remote sensing

MDPI

Article

A High-Accuracy Indoor-Positioning Method with Automated RGB-D Image Database Construction

Runzhi Wang [1,2], Wenhui Wan [1,*], Kaichang Di [1], Ruilin Chen [1] and Xiaoxue Feng [3]

[1] State Key Laboratory of Remote Sensing Science, Aerospace Information Research Institute, Chinese Academy of Sciences, No. 20A, Datun Road, Chaoyang District, Beijing 100101, China; wangrz@radi.ac.cn (R.W.); dikc@radi.ac.cn (K.D.); reline.chen@myzygroup.com (R.C.)

[2] College of Resources and Environment, University of Chinese Academy of Sciences, Beijing 100049, China

[3] Institute of Remote Sensing and GIS, School of Earth and Space Sciences, Peking University, Beijing 100871, China; fengxx@pku.edu.cn

* Correspondence: wanwh@radi.ac.cn; Tel.: +86-10-64807987

Received: 27 August 2019; Accepted: 30 October 2019; Published: 1 November 2019

Abstract: High-accuracy indoor positioning is a prerequisite to satisfy the increasing demands of position-based services in complex indoor scenes. Current indoor visual-positioning methods mainly include image retrieval-based methods, visual landmarks-based methods, and learning-based methods. To better overcome the limitations of traditional methods such as them being labor-intensive, of poor accuracy, and time-consuming, this paper proposes a novel indoor-positioning method with automated red, green, blue and depth (RGB-D) image database construction. First, strategies for automated database construction are developed to reduce the workload of manually selecting database images and ensure the requirements of high-accuracy indoor positioning. The database is automatically constructed according to the rules, which is more objective and improves the efficiency of the image-retrieval process. Second, by combining the automated database construction module, convolutional neural network (CNN)-based image-retrieval module, and strict geometric relations-based pose estimation module, we obtain a high-accuracy indoor-positioning system. Furthermore, in order to verify the proposed method, we conducted extensive experiments on the public indoor environment dataset. The detailed experimental results demonstrated the effectiveness and efficiency of our indoor-positioning method.

Keywords: visual positioning; indoor scenes; automated database construction; image retrieval

1. Introduction

Nowadays, position information has become key information in people's daily lives. This has inspired position-based services, which aim to provide personalized services to mobile users whose positions are changing [1]. Therefore, obtaining a precise position is a prerequisite for these services. The most commonly used positioning method in the outdoor environment is the Global Navigation Satellite System (GNSS). In most cases, however, people spend more than 70% of their time indoors [2]. Therefore, accurate indoor positioning has important practical significance. Although GNSS is a good choice for outdoor positioning, due to signal occlusion and attenuations, it is often useless in indoor environments. Thus, positioning people accurately in indoor scenes remains a challenge and it has stimulated a large number of indoor-positioning methods in recent years [3]. Among these methods, fingerprint-based algorithms are widely used. Their fingerprint databases include Wi-Fi [4–8], Bluetooth [9,10], and magnetic field strengths [11,12]. Although these methods are easy to implement, construction of a fingerprint database is usually labor-intensive and time-consuming. Moreover, it is difficult for their results to meet the needs of high-accuracy indoor positioning.

Given that humans use their eyes to see where they are, mobile platforms can also do this with cameras. A number of visual positioning methods have been proposed in recent years. These positioning methods are divided into three categories: image retrieval based methods, visual landmarks-based methods, and learning-based methods.

Image retrieval based methods treat the positioning task as an image retrieval or recognition process [13–15]. They usually have a database that are augmented with geospatial information, and every image in the database is described through the same specific features. These methods perform a first step to retrieve candidate images from the database according to a similarity search, and the coarse position information of the query image is then obtained based on the geospatial information of these candidate images. So the first step, similar image retrieval process, is critical. The brute-force approach, which is a distance comparison between feature descriptor vectors, is often used for similarity search. Some positioning methods based on feature descriptors [16–18] adopt brute-force comparison for the similarity search process of image retrieval. However, it is computationally intensive when the images of a database are described with high-dimensional features, limiting its scope of applications. Azzi et al. [19] use a global feature-based system to reduce the search space and find candidate images in the database, then the local feature scale-invariant feature transform (SIFT) [20] is adopted for points matching in pose estimation. Some researchers try to trade accuracy for rapidity by using approximate nearest neighbor search, such as quantization [21] and vocabulary tree [22]. Another common way to save time and memory of similarity search is principal component analysis (PCA), which has been used to reduce the size of feature vectors and descriptors [23,24]. Some works use correlation algorithms, such as sum of absolute difference (SAD), for computing similarity between query image and database images [25,26]. In recent studies, deep learning-based algorithms are an alternative to aforementioned methods. Razavian et al. [27] use features extracted from a network as an image representation for image retrieval in a diverse set of datasets. Yandex et al. [28] propose a method that aggregates local deep features to product descriptors for image retrieval. After a set of candidate images are retrieved, the position information of the query image is calculated according to the geospatial information of these candidate images through a weighting scheme or linear combination. However, because this position result is not calculated by strict geometric relations, it is rough in most cases and difficult to meet the requirement of high-accuracy positioning.

Visual landmarks-based positioning methods aim to provide a six degrees of freedom (DoF) pose of the query image. Generally, visual landmarks in the indoor environments includes natural landmarks and artificial landmarks. The natural landmarks refer to the geo-tagged 3D database, which is represented by feature descriptors or images with poses. This database could have been built thanks to the mapping module of simultaneous localization and mapping (SLAM) [29,30]. Then the pose of query image is estimated by means of re-localization module and feature correspondence [31–35]. Although the results of these methods are of good accuracy, it takes a long time to match the features of query image with geo-tagged 3D database, especially when the indoor scenes are large. In addition to natural landmarks, there are also positioning methods based on artificial landmarks, e.g., Degol et al. [36] proposed a fiducial marker and detection algorithm. In reference [37], the authors proposed a method to simultaneously solve the problems of positioning from a set of squared planar markers. However, positioning from a planar marker suffers from the ambiguity problem [38]. Since these methods require posting markers in the environments, they are not suitable for places such as shopping malls that maintain a clean appearance.

In addition to the traditional visual-positioning method based on strict geometric relations, with the rapid development of deep learning in recent years scholars have proposed many learning based visual-positioning methods [39–41]. The process of these methods are broken down into two steps: model training and pose prediction. They train models through given images with known pose information, and the indoor environments are expressed as the trained models. The pose of a query image is then regressed through the trained models. Some methods even learn the pose directly [42,43]. These methods, which are based entirely on learning, have better performance in weak-textured indoor scenes, but are less accurate or have lower generalization ability to large indoor environments than traditional visual-positioning methods [44]. Therefore, some methods use trained models to replace modules of traditional visual-positioning methods, such as depth estimation [45–47], loop detection [48], and re-localization [49]. The method proposed by Chen et al. [50] uses a pre-trained network for image recognition. It retrieves two geo-tagged red-green-blue (RGB) images from database, and then use traditional visual positioning method for pose estimation. This method performs well on public dataset, but its database images are hand-picked, which increases the workload of database construction. Moreover, the two retrieved geo-tagged RGB images should have favorable geometric configuration (e.g., sufficient intersection angle) for high-accuracy depth estimation. However, this favorable configuration is not guaranteed by the existing two-image methods. This is a potential disadvantage of these methods. Our RGB-D database method directly provides high accuracy depth information from only one image, this not only ensures high accuracy of positioning, but also improves the efficiency of image retrieval.

To overcome the limitations of the aforementioned methods, in this paper, a high-accuracy indoor visual-positioning method with automated RGB-D image database construction is presented. Firstly, we propose an automated database construction process, making the constructed database more objective than a hand-picked database and thus reducing the workload. The database is automatically constructed according to the rules, which reduces the redundancy of database and improves the efficiency of the image-retrieval process. Secondly, considering the requirement of real-time positioning, we introduce a convolutional neural network (CNN) model for a robust and efficient retrieving candidate images. Thirdly, different from aforementioned image retrieval based positioning methods, we replace rough combination of geospatial information with strict geometric relations to calculate the position of query image for high-accuracy positioning. Finally and most importantly, by combining the above three components into a complete indoor-positioning method, we obtain high-accuracy results in an indoor environment and the whole process is time efficient.

2. Methodology

In this section, the proposed indoor-positioning method consists of three major components: (1) RGB-D indoor-positioning database construction; (2) image retrieval based on the CNN feature vector; (3) position and attitude estimation. Detailed processes in each component are given in the following sub-sections.

2.1. RGB-D Indoor-Positioning Database Construction

In the proposed indoor visual positioning method, RGB-D images are used to build positioning database in an indoor scene. Since most RGB-D image acquisition devices, such as Microsoft Kinect sensor, can provide a frame rate of 30 Hz, images acquired over a period of time have redundancies. Note that a large number of database images need a lot of memory in storage, it takes longer for the image retrieval and positioning. However, if the images in the database are too sparse, it may not achieve high positioning accuracy. In order to meet the requirements of precise and real-time indoor positioning, an automated RGB-D image database construction process is proposed.

