

## A High Precision Visual Localization Method Optimized by Multi-Features

Yuchen Deng<sup>1</sup>, Shengjun Tang<sup>1,\*</sup>, Weixi Wang<sup>1</sup>, Xiaoming Li<sup>1</sup>, Renzhong Guo<sup>1</sup>

<sup>1</sup> School of Architecture and Urban Planning, Research Institute for Smart Cities, Shenzhen University & Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Natural Resources, Shenzhen, P.R. China –  
dengyuchen2019@email.szu.edu.cn, (shengjuntang, wangwx, lixm, guorz)@szu.edu.cn

### Commission IV, WG IV/5

**KEY WORDS:** Visual Localization, Image Retrieval, Place Recognition, Indoor Localization, Pose Estimation, Image Matching.

#### ABSTRACT:

The demand for indoor localization has increased in fields such as indoor navigation, virtual reality and emergency response. Traditionally, hardware-based indoor positioning methods require a large number of devices to be deployed and require high maintenance costs. Vision-based localization methods offer a low-cost option for this purpose. Visual Localization has two typical pipeline: end-to-end study and traditional pose estimation based on PnP(Perspective- n-point). However, the quality of the retrieved images and 2D-3D correspondences is vital to the precision and recall of the traditional method. In this paper we try to partly overcome the mentioned drawback by eliminate the error retrieval images with multi-features, and we use several retrieved images to collect enough 2D-3D correspondences to improve the robustness against error input. We also filter the outliers during forming the 2D-3D correspondences with RANSAC and Lowe's ratio test. As a supplement to the various indoor visual localization dataset production, we introduce a pipeline which can generate point clouds and mesh model via our integrated RGB-D cameras.

### 1. INTRODUCTION

LBS(Location Based Service) are now applied in everywhere in the daily life: automatic driving, augmented reality, smart cities, robotic navigation and so on. We can get GPS signal to know our location in outdoor environments easily. However, due to the shielding effect in indoor environment(e.g., ceilings, floors), it's hard to receive GPS signal stable enough to perform localization. New positioning methods like Bluetooth(Zafari et al. 2019), Wi-Fi(Yang and Shao, 2015), magnetic fields(Park and Myung, 2014) and ultra-wideband(Monica and Bergenti, 2019) are more and more employed in LBS application. These technology with their own specific advantages, however, are not suitable to some scenarios. Wi-Fi signals are not stable enough can be easily disturbed by the environment. Bluetooth can achieve high precision, but it is not stable and the range of it is limited. Ultra-wideband requires a number of additional devices and high costs, and can be easily disturbed due to temperature factors. Using magnetic fields to perform localization requires frequent update and maintenance of the dataset. Visual localization (Selvaraju et al., 2020) has a wide range of application and does not need to deploy additional facilities. This low-cost technology can also be rather accurate and robust.

There are two leading method of visual localization currently. The first one relies on gaining correspondences of 2D keypoints in the query image and 3D points in the real world. To achieve this goal, local feature matching is applied between the query image and the dataset image(or image retrieval result). Then the pose estimation is based on the correspondences and rigorous geometric deviation. With a rather high precision, though, the traditional method largely relies on the inliers of the 2D-3D correspondences. Another state-of-art method is end-to-end study. It makes use of deep learning to train a network which could estimate pose directly, regardless of procedures like local feature matching, image retrieval and geometric deviation. This method also requires a large training dataset, so virtual views generated from 3D models can create rather big dataset(Acharya et al., 2019).End-to-end study has a good performance when

facing a relative big scene, but its precision still needs improvement.

Hand-crafted features was popular in the past decades, and now more and more state-of-art research introduce features extracted by deep learning. Comparing to hand-crafted features, they have a good performance when trained for a specific scene or area, but most of them lack interpretability, which may not work when applied in a completely different scene. And hand-crafted features are also designed for certain usage, which means they have their own drawbacks in the same time.

