Real-time People Detection and Mapping System for a Mobile Robot using a RGB-D Sensor

Francisco F. Sales, David Portugal and Rui P. Rocha
Institute of Systems and Robotics, University of Coimbra, Coimbra, Portugal

Keywords: People Detection, Mapping, Mobile Robot, RGB-D Sensor, ROS.

Abstract: In this paper, we present a robotic system capable of mapping indoor, cluttered environments and, simultaneously, detecting people and localizing them with respect to the map, in real-time, using solely a Red-Green-Blue and Depth (RGB-D) sensor, the Microsoft Kinect, mounted on top of a mobile robotic platform running Robot Operating System (ROS). The system projects depth measures in a plane for mapping purposes, using a grid-based Simultaneous Localization and Mapping (SLAM) approach, and pre-processes the sensor's point cloud to lower the computational load of people detection, which is performed using a classical technique based on Histogram of Oriented Gradients (HOG) features, and a linear Support Vector Machine (SVM) classifier. Results show the effectiveness of the approach and the potential to use the Kinect in real world scenarios.

1 INTRODUCTION

One of the main use of robots is to replace humans in unpleasant situations, such as repetitive manufacturing tasks and dangerous environments. In these harsh scenarios, robots are usually mobile and should be able to explore, map and detect people, *e.g.* in the case of Search and Rescue (SaR) missions and after an industrial accident, involving the leakage of toxic substances, they can be used to assist human first responders (Rocha et al., 2013).

Such missions are critical and of extreme importance because their accomplishment might save many lives. As a consequence, human rescue teams are often subject to specialized training. However, they usually face a lack of technological equipment and risk themselves in this process. Thus, Robotics plays a fundamental role by reducing this risk, and can be a great resource to human rescue teams.

Detecting people and mapping the environment are key tasks in Robotics for SaR missions and other applications. Since these environments are usually dangerous, mobile robots must be endowed with appropriate locomotion skills, provide accurate results, and the whole system should be affordable due to the risk taken in such harsh environments.

This work has been supported by the CHOPIN research project (PTDC/EEA-CRO/119000/2010) funded by "Fundação para a Ciência e a Tecnologia".

In this work, we use a RGB-D sensor, the Kinect, on top of a mobile robotic platform running Robot Operating System (ROS) (Quigley et al., 2009), to map the environment and detect victims from visual cues. To do so, we project depth measurements to 2D and run a 2D Simultaneous Localization and Mapping (SLAM) algorithm. At the same time, we pre-process the point cloud and compute 3D clusters that might contain people. Afterwards, we run a HOG-based classifier on a corresponding portion of the coloured image to assess the presence of people. Finally, we associate the obtained map and the detections to localize people in the map. Although we use a Kinect sensor, our approach can be applied with any RGB-D sensor. Note however that, for outdoor scenarios, the Kinect is unusable due to infra-red interference induced by the sun, but if the depth measures are made available by more capable sensors under those conditions, our approach is still applicable.

This paper is organized as follows: Section 2 reviews important related work; Section 3 presents the proposed system; Section 4 describes the experimental setup and validates the mapping and people detection modules; Section 5 presents and discusses the results of the integrated system; and in Section 6 we draw conclusions and suggest future work.

F. Sales F., Portugal D. and P. Rocha R..
Real-time People Detection and Mapping System for a Mobile Robot using a RGB-D Sensor.
DOI: 10.5220/0005060604670474
In *Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2014), pages 467-474
ISBN: 978-989-758-040-6

2 RELATED WORK

There has been considerable research on SLAM and people detection with laser range finders (LRFs), stereo cameras and, recently, with RGB-D sensors. Surprisingly, it is not common to integrate both efforts, *i.e.* building a map of the environment, localizing the robot with respect to the map and, simultaneously, identifying people within the environment. A recent approach was proposed in (Soni and Sowmya, 2013). However, in contrast to our approach, it is not built around a single RGB-D sensor, which requires performance-oriented approaches to be able to conduct these tasks in near real-time while being able to obtain sufficiently accurate results.

