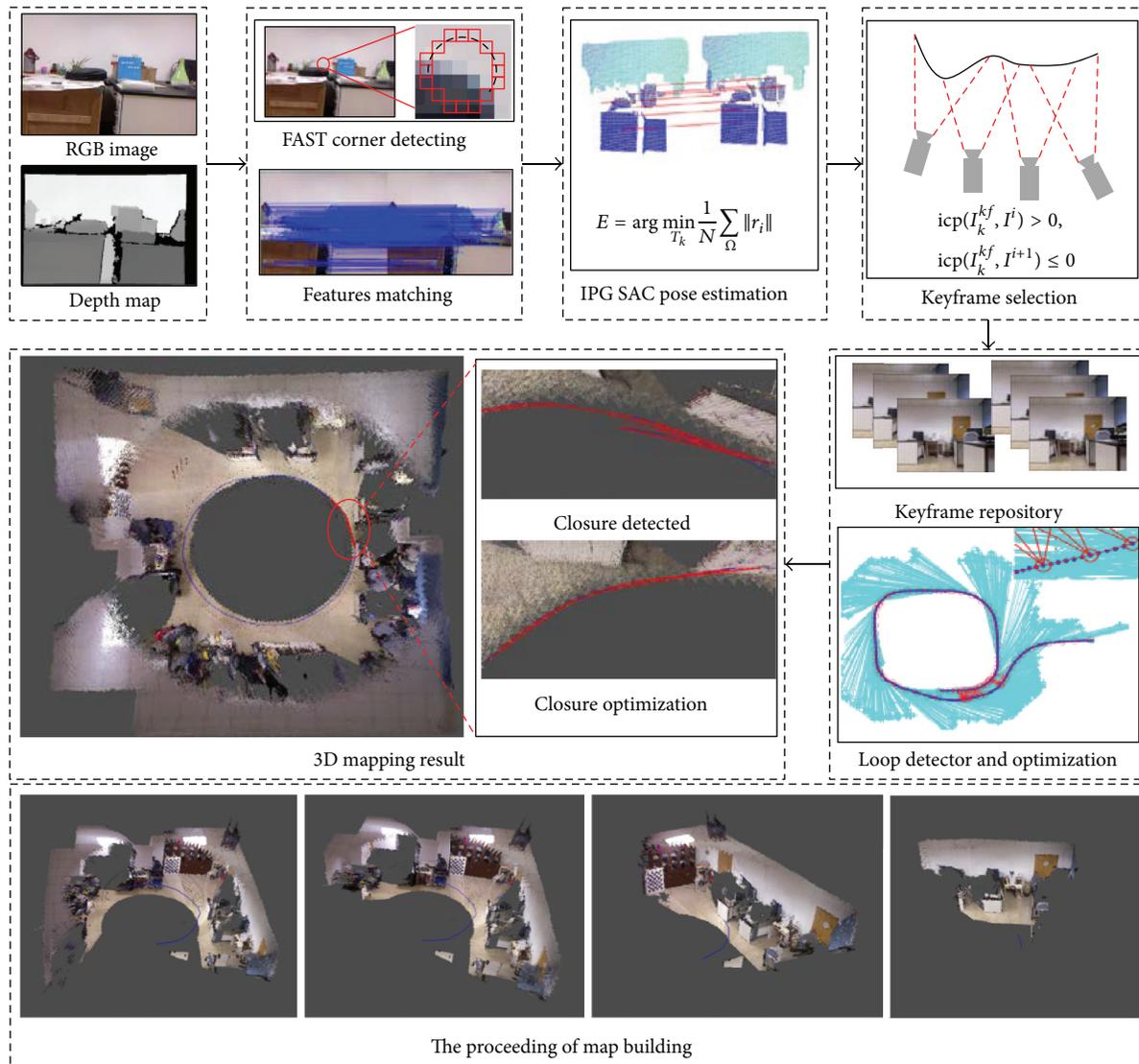


FIGURE 11: The 3D map building experimental result.

algorithms when the outlier rate is lower than 80%. In fact, the utmost outlier rate of this proposed algorithm is 85%. When the outlier rate rises to 90%, this proposed algorithm becomes unstable and unviable. In some extreme cases, it needs more than 2,000 iterations to achieve a correct solution. In future work, we will further improve the robustness and efficiency of our proposed algorithm to achieve a stable solution for the higher outlier rate. And an adapt threshold scheme will be proposed to enhance the robust of our method.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

The research work is financially supported by Promotion Project on Intelligent Robot in Beijing University of

Technology “Key Technologies on Modularization and Practicalization of Intelligent Service Robot” and the Natural Science Foundation of China (61175087).

References

- [1] G. Klein and D. Murray, “Parallel tracking and mapping for small AR workspaces,” in *Proceedings of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '07)*, pp. 225–234, IEEE, Nara, Japan, November 2007.
- [2] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, “RGB-D mapping: using Kinect-style depth cameras for dense 3D modeling of indoor environments,” *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 647–663, 2012.
- [3] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, “An evaluation of the RGB-D SLAM system,” in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '12)*, pp. 1691–1696, IEEE, St. Paul, Minn, USA, May 2012.

