remote sensing

MDPI

Article

A Precise Indoor Visual Positioning Approach Using a Built Image Feature Database and Single User Image from Smartphone Cameras

Ming Li [1,2,†], Ruizhi Chen [1,*], Xuan Liao [1,3], Bingxuan Guo [1], Weilong Zhang [4,†] and Ge Guo [1]

[1] State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China; lisouming@whu.edu.cn (M.L.); liaoxuan@whu.edu.cn (X.L.); 0020150@whu.edu.cn (B.G.); 15872439113@163.com (G.G.)

[2] School of Resource and Environmental Science, Wuhan University, Wuhan 430079, China

[3] Department of Land Surveying and Geo-informatics, The Hong Kong Polytechnic University, Hong Kong, China

[4] Three Gorges Geotechnical Engineering Co. Ltd., Wuhan 430074, China; zhangweilong@whu.edu.cn

* Correspondence: ruizhi.chen@whu.edu.cn; Tel.: +86-136-3863-6532

† These authors contributed equally to this work.

Received: 11 February 2020; Accepted: 6 March 2020; Published: 8 March 2020

check for updates

Abstract: Indoor visual positioning is a key technology in a variety of indoor location services and applications. The particular spatial structures and environments of indoor spaces is a challenging scene for visual positioning. To address the existing problems of low positioning accuracy and low robustness, this paper proposes a precision single-image-based indoor visual positioning method for a smartphone. The proposed method includes three procedures: First, color sequence images of the indoor environment are collected in an experimental room, from which an indoor precise-positioning-feature database is produced, using a classic speed-up robust features (SURF) point matching strategy and the multi-image spatial forward intersection. Then, the relationships between the smartphone positioning image SURF feature points and object 3D points are obtained by an efficient similarity feature description retrieval method, in which a more reliable and correct matching point pair set is obtained, using a novel matching error elimination technology based on Hough transform voting. Finally, efficient perspective-n-point (EPnP) and bundle adjustment (BA) methods are used to calculate the intrinsic and extrinsic parameters of the positioning image, and the location of the smartphone is obtained as a result. Compared with the ground truth, results of the experiments indicate that the proposed approach can be used for indoor positioning, with an accuracy of approximately 10 cm. In addition, experiments show that the proposed method is more robust and efficient than the baseline method in a real scene. In the case where sufficient indoor textures are present, it has the potential to become a low-cost, precise, and highly available indoor positioning technology.

Keywords: indoor visual positioning; smartphone; feature matching; SURF; camera pose

1. Introduction

Positioning is one of the core technologies used in location-based services, augmented reality (AR), internet of everything, customer analytics, guiding vulnerable people, robotic navigation, and artificial intelligence applications [1–5]. At present, outdoor GNSS-based smartphone positioning services can achieve centimeter-level accuracy. However, GNSS signals are unavailable indoors, and it is still difficult to achieve low-cost, high-availability, and high-precision indoor positioning effects with the existing indoor positioning technology [1,3,5]. In this area, vision-based indoor positioning of smartphones is an important indoor positioning technology, which does not require much extra consumption to change the indoor environment and only needs to use the existing decorative texture information in a room. Vision-based indoor positioning has the advantages of strong practicality and wide coverage [2–6]; moreover, it is an efficient expansion of positioning technologies based on Bluetooth/iBeacon [7], WIFI [8], UWB [9,10], PDR [11], INS [12], and Geomagnetic Fields [13], with the benefits of better accuracy and lower cost.

Visual localization has become an emerging research hotspot in the field of indoor positioning [1–6,14–20]. Most state-of-the-art methods [1,2,4,6,14–18] rely on local features such as SIFT or SURF [21,22] to solve the problem of image-based localization. These methods usually contain two steps, namely establishing 2D–3D matches between features extracted from the positioning image and 3D points via descriptor matching and perspective-n-point (PnP), which calculates the extrinsic parameters. Pose estimation can only succeed if enough appropriate matches have been found in the first stage; otherwise, it will cause positioning approaches to fail. Recently, some new approaches have tackled the problem of localization with end-to-end learning. They formulate localization as a classification problem with a deep-learning architecture, where the current position is matched to the best position in the training set [19,20]. Rather than precomputing feature points and building a 3D points model, as done in classical feature-based matching localization methods, they can handle the hard scenarios with textureless areas and repetitive structures. However, in the region of certain textures, the visual positioning methods based on feature matching has advantages, especially in positioning accuracy. In image invariant local features-based indoor visual positioning approaches, according to the research content and characteristics of the visual positioning technology, there are two key problems at the algorithm level: (1) how to calculate the precise spatial pose of the positioning image robustly and rapidly; and (2) how to generate a high-precision positioning feature library. In the case of no new observation sources (such as WIFI, Ins, magnetic, etc.) being available, there exist many problems to be solved in these two aspects. This makes the application of visual positioning in indoor scenes more difficult than in outdoor environments, especially for the problem of image-feature mismatch caused by the lack of decorative texture or texture repetition in indoor scenes. In this paper, we first modify and extend a method for indoor positioning feature database establishment based on existing classical matching algorithms and strategies; our main aim is to use an epipolar constraint based on the fundamental matrix and a matching image screening strategy based on image overlap during construction of the positioning feature database. Then, introducing the image feature retrieval strategy of Kd-Tree+BBF (K dimensional Tree, Kd-Tree; Best bin first, BBF) improves the retrieval efficiency of the positioning image features, and the PROSAC algorithm is used (instead of the commonly used RANSAC algorithm) for matching optimization. In addition, the final matching result is further optimization by our proposed novel mismatched elimination method based on Hough transform voting idea, thus improving the matching precision and speed of obtaining corresponding feature points. Finally, the efficient PnP algorithm and bundle adjustment are used to solve the camera pose with accuracy. The paper proceeds with a review of visual positioning methods and related works in Section 2. The theory and methodology for proposed visual positioning method are explained in Section 3. The experimental design and the evaluation results are discussed in Sections 4 and 5, followed by conclusions in Section 6.

2. Related Work

With the rapid development of photogrammetric computer vision and optical camera technology, it is possible to achieve fast and economical image acquisition; precise and efficient image feature extraction and image matching; and quick solution of the projection matrix and external orientation elements. Moreover, image-based visual positioning has the characteristics of good visualization effect, context-rich information, and better precision. Thus, it has potential as a low-cost, accuracy active indoor positioning technology. Therefore, visual positioning technologies have been widely studied by international researchers. Generally, methods for image-based visual positioning consist of two steps:

establishing a visual location feature library for place recognition and using perspective-n-point (PnP) for a camera pose estimation [1,2]. Many algorithms and solutions have emerged in applications in different fields. For example, [1] presented a two-step pipeline for performing image-based positioning of mobile devices in indoor environments. In the first step, it generated a sparse 2.5D georeferenced image database; in the second step, a query image was matched against the image database to retrieve the best-matching database image. In [2], an accurate indoor visual positioning method was proposed for smartphones, based on a high-precision 3D photorealistic map using PnP algorithms. It focused, in particular, on the research and comparison of camera pose estimation in the case of unknown mobile phone camera internal parameters. Similarly, [3] proposed a smartphone indoor positioning dynamic ground truth reference system using robust visual encoded targets for the real-time measurement of smartphone indoor positioning technologies, providing a new low-cost and convenient method for direct ground truth measurement in the research of smartphone indoor positioning technologies. In [5], a localization method was carried out by matching image sequences captured by a camera, using a 3D model of the building in a model-based visual tracking framework. The works [14,15] studied and proposed a wide baseline matching technique based on the SIFT algorithm to improve the accuracy of image matching between the positioning image and the database image. In [16], visual features of the identification images taken from a location space were extracted by studying the SIFT-based word of bag retrieval technology, and then matched them with massive images in the database to realize indoor visual positioning. In [17], a spatial visual self-localization method based on mobile platforms in urban environments was proposed, which was useful for exploring high-precision visual positioning of smartphones in outdoor spaces. In addition, with the emergence of some fast image matching algorithms (e.g., ORB, SURF, and so on) and clustering algorithms (e.g., Random forests, SVM, and so on), the real-time performance of visual positioning methods has been studied more and more [18,21–25]. In [26], the PnP method was used to solve the motion of a calibrated camera through a set of n 3D points in the world and their corresponding 2D projections in the image. A continuous camera pose estimation method for indoor monocular cameras was proposed, which improved the camera pose estimation accuracy. However, further research on the establishment of high-precision 3D maps and rapid image retrieval from the location image database is still needed. Simultaneously, research on SfM and SLAM technologies has provided a lot of reference for the establishment of high-precision positioning feature libraries, storage and retrieval of the information in them, and accurate solution of the camera pose. In [27–29], the authors considered how to use SfM to solve the projective transformation matrix and camera parameters more robustly. In [30–35], a variety of visual SLAM schemes were proposed. A large number of algorithms for continuous camera-pose estimation and global optimization in indoor environments have been studied, and visual-based indoor real-time 3D mapping and positioning technologies have gradually been improved. However, they generally require continuous input data and occupy a large amount of the computing resources in smartphones, and they have generally been used only in a small range of VR/AR applications. In [36–41], RGB-D depth cameras were used to study high-precision real-time indoor 3D surface model reconstruction and mapping technologies. The RGB-D depth camera can provide depth information directly to the sensor while acquiring images, which improves the ability of color camera-pose estimation. It has been widely used in carriers, such as robots. In [42–48], the latest image feature extraction and image retrieval technologies were discussed, along with an analysis of the state-of-the-art methods for image location recognition, using deep learning and visual positioning based on traditional image features. It was concluded that the positioning success rate of neural network models based on deep-learning training needs to be improved and that it is difficult for the positioning accuracy to reach the decimeter level. Furthermore, the model training time was long, so the portability to different scenarios is limited. In addition, these methods have higher hardware occupation and requirements. However, there were advantages for specific objects, or when the training data were sufficient. Methods based on image invariant local features do not rely on training using big data, and have advantages in the cases where the image has more occlusion, the image color information changes sharply, or the texture is sufficient. In summary, the review of the relevant literature reveals that visual positioning approaches using local feature matching and deep learning still suffer from drift and positioning failure in different scenes, in spite of improved accuracy and robustness. Furthermore, many studies have focused on the outdoors and dedicated special mobile terminals. Research on the indoor visual positioning of smartphones remains small. Meanwhile, the visual positioning methods based on local feature matching have absolute positioning accuracy advantage in the area with texture and can better adapt to the problem of indoor occlusion, which is very conducive to the application of precision positioning in indoor shopping malls, stations, and other environments. Therefore, based on the existing theories and techniques, this paper first studies an error elimination algorithm for a high-precision visual location feature library, an efficient feature library storage and retrieval strategy, and an algorithm for robust and accurate smartphone camera pose estimation. Then, an accurate indoor visual positioning approach based on a single image from a smartphone camera is proposed, which provides an effective method for the indoor visual positioning of smartphones, using local feature matching. Thus, this paper contributes to improving the level and status of visual positioning technology in indoor smartphone positioning applications.

3. Methodology

The proposed method uses images taken from an experimental indoor environment, using a Sony ILCE-5000 camera for the image database, and builds the positioning feature database by a precise database building strategy, implemented later in the paper. The positioning images are taken with smartphone cameras. By retrieving and registering with the positioning feature database, the position of the current image can be obtained. This paper proposes and implements a visual positioning method for smartphones, based on a single smartphone image under the C/S architecture. A workflow chart of an indoor visual positioning system under the C/S architecture is shown in Figure 1. In the positioning system, complete sequence images of the indoor scene are first obtained by an optical camera, following which, feature extraction and 3D object co-ordinate calculation are performed (on the server side) to establish an object feature library for visual positioning. Then, the user takes an image (as a positioning image) through the optical camera built into the smartphone, performs feature extraction on the smartphone end, and transmits the extracted image features and camera information to the server. The accurate pose of the positioning image is then calculated on the server side by using the positioning feature library established (on the server side) in advance. Finally, the accurate pose information of the positioning image is transmitted back to the user's smartphone and displayed, thereby realizing the self-positioning of the instantaneous pose of the smartphone camera.

