sensors

MDPI

Article

Indoor Visual Positioning Aided by CNN-Based Image Retrieval: Training-Free, 3D Modeling-Free

Yujin Chen [1], Ruizhi Chen [1,2,*], Mengyun Liu [1], Aoran Xiao [1], Dewen Wu [1] and Shuheng Zhao [1]

[1] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University, Wuhan 430079, China; yujin.chen@whu.edu.cn (Y.C.); amylmy@whu.edu.cn (M.L.); xiaoaoran@whu.edu.cn (A.X.); wudewen@whu.edu.cn (D.W.); photonmango@foxmail.com (S.Z.)

[2] Collaborative Innovation Center of Geospatial Technology (INNOGST), Wuhan 430079, China

* Correspondence: ruizhi.chen@whu.edu.cn; Tel.: +86-134-7603-1098

Received: 5 July 2018; Accepted: 10 August 2018; Published: 16 August 2018

Abstract: Indoor localization is one of the fundamentals of location-based services (LBS) such as seamless indoor and outdoor navigation, location-based precision marketing, spatial cognition of robotics, etc. Visual features take up a dominant part of the information that helps human and robotics understand the environment, and many visual localization systems have been proposed. However, the problem of indoor visual localization has not been well settled due to the tough trade-off of accuracy and cost. To better address this problem, a localization method based on image retrieval is proposed in this paper, which mainly consists of two parts. The first one is CNN-based image retrieval phase, CNN features extracted by pre-trained deep convolutional neural networks (DCNNs) from images are utilized to compare the similarity, and the output of this part are the matched images of the target image. The second one is pose estimation phase that computes accurate localization result. Owing to the robust CNN feature extractor, our scheme is applicable to complex indoor environments and easily transplanted to outdoor environments. The pose estimation scheme was inspired by monocular visual odometer, therefore, only RGB images and poses of reference images are needed for accurate image geo-localization. Furthermore, our method attempts to use lightweight datum to present the scene. To evaluate the performance, experiments are conducted, and the result demonstrates that our scheme can efficiently result in high location accuracy as well as orientation estimation. Currently the positioning accuracy and usability enhanced compared with similar solutions. Furthermore, our idea has a good application foreground, because the algorithms of data acquisition and pose estimation are compatible with the current state of data expansion.

Keywords: indoor positioning; image geo-localization; image retrieval; CNN features; pose estimation

1. Introduction

The increasing demand of location-based services (LBS) in recent years inspires the desire for accurate position information. The most common way for positioning with cell phone and other mobile platforms is GNSS (Global Navigation Satellite System). However, in most of the time, GNSS is only available for the outdoor environment. When it comes to indoor environment, GNSS signals are mostly blocked by obstacles. In recent years, a number of alternative technologies have been proposed for indoor positioning. Most of indoor positioning methods are focused on fingerprinting-based localization algorithms which are infrastructure-free [1–3]. In these methods, Wi-Fi received signal strengths (RSS) or magnetic field strengths (MFS) are collected and will be compared with data in a fingerprinting database during positioning period. This fingerprinting-based system is easy to establish and can achieve fine localization performance in the short term. Nevertheless, signal patterns will change over time due to the environment changes, which makes it hard to maintain the positioning performance. Additionally, construction of fingerprint database is time-consuming and labor-intensive. To overcome the defects of this scheme, many alternatives have been proposed, including Optical [4,5], RFID (Radio Frequency Identification) [6], Bluetooth Beacons [7], ZigBee [8,9], Pseudo Satellite [10,11], etc. Whereas the accuracy is not enough in intricate indoor environment, and these solutions may need artificial setting and additional infrastructures which may bring unbearable costs.

There are also some previous attempts at indoor visual positioning. **Recognition-based** image geo-localization methods are quite similar with the problem of image classification in computer vision, in which global or region features are used for image matching [12–15]. In image classification issue, similar images are labeled as the same category. Regarding the visual localization problem, relative images are identified as sharing similar geo-location information. As for recognition-based method, location of the target image is estimated by retrieving related images or scene classification [16–19]. Recognition-based methods apply an image retrieval strategy or a scene classification strategy at first, subsequently the location of the query image is estimated based on the localization information of the associate retrieved images or the classification labels. However, the mentioned methods above generally provide a rather coarse estimation of location, which hardly satisfies the need of accurate LBS. **Geometric matching-based** methods represent the scenes by geo-referenced 3D models, and then, estimate the pose of query image by directly matching 2D image features to 3D models or by matching 3D image features to 3D models when depth information is available. These approaches typically come with estimation of 6 degrees of freedom (DoF) camera parameters. However, geometric matching-based methods still have many challenges, which can be concluded as follows: (1) Superior difficulty in constructing high fidelity RGB-D scene models as well as employing 2D-to-3D matching for textured 3D models scheme; and (2) as for non-RGB point-only models scheme, the problem of geometric alignment between the query images and 3D point models can be hard to settle.

To overcome the limitations of recognition-based methods and geometric matching-based methods, a combination of these two strategies has been proposed in the devised scheme. In this paper, we demonstrate an image-based indoor localization scheme which is capable of not only achieving sub-meter level positioning accuracy but also determining orientations. At the same time, the proposed scheme merely uses RGB images in the course of the online localization period and a server is used to host the image database for the computing operation.

The main contributions of this paper can be concluded as follows:

(1) Inspired by the visual spatial cognition ability of human, an image-based visual positioning scheme is proposed. The target image is matched with database images to get the most similar image for localization computing.
(2) Our visual localization algorithm is 3D-modeling-free. Compared with visual localization methods that combine with image retrieval and image pose estimation from regional 3D reconstruction, regional 3D modeling is unrequired in our scheme since we recover camera pose from two sets of 2D-to-2D matches.
(3) Our spatial model is training-free for different scenarios. Owing to pre-trained deep learning models are stable and can be used as powerful feature extractors, we apply deep convolutional neural network (DCNN) pre-trained on ImageNet to extract features to represent images, thus we need not train a unique model for a specific scene.
(4) For localization purpose, we use a lighter model to represent the scene. CNN features extracted from images of database can represent the scene in image retrieval phase. Compared with CNN learning-based visual localization methods that require a large number of images during model training, much fewer images are required to represent the same scene in our scheme.

The paper proceeds as follows: Section 2 provides a brief overview of related work. The system architecture and methods are described in detail in Section 3. Experiments and performance evaluations are presented in Section 4. Sections 5 and 6 are discussion and suggestions for future work.

2. Related Work

The work presented in this paper relates to many fields, such as visual localization, image retrieval, and visual pose estimation.

At present, visual localization systems can be roughly divided into three categories.

Structure-based localization methods are the most common visual localization methods that utilize local features to estimate 2D-to-3D matches between features in a query image and points in 3D models, or employ 3D-to-3D matches between RGB-D images and 3D models. Then camera pose will be estimated from the correspondence. Similarly, Torsten et al. [20] compared 2D image-based localization with 3D structure-based localization, and they drew a conclusion that purely 2D-based methods achieve the lowest localization and 3D-based methods offer more precise pose estimation with more complex model construction and maintenance. They proposed a combination of 2D-based methods with local structure-from-motion (SfM) reconstruction which has both a simple database construction procedure and accurate pose estimation. However, the drawback of their method is significantly longer run-time during the location process.

Image-based localization methods were pushed by massive repositories of public geo-labeled images. These methods employ an image retrieval-based strategy [16–19], which match the query image with images from the database. Afterward the location of the query image is computed based on the pose information of the retrieved reference images [21–23]. Owing to the prosperity of social network and street view photos, quantity of images with geo-tags has emerged which can be used for reference to these data-driven image-based localization methods. *Image retrieval* is a visual search task that searches and retrieves images from a large database of digital images, which is commonly used in many image-based localization methods. Conventional methods retrieve images based on local descriptor matching and reorder with elaborate spatial verification [24–26]. Content based image retrieval search for images relies on visual content such as edges, colors, textures, and shape [27]. Recent works leverage deep convolution neural networks for image retrieval, the majority of them use a pre-trained network as local feature extractor. Moreover, some work even can address the problem of geometric invariance of CNN features [28,29], and to accurately represent images of different sizes and aspects ratios [30,31].

Learning-based localization methods emerged in the past few years, which benefited from the dramatic progress made in a variety of computer vision tasks. By training models from given images with pose information, scenes can be represented by these learned models. These learning-based localization methods either predict matches for pose estimation [32–35] or directly regress the camera pose such as PoseNet [36], PoseNet2 [37], and VlocNet [38]. PoseNet was the first approach to use DCNNs to solve the metric localization problem, and then Bayesian CNN implementation was utilized to address the pose uncertainty [39]. After that, architectures such as long-short term memory (LSTM) [40–42] and symmetric encoder-decoder [43] were utilized to facilitate the performance of DCNNs.

