2015 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)
Sept 14-16, 2015. San Diego, CA, USA

Fusing Low-cost Sensor Data for Localization and Mapping of Automated Guided Vehicle Fleets in Indoor Applications

Christina Ramer, Julian Sessner, Michael Scholz, Xu Zhang, and Jörg Franke

Abstract— **This paper describes the joint use of external and on-board low-cost sensors for small to mid-size automated guided vehicles (AGVs) in varying indoor applications. Localization and mapping for navigation purposes of multi-scale vehicle fleets is achieved by combining fixed, one-time calibrated cameras with odometry information and cheap LIDAR (Light detection and ranging) and collision avoidance sensors, which are mounted on the vehicles. Our approach is scalable and applies to different domains, as logistics in production environments or hospital facilities. Experiments show that the proposed system concept, based on low-cost units, reaches sufficient accuracy in real-time and could therefore be a promising solution for the target applications.**

I. Introduction

Automated guided vehicles (AGVs) are used to transport different kind of goods automatically as logistic support. The fields of application vary from manufacturing to hospital environments. Following fixed guidelines is one easy, commonly used method to perform navigation. The disadvantage of such systems is, that they are only able to move on restricted paths and special infrastructure is needed. Using sensors allows the AGV to map the surrounding environment, localize itself, detect obstacles and perform path planning tasks on its own. Usually complex sensor systems like expensive laser scanners are needed to meet the requirements. The overall costs of such AGVs usually rise drastically.

Using external fixed, relatively cheap cameras allows to reduce the vehicles' costs. The idea behind using more external sensors which can be used for several AGVs is, that only the infrastructural costs rise and the costs per AGV can be kept low. The camera images can be used for localization, mapping and, if necessary, destination detection based on optical markers. Nevertheless performing navigation only with external cameras is subject to several restrictions. Vehicles are only able to move in camera monitored areas. The accuracy of detection can be low. And if the communication between the camera and AGV is interrupted the AGV isn't able to perform any transportation task.

In this work we present a method to integrate external ceiling mounted cameras, newly available low-cost laser scanners and incremental sensors to map the environment and to localize AGVs. By adding a low-cost internal laser scanner the range of each AGV isn't limited to camera monitored areas. The presented concept can be used as a basis for different kinds of AGVs with different sizes and purposes. This enables to use AGVs flexibly.

The paper is structured as follows. The next section describes existing AGV concepts and shows the usage of AGVs in different situations, followed by the description of our system setup. In section IV the image processing and estimation of the vehicles' poses based on fixed cameras are explained. The use of low-cost laser scanner data using open source packages to set up navigation algorithms is described in section V, followed by the integration of the localization and mapping based on the camera data in section VI. In section VII the localization accuracies using the low-cost laser scanner and the external camera data are evaluated separately. Finally, based on the main results of our work possible future optimizations are discussed.

II. Related Work

The company Swisslog provides different kinds of AGVs for several industrial purposes [1]. For navigation the systems follow either fixed paths with additional near field sensors to detect obstacles in their path, or use high-cost laser scanners. Their payload can weigh up to 4000 kg. Another commercially available product is the autonomous indoor vehicle platform Lynx by Adept [2]. The mobile robot can be customized for different applications and payloads. The navigation is done using a high-cost laser scanner. Low-cost laser scanners are already in use in consumer products like service robots. For instance, the company Vorwerk provides a robotic vacuum cleaner Kobold VR200, originally based on the Neato Robotics platform, using such laser scanner for navigation [3].

External cameras as basis for navigation of mobile robots have been introduced previously. In [4] cameras are used to detect several robots and their positions in a playing field during the robotic competition RoboCup. The cameras detect the robots with the help of unique markers on the top. Their positions can be determined by calculating the distance to fixed markers on the playing field. The resulting data is provided to both participating teams. In [5] external cameras detect AGVs, too. The vehicles are identified by special markers and after their localization the path planning is realized by a central computer.