Our strategy for RGB-D image database construction is based on the relationships between pose error (i.e., position error and attitude error), number of matching points and pose difference (i.e., position difference and attitude difference). To determine their relationships, we selected several RGB-D image as the database images and more than 1000 other RGB-D images as the query images. These images all come from the Technical University of Munich (TUM) RGB-D dataset [51], which provides ground truth of pose. Figure 1a,b show an example of RGB-D images. The positions of database images and ground truth of trajectory are shown in Figure 1c.

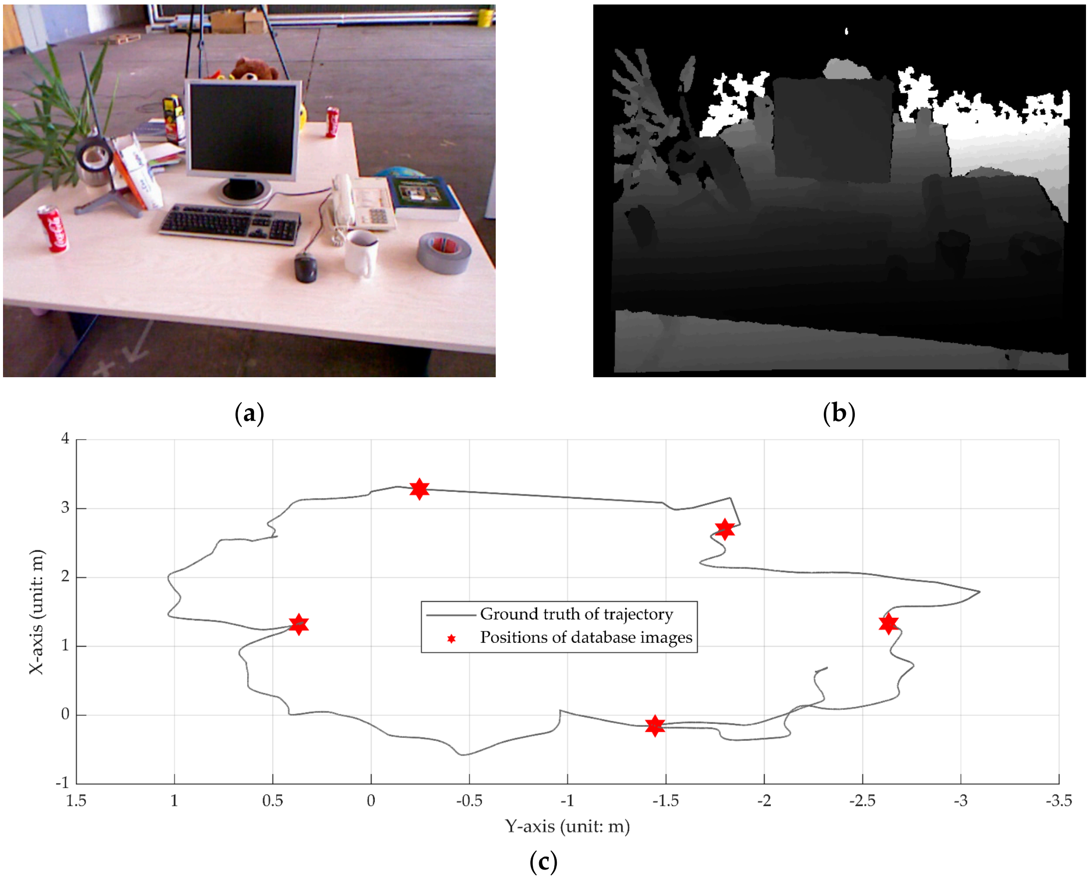
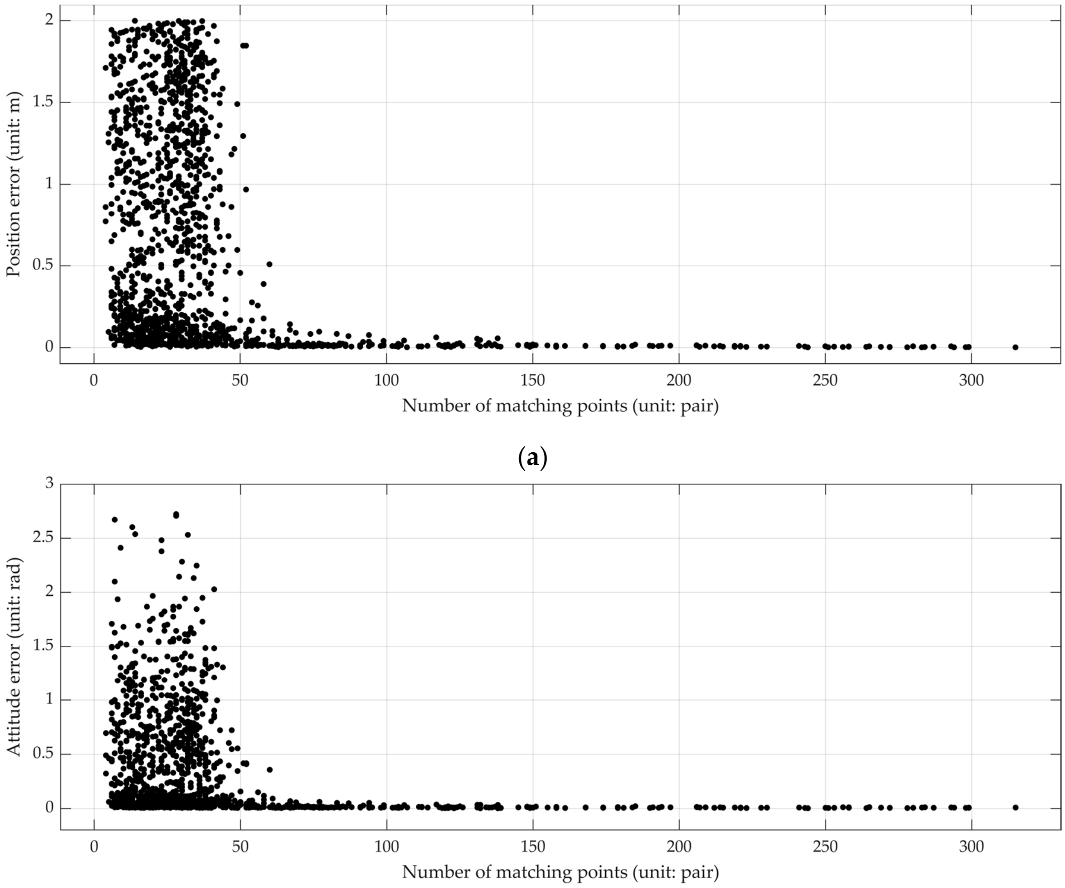


Figure 1. An example of RGB-D image and the positions of database images. (**a**) RGB image; (**b**) corresponding depth image of this RGB image; (**c**) positions of database images (shown by red hexagon) and ground truth of trajectory (shown by gray line).

First, the relationship between pose error and number of matching points is a key criterion of the proposed process. The pose of each query image was calculated by means of the visual-positioning process mentioned in Section 2.3. The number of matching points was recorded in this positioning process. The pose error was obtained by comparing the calculated pose with its ground truth. After testing all the query images, pose errors and corresponding number of matching points for each query image were collected and analyzed to determine their relationship (Figure 2). It is found from Figure 2a,b that both the position error and attitude error fluctuate greatly when the number of matching points is less than 50. However, when the number of matching points is more than 50, the pose errors are basically stable at a small value. In other words, our visual-positioning process can obtain precise and stable results when the number of matching points is more than 50. So we set 50 as the threshold T_{match} for the minimum number of matching points.



(**a**)

(**b**)

Figure 2. Relationship between pose error and corresponding number of matching points. (**a**) Position error vs. number of matching points; (**b**) attitude error vs. number of matching points.