Moreover, many localization methods [44–47] adopt a from-rough-to-precise idea. For example [44], to utilized scene recognition to locate in scene-level area, and then employed a multi-sensor fusion approach to give a specific location. Similarly, the purely visual-based methods have also been proposed by researchers. Reference [45] casts the localization as an alignment problem of the edges of the query image to a 3D model consisting of line segments. In Reference [46], recognition-based periods are utilized to give coarse localization and then matching can be employed in rather small region. Whereas, in their work, the accuracy and robustness are not sufficient for pervasive use, for the reason that their SIFT-based images retrieval is not stable for the complexity and diversity of indoor environments. To solve this problem, the proposed method adopts a robust CNN-based images retrieval scheme which can fully satisfy the requirement of image retrieval, which is efficient for indoor scenes. Moreover, 3D model is unnecessary in our strategy.

Compared with previous schemes, this paper combines image retrieval-based strategy with feature-based pose estimation period. During the image retrieval period, we utilize a network pre-trained on ImageNet as feature extractor. CNNs learn suitable feature representations for localization in indoor environments, and experiment shows that the performance of this strategy is sufficient to retrieve spatial adjacent images. Pose of the target image is estimated based on a selected geo-tagged image, this algorithm was inspired by similar procedure in monocular visual odometer which uses the images of nearby frames as well as the estimated pose of the first frame. Due to the procession of 3D modeling is complicated, we utilize a strategy that represents local scenario by two contiguous images and succeeding computed the query image's pose from one of the reference images. However, the performance of pose estimation is highly related to the similarity between the query images and the reference images. In other words, well-behaved image retrieval paves the way of valid precise pose estimation.

3. System Overview and Methods

In this section, firstly, we describe the proposed method at a high level. Then, key modules and important algorithms are described in detail, including data preparation, CNN-based image retrieval and pose estimation.

3.1. System Architecture

We demonstrate a single RGB image based localization system which is not only capable of reaching sub-meter localization accuracy but also estimating orientation.

The proposed system consists of three components, as shown in Figure 1:

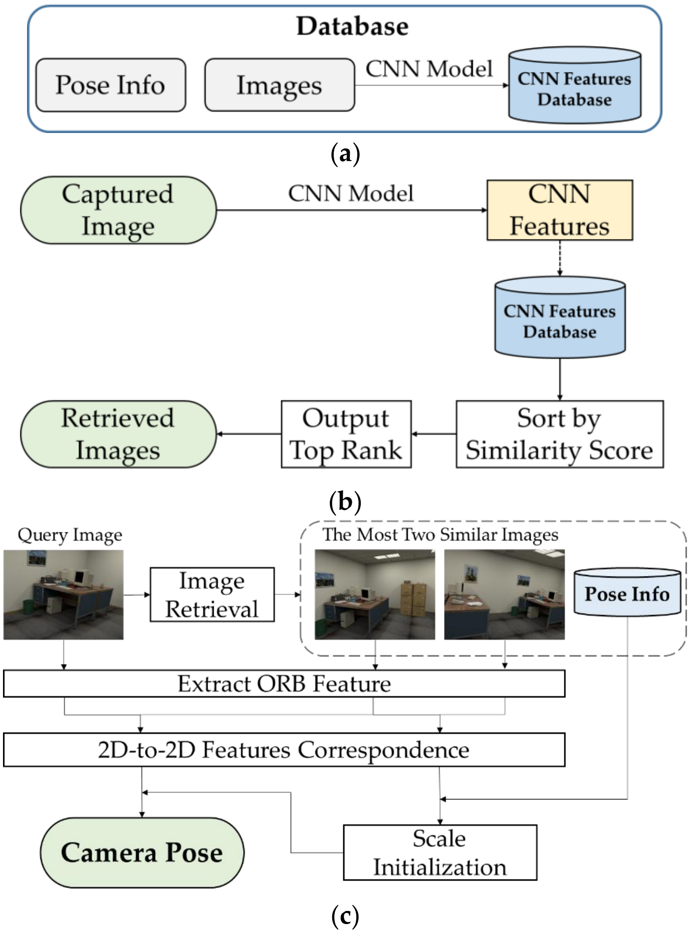


Figure 1. Overview of our visual indoor positioning method. The process is composed of (**a**) database construction; (**b**) image retrieval; and (**c**) pose estimation stages.

(1) Data preparation, shown in Figure 1a: We collected RGB images from target scenarios, then extracted CNN features from all RGB images through pre-trained CNN models. All of the work was done in offline period.

(2) Image retrieval, shown in Figure 1b: We loaded all of the CNN features of images in database, and ranked them according to their similarity from the CNN features extracted from captured image, and then output a set of images with top similarity. Pose estimation, shown in Figure 1c: We carried out image retrieval to the query image and got two of the most similar images as well as their poses. Then, feature points were extracted from the query image and retrieved images. We employed 2D-to-2D correspondence to feature points extracted from two retrieved images to compute the scale in monocular vision setting, and then applied the same procedure to feature points from the query image, and the matching image to compute the pose of the query image.

3.2. Data Preparation

In this part, structure of the database is described. The input of the proposed system is an RGB image which is captured either by a cellphone camera or other mobile platforms. In the database, the absolute 3D spatial coordinates (x, y, z) and quaternion (qx, qy, qz, qw) of all images are known with respect to a given local coordinate system. In addition, CNN features of each image are also included in image database.

Each image can locally represent the scene it belongs to, and image set contains the information of the scene. In the proposed method, two of the most similar images are applied to compute the scale of monocular vision during pose estimation period, therefore, adjacent images should have enough common area for feature matching. The more well-selected images to represent the scene, the better performance of the retrieval and pose estimation result would be. Besides, too many images result in increasing of the cost of data acquisition and computing time. We design the image set as follows.

As shown in Table 1, the database $\boldsymbol{S}$ of this experiment contains n different scenes as $\boldsymbol{S} = \{S_1, S_2, \ldots, S_n\}$. For each scene S_i, we need to get a set of images $\boldsymbol{I} = \{I_{ij}\}$ with associated pose information $\boldsymbol{P} = \{P_{ij}\}$, and their respective CNN features $\boldsymbol{C} = \{C_{ij}\}$ to create a global representation of this scene, where $P_{ij} = \{x_{ij}, y_{ij}, z_{ij}, qx_{ij}, qy_{ij}, qz_{ij}, qw_{ij}\}$ is the position and pose data of image I_{ij}.

Table 1. Composition of database.

Scene Labels	Color Images	Pose Information	CNN Features
S_1	$\{I_{11}, I_{12}, \ldots, I_{1k_1}\}$	$\{P_{11}, P_{12}, \ldots, P_{1k_1}\}$	$\{C_{11}, C_{12}, \ldots, C_{1k_1}\}$
S_2	$\{I_{21}, I_{22}, \ldots, I_{2k_2}\}$	$\{P_{21}, P_{22}, \ldots, P_{2k_2}\}$	$\{C_{21}, C_{22}, \ldots, C_{2k_1}\}$
...	...		
S_n	$\{I_{n1}, I_{n2}, \ldots, I_{nk_n}\}$	$\{P_{n1}, P_{n2}, \ldots, P_{nk_n}\}$	$\{C_{n1}, C_{n2}, \ldots, C_{nk_1}\}$

3.3. CNN-Based Image Retrieval

In this section, fundamentals of a deep convolutional neural network are described, as well as a pre-trained CNN model for deep feature extraction in following experiment.

3.3.1. Deep Convolutional Neural Networks (DCNNs)

As illustrated in Figure 2, the configuration of CNNs used in our proposed scheme is similar to VGG16 which achieved great performance in the large-scale image recognition tasks such as ILSVRC classification and localization. VGG-Nets apply the same principles as normal CNNs, and the key characteristic of this kind of method is increasing depth using an architecture with very small (3 × 3) convolution filters. [48] proposed six kinds of VGG-Nets, number of their layers varied from 16 to 24. In our proposed scheme, we use a 16-layer VGG-Net named VGG16, this network consists of thirteen convolutional layers (block1_conv1, block1_conv2, block2_conv1, block2_conv2, block3_conv1, block3_conv2, block3_conv3, block4_conv1, block4_conv2, block4_conv3, block5_conv1, block5_conv2, block5_conv3), five max pooling layers (block1_pool, block2_pool, block3_pool, block4_pool, block5_pool), three fully connected layers (fc1, fc2, fc3) and a soft-max layer.