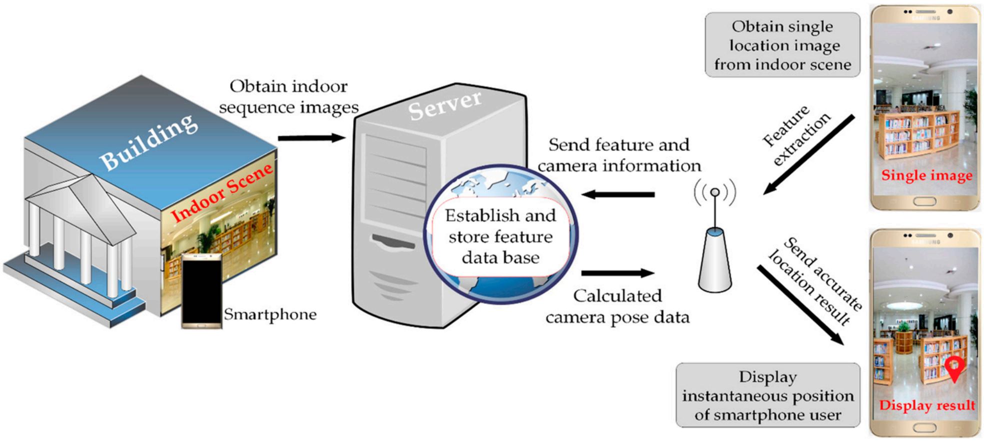

Building
Indoor Scene
Smartphone
Obtain indoor sequence images
Server
Establish and store feature data base
Send feature and camera information
Calculated camera pose data
Obtain single location image from indoor scene
Feature extraction
Single image
Send accurate location result
Display instantaneous position of smartphone user
Display result

Figure 1. The workflow chart of the indoor visual positioning system proposed in this paper.

3.1. Precise Positioning Feature Database Establishment

The use of pre-captured indoor images to establish the positioning feature database is a prerequisite for indoor visual positioning. The positioning feature database in this paper is a library file consisting of image point feature descriptors, image point co-ordinates, and 3D object point co-ordinates. It is used to provide 3D object points and image-matching information for positioning image matching and the EPnP algorithm, as shown in Figure 2. The establishment of the positioning feature database mainly includes image acquisition, bundle adjustment, and feature descriptor matching for SURF and 3D object co-ordinates.

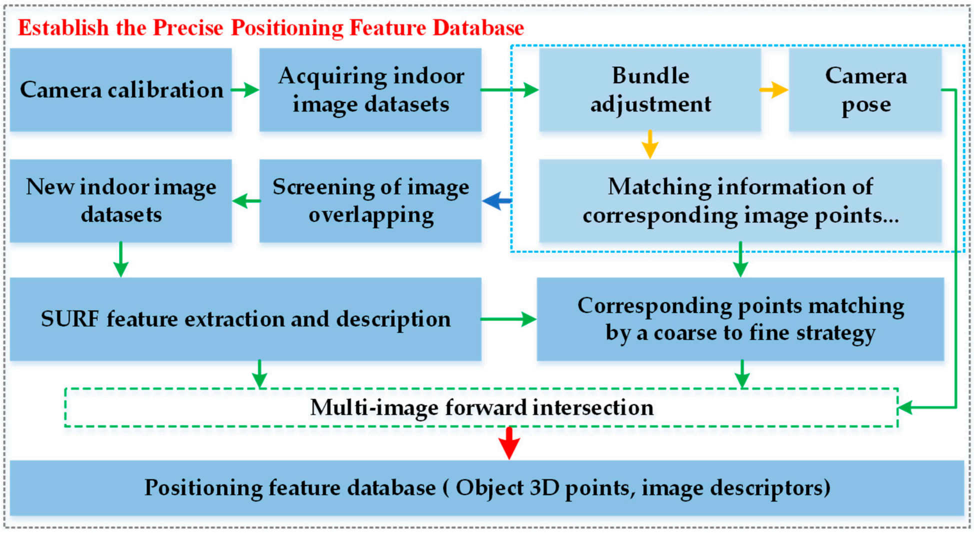

Establish the Precise Positioning Feature Database
Camera calibration
Acquiring indoor image datasets
Bundle adjustment
Camera pose
New indoor image datasets
Screening of image overlapping
Matching information of corresponding image points...
SURF feature extraction and description
Corresponding points matching by a coarse to fine strategy
Multi-image forward intersection
Positioning feature database (Object 3D points, image descriptors)

Figure 2. The workflow of the positioning feature database establishment.

3.1.1. Obtaining and Pre-Processing the Indoor Image Data Set

We first needed to take indoor images of the experimental environment before we established the positioning feature database. When shooting a complete indoor scene, we selected a commonly used camera—the Sony ILCE-5000 (Sony, Chonburi, Thailand)—which could capture photographs with a resolution of about 20 megapixels. It should be noted that, in order to reduce the influence of noise on image preprocessing, necessary texture information was needed in these images, and they needed to have a certain degree of overlap. Furthermore, camera calibration was done before shooting. After obtaining the indoor images, SfM [27–29] was used to preprocess these images, to achieve automatic bundle adjustment. Then, every camera pose of these images and every point's object space co-ordinates were obtained, and the projection matrix (PM) of every image could be simultaneously obtained [46]. According to the previous calculation, we could easily obtain the degree of overlap of the indoor images by using the pose information of every image; then, the images were selected in accordance with the principle of three-degree overlap (i.e., three adjacent images should have a certain overlap area), by the degree of overlap of the images. This served to effectively reduce redundant images from participating in subsequent image SURF feature extraction and matching.

3.1.2. Accelerated Image Feature Matching

In photogrammetric computer vision, high-precision image feature matching is a time-consuming and difficult procedure when the number of images is large. A good image feature matching algorithm and a coarse-to-fine matching strategy have typically been utilized to improve the computational efficiency and accuracy [48,49]. SIFT [21], ORB [21], and SURF [21,23] are three typical and representative image-matching algorithms for invariant local features. Among them, SURF has comprehensive advantages in computing speed, rotation robustness, fuzzy robustness, illumination invariance, and scale invariance, which means it has good time efficiency and robustness in simultaneous image matching. Therefore, to solve the problem that indoor images are easily affected by light, shooting angle, and regional environment (which results in a poor matching effect and difficulty in local area matching), this paper proposes an improved high-precision image feature matching strategy based on the SURF operator. In the experiment, after using the SURF operator to extract and describe the feature points of the indoor image dataset, instead of using the brute-force matching method, the matching information between the images obtained by bundle adjustment was used to assist the SURF feature matching, thereby avoiding the time spent searching for all the feature points in the image set (due to the feature point matching process). However, it is difficult to obtain a good matching effect with a single constraint. In order to improve the matching accuracy, this paper introduces an epipolar constraint to further improve the matching results of the corresponding image points. In the experiment, the fundament matrix (FM) is calculated by using the PM obtained by bundle adjustment, following which the epipolar lines of the corresponding image points can be solved by using the FMs. In this way, the epipolar lines of all image feature points can be calculated and used to eliminate mismatch. Thus, by increasing the degree of matching constraints, high-quality matching sets can be obtained at the same time.

3.1.3. Multi-Image Spatial Forward Intersection

The poses of images from the indoor image database are calculated by bundle adjustment, and the information of matching point pairs in the images is obtained by SURF matching algorithm and our strategy. These images are then used as database images. Thus, the forward intersection can obtain 3D object points, as object points have more than two observations from the images. Multi-image spatial forward intersection is adopted, because some observation are outlines; thus, RANSAC is used to estimate the optimal solution, as it performs better than least squares when there are many outliers. As shown in Figure 3, the yellow points are the top view of the 3D object points, and the red points are the camera exposure points of the positioning image captured by smartphone cameras. After a geometry check, there were many outliers in the point cloud.

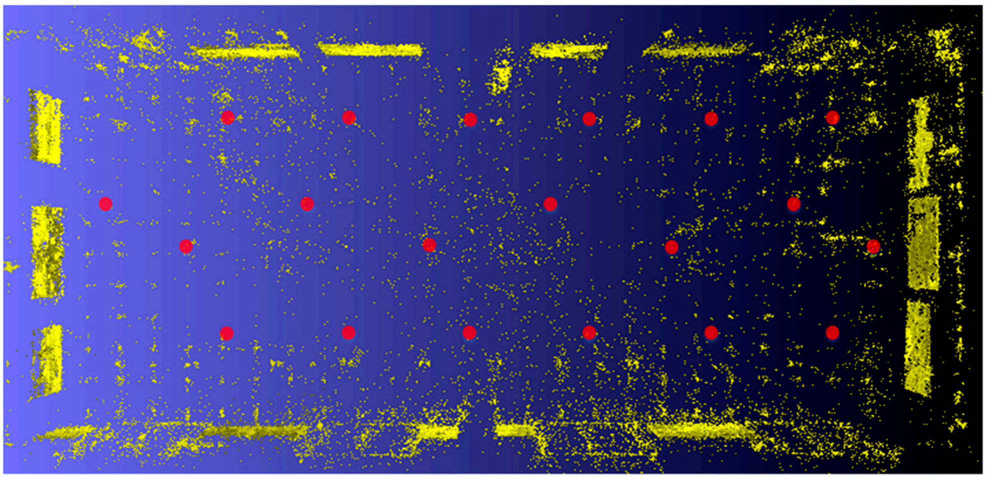


Figure 3. Three-dimensional object point cloud from multi-image spatial forward intersection.

After completing the above work, the positioning feature database can be established. It includes the 3D co-ordinates of the object points and descriptor information corresponding to each object point. The 3D co-ordinates of the object points can be expressed as $P_n(X_n, Y_n, Z_n)$ and the corresponding feature descriptors can be expressed as $(\text{feature}_{nn}, \text{feature}_{n(n+1)}, \text{feature}_{n(n+2)}, \ldots)$, where n is a positive integer.

3.2. Online Smartphone Indoor Visual Positioning

The process of online smartphone indoor visual positioning based on a single image from a smartphone camera includes the following steps: First, a single image is taken by the smartphone camera. Then, feature point extraction and description are performed, and similar feature descriptors are searched for in the positioning feature database. Finally, the pose of the smartphone is calculated and returned. The smartphone indoor visual positioning procedure is shown in Figure 4.

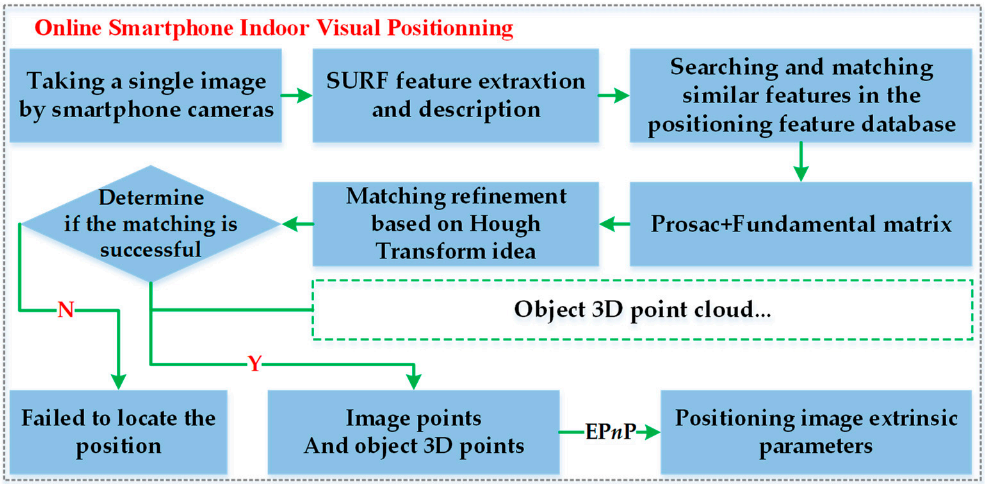

Online Smartphone Indoor Visual Positionning
Taking a single image by smartphone cameras
SURF feature extraxtion and description
Searching and matching similar features in the positioning feature database
Prosac+Fundamental matrix
Matching refinement based on Hough Transform idea
Determine if the matching is successful
Object 3D point cloud...
N
Y
Failed to locate the position
Image points And object 3D points
EP*n*P
Positioning image extrinsic parameters

Figure 4. The workflow of single smartphone image positioning.