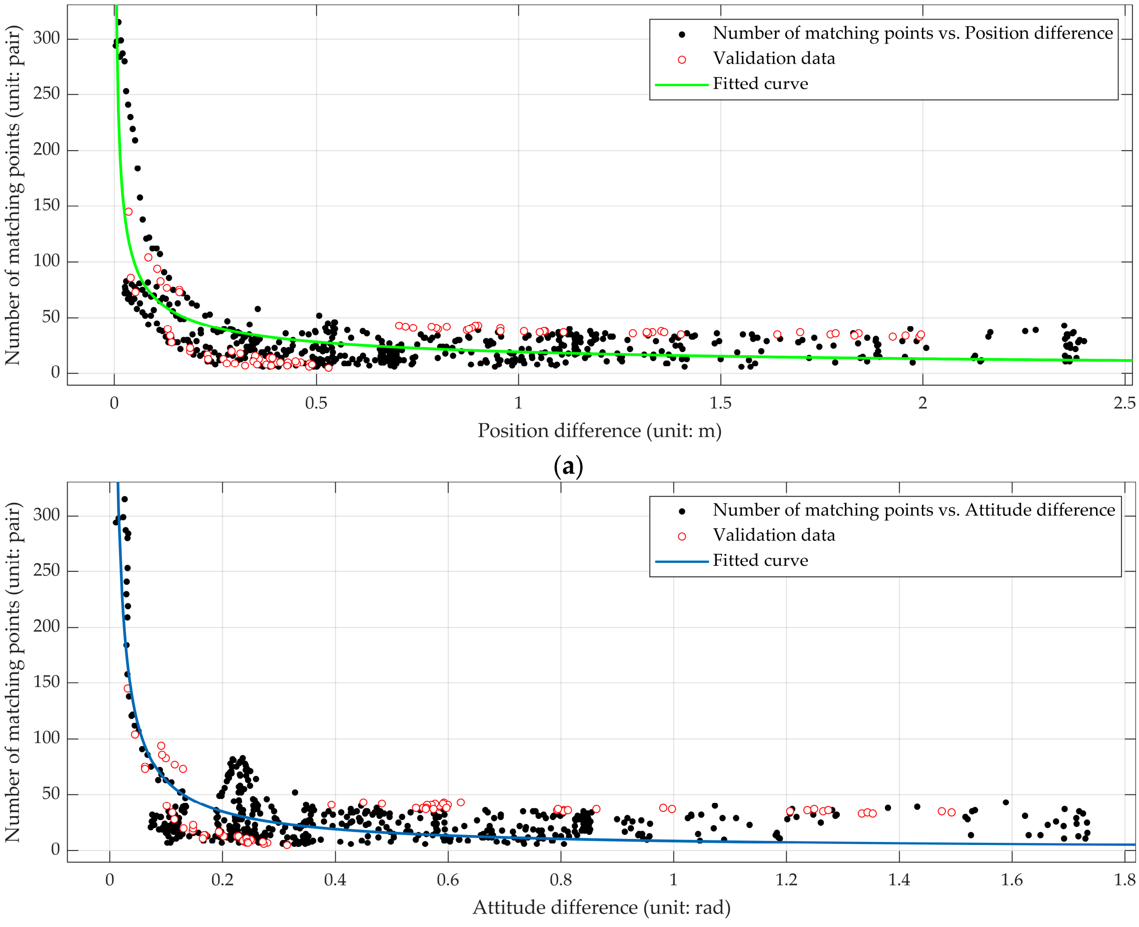
Second, the relationship between number of matching points and pose difference is also an important criterion in the RGB-D image database construction process. The pose difference was calculated by comparing the ground truth pose of each query image with corresponding database image. Then the pose differences of some images were combined with their number of matching points to fit their relationship. The green fitted curve in Figure 3a shows the fitted relationship between the number of matching points and position difference. Its expression is described as follows:

$$f_p(x) = 19.43 \times {x_p}^{-0.5435}. \tag{1}$$

Here x_p is position difference, $f_p(x)$ is the number of matching points. The blue fitted curve in Figure 3b shows the fitted relationship between the number of matching points and attitude difference. Its expression is described as Equation (2):

$$f_a(x) = 8.846 \times {x_a}^{-0.8503}. \tag{2}$$

Here x_a is attitude difference, $f_a(x)$ is the number of matching points. Then we used some other pose differences and number of matching points of more than seventy images to validate Equations (1) and (2). As shown in Figure 3a,b, the validation data are distributed near the fitted curve. The root mean square errors (RMSE) of fit and validation are shown in Table 1.



(**a**)

(**b**)

Figure 3. Relationship between number of matching points and corresponding pose difference. (**a**) Fitted curve of number of matching points vs. position difference; (**b**) fitted curve of number of matching points vs. attitude difference.

Table 1. The root mean square errors (RMSEs) of fitted curves and validation data in Figure 3.

Fitted Curves	RMSE of Fit	RMSE of Validation
Figure 3a	23.73	21.88
Figure 3b	23.96	24.08

We can see that the RMSE of validation data is close to the RMSE of fitted curve, which indicates that the fitted curves described in Equations (1) and (2) are applicable to different image data in the same scene. The empirical models are not sensitive to the selection of the query image, the established relationships are reliable to apply in the same scene.

From the trends of the fitted curves in Figure 3, the number of matched points decreases as both of the position difference and attitude difference increase. According to the threshold T_{match} for the number of matching points from Figure 2, the threshold of position difference $T_{\Delta position}$ (i.e., the x_p in Equation (1)), and the threshold of attitude difference $T_{\Delta attitude}$ (i.e., the x_a in Equation (2)), were obtained by substituting $f_p(x)$ and $f_a(x)$ with T_{match}. The results are as follows:

$$\begin{cases} T_{\Delta position} = 0.1757, \text{ unit} : \text{m} \\ T_{\Delta attitude} = 0.1304, \text{ unit} : \text{rad} \end{cases} . \quad (3)$$

Based on these three thresholds T_{match}, $T_{\Delta position}$ and $T_{\Delta attitude}$, the RGB-D image database construction process was proposed for indoor visual positioning (Figure 4).

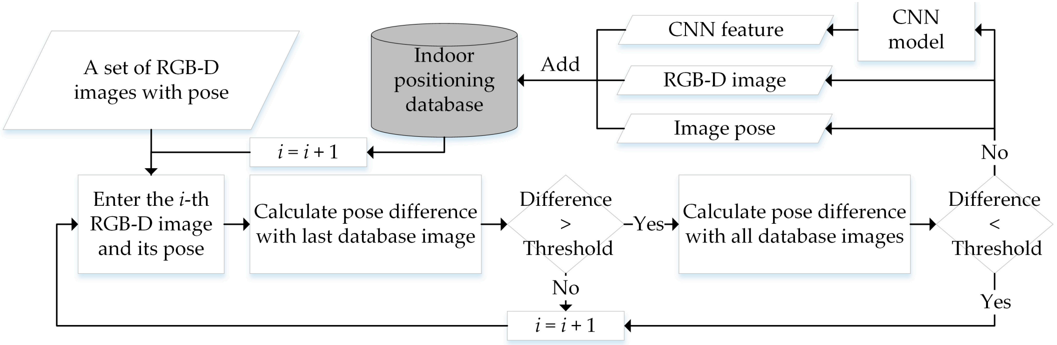

A set of RGB-D images with pose
Indoor positioning database
Add
CNN feature
CNN model
RGB-D image
Image pose
i = i + 1
No
Enter the i-th RGB-D image and its pose
Calculate pose difference with last database image
Difference > Threshold
Yes
Calculate pose difference with all database images
Difference < Threshold
No
i = i + 1
Yes

Figure 4. Flowchart of the proposed indoor positioning database construction process.

As shown in Figure 4, first we need to input a set of RGB-D images with known poses in the scene where conducting indoor positioning. The RGB-D images can be captured using a Microsoft Kinect sensor and the ground truth trajectory of camera pose can be obtained from a high-accuracy motion-capture system with high-speed tracking cameras as in reference [51]. In the database construction process, the first RGB-D image is considered as a database image and the other images will be input successively to determine whether they are required to join the database. From Figure 4, the *i*-th RGB-D image is compared with the last database image joining the database and then compared with all the existing database images to calculate the differences in position and attitude. If the differences meet the preset threshold condition (i.e., $T_{\Delta position}$ and $T_{\Delta attitude}$), the *i*-th image will be determined as a new database image. It will be also input into the CNN model mentioned in Section 2.2 to calculate its CNN feature vector for subsequent step of image retrieval. Finally, we add the three components of the eligible image into the indoor-positioning database, and then move on to the next RGB-D image until all images are judged.

The three components of the indoor positioning database $B = \{I, Pose, F\}$ are listed as follows. The first one is the RGB-D database images $I = \{RGB\text{-}D_1, \ldots, RGB\text{-}D_i, \ldots, RGB\text{-}D_n\}$ that meet the requirements. The second one is the corresponding pose information $Pose = \{Pose_1, \ldots, Pose_i, \ldots, Pose_n\}$ of database images. The $Pose_i$ here includes 3D position $\{x_i, y_i, z_i\}$ and quaternion form of attitude $\left\{qx_i, qy_i, qz_i, qw_i\right\}$. The last one is the CNN feature vector set $F = \{F_1, \ldots, F_i, \ldots, F_n\}$ of database images.

2.2. Image Retrieval Based on Convolutional Neural Network (CNN) Feature Vector

After building the indoor positioning database, it is important to know which RGB-D image in the database is the most similar to the input query RGB image acquiring by the mobile platform. The query RGB image and its most similar RGB-D image will be combined for conducting the subsequent visual positioning. In this sub-section, the CNN model and CNN feature vector-based image-retrieval algorithm were used in our indoor positioning method. We adopted the CNN architecture proposed in reference [52], the main component of which is a generalized vector of locally aggregated descriptors (NetVLAD) layer and it is readily pluggable into standard CNN architecture. The best performing network they trained was adopted to extract image deep features, i.e., CNN feature vector, for image retrieval in this study.

Figure 5 shows the process of image retrieval based on CNN feature vector. With this process, the RGB-D database image, which is the most similar to the input query image, and its pose information were retrieved. First, in Section 2.1, we have calculated and saved the database CNN feature vector set $F = \{F_1, \ldots, F_i, \ldots, F_n\}$ in the indoor positioning database. When a query color image C_q with the same size as the database images is input, the same CNN model is used to calculate its CNN feature vector. This output CNN feature vector F_q of query image has the same length with the feature vector F_i of the database image. In this research, the size of CNN feature vector is 4096×1.Then the distance between

F_q and each feature vector of $F = \{F_1, \ldots, F_i, \ldots, F_n\}$ is calculated to represent their similarity, which is defined as follows:

$$D_{iq} = (F_i - F_q)^{\mathrm{T}} \cdot (F_i - F_q). \quad (4)$$

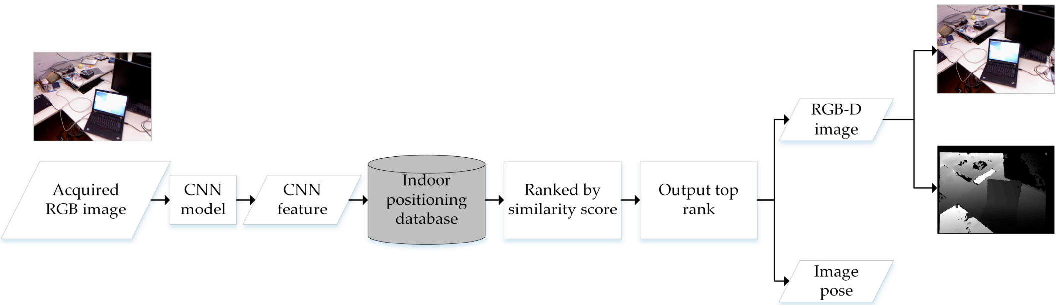


Figure 5. Process of the convolutional neural network (CNN) feature-based image retrieval.

