symmetry

MDPI

Article

Combining Obstacle Avoidance and Visual Simultaneous Localization and Mapping for Indoor Navigation

SongGuo Jin, Minhaz Uddin Ahmed, Jin Woo Kim, Yeong Hyeon Kim and Phill Kyu Rhee *

Department of Computer Engineering, Inha University, 100 Inha-Ro, Nam Gu, Incheon 402-751, Korea; sgkim735@gmail.com (S.J.); minhaz.ahmed@gmail.com (M.U.A.); lornpel@gmail.com (J.W.K.); ohpely@gmail.com (Y.H.K.)

* Correspondence: pkrhee@inha.ac.kr

Received: 7 October 2019; Accepted: 9 December 2019; Published: 7 January 2020

Abstract: People with disabilities (PWD) face a number of challenges such as obstacle avoidance or taking a minimum path to reach a destination while travelling or taking public transport, especially in airports or bus stations. In some cases, PWD, and specifically visually impaired people, have to wait longer to overcome these situations. In order to solve these problems, the computer-vision community has applied a number of techniques that are nonetheless insufficient to handle these situations. In this paper, we propose a visual simultaneous localization and mapping for moving-person tracking (VSLAMMPT) method that can assist PWD in smooth movement by knowing a position in an unknown environment. We applied expected error reduction with active-semisupervised-learning (EER–ASSL)-based person detection to eliminate noisy samples in dynamic environments. After that, we applied VSLAMMPT for effective smoothing, obstacle avoidance, and uniform navigation in an indoor environment. We analyze the joint approach symmetrically and applied the proposed method to benchmark datasets and obtained impressive performance.

Keywords: SLAM; obstacle avoidance; depth estimation; object detection

1. Introduction

It is very difficult for elderly individuals or people with disabilities (PWD) to walk long distances and ride on public transport. With the help of monocular simultaneous localization and mapping (SLAM), we can assist in the accessibility of elderly people and PWD by improving commuter mobility. A remarkable solution for improving mobility is to minimize walking by, for example, automated guiding, obstacle avoidance, and autonomous wheelchairs.

Our main research plan is to assist PWD. Specifically, we wanted to help visually impaired people so that they can move independently. In their day to day life, traveling in the airport or bus terminal is challenging work. In a large building, finding accurate objects and locations is very puzzling. Therefore, our proposed algorithm can bridge this gap and make the lives of PWD easier.

Avoiding people in a crowded area is a challenging problem due to multiple dynamic object movement. In this scenario, our algorithm consists of a number of steps such as place detection, object recognition, obstacle position alert through text to voice conversion (such as elevator is on your left side, turn left), obstacle avoidance, and provide a navigation service. Instead of covering the large scope of research work in this paper, we generally focused on object detection and obstacle avoidance where all of these objects were tracked eight meters from the camera. A very short navigation direction is sent to the user via voice for avoiding an obstacle. The flow diagram will give a detailed scenario of the workflow process. Here, a visually impaired person will avoid obstacles and find the elevator to go upstairs. We considered a university lobby for our experiment suitability as shown in Figure 1.