In this paper, we first want to introduce a repeatable framework based on RGB-D cameras, since the current researches mainly focus on specific section of visual localization. Secondly, we leverage several retrieved images rather than a single image, because error image retrieval result is fatal to the whole localization process and a single image is not stable enough. What's more, a single image may can not offer enough 2D-3D correspondence inliers to perform pose estimation. However, the increasing number of images involved in local feature matching would definitely lead to additional time and computational costs, so we aim to determine the best number of retrieved image which can balance the precision and speed simultaneously. Thirdly, we employed some tricks to filter outliers in the local feature steps, such as Lowe's ratio test(Lowe, 1999). Fourth, we make use of several hand-crafted features in image retrieval, trying to get more reliable results which would help get better precision.

### 2. RELATED WORK

Building a dataset is the initial task of image-based localization. SfM(Structure-from-Motion) is widely used (Svarm et al., 2016) to build 3D model when the acquired data are unordered image series(mostly by crowdsourcing). But the accumulative error of SfM is fatal and SfM requires the images series to have certain overlap areas, so it raise requirements for data collection. Author Li(2020) used one of the most classic way to build 3D models: taking photos from station to station and using electronic total

station to get the station pose. Then they build 3D point clouds from multi-imagespatial forward intersection. RGB-D (Endres et al., 2014) camera is also applied in the data collection step, but in most research the station poses are collected in advance.

In the earliest researches of image-based localization, query image is compared and matched with exactly the whole dataset((Robertson and Cipolla, 2004). With the expanding amount of data, this method is no longer suitable. Then the prototype of image retrieval is introduced into the field of image-based localization to improve efficiency, such as vocabulary tree (Nistér and Stewénius, 2006), inverted file scoring (Philbin et al., 2007). The idea of BoW(Bag of Words), originated from natural language processing area, is also introduced into image retrieval(DBOW3, 2017). HF-Net(Sarlin et al., 2018), NetVLAD(Arandjelovic et al., 2017) are both state-of-art algorithms to extract global descriptor. Zhang(2020) segment the whole scene into sub-spaces, and each sub- space is represented by an specific geo-tagged image, which also shows the idea of image retrieval.

Though challenged by features extracted by deep learning method like Superpoint (Detone et al., 2017), hand-crafted features remain irreplaceable. SIFT(Lowe, 1999) is of scale invariance and rotate invariance, but the dimension of SIFT is too high, which increase the time cost of the algorithm. ORB, FAST and BRISK are relatively simple features, and they can meet the real-time requirements of some applications(e.g. SLAM).

To obtain the pose, the most classic method is space resection(PnP) and optimized by bundle adjustment, but 2D-3D correspondences are necessary input of this algorithm. End-to-end study skip the long procedure and learn poses from the prebuilt neural networks. PoseNet (Kendall et al., 2015) is a typical model to estimate pose directly, but the mechanism of the network remains a black box, which is the common problems of end-to-end study. Pixloc(Sarlin et al., 2021) overcome the precision issue of end-to-end study to some extent, and the gap between the performance of Pixloc and classic local feature matching method is getting smaller.

### 3. METHODOLOGY

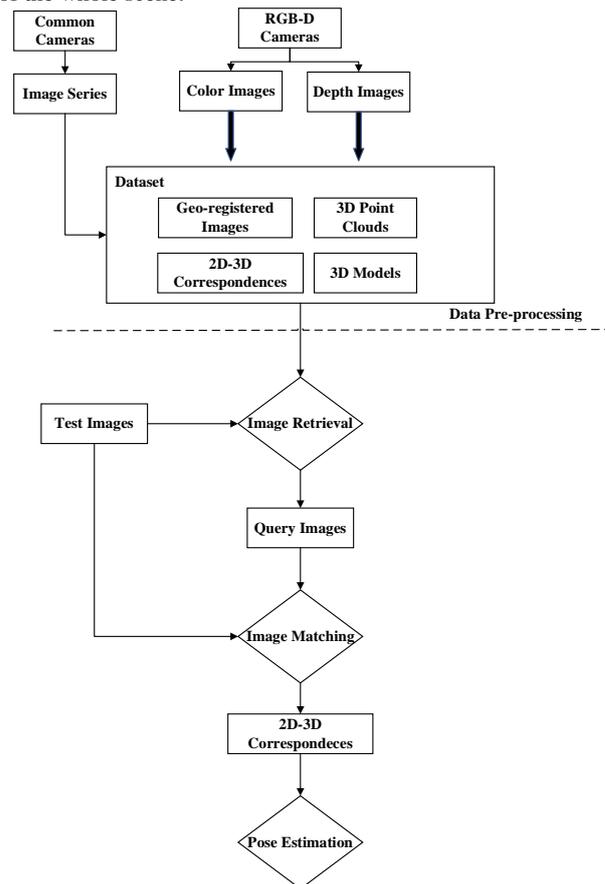
To improve the precision of traditional visual localization method and serve as a supplement of the pipeline, we introduce a relative complete visual localization frameworks from data preparation to pose estimation. The pipeline is shown in the **Figure 1**.