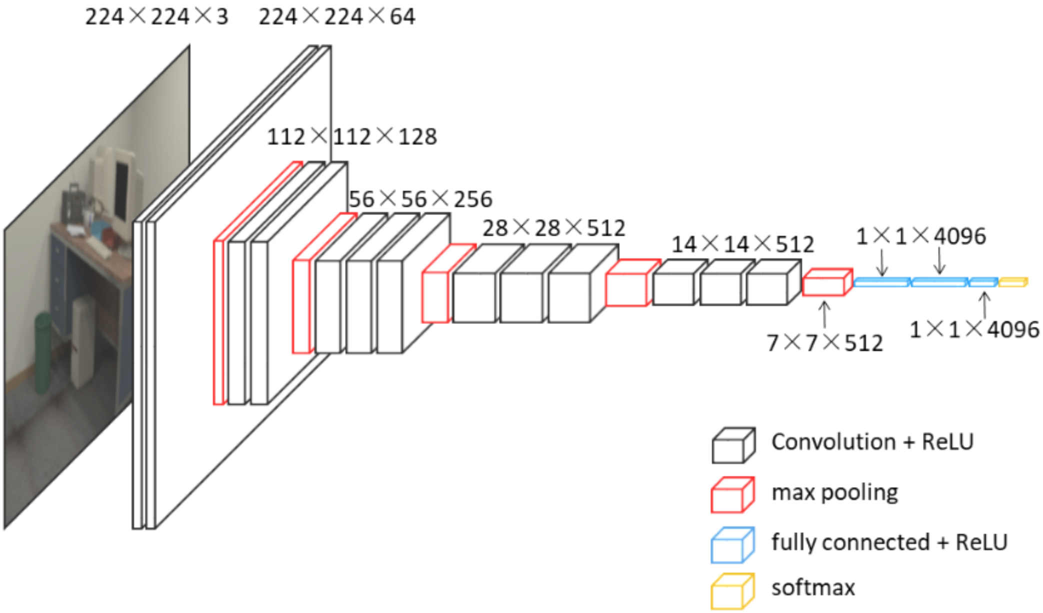


Figure 2. Architecture of VGG16.

It is hard to train a valid DCNN model only by data we collected since deep learning needs a mass of training data. In the proposed scheme, we use CNN for image feature extraction and apply the extracted features in a retrieval task, and then get the most similar images related to the query image. In view of the representation power of CNNs, pre-trained networks based on ImageNet can be used in our feature extraction period.

3.3.2. Deep Features Extracted by CNNs

As shown in Figure 1a,b, both the query images and images in database are processed by CNN model. From previous work [14], we know that deeper layers represent higher level of sematic information from the visualization of feature maps. In our experiment, deep features extracted from CNN can better represent the image, therefore competitive accuracy of image retrieval can be achieved.

Convolution layers (including responding ReLU and max pooling) are used to extract features from input images, and these features are robust to scale and translation. Subsequently image features are aggregated into a compact feature vector of fixed length. As is shown in Figure 3, we visualize the first 16 matrices of each layer. We scale layer maps to the same size when visualizing them, but their sizes as well as depths are different among layers as labeled at the left of the figure.

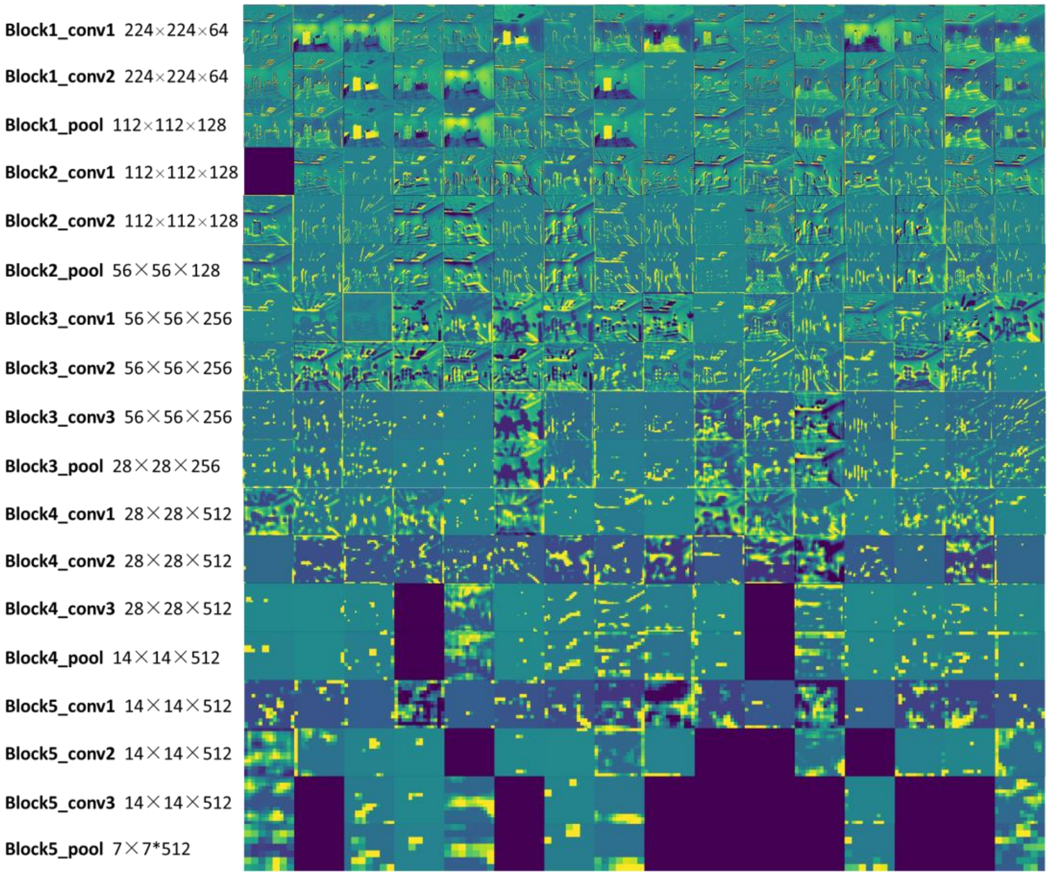


Figure 3. Convolution layers visualization. The first 16 matrices of each layer were visualized, and empty matrices corresponding to dropped out part in CNN. In order to better visualize features in layers, a viridis color map was employed, so layer maps looked greenish.

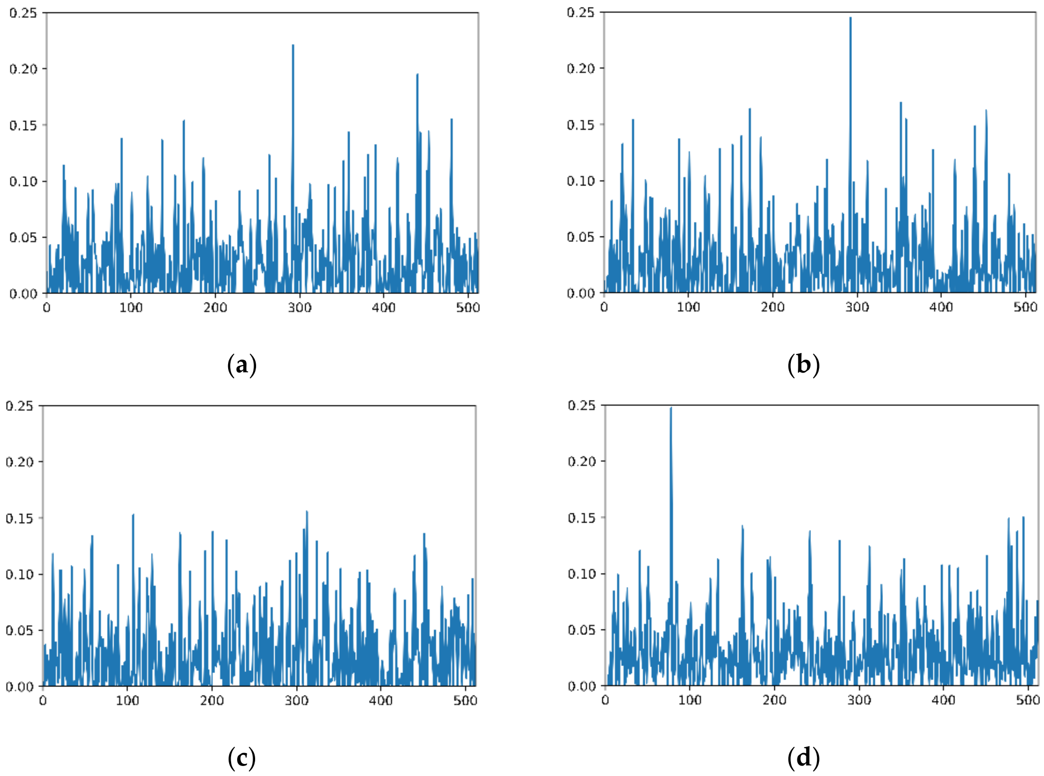
3.3.3. Image Retrieval Using Deep Features

Image features are aggregated into a vector of fixed length after feature extraction period. If we apply the same CNN model to extract features to the same size images, we will get the same fixed length of feature vectors, as shown in Figure 4. When comparing two images, we calculate the distance between image feature vector of retrieved image $(vector_i)$ and vector of the query image $(vector_q)$, where i $\in [1, \mathrm{n}]$ and n refers to the number of images in database. For a query image, we apply Equation (3) to calculate its scores with every image in database. Then, the images in the database are ranked in descending order of scores. At the end of this period, we output certain number of the most similar images as retrieved images related to the query image.

$$vector_i = \left[vector_i^0,\ vector_i^1, \ldots, vector_i^{512}\right], \tag{1}$$

$$vector_q = \left[vector_q^0,\ vector_q^1, \ldots, vector_q^{512}\right], \tag{2}$$

$$\mathrm{score}_i = vector_i * {vector_q}^T, \tag{3}$$



(**a**) (**b**) (**c**) (**d**)

Figure 4. Image feature vectors visualization. (**a**) A shows the vector (512 dimensions) of the query image; (**b**) shows the vector of retrieved image with a top score; (**c**) shows the vector of an unrelated image in the same scene; and (**d**) shows the vector of an image from a different scene.