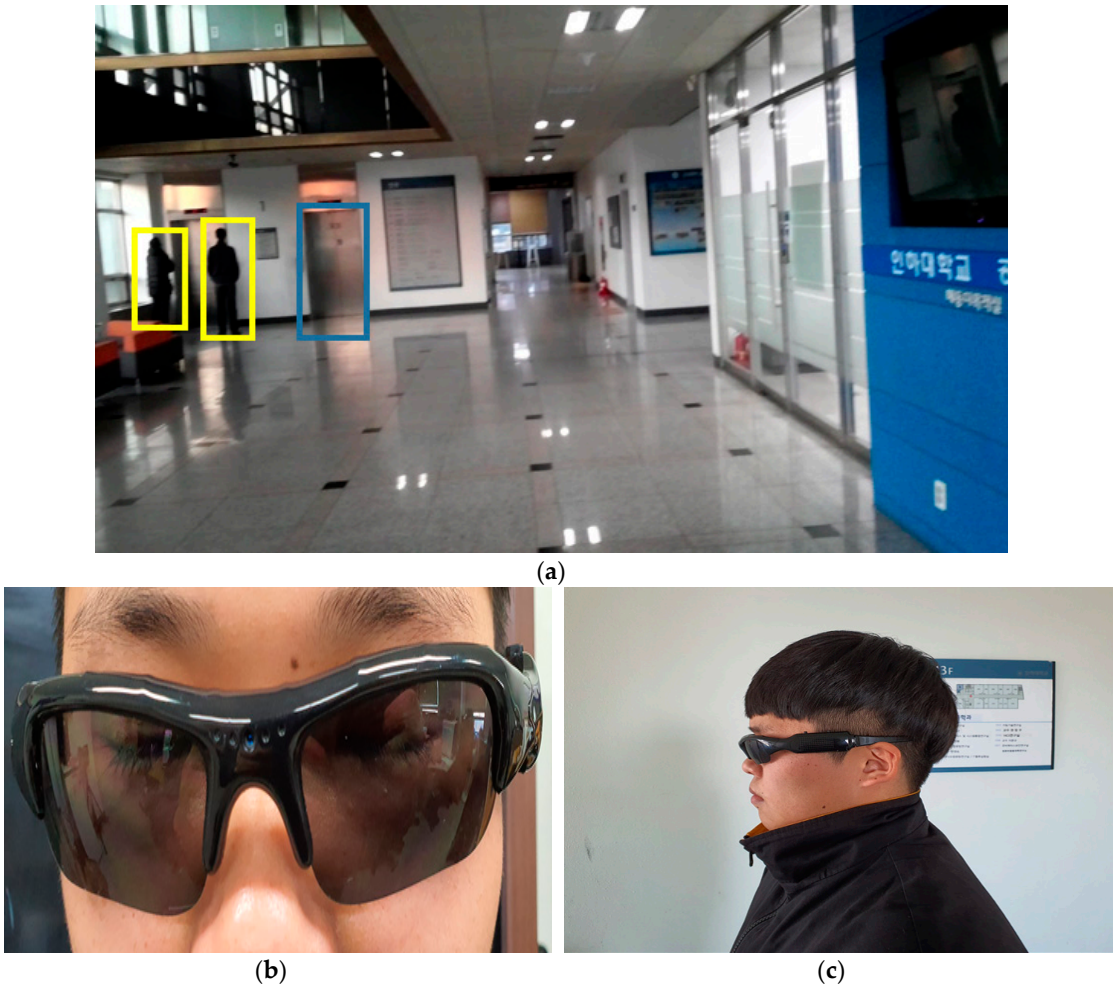


Figure 1. Visually impaired person searching for an elevator. (**a**) The university lobby where a blind person is searching for an elevator. The possible obstacles are the sofa and two people in front of the elevator. (**b**) Smart eyeglasses that capture the input image through a camera. (**c**) Person receives direction information through the eyeglass speaker.

Localization and concurrent mapping is a promising field of robotics studies. Visual-feature methods for landmark selection in visual SLAM (VSLAM) have been analyzed in a number of research works [1]. Three feature-detection methods are mainly used, namely, maximally stable extrema regions (MSER), scale invariant feature transform (SIFT), and sped-up robust features (SURF). In unknown settings, the visual SLAM scheme allows a robot agent to locate itself. SLAM, with the fast growth of computer-vision technology and the enhancement of processor efficiency, has been commonly used in a number of areas such as augmented reality, robots, drones, and driverless cars.

In most types of SLAM framework, research considers the environment as motionless, ignoring the influence of moving obstacles [2,3]. In fact, in real surroundings, like a moving car or a person coming from the opposite direction, these obstacles cannot be prevented. The majority of obstacle detection and removal methods have been developed on RGB-Depth cameras, for example, Kinect [4], as conventional SLAM instruments.

This paper has the following contributions: we applied a dynamic person-detection method, namely, the expected-error-reduction active-semi supervised-learning (EER–ASSL) method, which effectively perceives human actions in a cluttered environment. We applied VSLAM, which utilizes the cheapest monocular camera for extracting feature points that help to build a navigation path in real time for a PWD in an unknown environment. We briefly discuss recent related work in Section 2,

and describe our system in detail in Section 3. We show our results in Section 4, and outline our conclusions in Section 5.

2. Related Works

Indoor and outdoor travelling is challenging for PWD due to obstacles and navigation. A number of techniques have been applied to overcome these problems such as indoor navigation systems, wearable devices, and accessible maps. The removal of moving objects in order to avoid collision by using a moving camera is an important part of this research, as both foreground and background simultaneously change.

In the literature, VSLAM has been applied in a number of areas. Berat et al. proposed an RGB-D camera-based feature-detection method that was applied to a number of robots [5]. This process is known as cooperative SLAM, which makes a common map of an entire work environment. A challenge regarding cloud-based control is storage, due to the huge number of point clouds that a large number of images need to store. Moreover, separate hardware requires several implementations to make a global map by using a local map [5].

Yipu et al. introduced a new feature-selection method known as the MaxLogDet algorithm for pose optimization [6]. Later, they combined the MaxLogDet feature selection with VSLAM for pose estimation. A near-optimal heuristic approach was used for subset selection, and it improved performance [6].

Jihong et al. proposed a hybrid-vision-based SLAM and moving-object-tracking approach [7]. Their proposed approach combined two methods: (a) superpixel-based segmentation is that used to detect moving objects, and (b) a particle filter to estimate SLAM properties. They used the Markov random field (MRF) energy function in order to detect moving superpixels [7].

N-Danish et al. presented an approach for the accurate localization of moving objects where sparse-flow-based motion segmentation was done using a stereo camera [8]. Better pose accuracy was obtained by exploiting moving objects. This work was an improvement on camera-trajectory computation when compared with the standard methods [8].

Zhangfang et al. applied VSLAM on closed-loop detection [9]. First, they used a depth camera to gather environmental information. Then, the key frames were extracted, and they included the improved pyramid term frequency-inverse document frequency (TF-IDF) similarity-score function to reduce closed-loop perception ambiguity. Most effective closed-loop testing frames are performed on the basis of key-frame selection such as rotation and translation [9].

VSLAM was applied to the feature detection of aerial images where SLAM helped with better landmark selection [10]. Similarly, Valder et al. applied key-matching-point detection to calculate the movement of a robot by camera movement [11]. Autonomous-mobile-robot navigation is another popular VSLAM area, where a robot utilizes vision sensors to receive data from different locations [12].

Many methods perform well in removing obstacles with their own features [13,14], but a number of studies mostly rely on expensive sensors as input. After preprocessing input data with necessary depth information, the barrier is segmented from the scenes. Considering the commonly used technique of deep learning, we suggest a straightforward and low-cost technique for SLAM to locate and filter out the barrier [15]. Therefore, removing the barrier online is appropriate for SLAM as a part of the preprocessing phase. Our suggested framework addresses these above-mentioned problems, and works well on locating and mapping the actual dataset environment. In the experiment on the TUM Dynamic benchmark dataset, the suggested scheme was contrasted with raw ORB-SLAM2 that quantitatively evaluated our technique.

3. System Outline

The system's core function is egocentric 3D indoor navigation based on action-recognition technology. Unlike conventional indoor-navigation systems, this technology can detect obstacles or abnormal behaviors ahead of the user, and warn of dangers by connecting a fixed camera such as CCTV

and a camera equipped with a mobile device to configure the hybrid vision technology. The application of this technology is to guide the user safely to their destination when there are many people and complex obstacles like unmanned shops.