The set of distances is $D = \left\{D_{1q}, \ldots, D_{iq}, \ldots, D_{nq}\right\}$. Finally, we output a retrieved RGB-D database image $RGB\text{-}D_r$ and its pose information $Pose_r$, which has the minimum distance D_{rq} with the query color image.

2.3. Position and Attitude Estimation

After retrieving the most similar RGB-D database image with an acquired query RGB image, the visual positioning was achieved by estimation of the position and attitude of the query image based on the retrieved database image and its pose information (Figure 6).

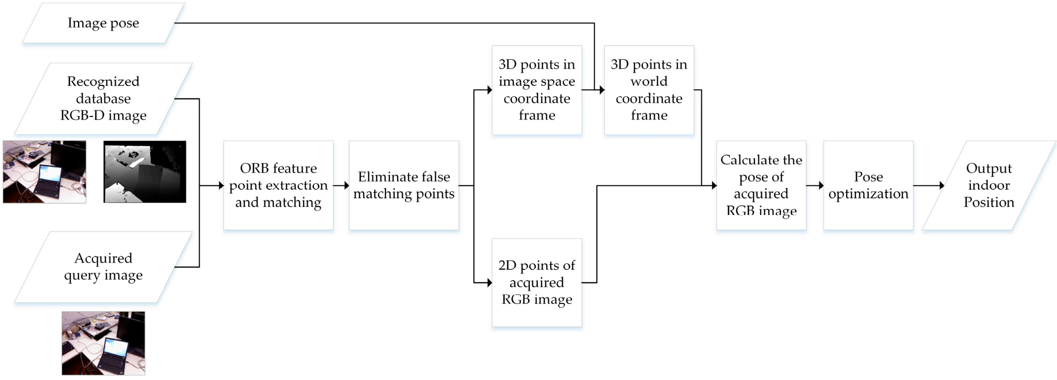


Image pose
Recognized database RGB-D image
Acquired query image
ORB feature point extraction and matching
Eliminate false matching points
3D points in image space coordinate frame
3D points in world coordinate frame
2D points of acquired RGB image
Calculate the pose of acquired RGB image
Pose optimization
Output indoor Position

Figure 6. Process of the pose estimation.

In Figure 6, feature point extraction and matching between the acquired query image and the retrieved database image is the first step in the visual-positioning process. The ORB algorithm was adopted in our method to extract 2D feature points and calculate binary descriptors for feature matching. Then, the fundamental matrix constraint [53] and random sample consensus (RANSAC) algorithm [54] were used to eliminate some false matching points. After that, two sets of good matching points pts_q and pts_r in the pixel coordinate frame were obtained. Figure 7 shows the result of good matching points between the acquired query image and the retrieved database image.

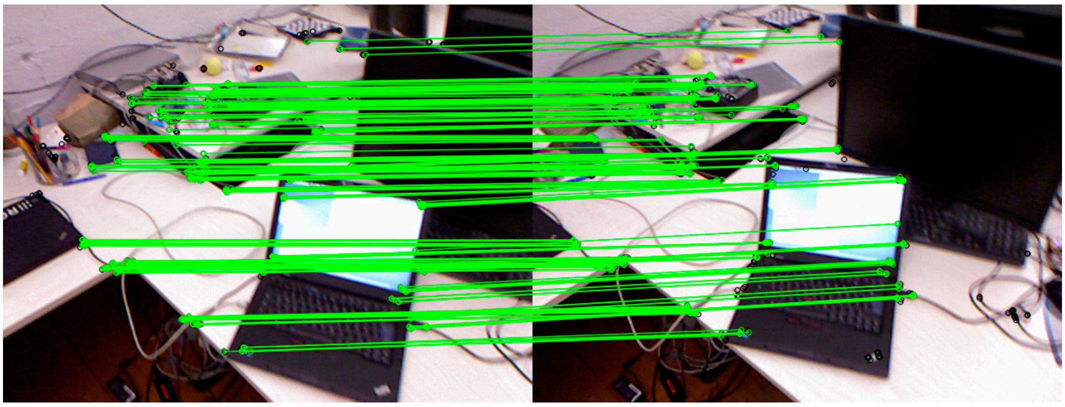


Figure 7. Good matching points of the acquired query image and the retrieved database image.

Second, 3D information in the world coordinate frame of the matching points was obtained by the retrieved RGB-D database image and its image pose. A feature point $p_r(u,v)$ belonging to pts_r in the retrieved database image is a 2D point in the pixel coordinate frame. Its form in the image plane coordinate frame $p_r(x,y)$ is obtained by Equation (5):

$$\begin{cases} x = (u - c_x)/f_x \\ y = (v - c_y)/f_y \end{cases}. \tag{5}$$

Here f_x, f_y, c_x and c_y belong to the intrinsic parameters K of camera, which was calculated through camera calibration process. Through the depth image of the retrieved RGB-D database image, we obtained the depth value $d_{(x,y)}$ of $p_r(x,y)$. Therefore, the 3D point $P_r(X',Y',Z')$ in the image space coordinate frame is obtained by Equation (6):

$$\begin{cases} X' = x \times d_{(x,y)} \\ Y' = y \times d_{(x,y)} \\ Z' = d_{(x,y)} \end{cases}. \tag{6}$$

Next, the input pose information $Pose_r$ of the retrieved image is used to translate $P_r(X',Y',Z')$ into the world coordinate frame. The $Pose_r$ here includes 3D position $\{x_r, y_r, z_r\}$ and quaternion form of attitude $\{qx_r, qy_r, qz_r, qw_r\}$. Usually $Pose_r$ is expressed as a transformation matrix T_{wr} from image space coordinate frame to world coordinate frame by Equation (7):

$$T_{wr} = \begin{bmatrix} R_{wr} & t_{wr} \\ 0 & 1 \end{bmatrix}, \tag{7}$$

where R_{wr} and $t_{wr} = [x_r, y_r, z_r]^{\mathrm{T}}$ are the rotation and translation parts of T_{wr} respectively. R_{wr} is defined as follows:

$$R_{wr} = \begin{bmatrix} 1 - 2qy_r{}^2 - 2qz_r{}^2 & 2qx_r \times qy_r - 2qw_r \times qz_r & 2qx_r \times qz_r + 2qw_r \times qy_r \\ 2qx_r \times qy_r + 2qw_r \times qz_r & 1 - 2qx_r{}^2 - 2qz_r{}^2 & 2qy_r \times qz_r - 2qw_r \times qx_r \\ 2qx_r \times qz_r - 2qw_r \times qy_r & 2qy_r \times qz_r + 2qw_r \times qx_r & 1 - 2qx_r{}^2 - 2qy_r{}^2 \end{bmatrix}. \tag{8}$$

Then T_{wr} is used to transform $P_r(X',Y',Z')$ into the world coordinate frame $P_w(X,Y,Z)$:

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = T_{wr} \cdot \begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} R_{wr} & T_{wr} \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix}. \tag{9}$$

So the relationship between a 3D point $P_w(X, Y, Z)$ in the world coordinate frame and its 2D point $p_r(u, v)$ in the pixel coordinate frame is expressed as follows:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \cdot T_{rw} \cdot P_w = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \left(\begin{bmatrix} R_{rw} & t_{rw} \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \right)_{(1:3)} . \tag{10}$$

Here, matrices T_{rw}, R_{rw} and t_{rw} are the inverse of matrices T_{wr}, R_{wr} and t_{wr} respectively. By using the relationship described in Equation (10), we calculated a set of 3D points Pts_w in the world coordinate frame corresponding to the set of 2D points pts_r in the retrieved image. Because pts_r and pts_q are two sets of matching points, Pts_w is also corresponding to pts_q.

Third, according to the 2D matching points and their 3D points in the query image, the efficient perspective-n-point (EPnP) method [55] was adopted to estimate the initial pose T_{qw} of the query image. The Levenberg–Marquardt algorithm implemented in g2o [56] was then used to optimize the camera pose iteratively. This process can be described as follows:

$$\{R_{qw}, t_{qw}\} = \underset{R_{qw}, t_{qw}}{\operatorname{argmin}} \sum_{i=1}^{n} \left(\| p_i - \frac{1}{\lambda_i} \cdot K \cdot (T_{qw} \cdot P_i)_{(1:3)} \|^2 \right). \tag{11}$$

Here p_i is the i-th 2D point of pts_q and P_i is the i-th 3D point of Pts_w. The number n is the length of pts_q. The poses got from the EPnP method were used as the initial values of T_{qw}.