3.4. Pose Estimation

As shown in Figure 1c, the last step of our visual indoor localization process is pose recovery of the query image. In image retrieval period, we get two of the most similar geo-tagged images related to the query image. In this period, the method for pose estimation mainly consists of the following four steps: Firstly, key-points and descriptors are extracted to mathematically express these three images. Secondly, the transformation from 2D-to-2D matches between images is calculated. Thirdly, both the transformation and the pose of the two retrieved images are utilized to compute the scale of monocular vision. Finally, pose of the query images is computed by applying the monocular vision scale and the transformation between the query image and the most similar image. It should be noted that, target images and images in database may captured by different cameras, and the difference in intrinsic parameters may affect the localization performance. Therefore, camera calibration is required beforehand in order to achieve more accurate positioning result. Furthermore, in consideration of top 2 retrieved images may fail to estimate the pose of the target image, all images in database are re-ranked in retrieval period. If the transformation from 2D-to-2D matches between images is calculated failed, we use the next ranked retrieved image for transformation computation, until pose of the target image could be figured out.

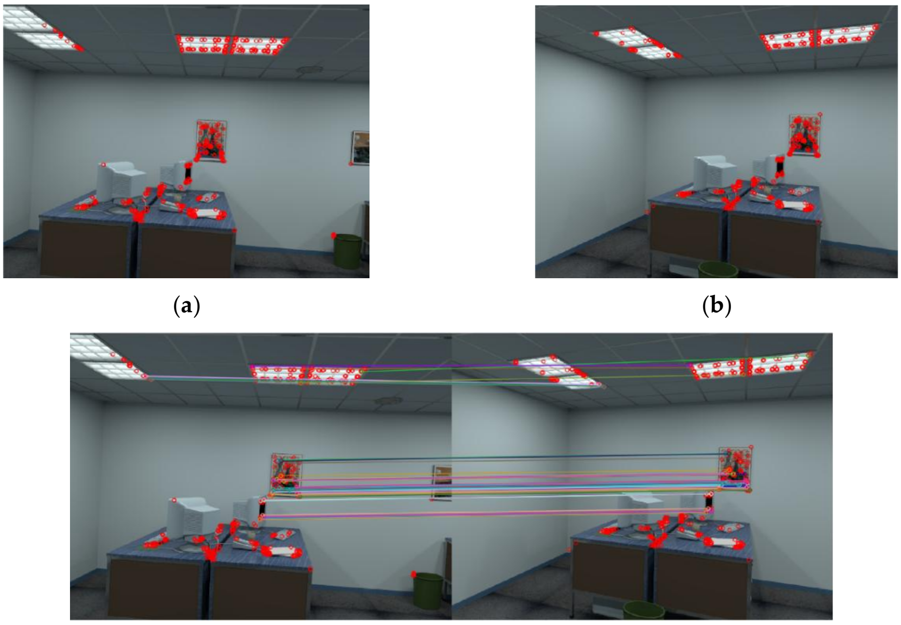
3.4.1. Feature Detection and Matching

The proposed system aims at recovering pose of the query image based on nearby image. However, the image is a matrix of brightness and color, and it is hard to compute the transformation between images by whole matrix. The most common approach to this issue is searching for salient key-points that can be used to match well in other images.

There are many point-feature detectors, such as corner detector (e.g., Moravec [49], FAST [50]) and blob detectors (SIFT [51], SURF [52], CENSURE [53]), their pros and cons can be found in Reference [54].

SIFT fully considers the illumination, scale, rotation and other changes in the image transformation, and achieves great performance in many positioning applications such as in Reference [46], however it can result in large computational cost. For the proposed system which is designed for real-time positioning and LBS, SIFT cannot satisfy the requirements. The time of extracting the same number of features from the same image by SIFT, SURF and ORB [55] is compared in Reference [55]. In that work, when extracting roughly 1000 features, SIFT takes 5228.7 ms, SURF takes 217.3 ms, while ORB only takes 15.3 ms. Accordingly, we adopt ORB (Oriented FAST and Rotated BRIEF) as our detector of point features.

After extraction of ORB features from a pair of images, we use Hamming distance as distance measurement to match features. The results of feature detection and matching from two images are shown in Figure 5.



(**a**) (**b**)

(**c**)

Figure 5. Results of feature detection and matching. (**a**,**b**) show key-points detected in a pair of images; and (**c**) shows the first 50 matches.

3.4.2. Motion from Image Feature Correspondences

In our proposed system, only RGB images are captured by cameras, therefore, feature correspondence is in two dimensions. This section explains the method of computing the transformation T_K between two images I_{K-1}, I_K from two sets of corresponding features f_{K-1}, f_K.

For calibrated cameras, the geometric relationship between two images I_{K-1} and I_K can be described by essential matrix E. The rotation and translation can directly be extracted from E which can be computed from 2D-to-2D feature correspondence. E contains an unknown scale factor of the transformation parameters in the following form:

$$E = \hat{t}R, \tag{4}$$

where $t = \left[t_x, t_y, t_z\right]^T$ and

$$\hat{t} = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}, \tag{5}$$

E can be computed from 2D-to-2D feature correspondence by using the epipolar constraint. Reference [56] illustrated a minimal case solution, which involves the correspondence of five pairs of points', and Reference [57] proposed an efficient implementation of this five-point-algorithm. In References [58,59], an eight-point-algorithm is created for the $n \geq 8$ noncoplanar points, which is summarized below. For one pair of points with normalized coordinates $x_1 = [u_1, v_1, 1]^T$ and $x_2 = [u_2, v_2, 1]^T$, according to epipolar constraint as Equation (6), where $E = [e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9]^T$.

$$[u_1u_2, u_1v_2, u_1, v_1u_2, v_1v_2, v_1, u_2, v_2, 1]E = 0, \tag{6}$$

Stacking the constraints from eight points gives the linear equation system as Equation (7), and the parameters of E can be computed by solving this system.

$$\begin{pmatrix} u_1^1u_2^1 & u_1^1v_2^1 & u_1^1 & v_1^1u_2^1 & v_1^1v_2^1 & v_1^1 & u_2^1 & v_2^1 & 1 \\ u_1^2v_2^2 & u_1^2v_2^2 & u_1^2 & v_1^2u_2^2 & v_1^2v_2^2 & v_1^2 & u_2^2 & v_2^2 & 1 \\ & \vdots & & \vdots & & & \vdots & & \\ u_1^8v_2^8 & u_1^8v_2^8 & u_1^8 & v_1^8u_2^8 & v_1^8v_2^8 & v_1^8 & u_2^8 & v_2^8 & 1 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \\ e_8 \\ e_9 \end{pmatrix} = 0, \tag{7}$$

The rotation and translation can be extracted from E using singular value decomposition (SVD). A valid essential matrix after SVD is $E = USV^T$. Generally, four different solutions for R, t for one E; however, by triangulation of a single point, the correct R, t can be determined. The four solutions are:

$$R = U\left(\pm W^T\right)V^T, \tag{8}$$

$$\hat{t} = U\left(\pm W^T\right)SU^T, \tag{9}$$

where

$$W^T = \begin{bmatrix} 0 & \pm 1 & 0 \\ \mp 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{10}$$

The transformation can be described by transform matrix T in the following form:

$$T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}, \tag{11}$$

3.4.3. Scale Determination

The main property of 2D-to-2D motion estimation is the epipolar constraint, which is based on the constraint of zero equation. Therefore, the equivalence is valid when essential matrix multiplies a multiplicative scalar. In other words, the essential matrix lacks scale to merely correspond to real scenario. This section explains the method to determine the scale by two reference images.

In the image retrieval periods, two most similar images I_1, I_2 are output. With camera calibrated, the scale can be computed from two images transformation and given pose.