Here, we demonstrate the necessity of removing obstacles with graph SLAM [16]. The error of a constraint relies on the comparative position of two neighboring poses. Once the SLAM graph is built, the specific aim is to discover a node setup that minimizes obstacle-generated mistakes [17].

Figure 2 shows the block diagram of the proposed system where VSLAM and motion person tracking (MPT) play a significant role for obstacle removal in a dynamic environment. Here, VSLAM receives a visual measurement (Z) and a motion measurement (U) to determine a map (M) and a pose (X). In the next step, the MPT receives the time measurement (Z) and outputs the position (P) of the moving person and the motion mode (S). Finally, VSLAM determines M and x composed of the static landmark by taking Z and U without assuming that the surrounding environment is static. Then, it outputs the dynamic object position (O) and (S).

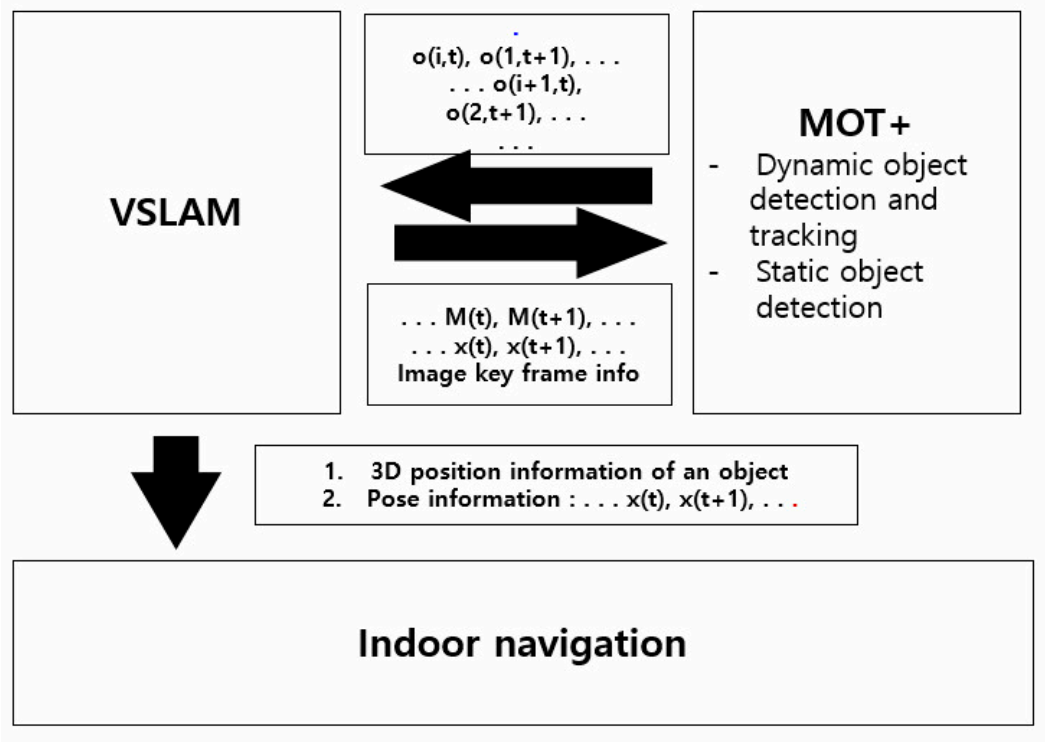


Figure 2. Block diagram of the proposed indoor-navigation system.

4. Proposed Method

Our method takes advantage of two monocular cameras by using VSLAM. We followed a similar structure as ORB-SLAM2 while taking the input. First, as an original input, we used monocular camera. Second, we used the picture sequence to extract features and create data association. The back end is based on maximum a posteriori estimation (MAP) likelihood inference theory [13]. We included the glossary of acronyms used in this paper in Table 1.

Table 1. The glossary of acronyms used in the paper.

Acronyms	Full Names
VSLAM	Visual simultaneous localization and mapping
MPT	Motion person tracking
EER-	Expected error reduction
ASSL	Active semi-supervised learning
MAP	Maximum a Posteriori Estimation
RGB-D	Red-green blue-Depth
MRF	Markov Random Field
CNN	Convolutional-neural-network

Visual simultaneous localization and mapping (VSLAM) combined with motion person tracking (MPT) effectively applied for person tracking shown in Figure 3.

$$P(x_k, M|u_1, u_2, \ldots u_k, z_0, z_1, \ldots z_k) \tag{1}$$

where S represents motion mode; P represents a moving disabled person; and Z is the time measurement.

$$P(p_k, S_k|Z_k) = P(p_k|S_k, Z_k)P(S_k|Z_k) \tag{2}$$

$$P(X_k, Y_k|Z_k, U_k) \; \alpha \; P(Z_k|X_k, Y_k)$$

$$\iint P\left(X_k|Y_{K-1,\, U_k}\right)P(Y_k|Y_{k-1})P(X_{k-1}, Y_{k-1}|Z_{k-1}, U_{k-1})dx_{k-1}dY_{k-1} \tag{3}$$

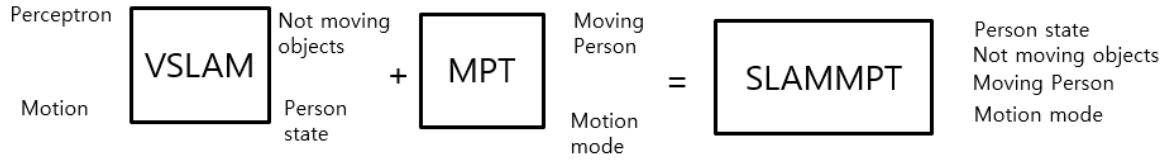


Figure 3. Combination of visual simultaneous localization and mapping (VSLAM) and motion person tracking (MPT) for person tracking in our proposed system.