The presented systems are either inflexible due to fixed paths, use expensive sensor systems or the applied cameras need to detect special markers being placed on the top of the vehicles. Systems with integrated low-cost sensors like the mentioned vacuum cleaner don't perform global localization in known maps and conduct path planning to chosen destinations, so that they can't be used as AGVs so far. In our approach we consistently use low-cost sensors, a method

C. Ramer, J. Sessner, M. Scholz, X. Zhang and J. Franke are with the Institute for Factory Automation and Production Systems (FAPS), Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 91058 Erlangen, Germany (corresponding author to provide phone: +49-9131-85-27711; fax: +49-9131-302528; e-mail: Christina.Ramer@faps.fau.de).

978-1-4799-7772-7/15/$31.00

which allows to localize AGVs without the need of unique markers in a decentralized flexible structure.

III. System Setup

Fig. 1 shows the relevant system and data processing components as well as the data flow between the components using one or more external cameras and AGVs. The main hardware parts are fixed ceiling cameras and the AGVs containing an internal computer, a LIDAR (Light detection and ranging) and incremental sensors. The image processing of the ceiling mounted cameras is conducted by embedded systems for each camera in a decentralized and easily expandable way. The AGVs' trajectories are received via WLAN, compared with the trajectories of detected obstacles and the resulting data is sent back to the AGVs. A more precise description of the image processing and trajectory comparison follows in the next chapter.

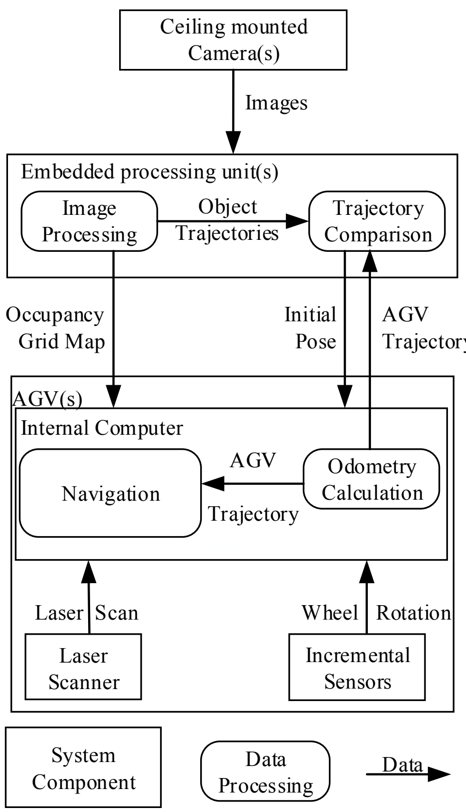

Figure 1: Communication between the relevant system components and data processing parts.

As mobile test platform a small scale automated guided vehicle, which has been developed as a versatile alternative for workpiece carriers at our institute and presented in [6], is used. Fig. 2 shows an image of the AGV with integrated low-cost sensors. As basic onboard environment sensor the laser scanner RPLidar by RoboPeak (Shanghai, China) is placed on the AGV's top in the test setup. For later use its installation position has to be changed according to the respective application. Furthermore, several range sensors and a depth camera are included on the front. A low-cost computer, the MIO-2261 by Advantech with a 1.6 GHz dual core processor, is integrated into each AGV. Here the main calculation of navigation tasks like mapping, path planning and localization takes place and the data from the AGV's internal sensors like laser scanner, incremental sensors or other additional sensors get processed and fused. Currently only the laser scanner and the incremental sensors are employed in our approach.

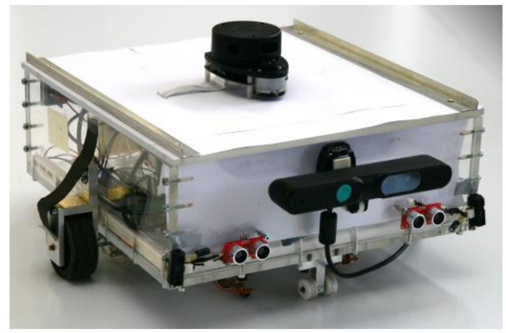

Figure 2: Mobile test platform with differential drive. As low-cost sensors a laser scanner, several range sensors and a depth camera are integrated.