By applying epipolar constraint to image I_1, I_2, we can get the transform matrix T_{12} from I_1 to I_2:

$$T_{12} = \begin{bmatrix} R_{12} & t_{12} \\ 0 & 1 \end{bmatrix}, \tag{12}$$

Images I_1, I_2 can be written in the form of homogeneous coordinate. Take image I_1 as example, the pose is $P_1 = \{x_1, y_1, z_1, qx_1, qy_1, qz_1, qw_1\}$. Its translation is $t_1 = [x_1, y_1, z_1]^T$ and the rotation R_1 can be donated as Equation (13). Then image I_1 can be represented by transform matrix T_1 as Equation (14).

$$R_1 = \begin{bmatrix} 1 - 2qy_1{}^2 - 2qz_1{}^2 & 2qx_1qy_1 - 2qw_1qz_1 & 2qx_1qz_1 + 2qw_1qy_1 \\ 2qx_1qy_1 + 2qw_1qz_1 & 1 - 2qx_1{}^2 - 2qz_1{}^2 & 2qy_1qz_1 - 2qw_1qx_1 \\ 2qx_1qz_1 - 2qw_1qy_1 & 2qy_1qz_1 + 2qw_1qx_1 & 1 - 2qx_1{}^2 - 2qy_1{}^2 \end{bmatrix}, \tag{13}$$

$$T_1 = \begin{bmatrix} R_1 & t_1 \\ 1 & 1 \end{bmatrix}, \tag{14}$$

Transformation T'_{12} from image I_1 to image I_2 can be computed from a function as Equation (15), where $inv(T_1)$ is the inverse of matrix T_1. The relationship of transformation T'_{12} and its corresponding rotation R'_{12} and translation t'_{12} is in Equation (16).

$$T'_{12} = inv(T_1)T_2, \tag{15}$$

$$T'_{12} = \begin{bmatrix} R'_{12} & t'_{12} \\ 0 & 1 \end{bmatrix}, \tag{16}$$

Comparing T'_{12} with transform matrix T_{12} computed from 2D-to-2D feature correspondence, we find the rotation R_{12} almost equals to R'_{12}, however t_{12} and t'_{12} show a great difference and contain a scale as Equation (17), where s is the scale.

$$t'_{12} = s\, t_{12}, \tag{17}$$

3.4.4. Pose Estimation of the Query Image

In Section 3.4.2, computation of transform matrix from two images is illustrated, and Section 3.4.3 gave the method of computing scale from two retrieved images. As the poses of retrieved images are known, the transform matrix can be calculated by one of the retrieved images. In our method, we use two of the most similar images I_1 and I_2 to determine the scale s, then estimate the query image's pose P_0 from the transform matrix T_{10} which is computed from the most similar image I_1 and the query image I_0, and the relationship between the pose P_0 of image I_0 and corresponding transform matrix from T_0 is illustrated in Equation (18). P_0 can be denoted as $P_0 = \{x_0, y_0, z_0, qx_0, qy_0, qz_0, qw_0\}$.

$$T_0 = \begin{vmatrix} 1 - 2qy_0{}^2 - 2qz_0{}^2 & 2qx_0qy_0 - 2qw_0qz_0 & 2qx_0qz_0 + 2qw_0qy_0 & x_0 \\ 2qx_0qy_0 + 2qw_0qz_0 & 1 - 2qx_0{}^2 - 2qz_0{}^2 & 2qy_0qz_0 - 2qw_0qx_0 & y_0 \\ 2qx_0qz_0 + 2qw_0qy_0 & 2qy_0qz_0 + 2qw_0qx_0 & 1 - 2qx_0{}^2 - 2qy_0{}^2 & z_0 \\ 0 & 0 & 0 & 1 \end{vmatrix}, \tag{18}$$

4. Experimental Evaluation

4.1. Data Acquisition

For our experiment setup, we utilize images and their poses from trajectories for visual odometer since the task of visual positioning is similar to visual odometer tasks. In this experiment, the ICL-NUIM dataset [60] and the TUM RGB-D dataset [61] are adopted.

ICL-NUIM: A dataset consists of RGB-D images from camera trajectories from two indoor scenes, *living room* and *office room*. The images were collected by a handheld Kinect RGB-D camera and the ground truth of trajectories was obtained by using Kintinuous [62]. The images were captured at 640 × 480 resolutions. Four trajectories were recorded in each scene, and images were taken at different positions for different trajectories. Images obtained at different pose are shown in Figures 6 and 7.

TUM RGB-D: A dataset contains the color and depth images of a Microsoft Kinect sensor and the ground-truth trajectory of camera pose with the goal of establishing a benchmark for the evaluation of visual SLAM systems. The images are at a resolution of 640 × 480 and ground-truth trajectory was obtained from high accuracy motion-caption system. The dataset consists of 89 sequences from different camera motions.

These datasets are employed to verify the performance of the proposed method, and images deal with different scales of captured area vary in capabilities of representing the scenario in different levels. Among these datasets, as shown in Figure 6, images in office scene of ICL-NUIM can represent a larger area such as half a room, whereas as shown in Figure 8, area represented in TUM RGB-D varies from a corner to part of room. As illustrated in Table 2, we choose a part of images in the dataset to represent scenarios, the number of train images and test images are also shown in this table. It should be noted that, database images are hand-picked to cover the application scenarios. The intrinsic parameters of the RGB camera can be obtained from Reference [63].

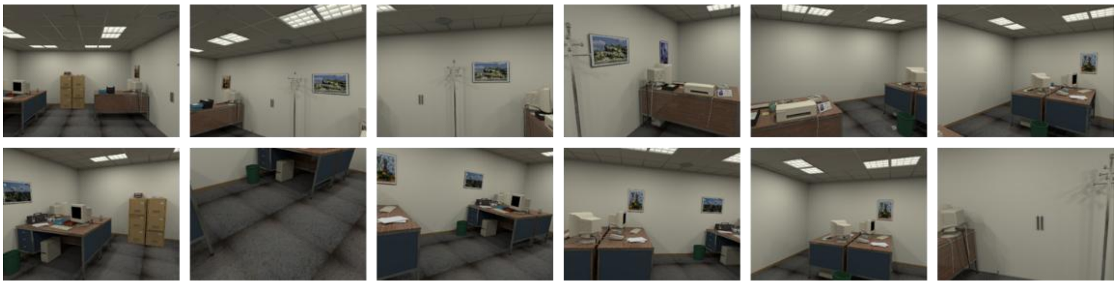


Figure 6. Images in office room scene of ICL-NUIM dataset.

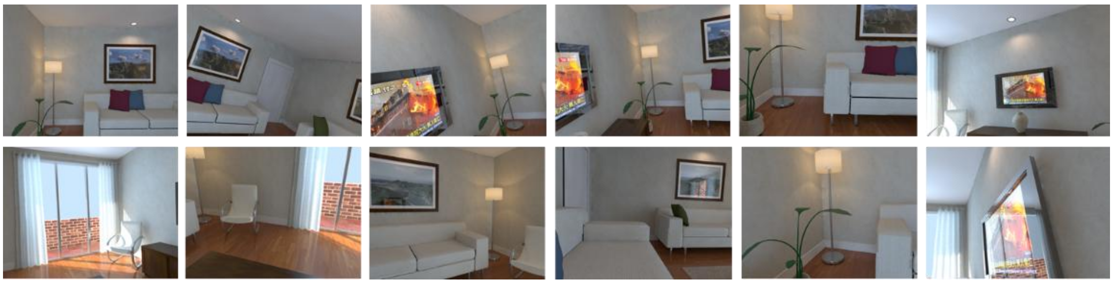


Figure 7. Images in living room scene of ICL-NUIM dataset.

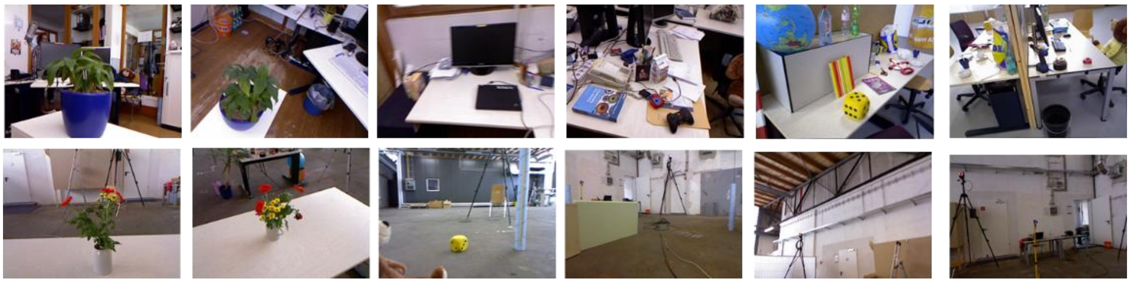


Figure 8. Images in TUM RGB-D dataset.

Table 2. We chose images from different scenarios of ICL-NUIM and TUM RGB-D to compose our own experiment dataset.