We formulated MPT in a generic way, where U denotes the motion-sensor-measurement observation continuous points; Z represents visual-sensor-measurement observation continuous points; X represents pose-hidden continuous points; Y conveys position of moving object hidden discontinuity point; and S is motion mode [2].

$$\underset{\text{Update}}{P(p_k, M, x_k|Z_k,\; U_k)\; \alpha \; P(Z_k|p_k, X_k)} \underset{\text{prediction}}{\int P(p_k|p_{k-1})P(p_{k-1}|Z_{k-1}, U_{k-1})\; dp_{k-1}}$$

$$\underset{\text{Update}}{P(z_k^m|M,\; x_k)} \underset{\text{prediction}}{\int P(x_k|u_k, x_{k-1})\; p(x_{k-1}, M|Z_{K-1}^m, U_{k-1})dx_{k-1}} \tag{4}$$

VSLAM and MPT Formulation: U represents the motion-sensor-measurement observation continuous points; z denotes the visual-sensor-measurement observation continuous points; x is the pose-hidden continuous points; S is the motion mode; and p denotes the moving disabled person. Hidden discontinuity points are represented by M.

4.1. People Detection for Obstacle Avoidance

The flow diagram for person detection shown in Figure 4. The collaborative sampling used for measurement of uncertainty and diversity. The details flow explained in Section 4.3.

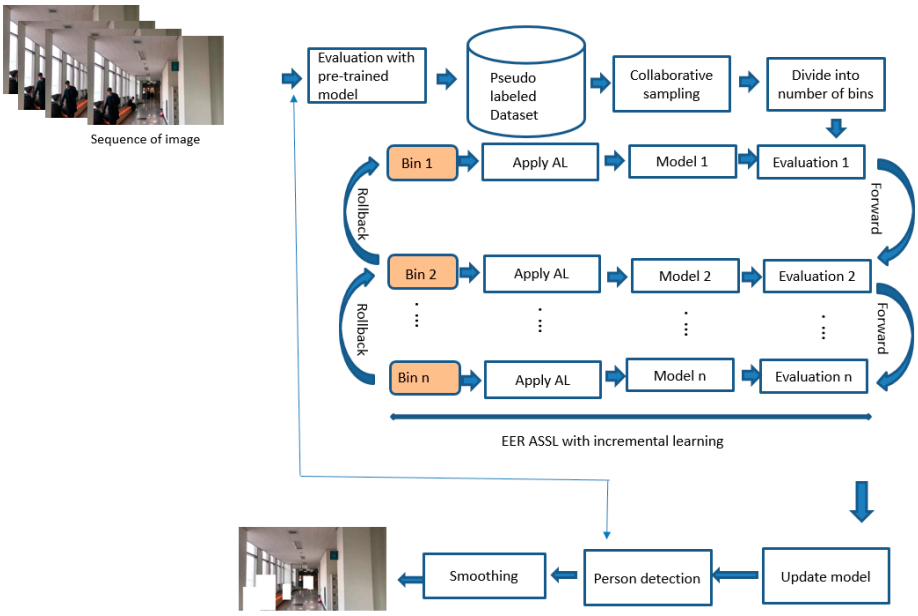


Figure 4. Flow diagram of expected-error-reduction active-semisupervised-learning (EER–ASSL)-based person detection.

4.2. Person Detection

Many convolutional neural network (CNN)-based person detectors are designed for static data distribution, and are unable to handle drift, fast motion, and occlusion problems [14]. Moreover, in a dynamic environment with complex settings, person detection becomes challenging due to aspect-ratio difference and vanishing problems. Our expected-error-reduction active-semisupervised-learning (EER–ASSL)-based method can overcome this problem where the prevailing person detector is applied in order to obtain the desired detection result. We adopted state-of-the-art object detector YOLOv2 [4] in our EER–ASSL method, which takes advantage of person detection. Most common detectors train their detector with thousands of images in order improve detection performance, but labeling images for detection is very expensive.

4.3. Human-Action Smoothing

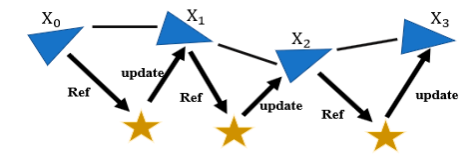
We can see from Figure 4 that person detection was applied to find out whether a human exists in the scenario or not, since labeling images for classification is expensive, but unlabeled data carry almost no information related to human-action recognition. Therefore, in order to improve person-detection accuracy, we required a semisupervised-learning (SSL) algorithm [14]. From a large volume of samples, we can find a way to gather informative samples that can contribute to EER–ASSL performance. Sampling bias or wrongly labeled samples can hamper detection performance. Therefore, AL helps our proposed system overcome wrongly labeled samples by relabeling. Moreover, expected-error-reduction sampling becomes handy.

The selected samples are divided into a number of bins in order to process the data. In each bin, we apply AL, and then train and evaluate. After that, Bin 1 is carried forward to Bin 2. This entire process continues until Bin N. If the Bin 1 model performance is better than that of the next bin, we consider this as forward learning; otherwise, it is considered rollback learning, and that leads to skipping the poorly scored bin. The entire process continues until Bin N.