#### 3.1 Data collection

First of all, we want to generate data using our integrated RGB-D cameras(GHO3D,**Figure 2**) instead of using a series of unordered images without other information, since that cumulative error is a vital drawback of the SfM algorithm. Our integrated RGB-D cameras consist of three Kinect RGB-D cameras with fixed relative poses. It could capture colour and depth images of three directions simultaneously, and rotate a horizon angle of 40 degrees automatically between two frames.

To ensure the frames of the same number of each station have similar orientation(which improves the efficiency of the upcoming pose estimation procedure), the initial orientations of the device of each station is restricted. In need of co-visible areas between two consecutive stations, the distances between adjacent stations are also limited to 1m-2m.

Secondly, we separate the colour images and calibrated depth images from the output data form of RGB-D camera. Then we generate 2D-3D correspondences of each image, and form point clouds(within the relative coordinate system of each station). Setting the first station as the initial station, we try to calculate the pose of each station within the coordinate system in which the initial station locates in origin. Starting from the initial station, local feature matching is carried out between the frames of the same sequence number between two consecutive station, and the point clouds in the co-visible area help perform space resection to retrieve the unknown pose of the station in our known coordinate system. As soon as the procedure is finished, we could get the poses of each station in the same coordinate system, and the point clouds could be merged by ICP, forming the 3D model of the whole scene.



**Figure 1.** Pipeline of our method

And some 3D points in the point clouds would be removed due to the possible inaccuracy caused by the RGB-D camera issue. For example, points with depth over a certain threshold(10 m) may not be good enough.

#### 3.2 Image Retrieval with multi-features

To get images with certain co-visible area is a vital issue to perform visual localization. Wrong retrieval results would cost additional computation time and may lead to the fail of pose estimation. In certain cases, the images of the whole dataset serve as candidates for local features matching, and the method ensure retrieval of the best images to some extent, but causes a long processing time and not proper for large dataset. In our method, based on DBOW3, we perform image retrieval with different features to obtain relative stable result. The pipeline is summarized as below:



Figure 2. Integrated RGB-D cameras(GHO3D)

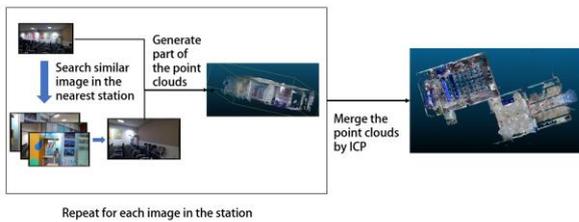


Figure 3. The procedure of generating a 3D model

**Visual Vocabulary Generation.** We first gather all local features of the images in the dataset and use k-means(k-means clustering algorithm) to divide the features into k clusters, and repeat the procedure for the k clusters until we finally get a vocabulary tree with  $l$  layers and each layer has  $k$  nodes( $k, l$  are parameters based on requirement). The nodes are the visual words, and typically we only use the nodes of the bottom layer as visual words.