In the experiment, Kd-Tree+BBF [50] was used to retrieve the similar image descriptors. After SURF feature matching, using minimum distance, and geometry check, using the fundamental matrix and PROSAC [51] to select the inliers, the final matching result was further purified by our proposed method, based on Hough Transform voting.

3.2.1. SURF Feature Retrieval and Matching in Positioning Feature Database

After extracting the SURF features in the positioning image and establishing the descriptors, the feature point set, $\boldsymbol{P}$, of the positioning image and the SURF descriptor subset, $\boldsymbol{D}$, corresponding to $\boldsymbol{P}$, are obtained. As we have an established positioning feature database and the smartphone camera interior parameters can be obtained from the Exif (Exchangeable image file format) file, the instantaneous shooting position of the smartphone can be calculated by matching the SURF features of positioning image with the SURF features in the pre-established positioning feature database. In the experiment, for the SURF descriptor d_i ($i = 0, 1, 2 \ldots\ n$, n is a positive integer) of a feature point p_i ($i = 0, 1, 2 \ldots\ n$, n is a positive integer) in the positioning image, if a brute force search method is used to search and match the descriptors in the positioning feature database, it traverses all descriptors in the positioning feature database each time, and the positioning time is greatly increased. Kd-tree is one of many high-dimensional spatial index structure and approximate query algorithms. It establishes an effective index structure by hierarchically dividing the search space, which greatly speeds up the retrieval. In image feature matching algorithms (e.g., SIFT and SURF), the standard Kd-tree index structure has been widely used for fast image feature comparison. However, its efficiency is closely related to the dimension of the feature vector. The higher the number of dimensions, the lower the efficiency. This is because the query completion process of each nearest neighbor eventually ends up falling back to the root node, resulting in unnecessary backtracking and node comparisons. When these extra losses occur in high-dimensional data lookups, the search efficiency becomes quite low. Incorporating BBF into the bilateral matching in the standard Kd-tree algorithm can significantly solve this problem. In short, its improvement to Kd-tree is to sequentially sort the nodes in the “query path” to shorten the search time.

In the experiment, the Kd-Tree+BBF similar feature search strategy was used for corresponding feature matching between the feature descriptors of positioning image and the feature descriptors in the positioning feature database. By traversing all the feature points in the positioning image and searching the corresponding matching descriptors for them in the positioning feature database, a series of matching feature point pair sets, $\boldsymbol{M}$, can be obtained. After the query procedure, corresponding feature matching by minimum distance and geometry check is carried out by using the fundamental matrix and PROSAC to select the inliers. In the experiment, PROSAC was used—instead of RANSAC—mainly because it can effectively reduce the number of iterations and time consumption when there are outliers in the matching points, as well as improving the time and robustness of the matching error elimination algorithm. In order to compare the effects of the two methods, this paper introduces the precision–recall curve, which is calculated by using Equations (1) and (2).

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

In Equations (1) and (2), TP is the number of real matching points which are predicted as matching points, FP is the number of real mismatching points which are predicted as matching points, and FN is the number of real feature matching points which are predicted as mismatching points.

3.2.2. Matching Error Elimination Based on Hough Transform Voting Idea

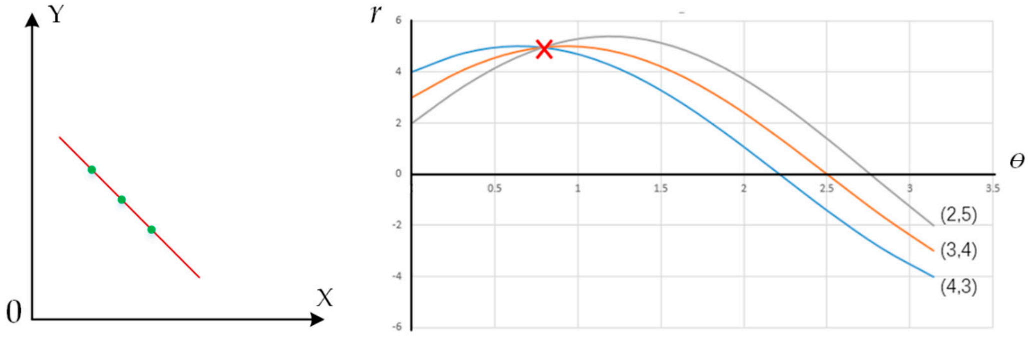
After matching the SURF feature descriptors extracted from the smartphone positioning image with the feature descriptors in the pre-established positioning feature database, a series of image points from pre-located smartphone images and their corresponding object points are obtained; that is, each matching point pair includes a two-dimensional image point of a smartphone positioning image and a corresponding object point in three-dimensional space.

As the similarity degree of the image feature descriptors is used in the point matching process to find the corresponding relationship between 2D image points and 3D object points, even if the feature matching results (as obtained in Section 3.2.1) eliminate a large number of mismatched points by matching optimization, there will still be a certain number of mismatched point pairs in the corresponding points due to similar textures and other factors in indoor space. If the matching error in the corresponding points is not eliminated and the subsequent smartphone camera pose is directly solved by P*n*P, the estimated camera pose may have a large error or may not even be solved. Therefore, in order to meet the smartphone camera-pose calculation requirements, this paper hopes to eliminate such mismatches as much as possible, to improve the success rate and accuracy of the smartphone camera-pose solving. A matching error elimination is proposed here based on Hough Transform Voting Idea (HTVI). This section concisely and clearly introduces the proposed mismatching elimination method based on the Hough transform voting idea, further purifying the matching point pairs (i.e., those obtained in Section 3.2.1).

The Hough transform is an image feature recognition and extraction technique which finds a particular type of shape by voting in the parameter space [52,53]. The simplest Hough transform is straight-line detection; a brief introduction follows. A straight line in two-dimensional space is shown in Figure 5a. The Equation of the line can be represented by polar co-ordinates:

$$r = x\cos\theta + y\sin\theta \tag{3}$$

where r is the distance from the origin to the nearest point on the red straight line (called the polar path), and θ is the angle between the blue dashed line and the X-axis (called the polar angle).



(a) **(b)**

Figure 5. An instance of points in a 2D space being transformed into sinusoids in Hough space: (**a**) three points in 2D space and (**b**) three sinusoids in Hough space.

Each straight line corresponds to a pair of parameters (r, θ). This two-dimensional parameter space is a Hough space, which can be used to represent the collection of all two-dimensional straight lines. According to the principle of the Hough transform, if the co-ordinate of a two-dimensional point is known, then all straight lines passing through this point become a sinusoid in the Hough space. For ease of understanding, let us use an example to illustrate the Hough transform straight-line detection method.

Suppose there are three points (4, 3), (3, 4), and (2, 5) in two-dimensional space. As shown in Figure 5a, these three points satisfy the collinearity condition. Converting these three points into the Hough space yields three sinusoids, as shown in Figure 5b. It can be seen from the figure that the sinusoids of the three points in the Hough space intersect at one point. According to such characteristics, the feature points extracted from the image can be converted into the Hough space, and the position of the intersection of the sinusoids gives the parameters of the straight-line equation. Therefore, finding the intersection of the sinusoids in a set of sinusoidal curves in the Hough parameter space is the key. Straight-line detection in Hough space essentially uses a voting idea, which can be divided into three steps: First, the Hough parameter space is quantized into a series of finite intervals (or accumulator boxes). Then, the points that may be straight lines are converted into a sinusoidal function in Hough space, and the number of votes in the corresponding accumulator box is increased, according to the areas where the sinusoids are distributed in Hough space. Finally, the object most likely to be a straight line is detected by looking up the local maximum value in the accumulator.

When a positioning image is taken with a smartphone camera, a line of light is formed from the object point, and the center of photography has an intersection with the image plane of the image. Moreover, this intersection point is the image point corresponding to this object point. Put simply, for an image, the center of the image at the time of image capture, the image point on the image, and the object point corresponding to the image point are on the same straight line. This idea is demonstrated in Figure 6. ***S*** is the photography center, P is the object point, and p is the image point

According to the previous introduction, when a smartphone captures an image for positioning, it first needs to match the feature descriptors extracted from the smartphone positioning image with the feature descriptors in the positioning feature database. Then, matching point pairs in which the image points of positioning image are in one-to-one correspondence with the object space 3D points are obtained. The object point and the image point in each matching point pair can be connected to obtain a straight line. Then, the matching point pairs (of Section 3.2.1) correspond to straight lines in 3D space. In order to simplify the computational complexity, we project the 3D space onto the ground and simplify it into a 2D plane; that is, we project these straight lines in space onto the ground to obtain a set of 2D straight lines. In this paper, we denote such a straight-line sequence by ***L***.

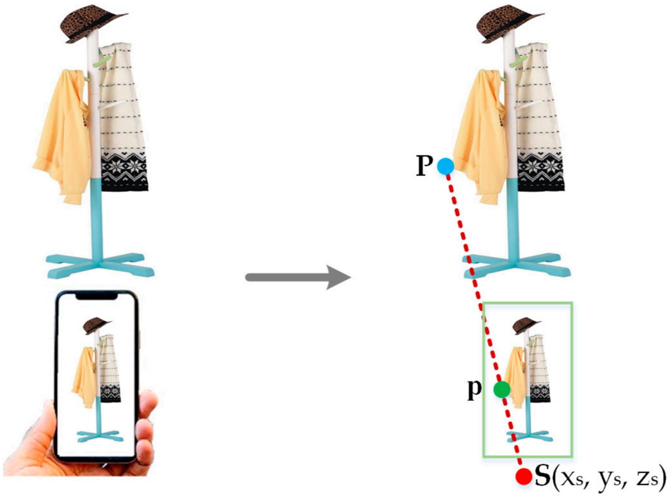


Figure 6. The schematic diagram of three-point collinearity in smartphone photography.

In these straight lines, there are straight lines which are connected by correct matching point pairs, but there are also straight lines that are connected by erroneous matching point pairs. If it is a straight line connected by the correct matching point pair, the line connecting an image point to the corresponding real object point must pass through the photography center, ***S***. If it is a straight line connected by a mismatched point pair, as the object point is not a real object point corresponding to an image point, the line will not pass through the photography center, ***S***. If all Fmatching point pairs are correct matches, then all the straight lines in ***L*** should pass through the same point, i.e., the photography center, ***S***, at the time of shooting.

The premise of this hypothesis is that most of the matching point pairs obtained are correct and that only a small number are mismatched. In fact, it is proved in the subsequent experiments that this assumption is true. In this way, we only need to remove the lines that do not pass through the photography center from the line sequence, L. This paper draws on the idea of voting in the Hough transform to perform mismatching point culling. The biggest different from the Hough transform is that the method proposed in this paper does not vote in the parameter space. Instead, the area where all the straight lines pass through is voted for directly in the projected two-dimensional plan space. The area with the highest voting value can be regarded as the area where the photography center is located. A line that does not pass through this area can be considered to be a mismatch for its corresponding matching point pair, which can be eliminated.