2.1 Mapping

Most popular 2D SLAM algorithms rely on probabilities to cope with noise and estimation errors. There are some popular implementations based on Kalman Filters and Particle Filters (Dissanayake et al., 2001). An alternative approach is graph-based SLAM. In this case, algorithms use the data to build a graph composed of estimated poses, local maps and their relations, in order to compute a consistent global map. ROS, the robotic framework used on this paper, has already available a set of 2D SLAM algorithms, such as *GMapping*, *HectorSLAM*, *KartoSLAM*, etc.

More recently, 3D mapping has also been intensively studied. However, it often relies on stereo cameras (Konolige and Agrawal, 2008), range scanners (Triebel and Burgard, 2005), (May et al., 2009), or monocular cameras (Clemente et al., 2007), thus requiring heavy computation, including aligning consecutive frames, detecting loop closures, and the globally consistent alignment of all processed frames.

The approach used for frame alignment depends on the data to process. However, the Iterative Closest Point (ICP) algorithm is a popular technique for 3D mapping applications (Droeschel et al., 2009). For stereo cameras, Scale-Invariant Feature Transform (SIFT) features (Lowe, 2004), as well as fast descriptors based on random trees (Michael Calonder et al., 2008) computed for keypoints, such as Features from Accelerated Segment Test (Rosten and Drummond, 2006), are often applied. Also sparse feature points can be aligned over consecutive frames via RANdom SAmple Consensus (RANSAC) (Fischler and Bolles, 1981).

Regarding the loop closure problem, most techniques rely on image matching between keyframes. In graph-based techniques, whenever a loop closure is detected, the correspondence between data frames can be used as a constraint in the pose graph, which represents the spatial relationship between frames. The optimization of these pose graphs originates a globally aligned set of frames. In this context, bundle adjustment (Triggs et al., 2000) simultaneously optimizes the pose graph and a map. Other alternatives have also been explored, such as the g2o framework (Kuemmerle et al., 2011).

With the recent massification of RGB-D sensors, most SLAM approaches were adapted to be used with sensors providing 3D dense depth data. This adaptation was required due to the limitation of the field of view (FoV), usually around 60°, and less precise depth measurements. The first constraint can cause problems in the ICP alignment due to the lack of spatial structure, and only a few approaches have been presented that can deal with this particular issue, *e.g.*, the combination of a time-of-flight camera and a CCD camera makes viable to localize the robot (Prusak et al., 2008).

Recently, with the popularization of RGB-D sensors, an approach was presented which uses sparse keypoint matches between consecutive RGB images as an initialization to the ICP algorithm (Henry et al., 2010). However, it has been concluded through experimentation that expensive ICP is not always required. Still, 3D mapping has clearly shown to require more computational effort than 2D mapping.

2.2 People Detection from Visual Cues

People detection is important for various Robotics applications. Much effort has been put in human-robot interaction for the past few years so that robots can engage and interact with people in a friendly way (Ferreira et al., 2013). Detecting and localizing people is essential before initiating such interaction. However, some of this research has relied solely on 2D visual information provided by cameras (Menezes et al., 2003). Some methods involve statistical training based on local features, such as HOG (Dalal and Triggs, 2005), Edge Orientation Histogram (EOH) (Levi and Weiss, 2004), while other methods involve extracting interest points in the image, such as SIFT features. Recently, with the popularization of 3D sensors, much research has been done on people detection. This is also important for intelligent vehicles to avoid collisions. In this context, there is interesting work, such as (Premebida et al., 2009), (Keller et al., 2011), (Llorca et al., 2012).

Another relevant approach using 3D information was proposed by (Satake and Miura, 2009), wherein depth templates are used to detect the upper human body. In (Bajracharya et al., 2009), a reduction of the

point cloud to a 2.5D map is performed to preserve the low computational effort so that detection is based on different 2D features.

Later on, a method that combines both depth information and color images to detect people was introduced (Spinello and Arras, 2011). A HOG-based detector is used to identify human bodies from image data and the Histogram of Oriented Depths (HOD) method is introduced for dense depth data that derives from HOG; and, finally, Combo-HOD probabilistically combines HOG and HOD.

Recently, a method that does not require a Graphics Processing Unit (GPU) implementation and still presents accurate and real-time results was presented (Munaro et al., 2012). The only drawback is that they assume people stand on the ground plane, consequently it does not present accurate results for people that stand considerably above or below that plane, *i.e.* performs poorly for people climbing stairs or sitting behind a table. It processes information from the point cloud by downsampling it. Then, it estimates the ground plane with a RANSAC-based least square method so that it can be removed, thus separating clusters that might contain people. For each of these clusters, a HOG-based people detector is applied to the corresponding part of the RGB image.