**Similarity Calculation.** Then we convert each images in the dataset into a vocabulary frequency vector. Considering that different features are not of the same importance, It should be distributed weights separately. TF-IDF(Term Frequency-Inverse Document Frequency) is widely used in this area. IDF(Inverse Document Frequency) distributed higher weights to the words

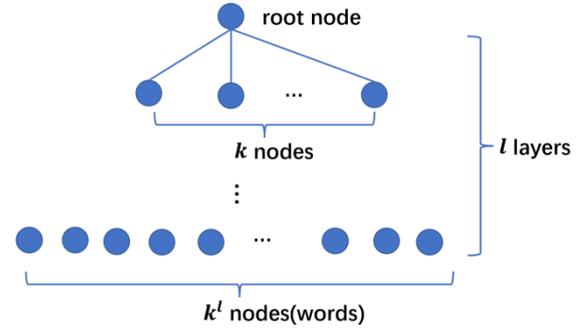


Figure 4. Structure of a vocabulary tree

appear not that frequent in the dataset, and TF(Term Frequency) considers features appear in the query image frequently as important features. Supposed that the total number of the features in the dataset is  $n$ , and  $n_i$  features could be clustered to word  $w_i$ , then the IDF for this word can be calculated as:

$$IDF_i = \log \frac{n}{n_i}, \quad (1)$$

In respect of a certain image, suppose that there are  $n$  features in total, and the word  $w_i$  appears in the image  $n_i$  times, then the TF for this word in the image is:

$$TF_i = \frac{n_i}{n}, \quad (2)$$

And the weight  $p_i$  for the word  $w_i$  is the product of  $IDF_i$  and  $TF_i$ . The image can be convert into a vocabulary vector  $v$  in the form of:

$$v = [f_1 \cdot p_1, f_2 \cdot p_2, \dots, f_n \cdot p_n], \quad (3)$$

where  $f_{i(i=1,2,\dots,n)}$  is the frequency of each word of the vocabulary tree in the image, and  $p_{i(i=1,2,\dots,n)}$  is the corresponding weight.  $n$  is the total number of words in the dictionary(vocabulary tree). The similarity score is determined by the distance between the vocabulary vectors of two images.

However, the performance of different features in image retrieval is various when applied scenes change. In order to obtain a rather stable result, we assign weights to the features type and, combine the similarity score of different feature types together, making use of multi-features to overcome the drawback of using single features. The weights are empirical distributed(decided by the recall of our test data).

### 3.3 Pose Estimation with Advanced Filtering

**Generating and Filtering the 2D-3D Correspondences:** We can get the rank and similarity scores of dataset images comparing to the query image, and we use top  $k$  images to perform local matching instead of using a single one to avoid the influence of error image retrieval results. Lowe's ratio test based on knn(k-nearest-neighbours) algorithm is also integrated into local feature matching to filter features too similar in the same image. For a feature  $p$  in the query image, whether another feature in the other image is good enough to match it can be decided by:

$$d_1 < k * d_2, \quad (4)$$

where  $d_1, d_2$  is the distance between the top 2 matched features and feature  $p$ , and  $k$  is an empirical parameters, which is 0.8 in

Lowe’s test, but the recent research shows that value from 0.4 to 0.6 would have better performance. Through this test, features which can not be significantly enough to separated from other features in the same image will be filtered, and these features may introduce disturbance in the following steps if not eliminated.

Still there may be outliers, so we perform RANSAC(Random Sample Consensus) to retrieve the affine transformation of the two images, and we filter features which can not suit the transformation. Then, some features(2D) in the query image can be matched to features in the retrieved image, which have corresponding 3D points in our dataset. So the features in the query image can be corresponded with certain 3D points to estimate the pose. And this procedure would be repeated for each top  $k$  images to get 2D-3D correspondences, irrespective of which image pairs they are obtained in.

**Obtain Pose.** We apply the traditional RANSAC and PnP(Perspective-n-Point) algorithms to obtain the pose of query image. Still RANSAC are used to build a model fit most 2D-3D correspondences and filter outliers, and PnP can obtain pose without an initial pose as input.