3.2.3. Single Image Positioning

In this section, using the matching feature points, the corresponding 3D object points of the feature points from the smartphone positioning image are obtained. The methods of camera-pose calculation can be used to calculate the extrinsic parameters of the positioning image. Perspective-*n*-point (P*n*P) is a method for solving 3D-to-2D point pair motion. It estimates the pose of the camera when shooting images by obtaining *n* 3D object spatial points and their projected positions in the image. A P3P problem is shown in Figure 7 that is one of the common methods for solving P*n*P problems. As it can obtain better motion estimation in few matching points, it has been considered to be the most important camera-pose estimation method.

In Figure 7, ***O*** is the camera's optical center; and *a*, *b*, and *c* are the 2D projection points on the image plane corresponding to 3D object points *A*, *B*, and *C*, respectively.

Although P3P is an important and common method for solving PnP problems, it cannot make full use of information and is susceptible to noise and mismatching points. In order to solve this problem, a better improvement method is to use EP*n*P (Efficient P*n*P) for pose solving. It can make use of more information and optimize the camera pose in an iterative way, in order to eliminate the influence of noise as much as possible. In this paper, the EPnP algorithm and bundle adjustment (BA) are used to solve the camera pose. In addition, other methods, such as UPnP (Uncalibrated PnP), have been widely used to estimate camera pose in different situations. We compare them later, in the Experimental Analysis section.

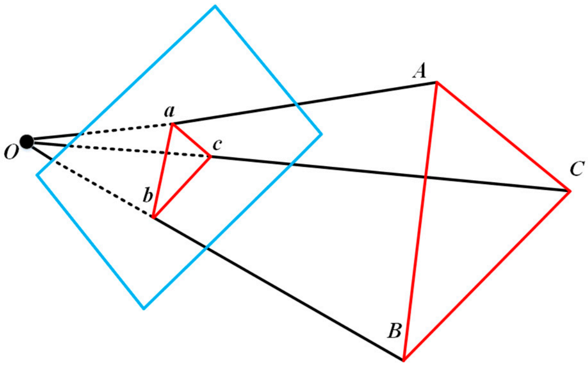


Figure 7. The schematic map of P3P.

4. Experimental Analysis

4.1. Test Data and Experimental Environment

In the experiment, three different indoor scenes were selected as indoor experimental environments to evaluate the proposed method. Among them, two indoor scenes with decorative pictures of different materials were set up as experimental fields, as well as a real conference room scene. Figure 8 shows the decorated experimental rooms. Figure 9 shows the real conference scene. The database images were taken with a Sony ILCE-5000, where the image size was 5456 × 3632. There were 149, 151, and 100 images in the building positioning feature databases for Room 212, Room 214, and the conference room, respectively. To evaluate the precision of the positioning, the non-prism total station (Leica TS60) was used to measure the camera position of the smartphone in the experiment; the value measured by the total station was taken as the ground truth. It was difficult to measure the smartphone camera, as the surface of the camera was a glass material. Therefore, a particular ring crosshair was affixed to the camera for aiming and automatic tracking measurement by the TS60. Figure 10 shows the Leica measurement robot, the ring crosshair affixed on the smartphone, and the interface of experimental app. The purple circle in Figure 10c shows the solved instantaneous 2D co-ordinates when the positioning image was taken. In the experiment, we evenly selected the position when shooting the positioning image in the experimental rooms and held smartphones to capture the positioning images in these positions, while using the TS60 to measure the ring crosshair on the smartphones. After measuring the offset, the smartphone camera position was acquired.

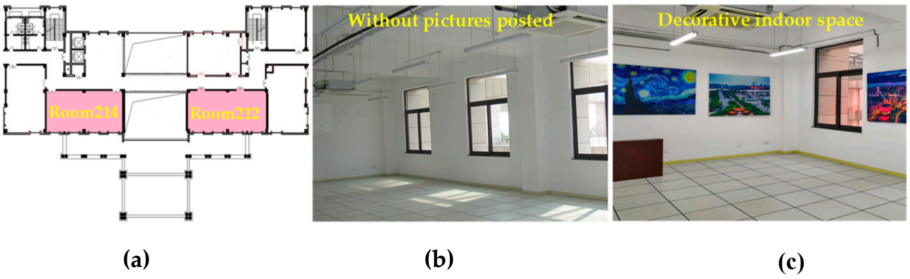


Figure 8. The decorated experimental rooms in a building: (**a**) location of rooms, (**b**) undecorated room, and (**c**) decorated room.

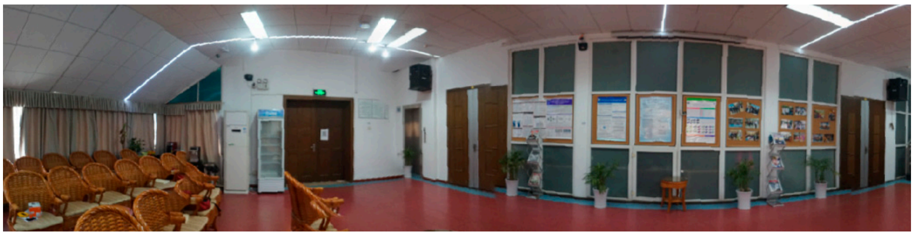


Figure 9. The real conference scene room.

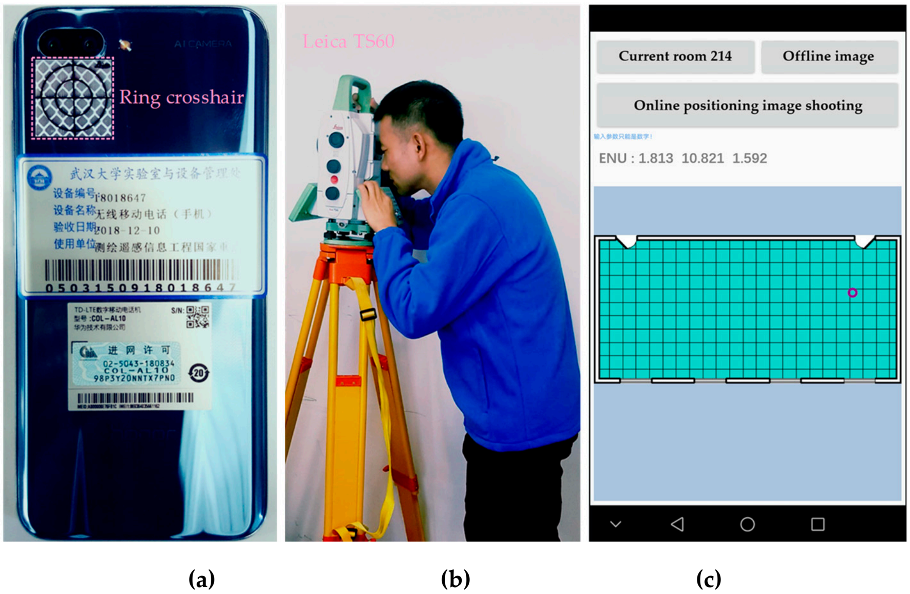
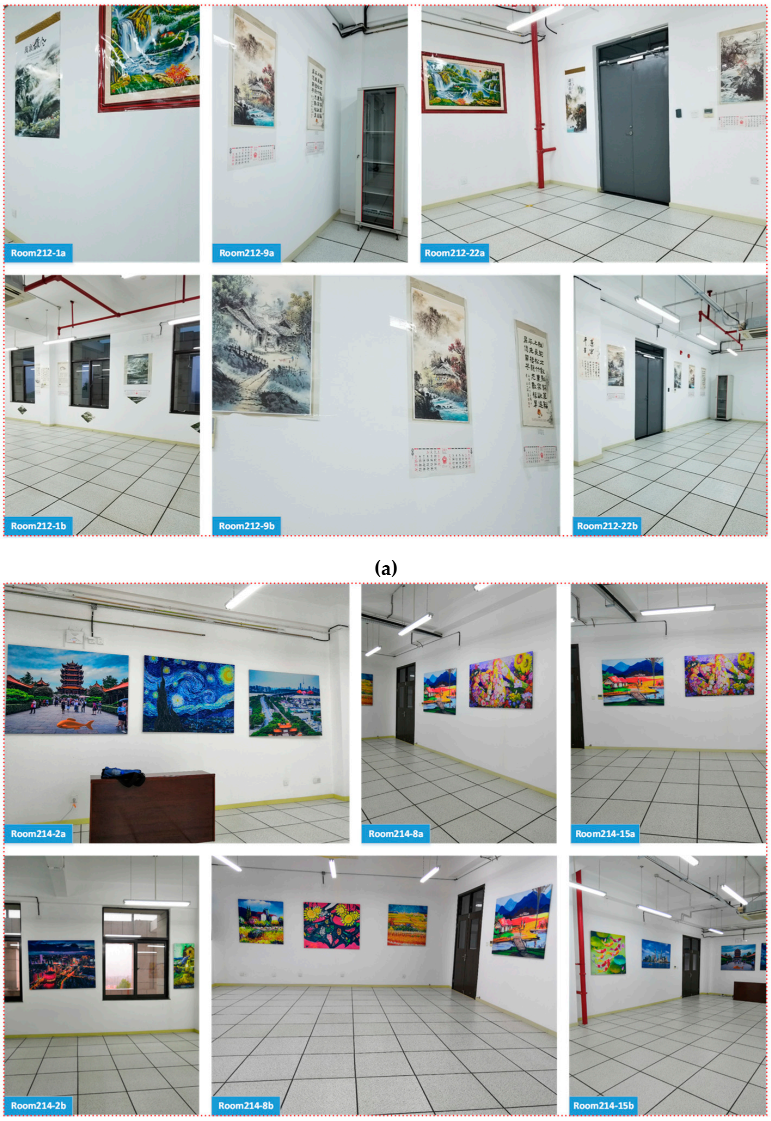


Figure 10. Experimental measure equipment: (**a**) ring crosshair, (**b**) Leica TS60, and (**c**) demo App.

The indoor area of each decorated experimental room was approximately 120 square meters; however, Room 212 has a little more space than Room 214. We selected 23 and 20 positioning image capture points in Rooms 212 and 214, respectively, and Rectangular Plane (X, Y) Co-ordinate Systems were established in the two rooms. Huawei Honor 10 and Samsung Galaxy S8 smartphones were used to capture the positioning images at these points to implement the smartphone positioning experiment, and two images were captured at different orientations at the same position of each point. Figure 11 shows some of the smartphone positioning images. In Room 212, the Huawei and Samsung smartphones each obtained 46 experimental positioning images; in Room 214, the Huawei and Samsung smartphones each obtained 40 experimental positioning images. The image resolution of the Huawei Honor 10 is 3456 × 4608, and the image resolution of the Samsung Galaxy S8 is 3024 × 4032. It should be noted that the current indoor positioning service is more concerned with our planar position on a certain floor of the indoor space, but not as concerned about the indoor height information. This is because, when people use a smartphone in a certain height space, its height space will fluctuate within a small range. People can easily estimate the height of a smartphone based on information such as their stature, and this estimate generally does not deviate too much from the true height. Therefore, when using a smartphone for positioning and navigation indoors, we often only need to know the spatial planar (i.e., ground) position. Hence, in the experiment, our focus was to verify the accuracy of the planar co-ordinate values for smartphone positioning in the indoor space. In addition, to prove how the present study advances the existing state-of-the-art, the original method using the image with the most matches to calculate the smartphone camera pose, as in [1], was used as a baseline method. We compared the two methods in a real scene. The experimental Desktop computing environment was the Windows 10 operating system with an Intel (R) Core (TM) i7-7820HK and 32 GB RAM.