- [4] B. K. Horn and B. G. Schunck, "Determining optical flow," in *Proceedings of the Technical Symposium East*, pp. 319–331, International Society for Optics and Photonics, 1981.
- [5] R. Ranftl, S. Gehrig, T. Pock, and H. Bischof, "Pushing the limits of stereo using variational stereo estimation," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV '12)*, pp. 401–407, Alcalá de Henares, Spain, June 2012.
- [6] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK, 2nd edition, 2004.
- [7] X. Armangué and J. Salvi, "Overall view regarding fundamental matrix estimation," *Image and Vision Computing*, vol. 21, no. 2, pp. 205–220, 2003.
- [8] C. Kerl, J. Sturm, and D. Cremers, "Dense visual SLAM for RGB-D cameras," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '13)*, pp. 2100–2106, IEEE, Tokyo, Japan, November 2013.
- [9] F. Steinbrücker, J. Sturm, and D. Cremers, "Real-time visual odometry from dense RGB-D images," in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV '11)*, pp. 719–722, IEEE, Barcelona, Spain, November 2011.
- [10] P. H. S. Torr and A. Zisserman, "MLESA: a new robust estimator with application to estimating image geometry," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000.
- [11] P. J. Rousseeuw, "Least median of squares regression," *Journal of the American Statistical Association*, vol. 79, no. 388, pp. 871–880, 1984.
- [12] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [13] P. H. S. Torr, "Bayesian model estimation and selection for epipolar geometry and generic manifold fitting," *International Journal of Computer Vision*, vol. 50, no. 1, pp. 35–61, 2002.
- [14] S. Obdrzalek, O. Chum, and J. Matas, "Enhancing RANSAC by generalized model optimization," in *Proceedings of the Asian Conference on Computer Vision (ACCV '04)*, 2004.
- [15] B. J. Tordoff and D. W. Murray, "Guided-MLESAC: faster image transform estimation by using matching priors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1523–1535, 2005.
- [16] O. Chum and J. Matas, "Matching with proSAC—progressive sample consensus," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 1, pp. 220–226, IEEE, June 2005.
- [17] D. Myatt, P. Torr, S. Nasuto, J. Bishop, and R. Craddock, "NAPSAC: high noise, high dimensional robust estimation—it's in the bag," in *Proceedings of the 13th British Machine Vision Conference (BMVC '02)*, pp. 44.1–44.10, BMVA Press, Cardiff, Wales, 2002.
- [18] O. Chum and J. Matas, "Randomized ransac with $T_{d,d}$ test," in *Proceedings of the British Machine Vision Conference*, vol. 2, pp. 448–457, September 2002.
- [19] J. Matas and O. Chum, "Randomized RANSAC with sequential probability ratio test," in *Proceedings of the 10th IEEE International Conference on Computer Vision (ICCV '05)*, vol. 2, pp. 1727–1732, IEEE, Beijing, China, October 2005.
- [20] A. Hast, J. Nysjö, and A. Marchetti, "Optimal RANSAC—towards a repeatable algorithm for finding the optimal set," *Journal of WSCG*, vol. 21, no. 1, pp. 21–30, 2013.
- [21] V. Rodehorst and O. Hellwich, "Genetic algorithm sample consensus (gasac)—a parallel strategy for robust parameter estimation," in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '06)*, p. 103, IEEE, June 2006.
- [22] M. T. El-Melegy, "Model-wise and point-wise random sample consensus for robust regression and outlier detection," *Neural Networks*, vol. 59, pp. 23–35, 2014.
- [23] G. Shafer, *A Mathematical Theory of Evidence*, vol. 1, Princeton University Press, Princeton, NJ, USA, 1976.
- [24] T. Botterill, S. Mills, and R. Green, "New conditional sampling strategies for speeded-up ransac," in *Proceedings of the 13th British Machine Vision Conference (BMVC '09)*, pp. 1–11, September 2009.
- [25] J. D. Hol, T. B. Schön, and F. Gustafsson, "On resampling algorithms for particle filters," in *Proceedings of the Nonlinear Statistical Signal Processing Workshop (NSSPW '06)*, pp. 79–82, IEEE, Cambridge, UK, September 2006.
- [26] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings F—Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.
- [27] J. Smisek, M. Jancosek, and T. Pajdla, "3D with kinect," in *Consumer Depth Cameras for Computer Vision*, pp. 3–25, Springer, 2013.
- [28] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Computer Vision—ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7–13, 2006. Proceedings, Part I*, vol. 3951 of *Lecture Notes in Computer Science*, pp. 430–443, Springer, Berlin, Germany, 2006.
- [29] E. Rosten, R. Porter, and T. Drummond, "Faster and better: a machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.
- [30] M. Calonder, V. Lepetit, M. Özuysal, T. Trzcinski, C. Strecha, and P. Fua, "BRIEF: computing a local binary descriptor very fast," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1281–1298, 2012.
- [31] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Robotics-DL Tentative*, pp. 586–606, International Society for Optics and Photonics, 1992.
- [32] G. Grisetti, S. Grzonka, C. Stachniss, P. Pfaff, and W. Burgard, "Efficient estimation of accurate maximum likelihood maps in 3D," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '07)*, pp. 3472–3478, IEEE, San Diego, Calif, USA, November 2007.
- [33] G. Grisetti, C. Stachniss, S. Grzonka, and W. Burgard, "A tree parameterization for efficiently computing maximum likelihood maps using gradient descent," in *Robotics: Science and Systems*, 2007.
- [34] Visual Geometry Group, "Oxford visual geometry group affine covariant regions datasets," 2005, <http://www.robots.ox.ac.uk/~vgg/data/data-aff.html>.
- [35] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proceedings of the 25th IEEE/RSJ International Conference on Robotics and Intelligent Systems (IROS '12)*, pp. 573–580, IEEE, Vilamoura, Portugal, October 2012.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