#### 4. EXPERIMENTS

##### 4.1 Datasets

We build the datasets by our integrated RGB-D cameras and our reconstruction pipeline. The datasets in the experiments consist of several indoor scenes: meeting rooms, factory and office. **Table 1** shows the basic information of the datasets. It is worth mentioning that the factory is quite empty and textureless, while the other two scenes are covered with common items of indoor environment. We also take pictures by our integrated RGB-D cameras as query image, and the size of query image is about 1/3 of the datasets. The vocabulary tree of our experiment has 5

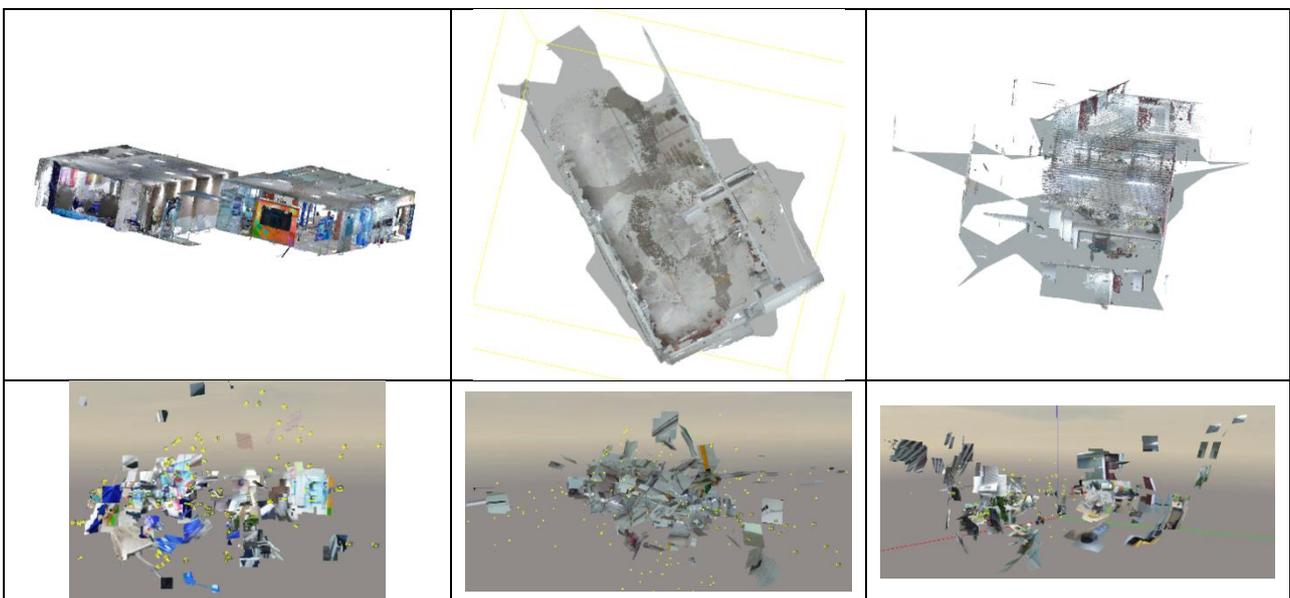
layers, and each father node has 10 son nodes, which means there are 10000 word in the dictionary.