2.3 Statement of Contributions

In this work, we aim at providing an insight on performing SLAM and human detection and localization simultaneously, while achieving reliable results and acceptable performance using solely one RGB-D sensor in the mobile robot. Even though much research has been conducted on mapping and people detection with RGB-D sensors, both subjects are not often integrated to assemble a functional system for applications such as SaR missions, where providing rescue teams with a map of the environment and localizing possible victims is of inestimable value.

3 SYSTEM OVERVIEW

As seen in Fig. 1, the proposed system uses a Kinect sensor and comprises two major modules: the People Detection module and the Mapping module. Additionally, it runs under ROS which is the most widely used robotics framework, providing a set of tools, libraries, drivers and other resources that make easier developing robot applications, and provide hardware abstraction (Quigley et al., 2009). The data from the Kinect was retrieved using the OpenNI driver[1] and the driver used for the Pioneer 3-DX mobile robot was ROSARIA[2], both already available in ROS.

3.1 Mapping

Although the Kinect allows to perform RGB-D mapping, our goal is to run a SLAM algorithm along with other tasks, such as people detection, and eventually autonomous exploration.

We opted to project the depth measurements provided by the sensor in the floor plane and simulate a 2D Laser Scan in order to reduce the computational cost. This is represented by the "Depth to LaserScan" block in Fig. 1. It processes the columns of the matrix and creates a vector with the minimum depth value per column, thus originating a vector of 640 distance measures, *i.e.* a 2D scan.

The 2D range measurements are used as an input to the *GMapping* algorithm (Grisetti et al., 2007), already available in ROS, along with odometry information provided by the robot's driver.

This SLAM algorithm was selected for several reasons. Firstly, considering our performance constraints, it does not present a high computational burden. Secondly, the Kinect has a low FoV, which can cause problems in scan matching, therefore the mobile robot's odometry can greatly improve results. Finally, it was shown to be robust in testing and experiments, when compared to other SLAM approaches.

3.2 People Detection

Several people detection algorithms do not take into consideration 3D information, while others use that information to improve results. However, the authors of (Munaro et al., 2012) proposed an algorithm that uses the point cloud generated to lower the computational load of classical people classifiers. Furthermore, ROS provides access to the Point Cloud Library (PCL) (Rusu and Cousins, 2011), which contains algorithms to process 3D data from RGB-D sensors. Therefore, the technical implementation of the algorithm becomes much simplified.

The algorithm firstly processes the point cloud, dividing the space into volumetric pixels (voxels) with an edge length of 0.06m, and reduces the 3D points into a common voxel according to the voxel's centroid. Therefore, we obtain a reduced number of

[1]ROS Wiki - openni_kinect, http://wiki.ros.org/openni_kinect (Accessed: 2014-06-21)

[2]ROS Wiki - ROSARIA, http://wiki.ros.org/ROSARIA (Accessed: 2014-06-21)

points and also a point cloud with approximately constant point density, avoiding its variation with the distance from the sensor.

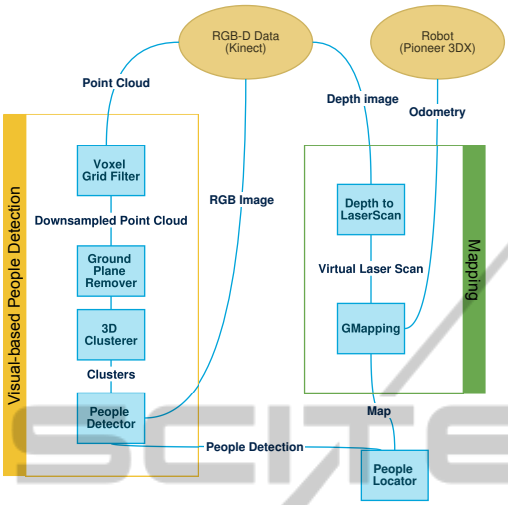

Figure 1: System Overview.