(a)

(b)

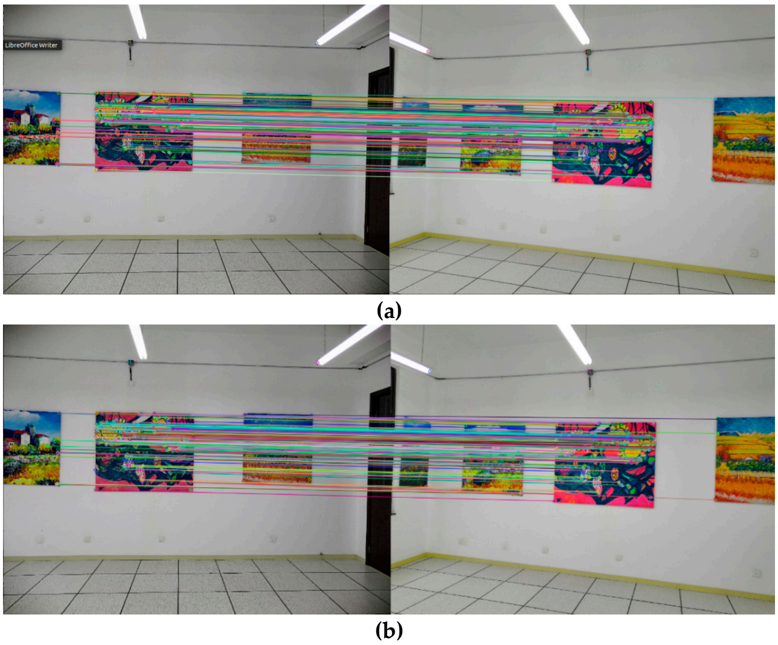
Figure 11. Positioning images: (**a**) images in Room 212 and (**b**) images in Room 214.

4.2. Optimization Matching Results Comparison of RANSAC and PROSAC Algorithms

Figure 12 shows the matching optimization results for the RANSAC and PROSAC, where the image in the lower half of the figure (PROSAC) has more feature matches and better distribution-match results.

Figure 13 is the precision–recall curve of RANSAC and PROSAC with the experimental images in Figure 12. It can be seen from Figure 13 that the PROSAC algorithm has advantages over the RANSAC algorithm, in terms of precision rate under the same recall rate, especially when the recall rate is between 0.55 and 0.8. In this range, there were enough interior points, and the precision rate was high. This was consistent with conclusions previously drawn in the literature—that the recommended recall value is around 0.65, which can ensure that the image-matching interior point set can satisfy both the requirements of number and quality [51].

Figure 14 shows the time–proportion in terms of interior points given by RANSAC and PROSAC on the pair of indoor matching images shown in Figure 12. It can be seen that the average time cost of PROSAC was significantly lower than that of RANSAC. This was because the random sampling in the RANSAC algorithm leads to more iterations. In general, the average number of iterations was greater than one, so the time cost of the RANSAC algorithm was relatively large. As PROSAC presorts the interior points, it can obtain better samples during the sampling process, such that the number of iterations is far less than that of the RANSAC algorithm. Generally, one iteration could obtain the correct model, and the corresponding number of iterations was small. As the percentage of interior points increased, the probability that RANSAC selected an interior point when randomly selecting samples became larger, and the success rate of obtaining the correct model increased correspondingly, such that the number of iterations decreased and the time cost became smaller. The running time of PROSAC was almost independent of the proportion of interior points, and it was more robust to sample error.



(a)

(b)

Figure 12. Matching optimization results of different algorithms: (**a**) RANSAC and (**b**) PROSAC.

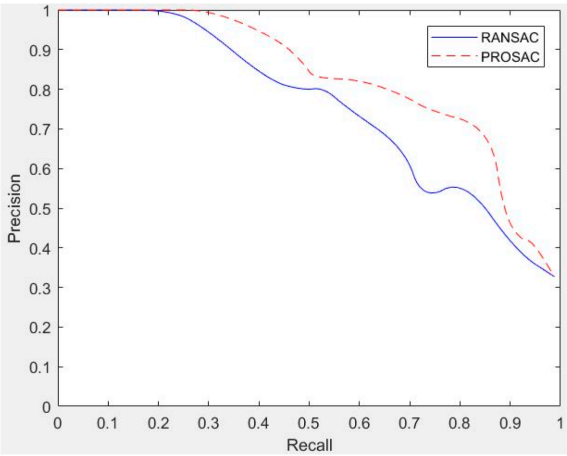


Figure 13. The precision-recall curves of RANSAC and PROSAC.

4.3. *Experimental Comparison Before and after Using the Mismatch Elimination Method Based on HTVI*

Figure 15 shows the 10 experimental smartphone positioning images in this experiment. Table 1 is a comparison of the planar positioning results obtained by the PnP method for the 10 smartphone positioning images, before and after the matching error elimination based on HTVI. If the location result is beyond the range of the test room, it is considered that the location result is wrong, and the case where the positioning result cannot be output is called positioning failure. In other cases, the positioning is successful.

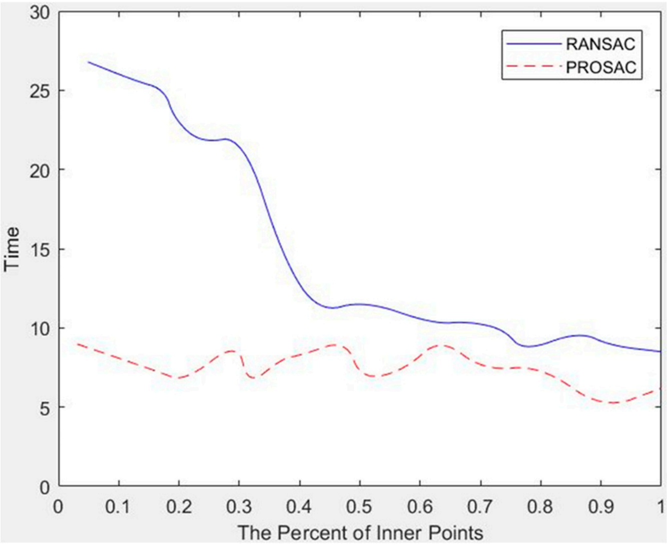


Figure 14. Time-proportion of interior points: comparison between RANSAC and PROSAC.

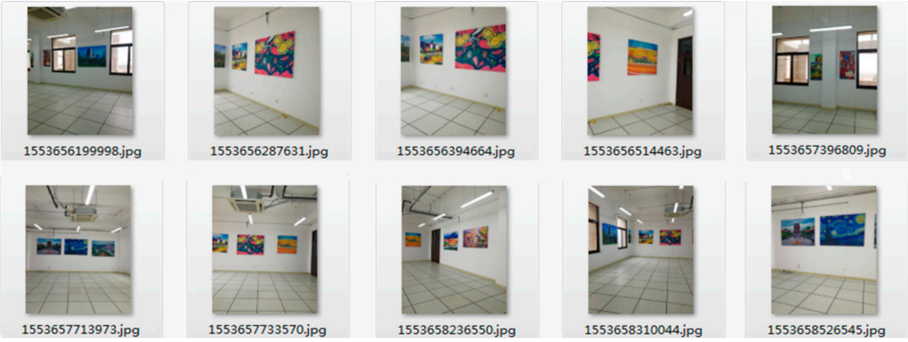


Figure 15. Ten experimental smartphone positioning images.

From the results of Table 1, it can easily be found that the positioning success rate was 80% and the correct rate was 50% before using the matching error elimination based on HTVI detailed in Section 3.2.2. After using the proposed method, the positioning success rate and correct rate are both 100%, and the average error of positioning decreased from 0.98 to 0.61 m. Although there were many factors affecting the success and accuracy of positioning, this experiment reflects the effect and significance of further elimination of mismatching.

Table 1. Comparison of positioning results.

No.	**Before (m)**		**After (m)**		**Ground Truth (m)**
	Measure	**Point Error**	**Measure**	**Point Error**	
1	No output	Failure	(1.931, 5.980)	0.88	(1.186, 5.517)
2	(3.156, 4.593)	1.26	(3.320, 4.521)	0.30	(3.030, 4.415)
3	(−3.230, 3.071)	Wrong	(2.141, 2.678)	0.89	(2.920, 3.110)
4	(2.850, 11.27)	Wrong	(3.061, 2.028)	0.43	(3.030, 2.440)
5	No output	Failure	(5.785, 3.130)	0.25	(6.025, 3.080)
6	(9.053, 2.847)	0.28	(8.506, 3.162)	0.48	(8.900, 3.110)
7	(9.620, 2.943)	0.67	(9.167, 2.536)	0.61	(8.900, 3.110)
8	(8.181, 1.582)	0.39	(7.885, 1.690)	0.56	(8.390, 1.915)
9	(3.832, 11.873)	Wrong	(3.679,1.591)	0.47	(4.060, 1.850)
10	(5.623, 4.128)	0.46	(5.661, 4.035)	0.41	(5.380, 3.725)

4.4. Comparison Experiment of Three Camera-Pose Estimation Methods

In order to compare the three most commonly used camera-pose estimation methods (i.e., P*n*P, EP*n*P, and UP*n*P), we carried out a relative pose recovery experiment, using the pair of indoor images presented in Figure 12. The experimental results obtained are shown in Table 2.

Table 2. Comparison of different camera pose estimation methods.

Method	Error	Time
P*n*P	4.277 mm	5.531 ms
EP*n*P	1.311 mm	5.111 ms
UP*n*P	2.796 mm	5.309 ms

In this experiment, *n* was 3. It can be seen that EP*n*P was superior to other methods in accuracy and time and so the EP*n*P method was selected for camera pose estimation in this paper.

4.5. Experimental Accuracy Evaluation with Decorated Indoor Scene

In the experiment, as the root mean square error (RMSE) can well reflect the precision of the measurement, this paper uses the RMSE for accuracy evaluation. The RMSE values of the X and Y direction and the total error were calculated, which are denoted by ΔX, ΔY, and ΔD, respectively. In addition, the mean square error of a point (MSEP) is used to calculate the offset between the measured value of each positioning image and the truth value of the point where it is located. Equations (4) and (5) are their respective mathematical expressions:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(M_i - G_i)^2}{n}} \tag{4}$$

$$MSEP = \sqrt{(\Delta X)^2 + (\Delta Y)^2} = \sqrt{\left(X_{measure,i} - X_{ground\_truth,i}\right)^2 + \left(Y_{measure,i} - Y_{ground\_truth,i}\right)^2} \tag{5}$$

In Equation (4), M_i is the measured value and G_i is the ground truth corresponding to M_i. In Equation (5), $X_{measure,i}$ and $Y_{measure,i}$ are the measured co-ordinate values of the positioning image, and $X_{ground\_truth,i}$ and $Y_{ground\_truth,i}$ are the ground truth, corresponding to the measured co-ordinate values. The value of i ranges from 1 to n, where n is a positive integer.

Tables 3 and 4 show the RMSE values of two smartphone positioning experiments in the experimental indoor spaces Room 212 and Room 214, respectively. From the perspective of overall co-ordinate accuracy, the positioning accuracy of the proposed method was at the decimeter or centimeter level, which is much better than other indoor positioning technologies, such as Bluetooth, PDR, and Wi-Fi. Of course, visual positioning is inherently a highly accurate positioning technique in an indoor space with sufficient image textures; the results of this paper also prove this. As mentioned above, in order to prove the effectiveness of the proposed method, we used different smartphones to capture the positioning image in different rooms, in different situations and environments, such as different viewpoints, positions, illumination, distance, indoor decorative textures, and materials. As shown in Tables 3 and 4, there were significant differences in the positioning accuracy of two brands of smartphones in different rooms, where the differences in positioning accuracy between the two brands of smartphones in the same room were much smaller. As can be seen from Figure 8a, in order to avoid the influence of outdoor ambient light in the rooms, we selected two symmetrical rooms on the same side of the building as the experimental spaces. Moreover, the indoor positioning images were taken under the same indoor lighting conditions, at the same time. Based on the above considerations, we believe that the main reasons causing the positioning accuracy in Room 214 to be significantly better than that in Room 212 were the factors of the interior decoration texture and room size. As the difference in the indoor space between the rooms was small, the most important influence on the positioning accuracy was the interior decoration texture. As shown in Figure 11, three pairs of positioning images were shown from two rooms.