Through iteration, an optimized pose result T_{qw} of the query image was obtained. And we inverted T_{qw} to get T_{wq}, because T_{wq} is more intuitive, from which we can directly get the pose of the query image. Finally, T_{wq} was saved in the form of $Pose_q = \left\{x_q, y_q, z_q, qx_q, qy_q, qz_q, qw_q\right\}$, where $\left\{x_q, y_q, z_q\right\}$ and $\left\{qx_q, qy_q, qz_q, qw_q\right\}$ are the position and attitude of the query image respectively.

With this process, the precise and real-time position and attitude of the acquired query color image were estimated. In the following experimental section, we performed abundant experiments to verify the accuracy of our indoor-positioning method in common indoor scenes.

3. Experimental Results

We have conducted a series of experiments to evaluate the effectiveness of the proposed indoor positioning method. The first sub-section describes the test data and computer configuration we used in the experiments. In the second sub-section, we evaluate qualitatively the proposed RGB-D image database construction strategy of our indoor positioning method. And the results are reported in the third sub-section. For a complete comparative analysis, the results of our indoor positioning method are also compared with an existing method in reference [50].

3.1. Test Data and Computer Configuration

In order to better evaluate the proposed indoor positioning method, six sequences of the public dataset TUM RGB-D were adopted as the test data. Every sequence of the dataset contains RGB-D images, i.e., RGB and depth images, captured by a Microsoft Kinect sensor at a frame rate of 30 Hz. The size of the RGB-D images was 640 × 480.

Figure 8 shows the six sequences of TUM RGB-D dataset. These six sequences can well represent the common indoor scenes in daily life. And the intrinsic parameters of the Microsoft Kinect sensor were found in reference [51].

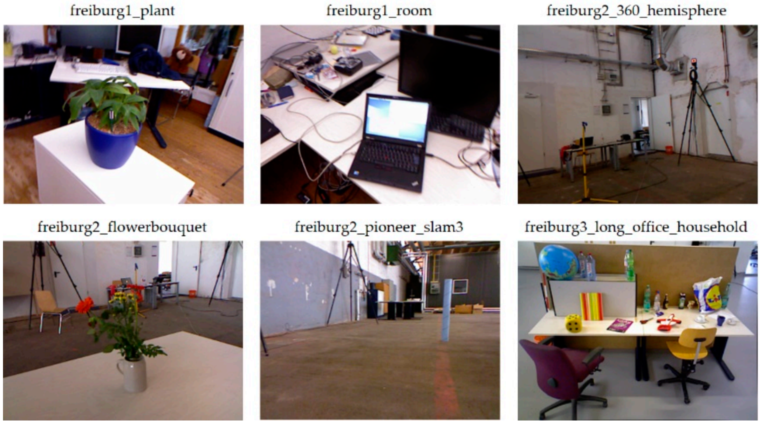


Figure 8. Six sequences of the TUM RGB-D dataset

Before the procedure of RGB-D image database construction, it was important to determine the availability of these sequences. If the associated depth image of a RGB image was missing, then the RGB image was discarded. After that, the remaining test images were used in the experiments of database construction and pose estimation. Then the database images corresponding to the query images with large pose errors were checked manually. If they were motion blur or poorly illuminated images, they were removed from the database. The number of test images in each of these six sequences is shown in Table 2.

Table 2. The number of test images in each of these six sequences of TUM RGB-D dataset.

Six Sequences of TUM RGB-D Dataset	The Number of Test RGB-D Images
freiburg1_plant	1126
freiburg1_room	1352
freiburg2_360_hemisphere	2244
freiburg2_flowerbouquet	2851
freiburg2_pioneer_slam3	2238
freiburg3_long_office_household	2488

We employed a computer with an Intel Core i7-6820HQ CPU @ 2.7 GHz and 16 GB RAM to conduct all the experiments. The procedure of image retrieval based on CNN feature vector was accelerated by a NVIDIA Quadro M2000M GPU. The details of the experimental results are described below.

3.2. Results of RGB-D Database Construction

According to the RGB-D database construction process described in Section 2.2, we got the constructed databases of six sequences shown in Figure 9. The gray lines represent the ground truth of trajectories when recording RGB-D images. The red hexagonal points on the gray lines are the positions of indoor-positioning database images. As can be seen from Figure 9, more database images are selected using the proposed database construction method at the corners where the position and attitude differences between neighboring recorded images change greatly. In these areas with smooth motion, the database images selected by our method are evenly distributed. After selecting the database images from the test images of six sequences, the remaining images were used as query images to conduct the subsequent visual-positioning experiments.

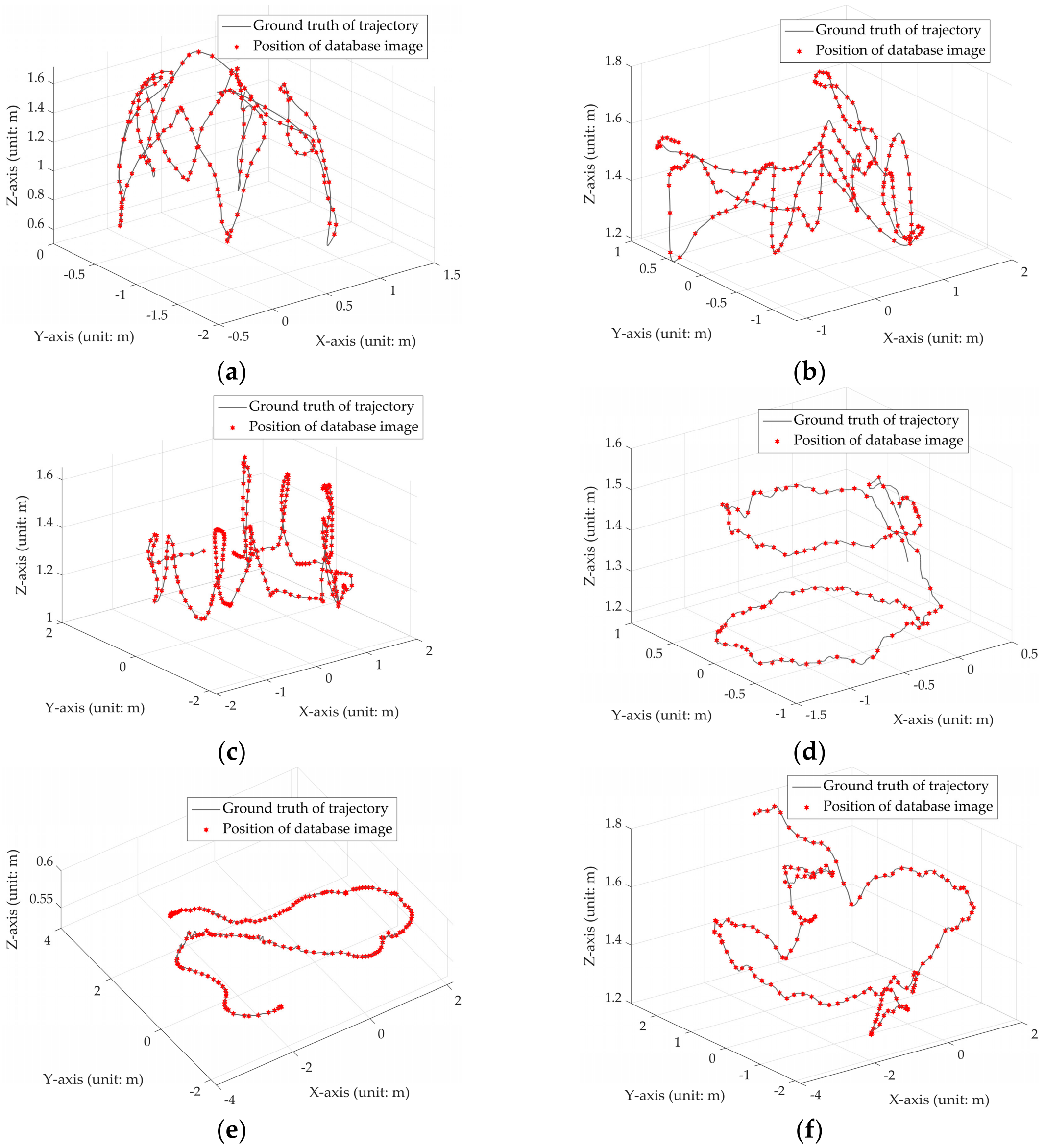


Figure 9. The results of RGB-D database construction for six sequences: (**a**) freibuig1_plant, (**b**) freiburg1_room, (**c**) freiburg2_360_hemisphere, (**d**) freiburg2_flowerbouquet, (**e**) freiburg2_pioneer_slam3 and (**f**) freiburg3_long_office_household.

Considering the redundancy of the RGB-D test images and the efficiency of the visual positioning process, most methods are implemented by selecting representative database images manually, as in reference [50]. The method of hand-pick is subjective and time-consuming. The quality of the database images depends largely on experience. When the number of captured images is large, the workload increases. In contrast, the database images selected by the proposed RGB-D image database construction process are not too redundant or sparse. The database image directly provides high accuracy depth information, this not only ensures high accuracy of positioning, but also improves the efficiency of image retrieval. Therefore, the proposed method can reduce the workload of selecting representative database images and meet the requirements of highly accurate and real-time indoor positioning.