With a filtered point cloud, and considering the assumption that people stand on the ground, the ground plane's coefficients are estimated and updated at every frame using a least square method, therefore it is robust to small changes such as those experienced when a mobile robot is moving. At this stage, points located in the ground plane are removed, by discarding every point located at a distance to the estimated ground plane lower than a threshold of 6cm. As a consequence, the remaining clusters become no longer connected by this common plane.

After this first stage of point cloud processing, the different clusters can now be computed by labelling neighbouring 3D points on the basis of their Euclidean distances. In our case, we started by considering that points closer than a threshold of 2 times the voxel edge belonged to the same cluster. However, this process may lead to errors, *e.g.* dividing partially occluded people into different clusters, or merging different people in the same cluster when they are near each other. As for the second issue, the algorithm uses the position of the heads, that generally are not so close and occluded, to divide these clusters into subclusters, so that people merged previously in a single cluster are separated into different clusters.

For the clusters obtained earlier, a HOG-based detector (Dalal and Triggs, 2005) is applied to the portion of the RGB image corresponding to the fixed aspect ratio bounding box that contains the whole cluster. This process includes the computation of HOG features and their application to a trained linear SVM[3]. The SVM is a learning model that allows us to classify the data based on its training.

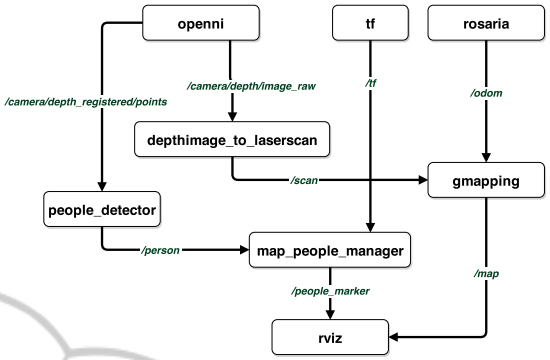

Figure 2: Summarized overview of the system in ROS. Boxes refer to ROS nodes and arcs to topics.

3.3 Integration

Although both modules run in the same system, their data is not in the same reference frame: detections are made on the Kinect frame which is different from the reference frame of the map. To deal with this issue, we created a an additional ROS node (see node *map_people_manager* in Fig. 2) that subscribes to the detections, transforms their coordinates to map coordinates, using ROS tools, and manages the detections, avoiding multiple detections of the same person in the same position. Also, it publishes the corresponding markers to allow the visualization of the map and the detections on the real relative position in the map. The significant portion of the *rqt_graph* is of the ROS system is presented if 2.

4 EXPERIMENTAL SETUP AND VALIDATION

In order to validate the people detection and mapping solution, we used the experimental setup depicted in Fig. 3, with the addition of a laptop on the robot's platform. The Kinect sensor was tilted 8° up so that the operating range for people detection is not affected by the relative position to the ground plane, *i.e.* point clouds will contain the whole person instead of half body at closer distances. The test scenario was indoor and was located in AP4ISR lab of the Institute of Systems and Robotics of the Univ. of Coimbra (ISR-UC). Our experimental work was divided into three stages: mapping method validation, people detection validation, and integrated system validation.

[3]The SVM was trained using the well known *INRIA Person Dataset.* (URL: http://pascal.inrialpes.fr/data/human)

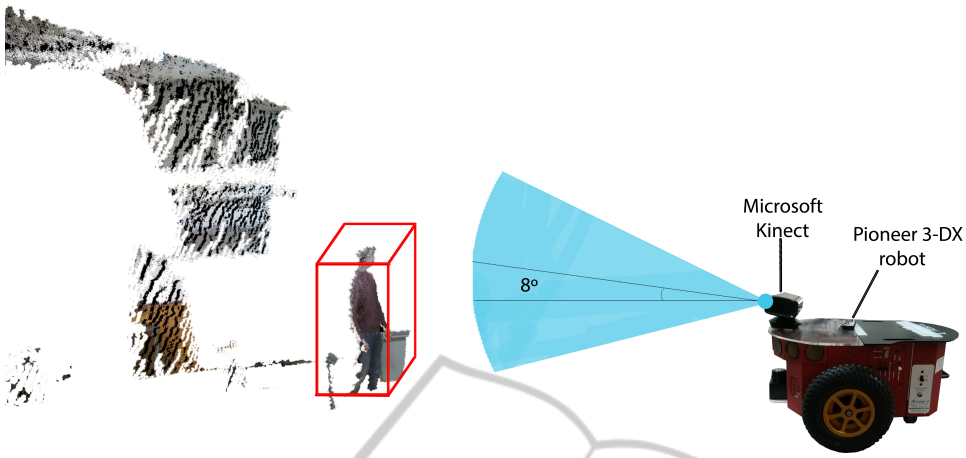

Figure 3: Mobile Robot (Pioneer 3-DX) with a Microsoft Kinect mounted on top.