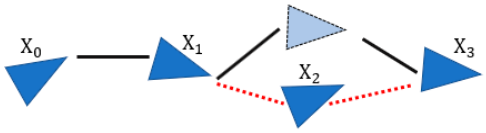
Human activities such as sports activities or real-world events have abrupt changes in the environment can be detected by comparing localized bounding boxes. If there is detected drift, we can estimate smoothing for validation, and apply the EER–ASSL updated model to overcome the drift problem. In this way, we selected the best possible model. In the next step, we extract the person features from an input image and apply smoothing for improved human-activity detection.

4.4. VSLAM with Dynamic Landmark Removal SLAMMPT

We selected SLAMMPT, which uses the feature-matching assessment method. In the issue of data association, it is about combining present characteristics (landmarks) with earlier observed characteristics to acknowledge past landmarks or locations. The presence of moving obstacles improves the likelihood of short- and long-term-information association-error accumulation [13]. Figure 5 indicates the strong lines in which a robot poses. Star marks represent landmarks or feature landmarks. Figure 5a demonstrates how landmarks make the robot pose. Robot poses X_0, X_1, X_2, and X_3 are disordered by a moving person [15]. Figure 5b shows that the camera pose in key frame X_2 is pointing in the wrong direction. The correct camera pose is in the triangle with the solid line. The miscalculated camera pose and position are shown in the dotted line in light blue.



(**a**)



(**b**)

Figure 5. Influence of moving obstacle in SLAM form graph. (**a**) Work principle of feature and camera pose. (**b**) Misdirected camera pose by an obstacle in a dynamic environment.

The error function can be presented as an error addition of static landmark $e_j(X)$ and dynamic landmarks $e_i(X)$.

$$e(X) = e_i(X) + e_j(X) \tag{5}$$

where X is the state vector. Error functions e_i and e_j denote the difference between predicted and actual measurements of static and dynamic landmarks. After the dynamic objects are removed, the equation becomes:

$$e(X) = e_i(X) = z_i - f_i(X) \tag{6}$$

where $f_i(X) = \hat{z}_n$ is the predicted measurements; z_i is a measurement of state X ; and $\hat{z}_i = f_i(X)$ is a function that maps X to predicted measurement $\hat{z}_i$. Then, measurements of noisy n become $z_{1:n}$. Assuming that the error has zero mean and is normally distributed, the Gaussian error function is given by information matrix Ω_i [15]. Scalar $e_i(X)$ is rewritten by $e_i(X)^T \Omega_i e_i(X)$. The problem of calculating the optimal X is formalized as follows:

$$X^* = \underset{X}{\operatorname{argmin}} \sum_i e_i(X) = \underset{X}{\operatorname{argmin}} \sum_i e_i(X)^T \Omega_i e_i(X) \tag{7}$$

5. Analysis Method and Experiment Results

5.1. Motion Person Tracking (MPT) and Performance

Figure 6 shows the mAP comparison of EER–ASSL with popular object detectors like Faster RCNN [3], SSD 300, and YOLOv2 [4]. Experiments were conducted by using incremental learning with a faster RCNN detector.

We used both local and benchmark datasets, and our baseline detector was YOLOv2. Both the test data for PASCAL VOC 2007 and our local data were used. The collaborative sampling parameters were set at 0.8, 0.6, and 0.8. The EER–ASSL-based object detector demonstrated much improvement over other object detectors.

Table 2 summarizes the mean average precision of state-of-the-art methods on PASCAL VOC and the local dataset.

Table 2. EER–ASSL object-detection performance on local dataset.

Framework	mAP	Dataset	Speed (fps)
Faster RCNN	67.7	07 + 12	5
SSD300	68.4	07 + 12	59
YOLOv2	71.0	07 + 12	67
Ours (EER–ASSL)	73.3	07 + 12 + local	36

Our proposed EER–ASSL-based person detector showed improvement when compared with popular object detectors like Faster RCNN, SSD300, and YOLOv2. Since EER–ASSL was jointly trained on PASCAL VOC and the local dataset after the network was fine-tuned, fast adaptive capacity was achieved regardless of velocity. Our EER–ASSL technique, as shown in Table 2, considerably increased the detection efficiency of the local dataset and its environment. For fair evaluation, each of the four objects had 100 labeled and 300 unlabeled samples. The detectors were trained for reasonable assessment by using the same benchmark dataset and local dataset. Table 2 demonstrates the comparison outcomes, where each column shows both the benchmark and local information composition percentage.

YOLOv2 trains the network by using classification dataset ImageNet 1000 [5], and then modifies the network for detection purposes. This joint classification and detection information is much more than that of the local dataset, which is restricted to only 100 pictures for each class such as a sofa or a ticket gate. As a consequence, in our experiment, reducing local training data had little impact on the YOLOv2 model.

Our framework worked well on a local dynamic environment. Obstacle instances were ideally removed by person detection in Figure 7a–d. We also showed the Detected person with SLAM features in Figure 8a–d.