The hand-picked database images of reference method [50] are used for comparison. The numbers of selected database images and used query images in the reference method and the proposed method are shown in Table 3. It can be seen from Table 3 that the number of query images used in our method is about twice the number of query images used in the reference method. This is because the proposed method automatically selects the database images and then takes the remaining images as query images, eliminating the workload of manually selecting images.

Table 3. The numbers of selected database images and query images in the reference method and the proposed method.

Six Sequences of TUM RGB-D Dataset	Images in the Reference Method		Images in the Proposed Method	
	Database	Query	Database	Query
freiburg1_plant	115	456	158	968
freiburg1_room	91	454	193	1159
freiburg2_360_hemisphere	273	1092	253	1991
freiburg2_flowerbouquet	149	1188	104	2747
freiburg2_pioneer_slam3	128	1017	173	2065
freiburg3_long_office_household	130	1034	152	2336

After building the RGB-D indoor-positioning database, we input query images of each sequence in turn. Through the process of image retrieval based on the CNN feature vector, we selected a database RGB-D image with pose information that is most similar to the input query image. Then we performed the visual-positioning process to estimate the position and attitude of each input query image.

3.3. Quantitative Analysis of Positioning Accuracy

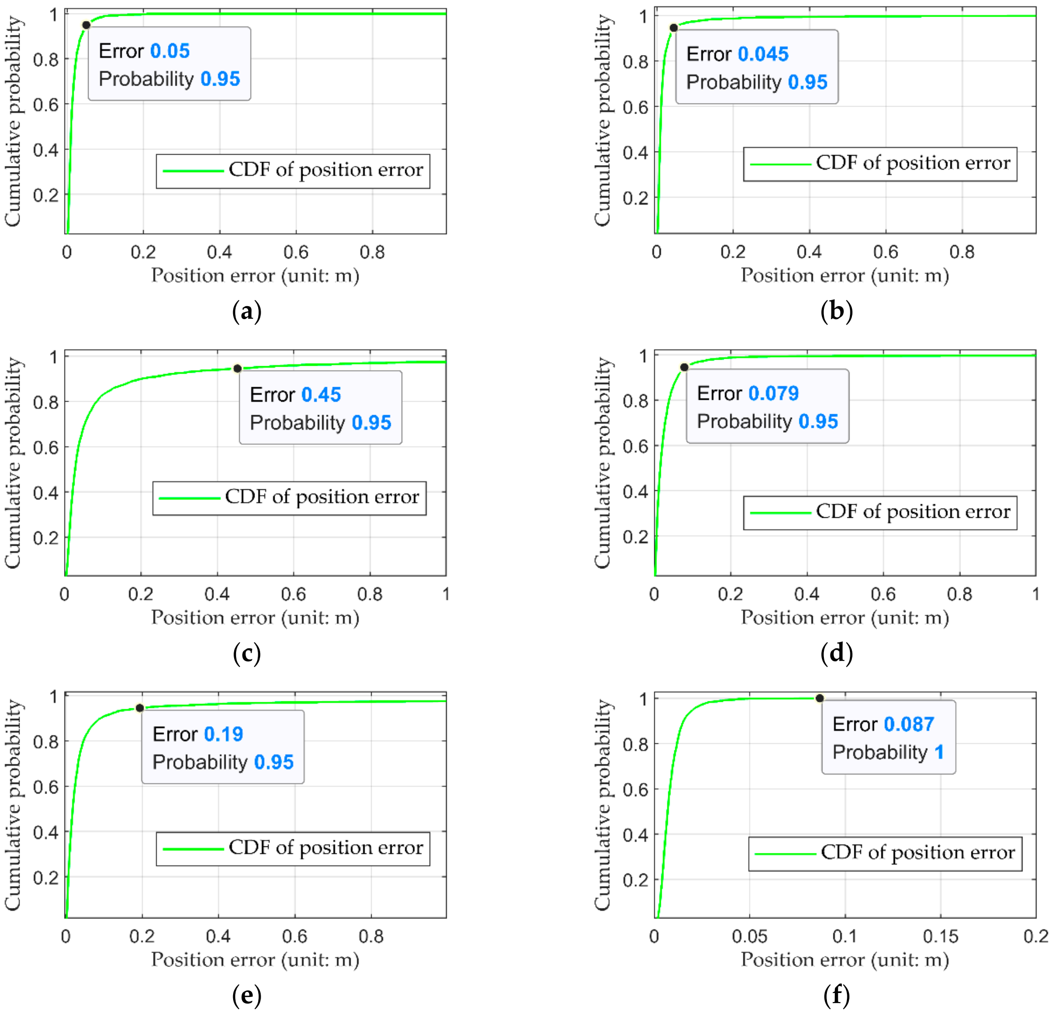
In this part, the performance of pose estimation in the proposed indoor-positioning method was evaluated by comparing it with the reference method mentioned in the previous section. The estimated pose results of each sequence was saved in a file. The mean pose error and median pose error of these two methods were obtained by comparing the estimated poses with the ground truth trajectory. In addition, both position error and attitude error were calculated as an evaluation of six DoFs.

The pose estimation results of each sequence using the reference method and proposed method are shown in Table 4. As can be seen from the results of the proposed method in Table 4, the mean values and median values of position errors are both at the half-meter level. As for attitude errors, the mean errors are within 5 degrees and the median errors are within 1 degree. These results demonstrate that the database we built in Section 3.2 meets the requirements of high-accuracy visual positioning well. By comparing the results of the reference method and the proposed method, we can see that most of the mean and median pose errors of our method are smaller than those of the reference method. This also indicates that the database constructed by our method can achieve or surpass the accuracy of the hand-picked database to some extent.

Table 4. The results of position error and attitude error of the six sequences using the reference method and the proposed method. Each term contains a mean error and a median error. The numbers in bold indicate that these terms are better than those of the other method.

Six Sequences of TUM RGB-D Dataset	Pose Error of the Reference Method		Pose Error of the Proposed Method	
	Mean	Median	Mean	Median
freiburg1_plant	0.38 m 3.37°	0.12 m **0.01°**	**0.02 m 0.61°**	**0.01 m** 0.44°
freiburg1_room	0.43 m 4.82°	0.17 m 0.54°	**0.02 m 1.14°**	**0.01 m 0.50°**
freiburg2_360_hemisphere	0.38 m 6.55°	0.05 m **0.16°**	**0.22 m 2.77°**	**0.03 m** 0.36°
freiburg2_flowerbouquet	0.15 m 5.32°	0.07 m **0.12°**	**0.04 m 1.57°**	**0.02 m** 0.51°
freiburg2_pioneer_slam3	0.34 m 8.80°	0.13 m **0.13°**	**0.18 m 4.10°**	**0.02 m** 0.43°
freiburg3_long_office_household	0.36 m 3.00°	0.15 m **0.21°**	**0.01 m 0.37°**	**0.01 m** 0.31°

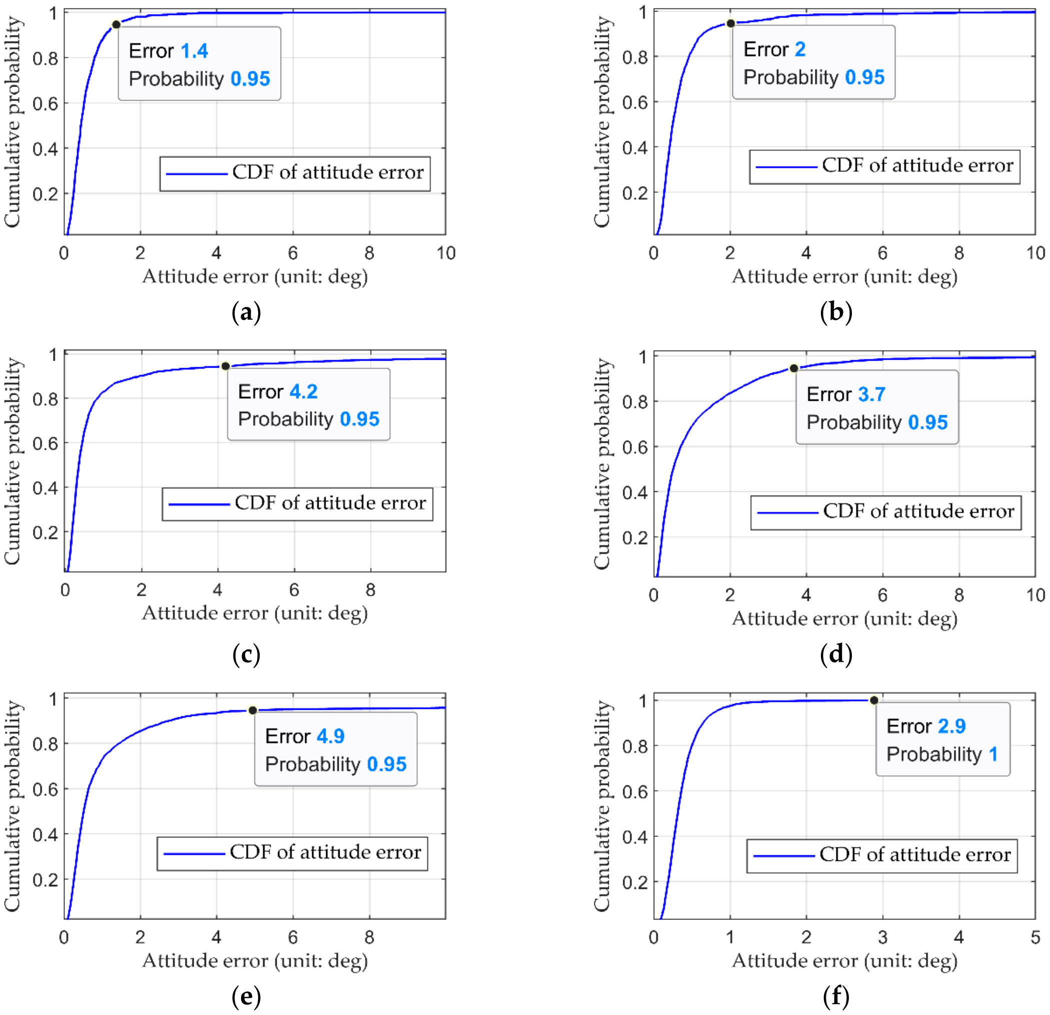
In order to demonstrate the pose estimation results of all the query images intuitively, cumulative distribution function (CDF) was adopted to analyze the positioning accuracy of the proposed method. Figure 10 shows the CDF of position error. From these CDF curves we can see that nearly 95% of the query images have a position error within 0.5 m. Furthermore, the position errors of all query images in sequence freiburg3_long_office_household are within 0.1 m, as shown in Figure 10f. These results show that the proposed method is able to localize the position of a query image well and achieve a high accuracy of better than 1 m in most cases.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