##### 4.2 Performance of 3D Reconstruction

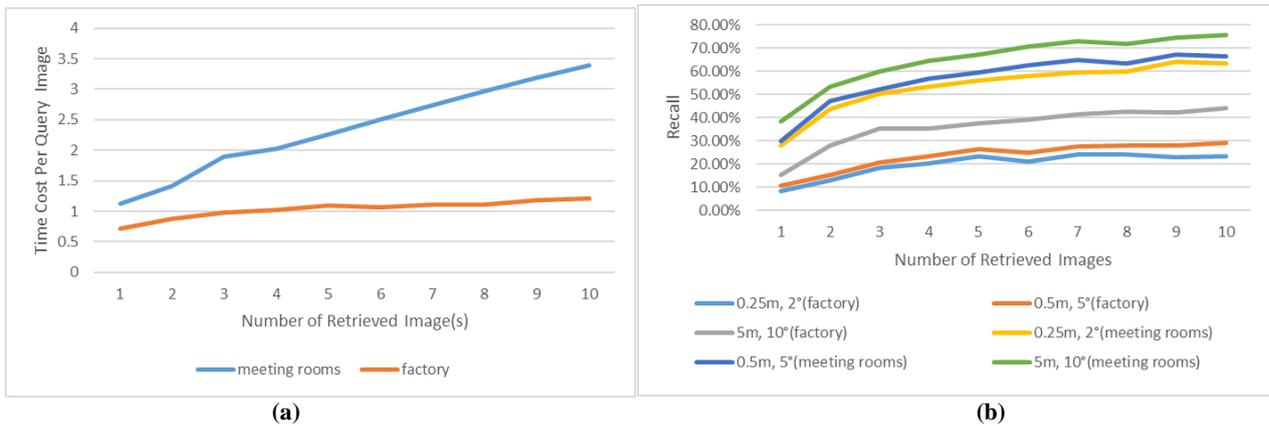
To compare the 3D reconstruction effect of our pipeline with a traditional SfM one, we use the colour images and depth images as the input of our pipeline, while the colour images as a series of unordered images input of SfM. SURF serves as the features used in this step. The results are shown in **Table 2**.

Scenes	Area	Number of images	Example image
Meeting rooms	26*11	459	
Factory	28*25	648	
Office	19*16	162	

**Table 1** Dataset Information



**Table 2.** The upper row is 3D point clouds generated by our pipeline, and the lower row is the SfM result. The three scenes from left to right are: meeting rooms, factory and office.



**Figure 5** (a) shows the relationship between the time cost and the number of retrieved images, and (b) shows the relationship between the recall(%) and the number of retrieved images. We use the three mainstream distance and orientation threshold.

The 3D model built through our pipeline is successful compared to a SfM method without optimization and additional pose information. Some parts of point clouds doesn't merge correctly, because the distances between the stations when collecting data exceed the threshold. Lack of enough overlap areas between consecutive stations is the main cause of failure.

### 4.3 Image Retrieval Optimization

As we mentioned, several images retrieved are involved in the following pose estimation procedure. We aim to find the best number of retrieved images which could get the most 2D-3D correspondences inliers with relative low time costs. We use the datasets of the meeting rooms and the factory we collected to carry out this experiment(Figure 5). BRISK is the deployed feature in this experiment for its relatively fast speed and stability. With regard to the meeting rooms scene, the time cost is positively correlated with number of retrieved images. And there is not much fluctuation of the time cost of factory scene. And the recall of this scene remains low due to the textureless environment. If the retrieval results are not good enough, the whole image would be filtered during the local feature matching steps, so the additional time cost is not explicit. In Figure 5(b), the recall of meeting rooms scene reaches over 75% when using more than six retrieved images within the 5m and 10 degrees using threshold. And the recall does not increase explicitly(even falls when using 8 retrieved images), because the top retrieved images are good enough for pose estimation, and the new entry retrieved image may not be of an positive influence. We can make a conclusion that using 7 retrieved images to perform the following pose estimation is the best way to balance speed and precision.

Since the hand-crafted features have their own limitations, we combined them in the image retrieval step. We also try to verify if using multiple features in image retrieval steps would have better performance than single feature. We experiment on SIFT, ORB and BRISK, and they are distributed different weights. We compare the recall of multi-feature image retrieval with single feature in the meeting room dataset. Table 3 shows the comparison.

BRISK has the best performance of recall within the 0.25 meters and 2 degrees threshold, comparing with ORB and SIFT. And the average time cost of one image is 1.46 second, so we use BRISK in most visual localization experiment we carry out. Image retrieval based on multi-features provides more reliable results, so there would be more inliers of 2D-3D correspondences, which improve the localization performance. However, the time cost of

this method is rather high, because image retrieval of different features is carried out instead of the classic single one. Though it reaches high precision, which feature strategy to choose is still based on the requirement of scene and experiment.