Table 3. Accuracy evaluation of positioning results in Room 212.

Room 212			
Phone Type		Huawei Honor 10	Samsung Galaxy S8
		Measure (m)	Measure (m)
RMSE	ΔX (m)	0.099	0.094
	ΔY (m)	0.069	0.097
	ΔD (m)	0.120	0.135

Table 4. Accuracy evaluation of positioning results in Room 214.

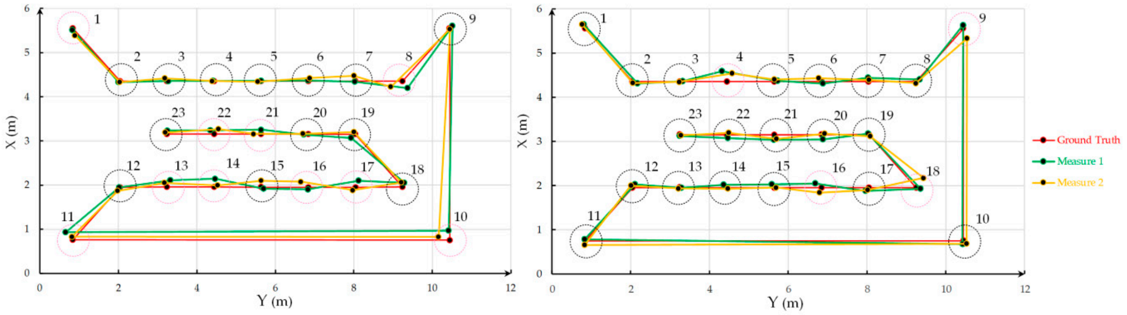
Room 214			
Phone Type		Huawei Honor 10	Samsung Galaxy S8
		Measure (m)	Measure (m)
RMSE	ΔX (m)	0.061	0.058
	ΔY (m)	0.059	0.075
	ΔD (m)	0.085	0.095

The decorative paintings posted in Room 212 were made of plastic paper and copper paper, and the decorative paintings in Room 214 were made of fabric. It is easy to see that the textures in Room 212 had noticeable reflections and that its wall decoration texture was not as rich as Room 214's. From the experimental results, these differences obviously affected the positioning accuracy. The difference in positioning accuracy between the Samsung Galaxy S8 and the Huawei Honor 10 is likely to be mainly due to the high imaging quality and resolution of the latter's camera. Therefore, the Huawei Honor 10 achieved slightly better positioning results in the experiment. It must be noted that, in terms of the RMSE metric, the proposed method achieves precision positioning results.

Figures 16 and 17 show the co-ordinate offset between the visual positioning measurements and the ground truth for the different smartphones in Rooms 212 and 214, with the method proposed in this paper. In Figure 16a, the errors of points 1, 8, 10, 11, 13, 14, 16, 17, 21, and 22 were larger than 15 cm. In Figure 16b, the errors of points 4, 9, 16, and 18 were larger than 15 cm. In Figure 17a, the errors of points 12, 14, 15, 19, and 20 were larger than 15 cm. In Figure 17b, the errors of points 4 and 8 were larger than 15 cm. The errors are given in Tables 5 and 6. In the corresponding smartphone positioning images in the experiment, the capture distances of these points were far and the image shooting angle was large. In addition, windows occupied the majority of the frame in some images, resulting in fewer available textures. These factors are problems that must be overcome in image matching, in order to conform to the fundamentals of image positioning technology. Although this paper has done a lot of work in image matching and proposed a reliable precision and fast image-matching strategy, it is still difficult to deal with all situations.

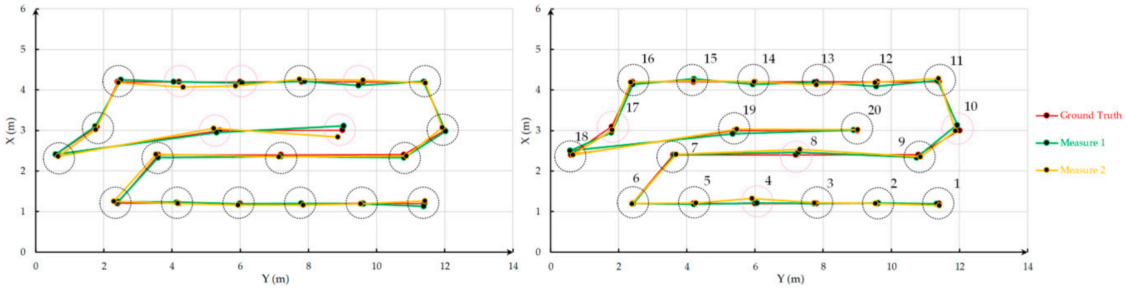
4.6. Experimental Accuracy Comparison in Real Indoor Scene

To prove the effectiveness of our proposed method, we conducted further experiments in a real indoor scene and compared the accuracy and robustness with one of the existing similar methods. The baseline method was the original method, using the image with the most matches to calculate the extrinsic parameters of the smartphone camera (as in [1]), which was proposed by scholars at the University of California, Berkeley. As shown in Figure 9, the experimental environment was a real conference scene room with different kinds of furniture and furnishings. In this experimental scene, we took 32 positioning images, using the Huawei smartphone. The MSEP of each positioning image was calculated for both our method and the baseline method. A statistical table of positioning error results is shown in Table 7.



(a) **(b)**

Figure 16. Positioning result co-ordinate offset in Room 212: (**a**) Samsung and (**b**) Huawei smartphones.



(a) **(b)**

Figure 17. Positioning result co-ordinate offset in Room 214: (**a**) Samsung and (**b**) Huawei smartphones.

Table 5. Positioning accuracy of the smartphones in Room 212.

No.	Room 212 (Huawei Honor 10)		Room 212 (Samsung Galaxy S8)	
	Measure Error 1	Measure Error 2	Measure Error 1	Measure Error 2
	MSEP (m)	*MSEP* (m)	*MSEP* (m)	*MSEP* (m)
1	0.096	0.118	0.039	0.179
2	0.130	0.032	0.040	0.020
3	0.073	0.009	0.032	0.084
4	0.268	0.223	0.028	0.056
5	0.087	0.047	0.007	0.093
6	0.057	0.101	0.020	0.081
7	0.088	0.039	0.017	0.130
8	0.104	0.037	0.203	0.323
9	0.076	0.242	0.083	0.022
10	0.081	0.104	0.218	0.292
11	0.035	0.097	0.257	0.082
12	0.097	0.063	0.022	0.100
13	0.041	0.041	0.173	0.113
14	0.116	0.030	0.188	0.094
15	0.103	0.045	0.070	0.143
16	0.186	0.317	0.050	0.230
17	0.106	0.060	0.167	0.098
18	0.119	0.287	0.108	0.112
19	0.026	0.055	0.143	0.061
20	0.110	0.078	0.115	0.134
21	0.111	0.105	0.105	0.196
22	0.075	0.061	0.130	0.156
23	0.028	0.038	0.083	0.056

Table 6. Positioning accuracy of the smartphones in Room 214.

No.	Room 214 (Huawei Honor 10)		Room 214 (Samsung Galaxy S8)	
	Measure Error 1	Measure Error 2	Measure Error 1	Measure Error 2
	MSEP (m)	*MSEP* (m)	*MSEP* (m)	*MSEP* (m)
1	0.071	0.052	0.075	0.065
2	0.017	0.070	0.013	0.068
3	0.072	0.054	0.018	0.059
4	0.062	0.150	0.036	0.071
5	0.025	0.077	0.073	0.035
6	0.030	0.006	0.042	0.123
7	0.047	0.089	0.075	0.067
8	0.099	0.183	0.042	0.079
9	0.086	0.076	0.070	0.076
10	0.149	0.113	0.035	0.100
11	0.062	0.083	0.022	0.042
12	0.117	0.076	0.162	0.046
13	0.063	0.070	0.091	0.087
14	0.083	0.022	0.057	0.165
15	0.070	0.028	0.154	0.183
16	0.059	0.053	0.107	0.031
17	0.146	0.087	0.051	0.087
18	0.103	0.067	0.014	0.076
19	0.093	0.075	0.104	0.185
20	0.115	0.023	0.115	0.219

Table 7. Positioning accuracy comparison using images from a real conference room.

Our Method's Measure Error (m)				Baseline Method's Measure Error (m)			
No.	*MSEP*	No.	*MSEP*	No.	*MSEP*	No.	*MSEP*
1	0.136	17	0.070	1	0.271	17	0.263
2	0.101	18	0.085	2	0.289	18	0.370
3	0.059	19	0.088	3	0.234	19	0.294
4	0.178	20	0.210	4	0.255	20	0.355
5	0.089	21	0.188	5	0.393	21	0.110
6	0.112	22	0.160	6	0.303	22	0.271
7	0.152	23	0.053	7	0.285	23	0.275
8	0.122	24	0.142	8	0.141	24	0.381
9	0.127	25	0.104	9	0.287	25	0.402
10	0.147	26	0.078	10	0.266	26	0.343
11	0.056	27	0.139	11	0.317	27	0.376
12	0.257	28	0.147	12	0.289	28	0.228
13	0.119	29	0.095	13	0.215	29	0.249
14	0.121	30	0.204	14	0.360	30	0.264
15	0.124	31	0.115	15	Failure	31	0.339
16	0.078	32	0.085	16	Failure	32	0.302

It can be seen that the accuracy of our method was higher than that of the baseline method. In addition, there were two image-localization failures in the comparison method. This shows the effectiveness and robustness of our method.

In Figure 18, we can more easily see the accuracy difference between the two positioning methods. In the RMSE metric, the proposed method was 0.132 m, and the baseline method was 0.289 m. In addition, we added people to the positioning image, and the presence of people in the image made the captured positioning image more consistent with the real indoor conference scene. As shown in

Figure 19, there were four control group images. Through this undesired occlusion phenomenon, we can evaluate the stability of local feature matching in actual image-based positioning.

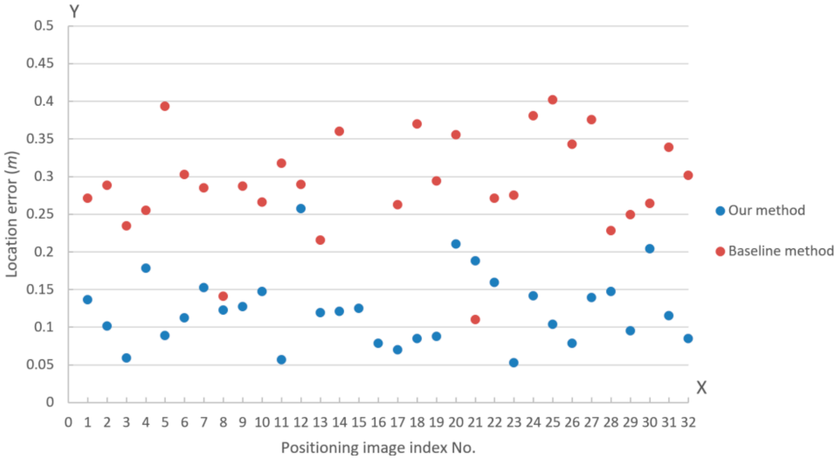


Figure 18. Scatter plot of the MSEP distribution.