Figure 10. Cumulative distribution function (CDF) of position error in the (**a**) freibuig1_plant, (**b**) freiburg1_room, (**c**) freiburg2_360_hemisphere, (**d**) freiburg2_flowerbouquet, (**e**) freiburg2_pioneer_slam3 and (**f**) freiburg3_long_office_household sequence. The green line of each sub-figure represents the CDF curve. The vertical coordinate represents the cumulative probability of position error.

In addition to CDF of position error, CDF of attitude error was also calculated as an evaluation of six DoFs. Figure 11 shows the CDF of attitude error. The blue line of each sub-figure represents the CDF curve. Similarly, we can see that the attitude errors of all query images in sequence freiburg3_long_office_household are within 3 degrees, as shown in Figure 11f. For the rest of the sequences, almost 95% of the query images have an attitude error within 5 degrees. These results show that the proposed method is able to calculate the attitude of a query image well with an accuracy of better than 5 degrees in most cases.



(**a**)

(**b**)

(**c**)

(**d**)

(**e**)

(**f**)

Figure 11. Cumulative distribution function (CDF) of attitude error in the (**a**) freibuig1_plant, (**b**) freiburg1_room, (**c**) freiburg2_360_hemisphere, (**d**) freiburg2_flowerbouquet, (**e**) freiburg2_pioneer_slam3 and (**f**) freiburg3_long_office_household sequence. The blue line of each sub-figure represents the CDF curve. The vertical coordinate represents the cumulative probability of attitude error.

The cumulative pose errors of the reference method and the proposed method were compared, as shown in Table 5. It was found that most of the results using our method outperformed those using the reference method. As can be seen from the cumulative accuracy of the reference method, 90% of the query images in each sequence are localized within 1 m and 5 degrees. The 90% accuracy of position error of our method is within 0.5 m and the attitude error is within 3 degrees, both of which are nearly half the pose error of the reference method. Specifically, all the cumulative position errors of the proposed method are better than those of the reference method. The cumulative attitude errors of the proposed method are better or comparable with those of the reference method. These good performances of the proposed indoor visual-positioning method also indicate the validity of the proposed database construction strategy.

Table 5. The cumulative pose errors of the reference method and the proposed method in six sequences. Each term contains a mean error and a median error. The unit of position error is meter and the unit of attitude error is degree. The numbers in bold indicate that these terms are better than those of the other method.

Six Sequences of TUM RGB-D Dataset	90% Accuracy of the Reference Method	90% Accuracy of the Proposed Method
freiburg1_plant	0.45 m 1.95°	**0.04 m 1.09°**
freiburg1_room	0.71 m 4.04°	**0.03 m 1.29°**
freiburg2_360_hemisphere	0.38 m **1.08°**	**0.21 m** 1.94°
freiburg2_flowerbouquet	0.26 m **2.54°**	**0.06 m** 2.76°
freiburg2_pioneer_slam3	0.66 m **1.54°**	**0.10 m** 2.75°
freiburg3_long_office_household	0.41 m 2.05°	**0.02 m 0.65°**

All the experiments were conducted by a laptop with an Intel Core i7-6820HQ CPU @ 2.7 GHz and 16 GB RAM. In the experiment, it takes about 0.4 s on average to implement the RGB-D database image-retrieval process in selecting the most similar database image with the input query image. The pose estimation process of one query image takes about 0.2 s on average. Considering that the RGB-D indoor positioning database is built offline, the time it costs is not taken into account. Therefore, our indoor-positioning program will take about 0.6 s to complete the two processes of image retrieval and pose estimation. Specifically, if we use a mobile platform to capture query image at a resolution of 640 × 480 and upload it into the laptop using 4G network, the process takes about 0.3 s. As for returning the location result from the laptop to the mobile platform, it takes about 0.1 s. Therefore, the whole procedure, which starts with capturing a query image by the mobile platform and finally obtains the position result from the laptop, takes about 1 s. In other words, the indoor-positioning frequency of the proposed method is about 1 Hz. This also shows that the proposed method has the ability of real-time indoor positioning while satisfying the need for high accuracy.

4. Conclusions

In this study, a novel indoor-positioning method with automated RGB-D image database construction was presented. The proposed method has two main innovations. First, the indoor-positioning database constructed by our method can reduce the workload of manually selecting database images and is more objective. The database is automatically constructed according to the preset rules, which reduces the redundancy of the database and improves the efficiency of the image-retrieval process. Second, by combining automatic database construction module, the CNN-based image retrieval module, and strict geometric relations based pose estimation module, we obtain a highly accurate indoor-positioning system.

In the experiment, the proposed indoor positioning method was evaluated with six typical indoor sequences of TUM RGB-D dataset. We presented the quantitative evaluation results of our method compared with a state-of-the-art indoor visual-positioning method. All the experimental results show that the proposed method obtains high-accuracy position and attitude results in common indoor scenes. The accuracy of the proposed method attained is generally higher than that of the reference method.

In the next version of our indoor-positioning method, we plan to combine the semantic information of the sequence to reduce the search space of visual positioning in large indoor scenes.

Author Contributions: R.W., K.D. and W.W. conceived the idea, designed the method and wrote the paper; R.W. and R.C. developed the software; R.W., W.W. and X.F. performed the experiments; W.W. and K.D. analyzed the data.

Funding: This research was funded by National Key Research and Development Program of China (No. 2016YFB0502102 and No. 2018YFB1305004).

Acknowledgments: We thank Computer Vision Group from Technical University of Munich for making the RGB-D SLAM dataset publically available.

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

Abbreviations

The following abbreviations are used in this manuscript:

RGB-D	red, green, blue and depth
GNSS	Global Navigation Satellite System
SIFT	scale-invariant feature transform
PCA	principal component analysis
SAD	sum of absolute difference
DoFs	degrees of freedom
SLAM	simultaneous localization and mapping
CNN	convolutional neural networks
TUM	Technical University of Munich
RMSE	root mean square errors
NetVLAD	vector of locally aggregated descriptors
RANSAC	random sample consensus
EPnP	efficient perspective-n-point method
CDF	cumulative distribution function

References

1. Jiang, B.; Yao, X. Location-based services and GIS in perspective. *Comput. Environ. Urban Syst.* **2006**, *30*, 712–725. [CrossRef]
2. Weiser, M. The computer for the 21st century. *IEEE Pervasive Comput.* **1999**, *3*, 3–11. [CrossRef]
3. Davidson, P.; Piché, R. A survey of selected indoor positioning methods for smartphones. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 1347–1370. [CrossRef]
4. Atia, M.M.; Noureldin, A.; Korenberg, M.J. Dynamic Online-Calibrated Radio Maps for Indoor Positioning in Wireless Local Area Networks. *IEEE Trans. Mob. Comput.* **2013**, *12*, 1774–1787. [CrossRef]
5. Du, Y.; Yang, D.; Xiu, C. A Novel Method for Constructing a WIFI Positioning System with Efficient Manpower. *Sensors* **2015**, *15*, 8358–8381. [CrossRef]
6. Moghtadaiee, V.; Dempster, A.G.; Lim, S. Indoor localization using FM radio signals: A fingerprinting approach. In Proceedings of the 2011 International Conference on Indoor Positioning & Indoor Navigation (IPIN), Guimarães, Portugal, 21–23 September 2011; pp. 1–7.
7. Bahl, P.; Padmanabhan, V.N. In RADAR: An in-building RF-based user location and tracking system. *IEEE Infocom* **2000**, *2*, 775–784.
8. Youssef, M.; Agrawala, A. The Horus WLAN location determination system. In Proceedings of the 3rd International Conference on Mobile Systems, Application, and Services (MobiSys 2005), Seattle, WA, USA, 6–8 June 2005; pp. 205–218.
9. Cantón Paterna, V.; Calveras Augé, A.; Paradells Aspas, J.; Pérez Bullones, M.A. A Bluetooth Low Energy Indoor Positioning System with Channel Diversity, Weighted Trilateration and Kalman Filtering. *Sensors* **2017**, *17*, 2927. [CrossRef]
10. Chen, L.; Pei, L.; Kuusniemi, H.; Chen, Y.; Kröger, T.; Chen, R. Bayesian Fusion for Indoor Positioning Using Bluetooth Fingerprints. *Wirel. Pers. Commun.* **2013**, *70*, 1735–1745. [CrossRef]
11. Huang, X.; Guo, S.; Wu, Y.; Yang, Y. A fine-grained indoor fingerprinting localization based on magnetic field strength and channel state information. *Pervasive Mob. Comput.* **2017**, *41*, 150–165. [CrossRef]
12. Kim, H.-S.; Seo, W.; Baek, K.-R. Indoor Positioning System Using Magnetic Field Map Navigation and an Encoder System. *Sensors* **2017**, *17*, 651. [CrossRef]
13. Gronat, P.; Obozinski, G.; Sivic, J.; Pajdla, T. Learning and Calibrating Per-Location Classifiers for Visual Place Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 907–914.
14. Vaca-Castano, G.; Zamir, A.R.; Shah, M. City scale geo-spatial trajectory estimation of a moving camera. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 1186–1193.