	0.25m,2°	0.5m,5°	5m,10°	Time cost per image
SIFT	48.4%	57.5%	65.4%	5.43s
ORB	37.3%	40.5%	56.9%	0.59s
BRISK	50.3%	52.3%	60.1%	1.46s
Multi-feature	60.8%	68.0%	74.5%	18.2s

**Table 3** The recall and time cost of different features.

### 4.4 Outliers Filtering

	0.25m,2°	0.5m,5°	5m,10°	Time cost per image
No filtering	12.4%	14.4%	29.4%	4.22s
With Lowe's ratio test(0.4)	51.0%	53.6%	58.8%	2.36s
With Lowe's ratio test(0.5)	54.9%	58.8%	66.7%	2.27s
With Lowe's ratio test(0.6)	64.1%	67.3%	75.2%	2.31s
With Lowe's ratio test(0.8)	66.7%	69.3%	74.5%	2.54s
Find homography by RANSAC	62.1%	64.1%	69.9%	2.90s
RANSAC+ ratio test(0.4)	45.8%	49.2%	57.5%	2.22s
RANSAC+ ratio test(0.5)	51.6%	54.9%	64.1%	2.23s
RANSAC+ ratio test(0.6)	55.6%	61.4%	67.3%	2.30s
RANSAC+ ratio test(0.8)	59.5%	64.9%	73.2%	2.73s

**Table 4** The recall and time cost of different outliers filtering strategy.

In this experiment we deployed Lowe's ratio test with different ratio and RANSAC to filter outliers of the homography, as shown in **Table 4** We use BRISK feature and 7 retrieved images per query image in this experiment. The dataset is the meeting rooms.

The time cost fluctuates a little each time we carry out the experiment due to the random sampling procedure of RANSAC. Both RANSAC(in local feature matching step) and ratio test could improve the precision and accelerate localization, because too much outliers will cause the RANSAC algorithms(in pose estimation) fail to converge. And with the ratio increasing, the recall is higher. Reasons are that the ratio test is filtering the outliers and some of the inliers simultaneously. If the number of 2D-3D correspondences is less than a certain threshold, PnP will fail to estimate a fine pose. The combination of two filtering strategy performs worse than single strategy due to the same reason.

## 5. CONCLUSION

In this paper, we introduced a full pipeline of visual localization from data preparation to pose estimation. We used RGB-D cameras integrated and programmed by ourselves to take color and depth images. Then we build 3D models based on the collected data through our method, and the models can be successfully reconstructed if followed our restriction of data collecting. Then we found out the best retrieved images number(7) which could balance the speed and precision, and applied it in our multi-features experiment. We distributed different weights for features and used image retrieval results based on the similarity scores and weights of several features. Though it may cost more time using multi-feature, the estimated pose is more precise than using single feature. We also added RANSAC algorithms to find homography of two matched images. This trick would help filter 2D matched keypoints which does not corresponded with the affine transformation. Lowe's test with different ratio is also compared with our tricks, and apply one of them is enough to improve the precision of localization, since applying too many filtering strategies will reduce the number of 2D-3D correspondences. Lack of enough correspondences will cause the failure of pose estimation, too.

## ACKNOWLEDGMENTS

This work was supported in part by the National Key Research and Development Program of China (Projects Nos. 2019YFB210310, 2019YFB2103104) and in part by a Research Program of Shenzhen S and T Innovation Committee grant (Projects Nos. JCYJ20210324093012033, JCYJ20210324093600002), the Natural Science Foundation of Guangdong Province grant (Projects No. 2121A1515012574), the Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation, MNR(Nos. KF-2021-06-125,KF-2019-04-014), the National Natural Science Foundation of China grant (Projects Nos. 71901147, 41901329, 41971354, 41971341) and the Foshan City to promote scientific and technological achievements of universities to serve industrial development support projects(Projects No. 2020DZXX04).