Dataset	Scenario	The Number of Raw Images	The Number of Database Images	The Number of Test Images
ICL-NUIM	office room	4602	289	1495
	living room	4602	304	1533
TUM RGB-D	freiburg1_plant	1141	115	456
	freiburg1_room	1362	91	454
	freiburg2_360_hemisphere	2729	273	1092
	freiburg2_flowerbouquet	2972	149	1188
	freiburg2_pioneer_slam3	2544	128	1017
	freiburg3_long_office_household	2585	130	1034

4.2. Performance of Image Retrieval

In this section, we evaluate the image retrieval performance by means of feature extraction methods on our database. It is important to note that, the process of evaluating the performance of image retrieval is time-consuming since feature extraction period is expensive, nevertheless, this process is not needed by our applying visual positioning algorithm.

Generally, mean average precision (mean AP) is applied to evaluate the performance of image retrieval task quantitatively, which compares the query image and the top retrieved images belonging to the same categories. The comparison of traditional image retrieval methods with CNN-based image retrieval methods has illustrated in Reference [29,64]. However, in the procedure of proposed image retrieval based visual positioning, mean AP cannot effectively demonstrate the retrieval result efficiently. Due to the feature extraction and matching period can affect the result of pose estimation heavily, pairs of images should share as many feature points as possible, which makes it essential for the query image and the retrieved images share some common areas. Therefore, we calculate the number of matched features between images to evaluate the result of image retrieval.

To evaluate the performance of image retrieval, we extract feature points and descriptors from each pair of images, and calculate the number of good-match. In our retrieval period, three of the most similar images are returned. We extract ORB features from these retrieved images together with the query image, then match features of the query image and its corresponding retrieved images. Noted that Hamming Distance is employed to compute the distance between ORB descriptors. Then the minimal distance of matched descriptors is computed in all image pairs, and matched feature points whose distance is less than a threshold value can be labeled as good-match. Moreover, when calculating the good-match in ORB descriptors, the threshold value is defined by the larger number between twice of the minimal distance and a constant, since sometimes the minimal Hamming distance can be quite small. As shown in Table 3, a great number of good-matches are detected, which is sufficient for eight-point-algorithm in pose estimation period. The experiment results show that top-ranked similar images share more good-match.

Table 3. The average number of ORB Oriented FAST and Rotated BRIEF) good-match in two datasets.

Similarity Rank	ICL-NUIM Dataset (Test on 3026 Images)	TUM RGB-D Dataset (Test on 5241 Images)
1	252.8	225.3
2	203.1	135.9
3	158.4	104.0
Average	204.8	155.1

In Reference [65], Jason et al. also developed an image based indoor localization scheme which uses FLANN search on SIFT features, and their experiment only successfully matched 78 out of 83 images to achieve a 94% retrieval accuracy. Whereas, in our proposed method, which aided by CNN-based image retrieval, achieved more than 99% image retrieval rate of 8267 images (the output images share the common area with the query image) owing to CNN features have more powerful representations for images.

4.3. Localization Results and Analysis

Figures 9 and 10 summarize the performance of pose estimation stage of proposed scheme. As shown in Figure 9a, our method is able to localize the position within sub-meter level of accuracy for over 90% of the query images in both datasets. Furthermore, more than 80% of the query images are successfully localized within 0.25 m of the ground truth position. As shown in Figure 9b, about 90% of the query images are localized within 3 degrees of ground truth position. We reported the performance in terms of the median errors of translation and orientation for each scene in the datasets, as shown in Table 4. The median errors of translation of our proposed method are around sub-meter level, which the 90% accuracy is 0.28 m in ICL-NUIM dataset and 0.45 m in TUM RGB-D dataset. The median error of orientation is within 1° and the 90% accuracy is 0.94° for ICL-NUIM dataset and 2.03° for TUM RGB-D dataset. It is important to note that the statistics in Table 4 has not removed the outliers, which enlarged the mean error of localization.

The proposed localization method combines CNN features and point features to estimate the pose. We compare the accuracy of the proposed method with the average pose estimation errors of three different CNN-based localization methods: (i) PoseNet which directly regress the camera pose by CNN; (ii) 4D PoseNet which was modified from PoseNet to accommodate the RGB-D input; (iii) CNN+LSTM [42], which utilize the PoseNet as a baseline pose estimator and the LSTM works as a temporal filter to process the estimated pose sequence. Table 5 summarizes statistics of the average pose estimation errors from those methods on ICL-NUIM dataset. We achieved better position accuracy on both scenarios, and achieved comparable accuracy in orientation.

More importantly, compared with state-of-the-art learning-based methods, the proposed scheme uses much fewer images in database construction period, which is significant for generalizing application of visual positioning. As is known, learning-based methods need quantity of images with poses to train a model (see Table 6). However, in proposed visual localization scheme, much fewer images are required. Less than 10% images are needed compared to learning-based methods, and we can still achieve comparable localization accuracy. Furthermore, CNN features can reduce the cost of model storage. Our CNN features model takes up 1.8 M storage for 886 images from TUM RGB-D database, which raw images occupy memory of 419.4 M, and 1.2 M storage is needed for 593 images from ICL NUIM database, which raw RGB images need 175.1 M.

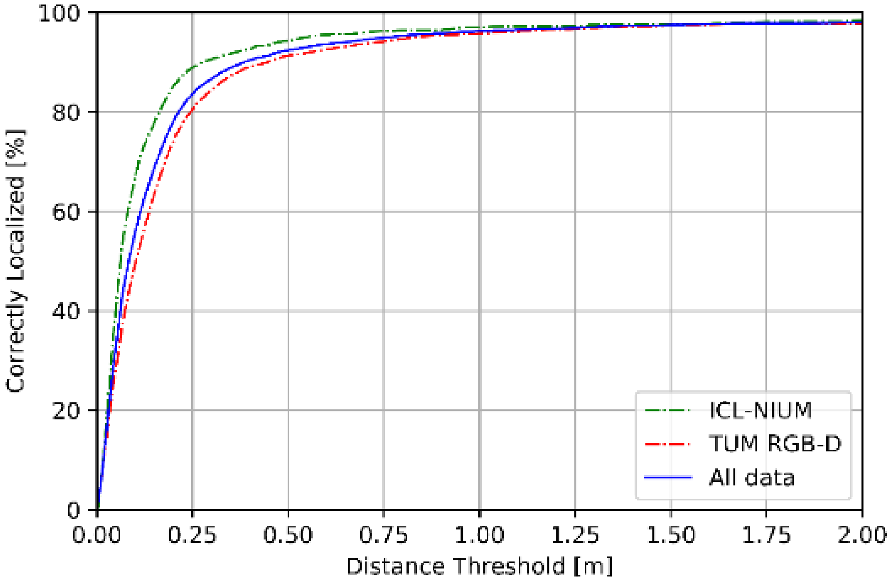


Figure 9. Cumulative distribution function of location error.

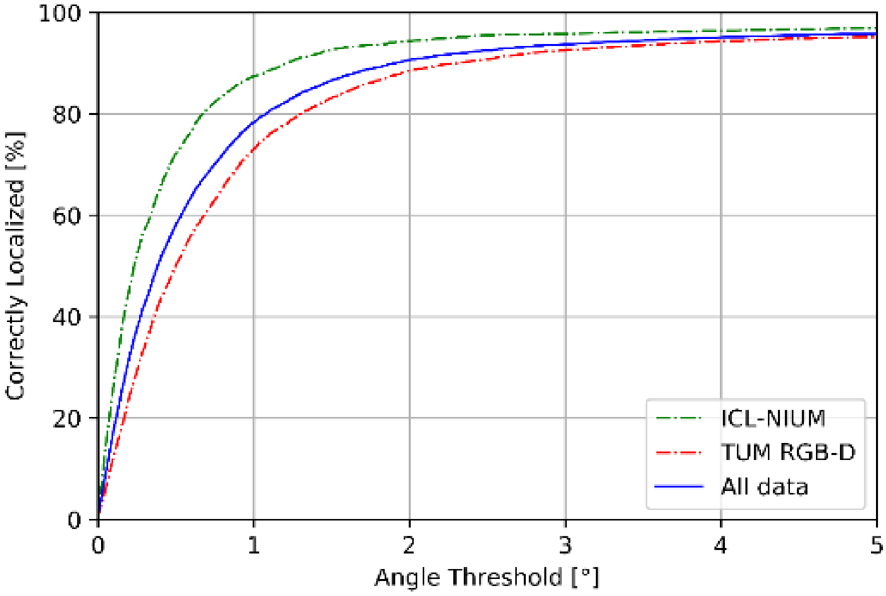


Figure 10. Cumulative distribution function of angle error.

Table 4. Localization performance in different scenarios from different datasets.