4.1 Mapping Validation

In order to validate the mapping task with the Kinect sensor, we attached to our robot a Hokuyo URG-04LX-UG01 LRF to produce maps to be compared with the ones obtained using the Kinect and the method described in sec. 3.1. The environment tested was a lab arena with approximately 4.6×4.0m, as illustrated in Fig. 4. The robot was teleoperated using an *ssh* remote connection, while running *GMapping*.

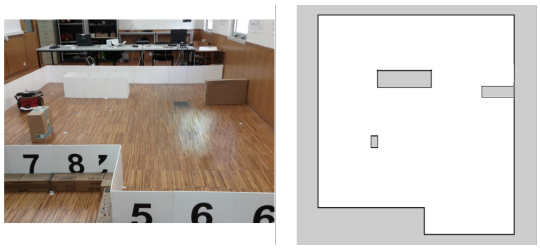

Figure 4: Photo of the test area (left) and ground truth map (right).

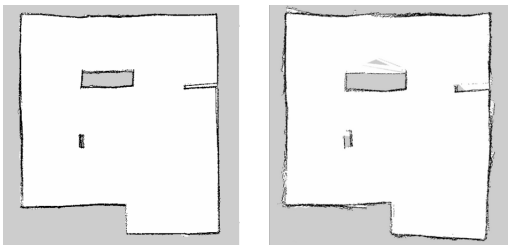

Figure 5: Maps produced with the Hokuyo URG-04LX-UG01 LRF sensor (left) and the Kinect sensor (right).

By visually comparing both maps, in Fig. 5, it becomes evident that the Kinect is not as accurate as the LRF and that its limited FoV, range, and lower accuracy have a negative impact on the results obtained. Nevertheless, both maps are easily interpretable by the human eye. We computed the absolute pixel-wise matching of all pixels in the maps generated to assess their quality, and obtained acceptable matching rates, as shown in Table 1. In order to compute the matching metric, we binarized the maps obtained and the ground truth, calculated the best fit alignment by rotating the maps, and computed the pixel-wise match of each pixel in the image.

Table 1: Pixel-wise matching rates.

Maps	Matching Rate
Ground truth - Laser	96.9 %
Ground truth - Kinect	94.3 %

4.2 People Detection Validation

Despite the availability of some datasets, they do not comply with the constraints and our hardware setup in Fig. 3, mostly because the Kinect is only 24cm above the ground, so it is tilted up to acquire visual information containing people. In order to validate our people detection method, we captured a dataset of about 100 frames that was manually annotated with the people present in each frame. It contains one person walking in several directions at a distance of 1 to 4 meters to the camera frame (an example is shown in Fig. 6). Therefore, we have a dataset of binary decision. This way, we were able to acquire data in similar conditions to the final intended applications.

We applied the people detection method implemented in ROS to process point clouds of the dataset, and extract results (true positives, true negatives, false positive and false negatives) in Receiver Operating Characteristic (ROC) curves (see Fig. 7 and Fig. 8). The ROC curve is a graphical plot which illustrates

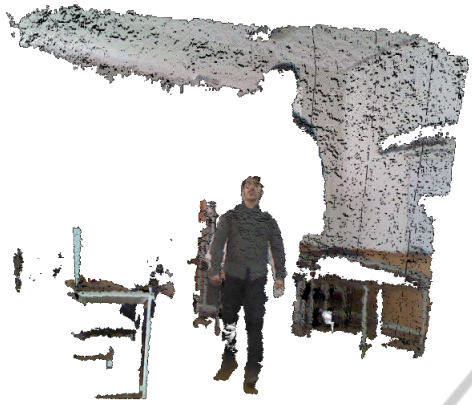

Figure 6: Example of a point cloud from the dataset.