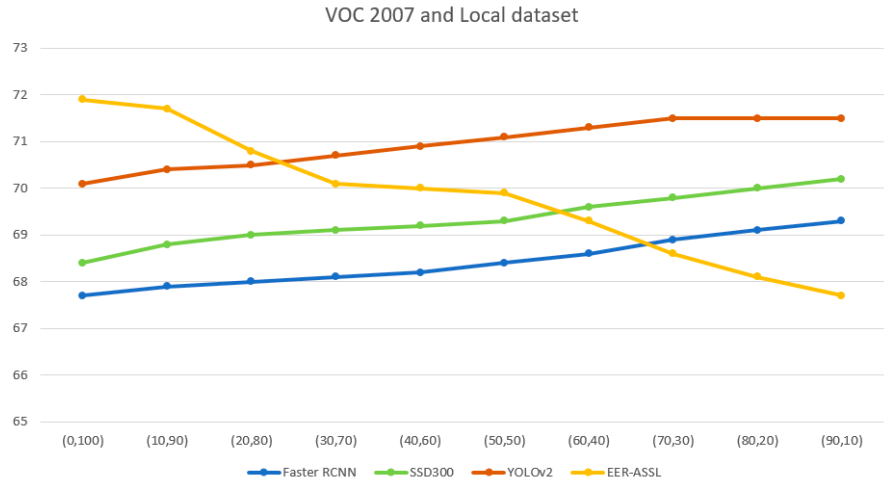
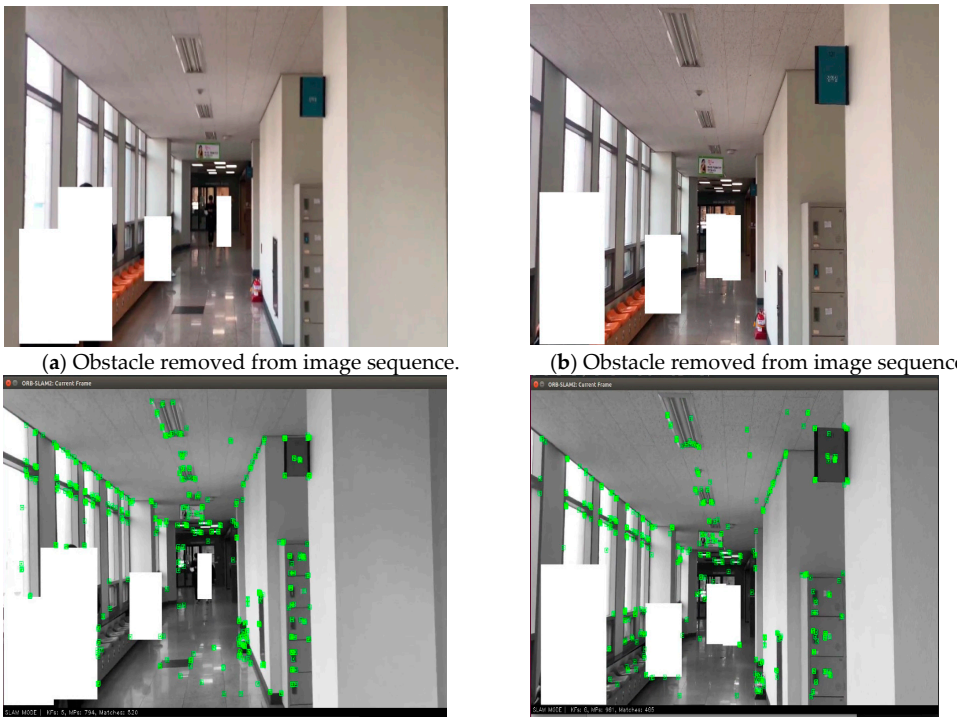


Figure 6. Performance comparison of the EER–ASSL object detector and state-of-the-art object detectors in terms of mAP measurements.

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

(**c**) (**d**)

Figure 7. Original image sequence of the local dataset and detected person. (**a**) Original image sequence. (**b**) Detected person in image sequence. (**c**) Original image sequence. (**d**) Detected person in image sequence.



(**a**) Obstacle removed from image sequence. (**b**) Obstacle removed from image sequence.

(**c**) Detected person with SLAM features. (**d**) Detected person with SLAM features.

Figure 8. (**a**,**b**) Local dataset with removed obstacle. (**c**,**d**) Image sequence with feature extraction in a SLAM system where uncertainty instance was successfully filtered out.

5.2. VSLAMMPT Performance Evaluation

We test our system on the Technical University of Munich (TUM) dynamic dataset. Due to a lack of person data in the precondition of the monocular-camera input and the benchmark-dataset evaluation, we made our performance table with a comparison with the raw ORB-SLAM2 system where there is person interaction.

We considered the relative pose error (RPE) and the root mean squared error (RMSE) for analysis and evaluation [18].

5.3. Root Mean Square Error (RMSE)

From the comparative pose mistake, we calculated the RMSE over all translation-component time indices:

$$RMSE(P_{1:n},\ \delta) := \left(\frac{1}{m} \sum_{i=1}^{m} \|trans(P_i)\|^2 \right)^{\frac{1}{2}} \tag{8}$$

where $trans(P_i)$ refers to the translation parts of relative pose error P_i ; and δ is generally set to a fixed time interval to 1.

We integrated person detection and ORB-SLAM2 feature extraction in order to stabilize the rotation along the r–p–y axes. We supposed an individual was an unstable and moving barrier to the SLAM scheme in our experiment. We used open-source software Evo, a Python package to evaluate odometry, and SLAM with the assessment technique described above [15].

The TUM Dynamic dataset is about individuals interacting in a scene with a desk, chair, and telephone. An individual walks into the scene at the desk and sits at a desk. This can be considered as a vibrant slow-motion dataset and is regarded as a fast-motion dynamic dataset in the walking sequence. Datasets of TUM Dynamic Objects were selected to test our scheme. The dataset of TUM Dynamic Objects includes RGB picture sequences, image-depth data, and ground-truth trajectory.

We selected seven out of ten dataset sequences in our experiment because some parts of the data sequences in camera movements like r–p–y and x–y–z are important. The number of valid key frames increases in more stable and precise localization performance, as shown in Table 3.

Table 3. Performance of root mean squared error (RMSE) in the TUM Dynamic dataset.