Dataset	Scenario	The Median Error	The Mean Error	90% Accuracy
ICL-NUIM	office room	0.07 m 0.01°	0.31 m 2.47°	0.35 m 0.83°
	living room	0.05 m 0.02°	0.36 m 4.36°	0.23 m 1.03°
	All images	0.06 m 0.01°	0.34 m 3.43°	0.28 m 0.94°
TUM RGB-D	freiburg1_plant	0.12 m 0.01°	0.38 m 3.37°	0.45 m 1.95°
	freiburg1_room	0.17 m 0.54°	0.43 m 4.82°	0.71 m 4.04°
	freiburg2_360_hemisphere	0.05 m 0.16°	0.38 m 6.55°	0.38 m 1.08°
	freiburg2_flowerbouquet	0.07 m 0.12°	0.15 m 5.32°	0.26 m 2.54°
	freiburg2_pioneer_slam3	0.13 m 0.13°	0.34 m 8.80°	0.66 m 1.54°
	freiburg3_long_office_household	0.15 m 0.21°	0.36 m 3.00°	0.41 m 2.05°
	All images	0.10 m 0.16°	0.32 m 5.58°	0.45 m 2.03°

Table 5. Comparison of average pose estimation error in ICL-NUIM dataset.

Method	Living Room	Office Room
PoseNet	0.60 m, 3.64°	0.46 m, 2.97°
4D PoseNet	0.58 m, 3.40°	0.44 m, 2.81°
CNN+LSTM [42]	0.54 m, 3.21°	0.41 m, 2.66°
ours	**0.36 m**, 4.36°	**0.31 m, 2.47°**

Table 6. Comparison of database sizes.

Method	Database Image per Scene	Median Localization Error
PoseNet	3000	0.47 m 14.40°
	6000	0.48 m 7.68°
NNnet [66]	2000	0.27 m 11.82°
	4000	0.24 m 6.35°
VLocNet	2000	0.097 m 6.48°
	4000	0.036 m 1.71°
ours	148	0.102 m 0.164°
	289	0.062 m 0.011°

We implemented the proposed localization scheme on Intel Core i7-7700 CPU @ 3.60 GHz. It takes 324.5 ms on average to find 10 best matches for a single image on 593 images of ICL NUIM database, and 336.5 ms to find 10 best matches on 886 images of TUM RGB-D database. Pose estimation costs 88.5 ms on average. The whole procedure from image retrieval to pose estimation takes ~0.45 s to output a location for a single image. We also employed a NVIDIA TITAN XP GPU to accelerate the computation of image retrieval, and 20 ms and 12 ms are taken to find 10 best matches of ICL NUIM and TUM RGB-D respectively. The whole procedure takes ~0.1 s for a single image. Computation and storing the CNN presentation of database images are done offline, and the period of retrieval evaluation, which is illustrated in Section 4.2, is unnecessary in localization procedure.

5. Discussion

In this study, we presented an image retrieval aided approach for indoor visual localization. A CNN-based image retrieval method was adopted to recognize the given query images by retrieving the matching images that were geo-tagged. The CNN-based strategy not only provides the output images with high spatial correlation, but also gives a new idea of scene representation. To put it another way, we no longer have to represent the space by its 3D model, instead we can design its spatial model representing methods in line with the usage of the spatial model. For instance, a group of CNN features with original images and poses can represent the whole area for visual localization purpose. A feature-points-correspondences strategy was then applied to estimate the precise location and pose of the query image. Experimental results demonstrated that our *monocular visual odometer*-inspired pose estimation methods resulted in high-precision localization consequence.

It is obvious that our result outperformed that of retrieval-based methods without CNN feature extraction in robustness, due to the complex and unstable indoor environments. Compared with 2D-to-3D and 3D-to-3D methods in pose estimation methods, our strategy only depends on calibrated monocular camera in online localization period. Furthermore, our method is free of building 3D models, which the process is considered as an expensive process. Compared with end-to-end learning-based methods that directly regress the pose from input images, the advantage of our method exists in offline preparing period. As is known, deep learning has become an extremely powerful tool in computer vision tasks, but due to deep learning is a kind of data-hungry method, a massive of high-quality training data is required. Experimental results have shown that when achieving comparable localization accuracy, the number of database images of our method is far smaller than that in learning-based methods. Besides, learning-based strategies heavily rely on the training stage with quantity of geo-tagged images, it is assumed that when expanding the applicant area will cause the whole model retrained as well as the growth of model size, whereas in our methods, the increasing of raw data has no effect on the extractor, but only correspondingly add to the database.

Moreover, owing to the scheme of data acquisition and the process of image-based localization, our method has massive potential to extend to a crowdsourcing-based method. The raw data from different resources can be integrated to compose the database. In the indoor environment, images are captured by cameras on cellphones, robots or other platforms, and the pose information can be obtained through pose measuring infrastructures. In fact, our proposed scheme is not a data-hungry solution as DCNNs, a set of limited images with high-precision pose is the key. In the future, the problems we need to address are the strategy of defining the space by a set of images and the approach to get high-precision pose information for database images. For image retrieval phase, a more efficient and robust method as well as more complicated and larger scale of environment needs to be considered in the future work.

6. Conclusions

In summary, our solution is highly available to different and complex environments and easily extendable to the change of raw data. We utilize a CNN-based image retrieval strategy which represents the scene by CNN features, and match the query image with database images. After that, the pose of the query image is recovered from the ORB feature points' correspondence, which is efficient and effective.

Based on the state-of-the-art studies of indoor visual localization systems, to the best of our knowledge, this work is the first to adopt both CNN-based image retrieval strategy and merely RGB images for accurate localization which is highly applicable to monocular vision positioning task. We think the image-based localization methods may become the mainstream owing to the scheme of data acquisition and the algorithm of pose estimation accorded with the current state of data expansion. The from-coarse-to-accurate strategy will be efficiently adopted to much larger applied range.

Author Contributions: Conceptualization, Y.C. and R.C.; Data curation, Y.C., A.X. and D.W.; Formal analysis, Y.C.; Funding acquisition, R.C.; Investigation, Y.C. and M.L.; Methodology, Y.C.; Project administration, R.C.; Resources, R.C.; Software, Y.C., M.L. and A.X.; Supervision, R.C.; Validation, Y.C., M.L., A.X. and D.W.; Visualization, Y.C.; Writing—original draft, Y.C.; Writing—review & editing, Y.C., R.C., M.L. and S.Z.

Funding: This research was funded by the National Key Research and Development Program of China (grant number 2016YFB0502201).

Acknowledgments: In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).

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

References

1. Liu, J.; Chen, R.; Pei, L.; Guinness, R.; Kuusniemi, H. A Hybrid Smartphone Indoor Positioning Solution for Mobile LBS. *Sensors* **2012**, *12*, 17208–17233. [CrossRef] [PubMed]
2. Youssef, M.; Agrawala, A. The Horus WLAN location determination system. In Proceedings of the International Conference on Mobile Systems, Applications, and Service, Seattle, WA, USA, 6–8 June 2005; pp. 205–218.
3. Yang, S.; Dessai, P.; Verma, M.; Gerla, M. Freeloc: Calibration-free crowdsourced indoor localization. In Proceedings of the 32nd IEEE International Conference on Computer Communications, Turin, Italy, 14–19 April 2013; pp. 2481–2489.
4. Mautz, R.; Tilch, S. Survey of optical indoor positioning systems. In Proceedings of the International Conference on Indoor Positioning and Indoor Navigation, Guimaraes, Portugal, 21–23 September 2011; pp. 1–7.
5. Nirmalathas, A.; Lim, C.; Skafidas, E.; Alameh, K.; Wang, K. Optical Wireless-Based Indoor Localization System Employing a Single-Channel Imaging Receiver. *J. Lightw. Technol.* **2016**, *34*, 1141–1149. [CrossRef]
6. Xu, H.; Ding, Y.; Li, P.; Wang, R.; Li, Y. An RFID Indoor Positioning Algorithm Based on Bayesian Probability and K-Nearest Neighbor. *Sensors* **2017**, *17*, 1806. [CrossRef] [PubMed]
7. Pei, L.; Chen, R.; Liu, J.; Tenhunen, T.; Kuusniemi, H. Inquiry-Based Bluetooth Indoor Positioning via RSSI Probability Distributions. In Proceedings of the 2nd International Conference on Advances in Satellite and Space Communications, Athens, Greece, 13–19 June 2010; pp. 151–156.
8. Larranaga, J.; Muguira, L.; Lopez-Garde, J.M.; Vazquez, J.I. An environment adaptive ZigBee-based indoor positioning algorithm. In Proceedings of the 2010 International Conference on Indoor Positioning and Indoor Navigation, Zurich, Switzerland, 15–17 September 2010; pp. 1–8.
9. Uradzinski, M.; Guo, H.; Liu, X.; Yu, M. Advanced Indoor Positioning Using Zigbee Wireless Technology. *Wireless. Pers. Commun.* **2017**, *3*, 1–10. [CrossRef]
10. Niwa, H.; Kodaka, K.; Sakamoto, Y.; Otake, M. GPS-based indoor positioning system with multi-channel pseudolite. In Proceedings of the IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 905–910.
11. Yun, S.; Yao, Z.; Wang, T.; Lu, M. High accuracy and fast acquisition algorithm for pseudolites-based indoor positioning systems. In Proceedings of the Fourth International Conference on Ubiquitous Positioning, Indoor Navigation and Location Based Services, Shanghai, China, 2–4 November 2016; pp. 51–60.
12. Quattoni, A.; Torralba, A. Recognizing Indoor Scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009), Miami, FL, USA, 20–25 June 2009.