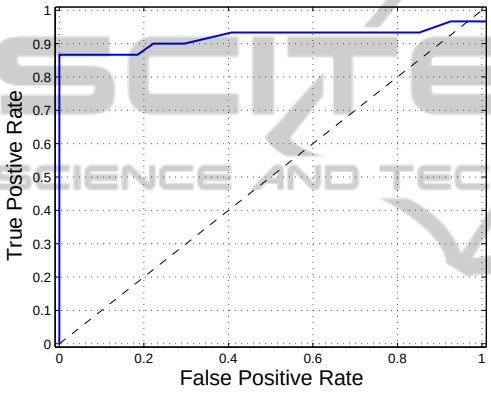

Figure 7: ROC curve on our dataset.

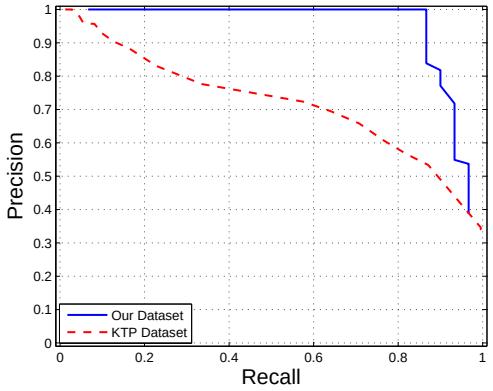

Figure 8: Precision-Recall for our dataset and for the KTP dataset.

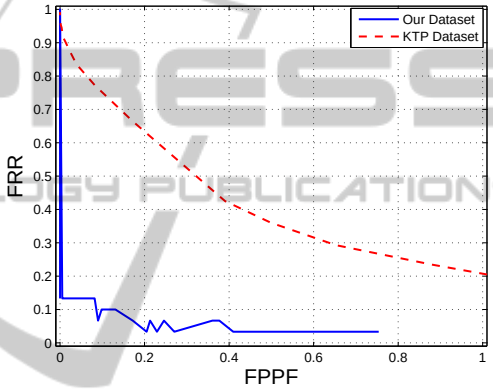

Figure 9: Detection Error Tradeoff (DET) for our dataset and for the KTP dataset.

the performance of a binary classifier system as the discrimination threshold is varied.

Afterwards, we applied the method to cross validate with the Kinect Tracking Precision (KTP) Dataset from (Munaro et al., 2012), which contains sequences of multiple people captured from a static camera.

Fig. 7 relates the True Positive and False Positive Rates (TPR and FPR). Perfect results are near the top left corner with 0% FPR and 100% TPR. Fig. 8 shows the precision and recall percentages for each experiment. The ideal result is situated on the top right corner with 100% recall and precision. Fig. 9 is the Detection Error Tradeoff (DET) curve which relates the False Rejection Rate (FRR) in percentage and the number of False Positives per Frame (FPPF). The best result is located on the bottom left corner.

We observed in the results obtained with our dataset that the method is very robust in terms of false negatives, showing a low FPR for high enough TPR, *e.g.* 86.87% TPR for $\approx$ 0.00% FPR. This is also visible in the high precision shown in Fig. 8, even for high recall values, *e.g.* 100% precision for 86.67% recall. We were able to achieve an accuracy of 92.98% which shows the reliability of the method.

Note however that this analysis is performed independently for each processed point cloud (each frame provided by Kinect). In real world applications, with depth data from Kinect at 30 *fps*, we will capture several frames for each person, which allows us to gain certainty when detecting a person in short time intervals. In the case of SaR missions, it is very important to lower the false positives as much as possible to avoid wasting resources and time while keeping a high FPR to be capable of detecting all the victims.

The low number of frames and the presence of only one person is clear in the curves and led us to run the method with the KTP dataset. The results obtained on the KTP dataset were comprehensively not as good as the ones with our dataset, since the former is a more complex dataset containing up to 5 people in the same sequence. Still, the accuracy of the method for a single frame is enough considering the amount of frames available that we can process for detecting each person. We did not compute the ROC curve for this dataset because it aims to assess a bi-

nary classification problem and it barely contains binary decisions due to the nature of the tracking problem. Still, we can conclude from the Precision-Recall and DET curves that the results are accurate enough for our intended applications, *e.g.* 72.39% precision for 57.51% recall and 0.39 FPPF for 42.49% FRR.