15. Arandjelović, R.; Zisserman, A. Three things everyone should know to improve pbject retrieval. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2911–2918.
16. Zamir, A.R.; Shah, M. Accurate Image Localization Based on Google Maps Street View. In Proceedings of the 11th European Conference on Computer Vision, Heraklion, Crete, Greece, 5–11 September 2010; pp. 255–268.
17. Zamir, A.R.; Shah, M. Image Geo-Localization Based on MultipleNearest Neighbor Feature Matching UsingGeneralized Graphs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 1546–1558. [CrossRef] [PubMed]
18. Babenko, A.; Slesarev, A.; Chigorin, A.; Lempitsky, V. Neural Codes for Image Retrieval. In Proceedings of the 13th European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 584–599.
19. Azzi, C.; Asmar, D.; Fakih, A.; Zelek, J. Filtering 3D keypoints using GIST for accurate image-based positioning. In Proceedings of the 27th British Machine Vision Conference, York, UK, 19–22 September 2016; pp. 1–12.
20. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
21. Hervé, J.; Douze, M.; Schmid, C. Product Quantization for Nearest Neighbor Search. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *33*, 117–128.
22. Nistér, D.; Stewénius, H. Scalable Recognition with a Vocabulary Tree. In Proceedings of the Computer Vision and Pattern Recognition, New York, NY, USA, 17–22 June 2006; pp. 2161–2168.
23. Kim, H.J.; Dunn, E.; Frahm, J.M. Predicting Good Features for Image Geo-Localization Using Per-Bundle VLAD. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 13–16 December 2015; pp. 1170–1178.
24. Torii, A.; Arandjelovic, R.; Sivic, J.; Okutomi, M.; Pajdla, T. 24/7 place recognition by view synthesis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1808–1817.
25. Milford, M.J.; Lowry, S.; Shirazi, S.; Pepperell, E.; Shen, C.; Lin, G.; Liu, F.; Cadena, C.; Reid, I. Sequence searching with deep-learnt depth for condition-and viewpoint-invariant route-based place recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 18–25.
26. Poglitsch, C.; Arth, C.; Schmalstieg, D.; Ventura, J. A Particle Filter Approach to Outdoor Localization Using Image-Based Rendering. In Proceedings of the IEEE International Symposium on Mixed & Augmented Reality, Fukuoka, Japan, 29 September–3 October 2015; pp. 132–135.
27. Razavian, A.S.; Azizpour, H.; Sullivan, J.; Carlsson, S. CNN features off-the-shelf: An astounding baseline for recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern RecognitionWorkshops, Columbus, OH, USA, 23–28 June 2014; pp. 512–519.
28. Yandex, A.B.; Lempitsky, V. Aggregating Local Deep Convolutional Features for Image Retrieval. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 13–16 December 2015; pp. 1269–1277.
29. Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J.J. Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Trans. Robot.* **2016**, *32*, 1309–1332. [CrossRef]
30. Younes, G.; Asmar, D.; Shammas, E.; Zelek, J. Keyframe-based monocular SLAM: Design, survey, and future directions. *Robot. Auton. Syst.* **2017**, *98*, 67–88. [CrossRef]
31. Campbell, D.; Petersson, L.; Kneip, L.; Li, H. Globally-optimal inlier set maximisation for simultaneous camera pose and feature correspondence. In Proceedings of the IEEE Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1–10.
32. Liu, L.; Li, H.; Dai, Y. Efficient Global 2D-3D Matching for Camera Localization in a Large-Scale 3D Map. In Proceedings of the IEEE Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2391–2400.
33. Yousif, K.; Taguchi, Y.; Ramalingam, S. MonoRGBD-SLAM: Simultaneous localization and mapping using both monocular and RGBD cameras. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 4495–4502.
34. Turan, M.; Almalioglu, Y.; Araujo, H.; Konukoglu, E.; Sitti, M. A non-rigid map fusion-based rgb-depth slam method for endoscopic capsule robots. *Int. J. Intell. Robot. Appl.* **2017**, *1*, 399. [CrossRef]

35. Sattler, T.; Torii, A.; Sivic, J.; Pollefeys, M.; Taira, H.; Okutomi, M.; Pajdla, T. Are Large-Scale 3D Models Really Necessary for Accurate Visual Localization? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6175–6184.
36. Degol, J.; Bretl, T.; Hoiem, D. ChromaTag: A Colored Marker and Fast Detection Algorithm. In Proceedings of the IEEE Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1481–1490.
37. Muñoz-Salinas, R.; Marín-Jimenez, M.J.; Yeguas-Bolivar, E.; Medina-Carnicer, R. Mapping and Localization from Planar Markers. *Pattern Recognit.* **2017**, *73*, 158–171. [CrossRef]
38. Schweighofer, G.; Pinz, A. Robust pose estimation from a planar target. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *28*, 2024–2030. [CrossRef]
39. Valentin, J.; Niebner, M.; Shotton, J.; Fitzgibbon, A.; Izadi, S.; Torr, P. Exploiting uncertainty in regression forests for accurate camera relocalization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4400–4408.
40. Tateno, K.; Tombari, F.; Laina, I.; Navab, N. CNN-SLAM: Real-Time Dense Monocular SLAM with Learned Depth Prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6565–6574.
41. Shotton, J.; Glocker, B.; Zach, C.; Izadi, S.; Criminisi, A.; Fitzgibbon, A. Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 2930–2937.
42. Kendall, A.; Cipolla, R. Geometric loss functions for camera pose regression with deep learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6555–6564.
43. Tekin, B.; Sinha, S.N.; Fua, P. Real-Time Seamless Single Shot 6D Object Pose Prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 292–301.
44. Piasco, N.; Sidibé, D.; Demonceaux, C.; Gouet-Brunet, V. A survey on Visual-Based Localization: On the benefit of heterogeneous data. *Pattern Recognit.* **2018**, *74*, 90–109. [CrossRef]
45. Ummenhofer, B.; Zhou, H.; Uhrig, J.; Mayer, N.; Ilg, E.; Dosovitskiy, A.; Brox, T. DeMoN: Depth and motion network for learning monocular stereo. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5622–5631.
46. Mahjourian, R.; Wicke, M.; Angelova, A. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 5667–5675.
47. Zhan, H.; Garg, R.; Weerasekera, C.S.; Li, K.; Agarwal, H.; Reid, I. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 340–349.
48. Gao, X.; Zhang, T. Unsupervised learning to detect loops using deep neural networks for visual SLAM system. *Auton. Robot.* **2017**, *41*, 1–18. [CrossRef]
49. Wu, J.; Ma, L.; Hu, X. Delving deeper into convolutional neural networks for camera relocalization. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 5644–5651.
50. Chen, Y.; Chen, R.; Liu, M.; Xiao, A.; Wu, D.; Zhao, S. Indoor Visual Positioning Aided by CNN-Based Image Retrieval: Training-Free, 3D Modeling-Free. *Sensors* **2018**, *18*, 2692. [CrossRef] [PubMed]
51. Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; Cremers, D. A benchmark for the evaluation of RGB-D SLAM systems. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Algarve, Portugal, 7–12 October 2012; pp. 573–580.
52. Arandjelovic, R.; Gronat, P.; Torii, A.; Pajdla, T.; Sivic, J. Netvlad: CNN architecture for weakly supervised place recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 1437–1451. [CrossRef] [PubMed]
53. Richard, H.; Andrew, Z. *Multiple View Geometry in Computer Vision*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2003; pp. 241–253.
54. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. Acm* **1981**, *24*, 381–395. [CrossRef]

55. Lepetit, V.; Moreno-Noguer, F.; Fua, P. EPnP: An accurate O(n) solution to the PnP problem. *Int. J. Comput. Vis.* **2009**, *81*, 155–166. [CrossRef]
56. Kümmerle, R.; Grisetti, G.; Strasdat, H.; Konolige, K.; Burgard, W. G2o: A general framework for graph optimization. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 3607–3613.