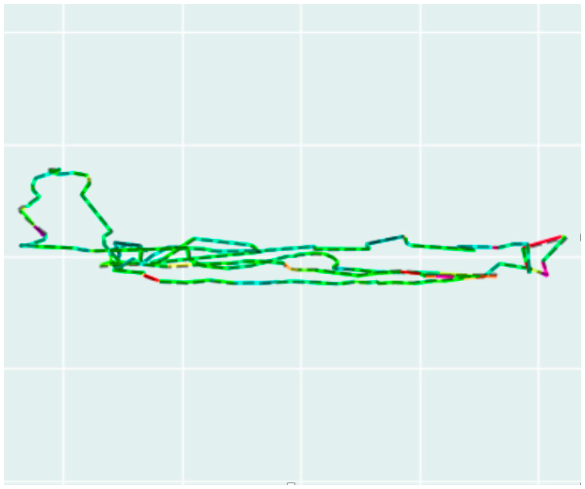
Sequences	ORB-SLAM2	Method 1	Method 2	Ensemble Performance
fr2_desk_with_person	0.006608	0.006375	0.00683	0.006375
fr3_sitting_halfsphere	0.014971	0.021956	0.013492	0.013492
fr3_sitting_xyz	0.014979	0.0146	0.014944	0.0146
fr3_walking_halfsphere	0.266141	0.029677	0.059813	0.029677
fr3_walking_xyz	0.010298	0.017487	0.059813	0.010298

Table 3 shows that our strategy helped to stabilize the system by more or less piling up key frames. In particular, it demonstrates excellent efficiency in low- and high-dynamic environments in the dataset sequence *fr2_desk_with_person* and *fr3_sitting_halfsphere*.

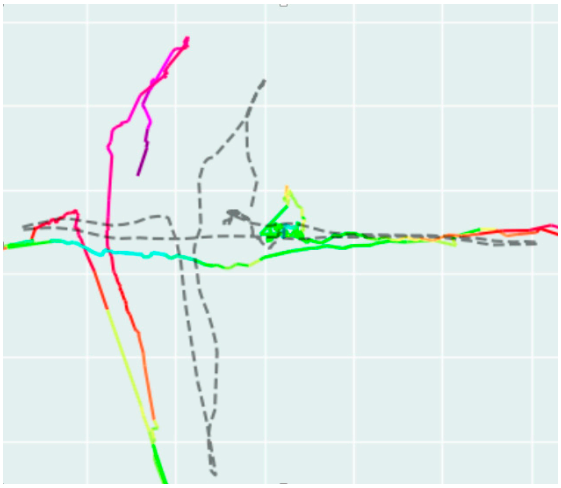
In some outcomes such as *fr3_sitting_halfsphere* and *fr3_walking_xyz*, there were only a few key frames. The ORB-SLAM2 scheme was suppressed by our technique because the actor was walking too close to the camera, which requires too much scene space to ratify function points.

The key-frame-based SLAM has a natural weakness due to powerful rotation, rapid motion, blurred pictures, and the absence of feature points [19]. However, the camera-rotation error was considerably decreased with our strategy, particularly in the *fr3_walking_halfsphere* sequence. Our proposed method had significant improvement on ensemble performance over existing methods, as shown in the last column of Table 2. This improvement came from the ensemble of ORB-SLAM2, and Methods 1 and 2, according to image-complexity analysis. Image-complexity analysis was calculated on the basis of

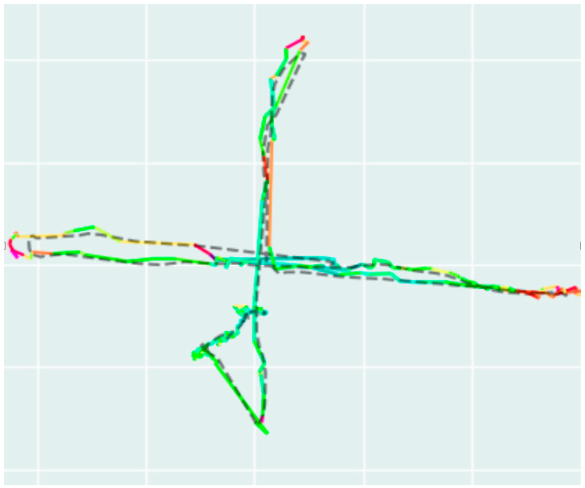
the number of objects in a scene and the size of each object. Plots on TUM dynamic dataset shown in Figure 9.



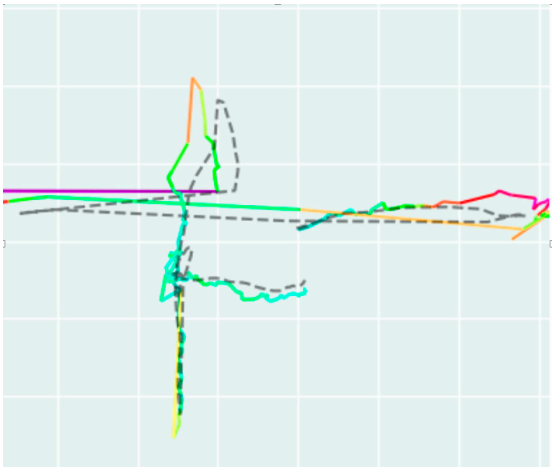
(**a**) fr2_desk_with_person



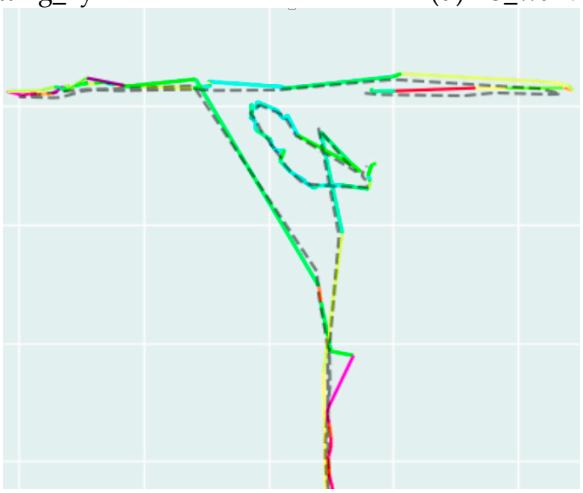
(**b**) fr3_sitting_halfsphere



(**c**) fr3_sitting_xyz



(**d**) fr3_walking_halfsphere



(**e**) fr3_walking_xyz

Figure 9. Plots of TUM pose files on ORB-SLAM and ground truth.