5 EXPERIMENTAL TESTS WITH A MOBILE ROBOT

To validate the system, we used the previously presented setup in an indoor uncontrolled and cluttered environment in the AP4ISR lab of ISR-UC. The environment contains desks, tables, placards, chairs, hardware and all kinds of objects (see Fig. 10). The systems performed all the computation and also displays the results in real-time.

Figure 10: Photo of the environment used to test the whole system.

The environment was reliably mapped (see Fig. 11). We note that the 2D mapping method uses the desks top edges in the mapping so the space under them are not considered due to the point cloud downsampling. On the other hand, for people detection purposes, the whole space is considered, therefore if a detection is made under a table, it would still be represented in the map.

In our experiments, we used three subject standing on different locations, two real humans (one male and one female) and a human model (seen in Fig. 10). In all experiments, the robot was able to map fairly accurately and detect all of them. Fig. 11 shows a picture of the ROS visualization software *rviz* with the ongoing construction of the map and the detections made so far. The experiment depicted lasted 4,3 minutes. The robot was teleoperated with a Wii Remote Control connected via bluetooth to the laptop mounted on top of the robot. The system depicted in Fig. 1 and in Fig. 2 was run on a laptop with an Intel Core i7-4700MQ CPU, 16GB of RAM, Ubuntu 12.04, and ROS Hydro. We computed the average CPU load along the experiment, which resulted in 44.71% of CPU usage and an average of 16.07 *fps* was processed. This frame rate could be increased through the parallelization of the code in a GPU. Also the results demonstrate that the system performs well in real word scenarios and its computational load leaves room to incorporate further modules in the system, such as additional sensory cues, *e.g.* audio input, and perform other tasks in parallel, such as autonomous navigation and exploration.

Figure 11: Map obtained and people detected.

6 CONCLUSION

This paper proposed an integrated system that is able to successfully map the environment, localize the robot with respect to the map and, simultaneously, detect and localize people within the environment, while relying solely in a RGB-D sensor. However, we intend in our future work to have a system that is also able to autonomously explore the environment. This will probably require an upgrade of the current hardware setup of Fig. 3, due to the narrow FoV of the Kinect which may cause unreliable navigation. Also, our goal is a system that can be used to perform SaR missions, the robot should be able to navigate towards victims, to eventually interact with them. We intend to study the use of a second Kinect sensor to achieve a wider FoV. This improvement does not imply great costs and should yield a safer navigation.

In the future, we also intend to take advantage of other capabilities of the Kinect sensor, such as processing audio information from its microphone array to improve people detection results. Furthermore, we would like to test the system in other applications such as automated patrolling and surveillance with robotic teams (Portugal and Rocha, 2013).

REFERENCES

Bajracharya, M., Moghaddam, B., Howard, A., Brennan, S., and Matthies, L. H. (2009). A fast stereo-based system for detecting and tracking pedestrians from a moving vehicle. *Int. Journal of Robotics Research*, 28(11-12):1466–1485.

Clemente, L., Davison, A., Reid, I., Neira, J., and Tardós, J. D. (2007). Mapping large loops with a single hand-held camera. *Proc. Robotics: Science and Systems Conf.*

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. *Int. Conf. on Computer Vision & Pattern Recognition*, 2:886–893.

Dissanayake, M. G., Newman, P., Clark, S., Durrant-Whyte, H. F., and Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem. *Rob. & Automation, IEEE Tr. on*, 17(3):229–241.

Droeschel, D., May, S., Holz, D., Ploeger, P., and Behnke, S. (2009). Robust ego-motion estimation with ToF cameras. *European Conf. on Mobile Robots*, pages 187–192.

Ferreira, J. F., Lobo, J., Bessière, P., Castelo-Branco, M., and Dias, J. (2013). A Bayesian framework for active artificial perception. *IEEE Trans. on Cybernetics (Part B)*, 43(2):699–711.

Fischler, M. A. and Bolles, R. C. (1981). RANdom SAmple Consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM*, 24(6):381–395.