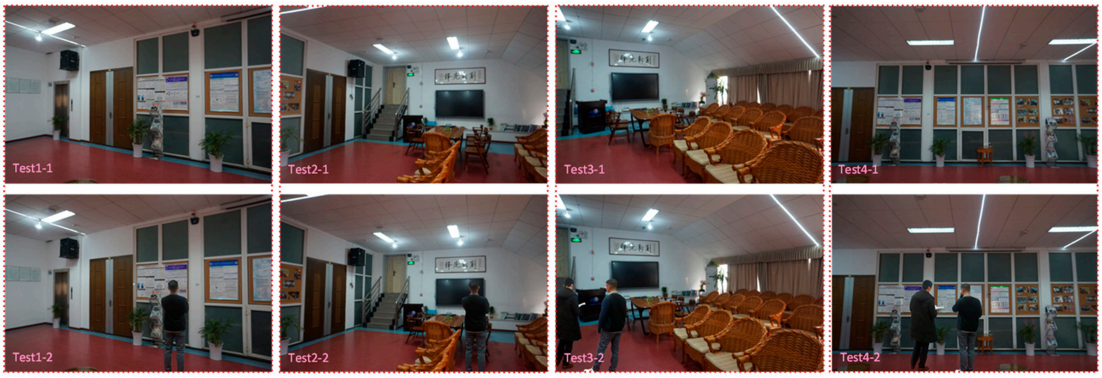


Figure 19. Four control group positioning images.

Table 8 shows the positioning error results calculated by two comparison methods for four groups of control data. We can see that the positioning accuracy changed. In terms of the RMSE metric, after adding new occlusions, the overall positioning accuracy of the two methods was reduced. The RMSE values of our method were 0.084 and 0.135 m before and after the occlusion was added. The RMSE values of the Baseline method were 0.309 and 0.323 m before and after the occlusion was added.

Table 8. Comparison of the positioning accuracy of different image content.

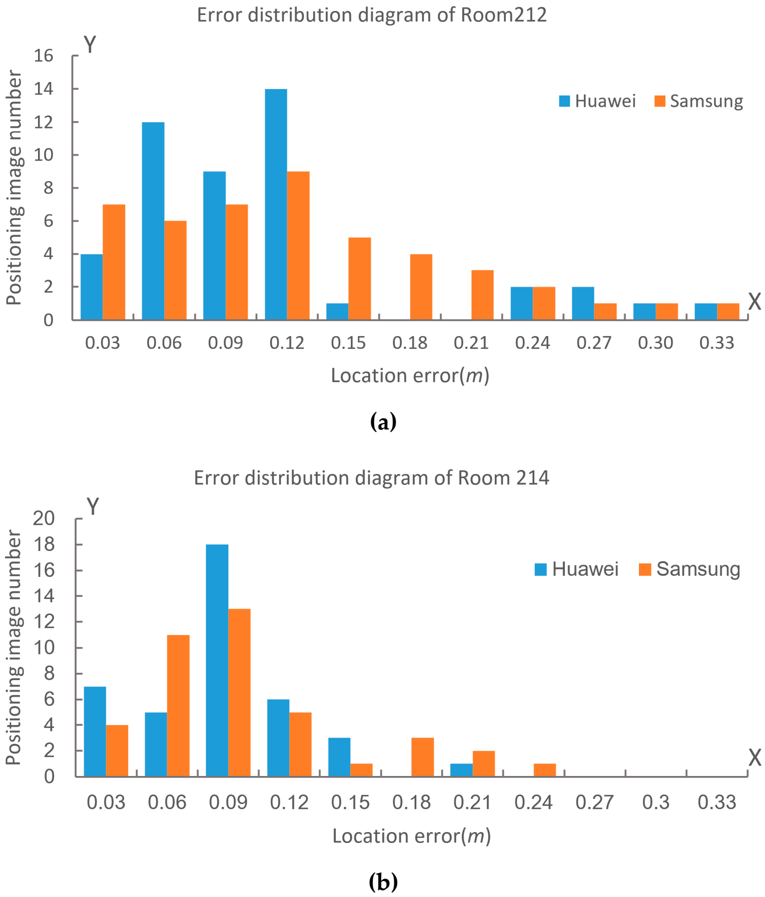
No.	Our Method (m)			Baseline Method (m)		
	ΔX	ΔY	*MSEP*	ΔX	ΔY	*MSEP*
1-1	0.025	0.098	0.101	0.253	−0.082	0.266
1-2	0.026	0.110	0.113	0.284	−0.125	0.310
2-1	0.081	0.037	0.089	−0.394	−0.080	0.402
2-2	0.038	0.168	0.172	−0.382	−0.107	0.397
3-1	0.084	−0.023	0.087	−0.218	0.176	0.280
3-1	0.034	−0.160	0.164	−0.257	0.095	0.274
4-1	−0.002	0.048	0.048	0.013	−0.268	0.268
4-2	−0.046	0.044	0.064	−0.023	0.298	0.299

5. Discussion

In the method that relies on the 3D point cloud, the accuracy of the positioning feature database has a great influence on the positioning result on the smartphones. Moreover, indoor environments are challenging for visual positioning because there are repetitive/similar texture, weak texture, or textureless regions. In the weak texture or textureless scenes, there is no result, or results are inaccurate. In the repetitive texture scene, because there are repetitive features, the number of overlap images is a key. Although the positioning accuracy and stability of the proposed work are proved in different experimental scenes, it is a prototype system for smartphone indoor visual positioning. Specifically, the processing of positioning feature database is offline. In this part, based on the existing classical matching algorithms and strategies, our main aim is to add an epipolar constraint based on the fundamental matrix and a matching image-screening strategy based on image overlap during construction of the positioning feature database, which is conducive to help reduce noise points in the feature point cloud. Matching images with the feature point cloud instead of database images improves the efficiency of the localization procedure [22]. In online Smartphone Indoor Visual Positioning, a strategy of Kd-Tree+BBF ensures the retrieval efficiency of the positioning image features and the PROSAC algorithm is used instead of the RANSAC algorithm for matching optimization. In addition, the final matching points is generated by our proposed novel mismatched elimination method based on HTVI, thus improved the inlier ratio, time cost, and matching point distribution. We can easily see these changes in Figures 12–14. At the same time, it can be easily found from Table 1 that the robustness and accuracy of the positioning were significantly improved after using the matching error elimination based on HTVI.

From the experimental results accuracy evaluation and analysis of decorated indoor scene, the position error is not uniform distributed, as shown in Figure 20. The X-axis in the figures is the location error in the range of, for example, 0–0.03 m, 0.03–0.06 m, 0.06–0.09 m, 0.09–0.12 m, and so on. The Y-axis is the positioning image number. In Figure 20a, the location error distribution is divergent. There exist large errors, such as 0.3 and 0.33 m, but the location error of 80.4 percent of all positioning images in the proposed method is smaller than 0.15 m. Among them, the location error of 73.9 percent of the Samsung smartphone's positioning images in the proposed method was smaller than 0.15 m, the location error of 87 percent of the Huawei smartphone's positioning images in the proposed method was smaller than 0.15 m, and the divergent errors contributed to a large RMSE. In Figure 20b, the location error of most positioning images was small. The location error of 91.3 percent of all positioning images in the proposed method was smaller than 0.15 m. Among them, the Samsung smartphone had 85 percent and the Huawei smartphone had 97.5 percent. The max error was only 0.219 m.

From the comparison of different experimental environments and conditions in decorated indoor scenes, we further found that the positioning error of the Huawei smartphone is smaller than that of the Samsung in the same experimental scene, and the positioning error is also significantly lower in the experimental scene with richer texture. In the experiments of these two scenes, the location method and the image of establishing the positioning feature library are the same, and all the images were taken by following the same rules. The difference is that the positioning image uses from two different smartphone cameras, and the texture in the two scenes is different. The Huawei smartphone images not only have a higher resolution, but there are more feature points detected in its image; and the Huawei smartphone image scale is closer to that of the database images. Thus, better camera resolution and richer texture can get better positioning accuracy. This is also the reason why the location error can be significantly different in different situations when using the method proposed in this paper. In the experiment, the number of database images in Rooms 212 and 214 were 149 and 151, respectively. The size of the positioning feature datasets generated using the images of the two rooms are all about 30 MB. If the number of dataset images is larger, more time is needed for positioning. To reduce the computational time, a coarse position can be useful when only the adjacent images are compared in similar features lookup; alternatively, GPU acceleration can be used on the smartphone side. Using down-sampled images can also improve the computational efficiency in engineering.



(a)

(b)

Figure 20. The location error distribution in different Rooms: (**a**) Room 212 and (**b**) Room 214.

From the experimental results' accuracy evaluation and analysis of real conference scene, it is easy to find that our method is more accurate and has a higher success rate than the baseline method. Moreover, after the control experiment of occlusion, although the positioning accuracy of both methods reduced, our method is still better than the baseline method. For further analysis, we compare the changes of the number of inliers used to calculate the camera pose by different methods in Table 9. It can be seen that the new matching point error elimination algorithm proposed by us played an important role. In all comparative experiments, although our method finally obtained fewer inliers, it has a better correct inlier rate, which helped to obtain more accurate positioning results. At the same time, we also found that the matching points obtained by the two methods had basically decreased after adding people occlusions; this is caused by occlusion. These results verify the effectiveness and robustness of the method and strategy proposed in this paper.

Table 9. Inliers comparison of smartphone camera-pose calculation.

Our Method				**Baseline Method Inliers**			
No.	**Inliers**	**No.**	**Inliers**	**No.**	**Inliers**	**No.**	**Inliers**
1-1	112	2-1	135	1-1	125	2-1	142
1-2	106	2-2	112	1-2	119	2-2	142
3-1	73	4-1	148	3-1	95	4-1	158
3-2	65	4-2	134	3-2	91	4-2	154

It should be noted that the proposed method is only a prototype system for smartphone indoor visual positioning. When we transmit the positioning information through the 4G network for server-side positioning solution after the positioning image is captured, the total positioning time is

2–5 seconds. In some cases, it can reach eight seconds. When we download the positioning feature database to the smartphones, the entire positioning calculation is completed on the smartphone. After the positioning image is taken, the total positioning time is between 0.3 and 1 second. The time difference between the two positioning modes is mainly due to the server-side positioning time being heavily dependent on the efficiency of the 4G network in transmitting the positioning information.

6. Conclusions

In this paper, an efficient automatic smartphone indoor visual positioning method was proposed, using local feature matching, which uses images with known intrinsic and extrinsic parameters to locate smartphones indoors. For the establishment of a precise positioning feature database, the proposed method uses a modified and extended high-precision SURF feature matching strategy and the multi-image spatial forward intersection to obtain a point cloud. For online smartphone indoor visual positioning, a robust and efficient similarity feature retrieval method was proposed, in which a more reliable and correct matching point pair set is obtained through the use of a novel matching error elimination technology based on Hough transform voting. Finally, an online indoor visual positioning experiment for smartphones was realized by the fast and stable camera-pose estimation algorithm in this paper. In decorated experimental scenes, the results show that 88.6 percent of the positioning images achieved location errors smaller than 0.15 m—there were only two positioning images with location errors exceeding 0.3 m—proving that the proposed method can achieve a precise positioning effect. Even in the more challenging scene of Room 212, 73.9 percent of the Samsung smartphone positioning images and 87 percent of the Huawei smartphone positioning images achieved location errors smaller than 0.15 m. For the real experimental scenes, the results show that the positioning accuracy of our method was more than double that of the comparison method. In terms of the RMSE metric, the overall positioning accuracy was still better than 15 cm. In addition, the success rate of our method was better than the baseline method. These all confirm the effectiveness and robustness of the proposed method.

Of course, the proposed method has some limitations. The object 3D points are obtained from feature point matching, as well as the relationships between the positioning image points and the object 3D points. Although much effort has been put into accurate and reliable image feature point matching to ensure that camera pose estimation is less affected by mismatched points, which has a good effect when we have similar indoor textures and small illumination and perspective changes. Thus, in weak or invalid texture regions, there will be either no result or inaccurate positioning results. Furthermore, when the interior decoration and furnishings change greatly, we need to update the location feature library in a timely manner; otherwise, the location will fail or the location accuracy will be poor. In future research, it is worth our consideration and exploration to improve the positioning accuracy and success rate by using more stable line features from indoor textures or indoor building frame structure information, which rarely changes.