6. Conclusions

This study focused on providing guidance to people with disability (PWD) in their everyday life, mostly helping them navigate through large indoor areas such as airports or bus terminals. We proposed a novel method that is very cheap and effective for PWD, in particular, visually impaired people, or the person who use wheelchair in order to avoid obstacles in a dynamic environment. Existing technology such as global positioning systems (GPS) do not work well in an indoor environment where our proposed visual simultaneous localization and mapping for moving-person tracking (VSLAMMPT) will be a significant improvement. While navigating through an indoor environment, we applied an EER–ASSL-based person detector for smooth person detection. However, the person detection performance decreased with lighting conditions and speed. In the future, we wish to further investigate to overcome these challenges. At present, we can only provide users very limited instructions to find the destination location such as turn left, right, or turn around. In the future, we would like to give the user a more precise object location and distance. We also demonstrated that our technique had significantly better performance than that of ORB-SLAM2.

Author Contributions: Conceptualization, S.J., Y.H.K. and M.U.A.; Writing—original-draft preparation, P.K.R.; Writing—review and editing, M.U.A. and J.W.K.; Supervision, P.K.R.; Project administration, Y.H.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This work was supported by the ICT R and D program of MSIP/IITP (2017-0-00543, Development of Precise Positioning Technology for the Enhancement of Pedestrian's Position/Spatial Cognition and Sports Competition Analysis). A Graphics Processing Unit (GPU) used in this research was generously donated by the NVIDIA Corporation.

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

References

1. Kadir, H.A.; Arshad, M.R. Features detection and matching for visual simultaneous localization and mapping (VSLAM). In Proceedings of the 2013 IEEE International Conference Control System Computing Engineering (ICCSCE), Mindeb, Malaysia, 29 November–1 December 2013; pp. 40–45.
2. Al-Mutib, K.N.; Mattar, E.A.; Alsulaiman, M.M.; Ramdane, H. Stereo vision SLAM based indoor autonomous mobile robot navigation. In Proceedings of the 2014 IEEE International Conference Robotics and Biomimetics, ROBIO 2014, Bali, Indonesia, 5–10 December 2014; pp. 1584–1589.
3. Wang, C.C.; Thorpe, C.; Thrun, S.; Hebert, M.; Durrant-Whyte, H. Simultaneous localization, mapping and moving object tracking. *Int. J. Rob. Res.* **2007**, *26*, 889–916. [CrossRef]
4. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
5. Erol, B.A.; Vaishnav, S.; Labrado, J.D.; Benavidez, P.; Jamshidi, M. Cloud-based control and vSLAM through cooperative Mapping and Localization. In Proceedings of the 2016 World Automation Congress (WAC), Rio Grande, Puerto Rico, 31 July–4 August 2016; pp. 1–6.
6. Zhao, Y.; Vela, P.A. Good Feature Selection for Least Squares Pose Optimization in VO/VSLAM. In Proceedings of the IEEE International Conference Intelligent Robots and Systems, Madrid, Spain, 1–5 October 2018; pp. 1183–1189.
7. Min, J.; Kim, J.; Kim, H.; Kwak, K.; Kweon, I.S. Hybrid vision-based SLAM coupled with moving object tracking. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 867–874.
8. Dinesh Reddy, N.; Abbasnejad, I.; Reddy, S.; Mondal, A.K.; Devalla, V. Incremental real-time multibody vslam with trajectory optimization using stereo camera. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, South Korea, 9–14 October 2016; pp. 4505–4510.
9. Hu, Z.; Qi, B.; Luo, Y.; Zhang, Y.; Chen, Z. Mobile robot V-SLAM based on improved closed-loop detection algorithm. In Proceedings of the 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC) 2019, Chongqing, China, China, 24–26 May 2019; pp. 1150–1154.

10. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
11. Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-scale direct monocular SLAM. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 834–849.
12. Endres, F.; Hess, J.; Sturm, J.; Cremers, D.; Burgard, W. 3D mapping with an RGB-D camera. *IEEE Trans. Robot.* **2013**, *30*, 177–187. [CrossRef]
13. Yi, K.M.; Yun, K.; Kim, S.W.; Chang, H.J.; Choi, J.Y. Detection of moving objects with non-stationary cameras in 5.8 ms: Bringing motion detection to your mobile device. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Portland, OR, USA, 23–28 June 2013; pp. 27–34.
14. Heinly, J.; Dunn, E.; Frahm, J.M. Comparative evaluation of binary features. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 759–773.
15. 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]
16. Leutenegger, S.; Lynen, S.; Bosse, M.; Siegwart, R.; Furgale, P. Keyframe-based visual-inertial odometry using nonlinear optimization. *Int. J. Robot. Res.* **2015**, *34*, 314–334. [CrossRef]
17. Sobral, A.; Vacavant, A. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Comput. Vis. Image Underst.* **2014**, *122*, 4–21. [CrossRef]
18. Konolige, K.; Bowman, J.; Chen, J.D.; Mihelich, P.; Calonder, M.; Lepetit, V.; Fua, P. View-based maps. *Int. J. Robot. Res.* **2010**, *29*, 941–957. [CrossRef]
19. Grisetti, G.; Kummerle, R.; Stachniss, C.; Burgard, W. A tutorial on graph-based SLAM. *IEEE Intell. Trans. Syst. Mag.* **2010**, *2*, 31–43. [CrossRef]