13. Espinace, P.; Kollar, T.; Roy, N.; Soto, A. Indoor scene recognition by a mobile robot through adaptive object detection. *Rob. Auton. Syst.* **2013**, *61*, 932–947. [CrossRef]
14. Zhou, B.; Lapedriza, A.; Xiao, J.; Torralba, A.; Oliva, A. Learning deep features for scene recognition using places database. In Proceedings of the IEEE International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 487–495.
15. Khan, S.; Hayat, M.; Bennamoun, M.; Sohel, F.; Togneri, R. A Discriminative Representation of Convolutional Features for Indoor Scene Recognition. *IEEE Trans. Image. Process.* **2016**, *25*, 3372–3383. [CrossRef] [PubMed]
16. 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]
17. 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.
18. Hays, J. IM2GPS: Estimating geographic information from a single image. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
19. 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.
20. 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, Honolulu, HI, USA, 21–26 July 2017; pp. 6175–6184.
21. Li, Y.; Snavely, N.; Dan, H.; Fua, P. Worldwide pose estimation using 3D point clouds. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 15–29.
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. [CrossRef] [PubMed]
23. Svarm, L.; Enqvist, O.; Kahl, F.; Oskarsson, M. City-Scale Localization for Cameras with Known Vertical Direction. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1455–1461. [CrossRef] [PubMed]
24. Mikulík, A.; Perdoch, M.; Chum, O.; Matas, J. Learning a Fine Vocabulary. In Proceedings of the European Conference on Computer Vision, Heraklion, Greece, 5–11 September 2010; pp. 1–14.
25. Tolias, G.; Avrithis, Y. Image Search with Selective Match Kernels: Aggregation across Single and Multiple Images. *Int. J. Comput. Vis.* **2016**, *116*, 262. [CrossRef]
26. Tolias, G.; Jégou, H. Visual query expansion with or without geometry: Refining local descriptors by feature aggregation. *Pattern Recognit.* **2014**, *47*, 3466–3476. [CrossRef]
27. Liew, W.C.; Law, N.F. Content-Based Image Retrieval. *Comput. Syst. Appl.* **2009**, *43*, 85.
28. Gong, Y.; Wang, L.; Guo, R.; Lazebnik, S. Multi-scale Orderless Pooling of Deep Convolutional Activation Features. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 392–407.
29. 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 Recognition Workshops, Columbus, OH, USA, 23–28 June 2014; pp. 512–519.
30. Babenko, A.; 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.
31. Tolias, G.; Sicre, R.; Jégou, H. Particular object retrieval with integral max-pooling of CNN activations. *Comput. Sci.* **2015**.
32. Brachmann, E.; Krull, A.; Michel, F.; Gumhold1, S.; Shotton, J.; Rother, C. Learning 6D Object Pose Estimation Using 3D Object Coordinates. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 536–551.
33. Brachmann, E.; Krull, A.; Nowozin, S.; Shotton, J.; Michel, F.; Gumhold, S.; Rother, C. DSAC—Differentiable RANSAC for Camera Localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2492–2500.

34. 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.
35. 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, Portland, OR, USA, 23–28 June 2013; pp. 2930–2937.
36. Kendall, A.; Grimes, M.; Cipolla, R. PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization. *Educ. Inf.* **2015**, *31*, 2938–2946.
37. 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.
38. Valada, A.; Radwan, N.; Burgard, W. Deep Auxiliary Learning for Visual Localization and Odometry. In Proceedings of the IEEE International Conference on Robotics and Automation, Brisbane, Australia, 21–25 May 2018.
39. Kendall, A.; Cipolla, R. Modelling uncertainty in deep learning for camera relocalization. In Proceedings of the IEEE International Conference on Robotics and Automation, Stockholm, Sweden, 16–21 May 2016; pp. 4762–4769.
40. Irschara, A.; Zach, C.; Frahm, J.M.; Bischof, H. From structure-from-motion point clouds to fast location recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 2599–2606.
41. Atanasov, N.; Zhu, M.; Daniilidis, K.; Pappas, G.J. Localization from semantic observations via the matrix permanent. *Int. J. Rob. Res.* **2016**, *35*, 73–99. [CrossRef]
42. Guo, F.; He, Y.; Guan, L. RGB-D camera pose estimation using deep neural network. In Proceedings of the IEEE Global Conference on Signal and Information Processing. Montreal, QC, Canada, 14–16 November 2017; pp. 408–412.
43. Cohen, A.; Schönberger, J.L.; Speciale, P.; Sattler, T.; Frahm, J.; Pollefeys, M. Indoor-outdoor 3D reconstruction alignment. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 285–300.
44. Liu, M.; Chen, R.; Li, D.; Chen, Y.; Guo, G.; Cao, Z.; Pan, Y. Scene Recognition for Indoor Localization Using a Multi-Sensor Fusion Approach. *Sensors* **2017**, *17*, 2847. [CrossRef] [PubMed]
45. Micusik, B.; Wildenauer, H. Descriptor free visual indoor localization with line segments. In Proceedings of the IEEE Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3165–3173.
46. Liang, J.Z.; 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 3013; pp. 71–75.
47. Xiao, A.; Chen, R.; Li, D.; Chen, Y.; Wu, D. An Indoor Positioning System Based on Static Objects in Large Indoor Scenes by Using Smartphone Cameras. *Sensors* **2018**, *18*, 2229. [CrossRef] [PubMed]
48. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Comput. Sci.* **2014**.
49. Moravec, H.P. Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover. Ph.D. Thesis, Stanford University, Stanford, CA, USA, 1981.
50. Rosten, E.; Drummond, T. Machine learning for high-speed corner detection. In Proceedings of the European Conference on Computer Vision, Graz, Austria, 7–13 May 2006; pp. 431–443.
51. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
52. Bay, H.; Ess, A.; Tuytelaars, T.; Tuytelaars, T.; Van Gool, L. Speeded-Up Robust Features. *Comput. Vis. Image Underst.* **2008**, *110*, 404–417. [CrossRef]
53. Agrawal, M.; Konolige, K.; Blas, M.R. Censure: Center surround extremas for realtime feature detection and matching. In Proceedings of the European Conference on Computer Vision, Marseille, France, 12–18 October 2008; pp. 112–115.
54. Siegwart, R.; Nourbakhsh, I.; Scaramuzza, D. *Introduction to Autonomous Mobile Robots*, 2nd ed.; MIT Press: Cambridge, MA, USA, 2011; ISBN 9780262195027.

55. 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, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571.
56. Kruppa, E. *Zur Ermittlung Eines Objektes aus Zwei Perspektiven mit Innerer Orientierung*; Abteilung IIa: Wien, Austria, 1913; Volume 122, pp. 1939–1948.
57. Nister, D. An efficient solution to the five-point relative pose problem. In Proceedings of the Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 195–212.
58. Hartley, R.I. In defense of the eight-point algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 580–593. [CrossRef]
59. Longuet-Higgins, H.C. A computer algorithm for reconstructing a scene from two projections. *Read. Comput. Vis.* **1987**, *293*, 61–62.
60. Handa, A.; Whelan, T.; Mcdonald, J.; Davison, A.J. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In Proceedings of the IEEE International Conference on Robotics and Automation, Hong Kong, China, 31 May–7 June 2014; pp. 1524–1531.
61. Sturm, J.; Magnenat, S.; Engelhard, N.; Pomerleau, F.; Colas, F. Towards a benchmark for RGB-D. SLAM evaluation. In Proceedings of the RGB-D Workshop on Advanced Reasoning with Depth Cameras at Robotics: Science & Systems Conference, Shanghai, China, 9–13 May 2011.
62. Whelan, T.; Kaess, M.; Johannsson, H.; Fallon, M.; Leonard, J.J. Real-time large-scale dense RGB-D SLAM with volumetric fusion. *Int. J. Rob. Res.* **2015**, *34*, 598–626. [CrossRef]
63. Camera Parameters. Available online: https://vision.in.tum.de/data/datasets/rgbddataset/file_formats#intrinsic_camera_calibration_of_the_kinect (accessed on 17 June 2018).
64. Babenko, A.; Slesarev, A.; Chigorin, A.; Lempitsky, V. Neural Codes for Image Retrieval. *Comput. Sci.* **2014**, *8689*, 584–599. [CrossRef]
65. Gordo, A.; Almazán, J.; Revaud, J.; Larlus, D. Deep Image Retrieval: Learning Global Representations for Image Search. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 241–257.
66. Zakaria, L.; Iaroslav, M.; Surya, K.; Juho, K. Camera Relocalization by Computing Pairwise Relative Poses Using Convolutional Neural Network. In Proceedings of the IEEE International Conference on Computer Vision Workshop, Venice, Italy, 22–29 October 2017; pp. 920–929.