Author Contributions: M.L., R.C., W.Z., and B.G. proposed the methodology; M.L. wrote the paper; X.L., W.Z., and M.L. conceived of and designed the experiments; and M.L., X.L., and G.G. performed the experiments. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key R&D Program of China, grant numbers 2016YFB0502200 and 2018YFB0505400; the National Natural Science Foundation of China (NSFC), grant number 41901407; the Fundamental Research Funds for the Central universities, grant number 2042018kf0012; and the China Postdoctoral Science Foundation, grant number 2017M622520.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Liang, J.; Corso, N.; Turner, E.; Zakhor, A. Image based localization in indoor environments. In Proceedings of the Fourth International Conference on Computing for Geospatial Research and Application, San Jose, CA, USA, 22–24 July 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 70–75.

2. Wu, T.; Liu, J.; Li, Z.; Liu, K.; Xu, B. Accurate Smartphone Indoor Visual Positioning Based on a High-Precision 3D Photorealistic Map. *Sensors* **2018**, *18*, 1974. [CrossRef] [PubMed]
3. Liao, X.; Chen, R.; Li, M.; Guo, B.; Niu, X.; Zhang, W. Design of a Smartphone Indoor Positioning Dynamic Ground Truth Reference System Using Robust Visual Encoded Targets. *Sensors* **2019**, *19*, 1261. [CrossRef] [PubMed]
4. Huitl, R.; Schroth, G.; Hilsenbeck, S.; Schweiger, F.; Steinbach, E. TUMindoor: An extensive image and point cloud dataset for visual indoor localization and mapping. In Proceedings of the 19th IEEE International Conference on Image Processing, Lake Buena Vista, FL, USA, 30 September–3 October 2012.
5. Acharya, D.; Ramezani, M.; Khoshelham, K.; Winter, S. BIM-Tracker: A model-based visual tracking approach for indoor localisation using a 3D building model. *ISPRS J. Photogramm. Remote Sens.* **2019**, *150*, 157–171. [CrossRef]
6. Liao, X.; Li, M.; Chen, R.; Guo, B.; Wang, X. An Image-based Visual Localization Approach to Urban Space. In Proceedings of the 2018 Ubiquitous Positioning, Indoor Navigation and Location-Based Services (UPINLBS), Wuhan, China, 22–23 March 2018; pp. 1–5.
7. Alfian, G.; Syafrudin, M.; Ijaz, M.F.; Syaekhoni, M.A.; Fitriyani, N.L.; Rhee, J. A Personalized Healthcare Monitoring System for Diabetic Patients by Utilizing BLE-Based Sensors and Real-Time Data Processing. *Sensors* **2018**, *18*, 2183. [CrossRef]
8. Xia, S.; Liu, Y.; Yuan, G.; Zhu, M.; Wang, Z. Indoor fingerprint positioning based on wi-fi: An overview. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 135. [CrossRef]
9. Monica, S.; Ferrari, G. UWB-based localization in large indoor scenarios: Optimized placement of anchor nodes. *IEEE Aerosp. Electron. Syst. Mag.* **2015**, *51*, 987–999. [CrossRef]
10. Liu, F.; Li, X.; Wang, J.; Zhang, J. An Adaptive UWB/MEMS-IMU Complementary Kalman Filter for Indoor Location in NLOS Environment. *Remote Sens.* **2019**, *11*, 2628. [CrossRef]
11. Pratama, A.R.; Widyawan; Hidayat, R. Smartphone-based Pedestrian Dead Reckoning as an indoor positioning system. In Proceedings of the International Conference on System Engineering and Technology, Bandung, Indonesia, 11–12 September 2012.
12. Zhuang, Y.; El-Sheimy, N. Tightly-Coupled Integration of WiFi and MEMS Sensors on Handheld Devices for Indoor Pedestrian Navigation. *IEEE Sens. J.* **2015**, *16*, 224–234. [CrossRef]
13. Kuang, J.; Niu, X.; Chen, X. Robust Pedestrian Dead Reckoning Based on MEMS-IMU for Smartphones. *Sensors* **2018**, *18*, 1391. [CrossRef]
14. Lu, Y.H.; Delp, E.J. An overview of problems in image-based location awareness and navigation. In Proceedings of the Visual Communications and Image Processing, San Jose, CA, USA, 18 January 2004.
15. Zhang, W.; Kosecka, J. Image Based Localization in Urban Environments. In Proceedings of the International Symposium on 3D Data Processing, Chapel Hill, NC, USA, 14–16 June 2006; pp. 33–40.
16. Li, L.; Yu, H. Improved SIFT performance evaluation against various image deformations. In Proceedings of the IEEE Information Technology and Artificial Intelligence Conference, Liverpoo, UK, 26–28 October 2015; pp. 172–175.
17. Zhang, C.; Wang, X.; Guo, B. Space Location of Image in Urban Environments Based on C/S Structure. *Geomat. Inf. Sci. Wuhan Univ.* **2018**, *43*, 978–983.
18. Wang, J.; Zha, H.; Cipolla, R. Coarse-to-fine vision-based localization by indexing scale-Invariant features. *IEEE Trans. Syst. Man Cybern. Part B* **2006**, *36*, 413–422. [CrossRef] [PubMed]
19. Walch, F.; Hazirbas, C.; Leal, L.; Sattler, T.; Hilsenbeck, S.; Cremers, D. Image-based localization using LSTMs for structured feature correlation. In Proceedings of the IEEE International Conference on Computer Vision, Las Vegas, NV, USA, 27–30 June 2016; pp. 627–637.
20. Kendall, A.; Grimes, M.; Cipolla, R. PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2938–2946.
21. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the IEEE International Conference on Computer Vision, Providence, RI, USA, 16–21 June 2012; pp. 2564–2571.
22. Sattler, T.; Leibe, B.; Kobbelt, L. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1744–1756. [PubMed]

23. Bay, H.; Tuytelaars, T.; Gool, L.V. SURF: Speeded up robust features. In Proceedings of the European Conference on Computer Vision, Graz, Austria, 7–13 May 2006; pp. 404–417.
24. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
25. Zhang, H.; Berg, A.C.; Maire, M.; Malik, J. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), New York, NY, USA, 17–22 June 2006; pp. 2126–2136.
26. Feng, G.; Ma, L.; Tan, X.; Qin, D. Drift-Aware Monocular Localization Based on a Pre-Constructed Dense 3D Map in Indoor Environments. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 299. [CrossRef]
27. Xu, Z.; Wu, L.; Liu, J.; Shen, Y.; Li, F.; Wang, R. Modification of SFM Algorithm Referring to Image Topology and Its Application in 3-Dimension Reconstruction of Disaster Area. *Geomat. Inf. Sci. Wuhan Univ.* **2015**, *40*, 599–606.
28. Wu, X.; Zhang, Y.; Zhao, L.; Yu, Y.; Wang, T.; Li, L. Comparison of the Accuracy of Incremental SFM with POS-aid Bundle Adjustment. *Acta Geod. Et Cartogr. Sinia* **2017**, *46*, 198–207.
29. Heller, J.; Havlena, M.; Jancosek, M.; Torii, A. 3D reconstruction from photographs by CMP SfM web service. In Proceedings of the 14th IAPR International Conference on Machine Vision Applications, Tokyo, Japan, 18–22 May 2018; pp. 30–34.
30. Davison, A.; Reid, I.; Molton, N.; Stasse, O. Monoslam: Real-time single camera slam. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 1052–1067. [CrossRef]
31. Klein, G.; Murray, D. Parallel Tracking and Mapping for Small AR Workspaces. In Proceedings of the IEEE and ACM International Symposium on Mixed and Augmented Reality, Cambridge, UK, 15–18 September 2008; pp. 1–10.
32. Mur-Artal, R.; Montiel, J.; Tardós, J. Orb-slam: A versatile and accurate monocular slam system. *IEEE Trans. Robot.* **2017**, *31*, 1147–1163. [CrossRef]
33. Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-Scale Direct Monocular SLAM. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 834–849.
34. Forster, C.; Pizzoli, M.; Scaramuzza, D. SVO: Fast semi-direct monocular visual odometry. In Proceedings of the IEEE International Conference on Robotics and Automation, Hongkong, China, 31 May–5 June 2014; pp. 15–22.
35. Newcombe, R.; Izadi, S.; Hilliges, O.; Molyneaux, D.; Kim, D.; Davison, A. Kinect Fusion: Real-time dense surface mapping and tracking. In Proceedings of the IEEE International Symposium on Mixed and Augmented Reality, Atlanta, GA, USA, 5–8 November 2012; pp. 127–136.
36. Audras, C.; Comport, A.; Meilland, M.; Rives, P. Real-time dense appearance-based slam for RGB-D sensors. In Proceedings of the Australasian Conference on Robotics and Automation, Melbourne, Australia, 7–9 December 2011.
37. Endres, F.; Hess, J.; Engelhard, N.; Sturm, J.; Cremers, D.; Burgard, W. An evaluation of the RGB-D SLAM system. In Proceedings of the IEEE International Conference on Robotics and Automation, St Paul, MN, USA, 14–19 May 2012; pp. 1691–1696.
38. Matthias, N.; Izadi, S.; Stamminger, M. Real-time 3D reconstruction at scale using voxel hashing. *ACM Trans. Graph.* **2013**, *32*, 1–11.
39. Kerl, C.; Sturm, J.; Cremers, D. Robust odometry estimation for RGB-D cameras. In Proceedings of the IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 3748–3754.
40. Vestena, K.; Dos Santos, D.; Oilveira, E.; Pavan, N.; Khoshelham, K. A weighted closed-form solution for Rgb-D data registration. *ISPRS-Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *XLI-B3*, 403–409. [CrossRef]
41. Qin, J.; Li, M.; Liao, X.; Zhong, J. Accumulative Errors Optimization for Visual Odometry of ORB-SLAM2 Based on RGB-D Cameras. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 581. [CrossRef]
42. Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436. [CrossRef] [PubMed]
43. Kendall, A.; Cipolla, R. Modelling Uncertainty in Deep Learning for Camera Relocalization. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 4762–4769.
44. Ye, F.; Su, Y.; Xiao, H. Remote Sensing Image Registration Using Convolutional Neural Network Features. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 232–236. [CrossRef]

45. García-Olalla, O.; Alegre, E.; Fernández-Robles, L.; Fidalgo, E.; Saikia, S. Textile Retrieval Based on Image Content from CDC and Webcam Cameras in Indoor Environments. *Sensors* **2018**, *18*, 1329.
46. Zheng, L.; Yang, Y.; Tian, Q. Sift meets cnn: A decade survey of instance retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *99*, 1224–1244. [CrossRef]
47. Acharya, D.; Khoshelham, K.; Winter, S. BIM-PoseNet: Indoor camera localization using a 3D indoor model and deep learning from synthetic images. *ISPRS J. Photogramm. Remote Sens.* **2019**, *150*, 245–258. [CrossRef]
48. Sharma, A.; Paliwal, K. Linear discriminant analysis for the small sample size problem: An overview. *Int. J. Mach. Learn. Cybern.* **2014**, *6*, 443–454. [CrossRef]
49. Agarwal, S.; Furukawa, Y.; Snavely, N.; Simon, I.; Curless, B.; Seitz, M.; Szeliski, R. Building Rome in a day. *Commun. ACM* **2011**, *54*, 105–112. [CrossRef]
50. Batur, A.; Yadikar, N.; Mamat, H.; Aysa, A.; Ubul, K. Complex Uyghur document image matching and retrieval based on modified SURF feature. *CAAI Trans. Intell. Syst.* **2019**, *14*, 296–305.
51. Lun, L. Research on Indoor Positioning Algorithm Based on PROSAC Algorithm. Master's Thesis, Harbin Institute of Technology, Harbin, China, 2018.
52. Hough, P. Method and Means for Recognizing Complex Patterns. U.S. Patent 3,069,654, 18 December 1962.
53. Duda, R.; Hart, H. Use of the Hough transformation to detect lines and curves in pictures. *Commun. ACM* **1972**, *15*, 11–15. [CrossRef]