Grisetti, G., Stachniss, C., and Burgard, W. (2007). Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Trans. on Robotics*, 23:2007.

Henry, P., Krainin, M., Herbst, E., Ren, X., and Fox, D. (2010). RGB-D mapping: Using depth cameras for dense 3D modeling of indoor environments. *Experimental Robotics*, 79:477–491.

Keller, C. G., Enzweiler, M., Rohrbach, M., Fernandez Llorca, D., Schnorr, C., and Gavrila, D. M. (2011). The benefits of dense stereo for pedestrian detection. *ITS, IEEE Trans. on*, 12(4):1096–1106.

Konolige, K. and Agrawal, M. (2008). FrameSLAM: From bundle adjustment to real-time visual mapping. *Robotics, IEEE Trans. on*, 24(5):1066–1077.

Kuemmerle, R., Grisetti, G., Strasdat, H., Konolige, K., and Burgard, W. (2011). g2o: A general framework for graph optimization. *ICRA*, pages 3607–3613.

Levi, K. and Weiss, Y. (2004). Learning object detection from a small number of examples: the importance of good features. *CVPR*, pages 53–60.

Llorca, D., Sotelo, M., Hellín, A., Orellana, A., Gavilan, M., Daza, I., and Lorente, A. (2012). Stereo regions-of-interest selection for pedestrian protection: A survey. *Transportation research part C: emerging technologies*, 25:226–237.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. Journal Computer Vision*, 60(2):91–110.

May, S., Droeschel, D., Holz, D., Fuchs, S., Malis, E., Nüchter, A., and Hertzberg, J. (2009). 3D mapping with ToF cameras. *Journal of Field Robotics, sp. issue on 3D Mapping*.

Menezes, P., Brethes, L., Lerasle, F., Danes, P., and Dias, J. (2003). Visual tracking of silhouettes for human-robot interaction. pages 971–976.

Michael Calonder, Vincent Lepetit, and Pascal Fua (2008). Keypoint signatures for fast learning and recognition. *In European Conf. on Computer Vision*.

Munaro, M., Basso, F., and Menegatti, E. (2012). Tracking people within groups with RGB-D data. *IROS*, pages 2101–2107.

Portugal, D. and Rocha, R. P. (2013). Distributed multi-robot patrol: A scalable and fault-tolerant framework. *Robotics & Auton. Syst.*, 61(12):1572–1587.

Premebida, C., Ludwig, O., and Nunes, U. (2009). Lidar and vision-based pedestrian detection system. *Journal of Field Robotics*, 26(9):696–711.

Prusak, A., Melnychuk, O., Roth, H., Schiller, I., and Koch, R. (2008). Pose estimation and map building with a time-of-flight camera for robot navigation. *Int. Journal Intell. Syst. Technol. Appl.*, 5(3/4):355–364.

Quigley, M., Conley, K., Gerkey, B. P., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. Y. (2009). ROS: an open-source robot operating system. *ICRA Workshop on Open Source Software*.

Rocha, R. P., Portugal, D., Couceiro, M., Araujo, F., Menezes, P., and Lobo, J. (2013). The CHOPIN project: Cooperation between Human and rObotic teams in catastroPhic INcidents. *SSRR*, pages 1–4.

Rosten, E. and Drummond, T. (2006). Machine learning for high-speed corner detection. *In European Conf. on Computer Vision*, pages 430–443.

Rusu, R. and Cousins, S. (2011). 3D is here: Point Cloud Library (PCL). *ICRA*, pages 1–4.

Satake, J. and Miura, J. (2009). Multiple-Person Tracking for a Mobile Robot Using Stereo. *MVA Conf.*, pages 273–277.

Soni, B. and Sowmya, A. (2013). Victim detection and localisation in an urban disaster site. *ROBIO*, pages 2142–2147.

Spinello, L. and Arras, K. O. (2011). People detection in RGB-D data. *IROS*, pages 3838–3843.

Triebel, R. and Burgard, W. (2005). Improving simultaneous localization and mapping in 3D using global constraints. *AAAI*, 20(3):1330.

Triggs, B., Mclauchlan, P., Hartley, R., and Fitzgibbon, A. (2000). Bundle adjustment – a modern synthesis. *Vision Algorithms: Theory and Practice, LNCS*, pages 298–375.