## REFERENCES

Acharya, D., Khoshelham, K., Winter, S., 2017 . BIM-PoseNet: Indoor camera localisation using a 3D indoor model and deep learning from synthetic images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150, 245-258. doi: 10.1016/j.isprsjprs.2019.02.020.

Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J. 2017. NetVLAD: CNN architecture for weakly supervised place recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 1

DBOW3, 2017. <https://github.com/rmsalinas/DBow3> (17 Feb 2017).

Detone, D., Malisiewicz, T., Rabinovich, A. , 2017 . SuperPoint: Self-Supervised Interest Point Detection and Description.

Endres, F., Hess, J., Sturm, J., Cremers, D., Burgard, W., 2014. 3-D mapping with an RGB-D camera. *IEEE Transactions on Robotics*, 30(1), 177-187.

Galvez-Lpez, D., Tardos, J. D. ,2012. Bags of Binary Words for Fast Place Recognition in Image Sequences. *IEEE Transactions on Robotics*, 28(5), 1188-1197.

Kendall, A., Grimes, M., Cipolla, R. ,2015. PoseNet: A convolutional network for real-time 6-dof camera relocation. *IEEE*.

Li, M., Chen, R., Liao, X., Guo, B., Zhang, W.,... Guo, G. ,2020. A Precise Indoor Visual Positioning Approach Using a Built Image Feature Database and Single User Image from Smartphone Cameras. *Remote Sensing*, 12(5), 869. doi: 10.3390/rs12050869.

Lowe, D.G., 1999. Object recognition from local scale-invariant features. In: *Proc. 7th IEEE international conference on Computer vision, IEEE*, pp. 1150-1157.

Monica, S., Bergenti, F. ,2019. Hybrid Indoor Localization Using WiFi and UWB Technologies. *ELECTRONICS*, {8}({3}). doi:10.3390/electronics8030334.

Nistér, D., Stewénius, H., 2006. Scalable Recognition with a Vocabulary Tree. *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference*.

Park, B., Myung, H. ,2014. Underground localization using dual magnetic field sequence measurement and pose graph SLAM for directional drilling. *MEASUREMENT SCIENCE AND TECHNOLOGY*, {25}({12}). doi: 10.1088/0957-0233/25/12/125101.

Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A. ,2007. Object retrieval with large vocabularies and fast spatial matching. *IEEE Computer Society Conference on Computer Vision & Pattern Recognition*.

Robertson, D., Cipolla, R.,2004. An Image-Based System for Urban Navigation. *bmvc*.

Sarlin, P., Cadena, C., Siegwart, R., Dymczyk, M. ,2019. From Coarse to Fine: Robust Hierarchical Localization at Large Scale. *CVPR 2019*.

Sarlin, P. E., Unagar, A., Larsson, M., Germain, H., Toft, C., Larsson, V.,... Kahl, F. ,2021. Back to the Feature: Learning Robust Camera Localization from Pixels to Pose. *CVPR 2021*.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D.,... Batra, D., 2020. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *INTERNATIONAL JOURNAL OF COMPUTER VISION*, {128}({2}), 336-359. doi: 10.1007/s11263-019-01228-7.

Svarm, L., Enqvist, O., Kahl, F., Oskarsson, M., 2016. City-Scale Localization for Cameras with Known Vertical Direction. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 1455-1461.

Yang, C., Shao, H., 2015. WiFi-Based Indoor Positioning. *IEEE COMMUNICATIONS MAGAZINE*, {53}({3}), 150-157. doi: 10.1109/MCOM.2015.7060497.

Zafari, F., Gkelias, A., Leung, K. K., 2019. A Survey of Indoor Localization Systems and Technologies. *IEEE COMMUNICATIONS SURVEYS AND TUTORIALS*, {21}({3}), 2568-2599. doi: 10.1109/COMST.2019.2911558.

Zhang, X., Lin, J., Li, Q., Liu, T., Fang, Z., 2020. Continuous Indoor Visual Localization Using a Spatial Model and Constraint. *IEEE Access*, 8, 69800-69815. doi: 10.1109/ACCESS.2020.2986044