IV. External Camera Image Processing

The method used in this work for assigning detected objects in camera images to known vehicles and obstacles, called the Hungarian Method, has been introduced in [7] and [8] to mobile robotics. This algorithm allows to localize vehicles in camera monitored areas without the need of special markers or structures. This enables to carry different kinds of payloads on the top of the AGVs and the AGVs' sizes may vary. The localization is realized by comparing the trajectory of detected objects with the vehicles' trajectories. All other detected objects, which cannot be recognized as AGVs, are treated as obstacles, or destinations if necessary. In this work the localization based on fixed, ceiling mounted cameras is used as initial localization, as the global localization using only the laser scanner data is usually inaccurate and time consuming, especially due to the short range of the used LIDAR compared to high-cost sensors. Furthermore, the detected obstacles in the camera monitored areas are used to update the static map in these regions.

A. Object Detection

In our system setup low-cost cameras are mounted on the ceiling of the manufacturing area facing towards the floor. The general aim of the associated image processing is to detect the contours of objects in the current camera image. This has several advantages over detecting special markers on the AGVs, as there is no need of expensive high resolution cameras to detect small markers and the payload can cover the whole vehicles. Based on a static background image of the monitored area and a simple frame differencing with current images, object detection is conducted. Using further image processing routines like smoothing, edge and contour detection, the positions of the objects in the image are calculated. Finally two data outputs of different file sizes are provided: one lightweight contour image of all identified objects with their current positions; and one, more resource consuming, grayscale image as a running average of foreground areas for static obstacles in order to update the occupancy grid map, as explained later.

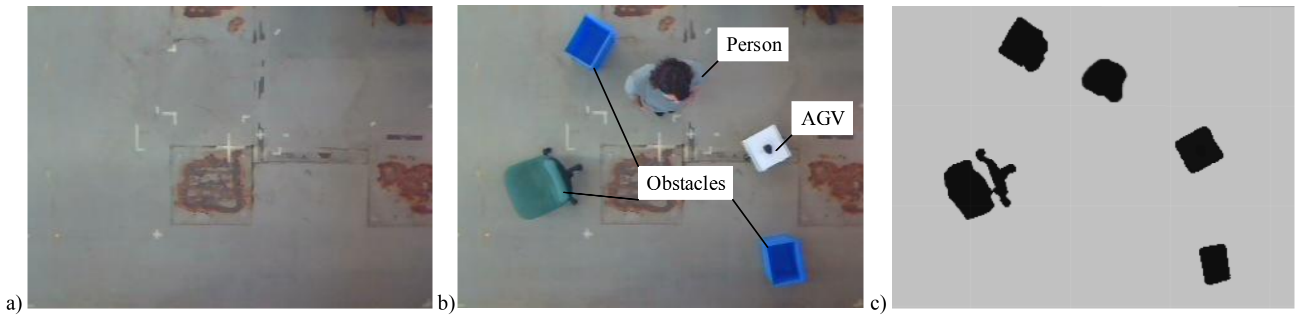

Figure 3: a) Background image without obstacles. b) Current image containing an AGV, a person and several obstacles. c) Grayscale image, used for occupancy grid map updates, created by subtraction of a) from b) and further image processing.

Fig. 3 a) shows an exemplary background image of a camera monitored area, which is subtracted from the current camera image in b). The resulting grayscale image, used for occupancy grid map updates, is shown in Fig. 3 c).

B. Algorithm of the Hungarian Method for the vehicles' state estimation

To separate vehicles and obstacles from the detected objects, the AGVs send their trajectory, calculated by incremental sensors, to the cameras' embedded system units.

The adapted Hungarian Method, explained in [9], localizes the AGVs by comparing detected objects' trajectories with the AGVs' trajectories. The objective is to calculate the pose of each vehicle's internal coordinate system in the camera's coordinate system. Let x' and y' be one AGV's inertial coordinate system and x^*, y^* be one camera coordinate system. The transformation between these coordinate systems can be described by translations x_0^* and y_0^*, a rotation θ_0 and a sizing factor p_0. Fig. 4 shows the relation between those two coordinate systems.

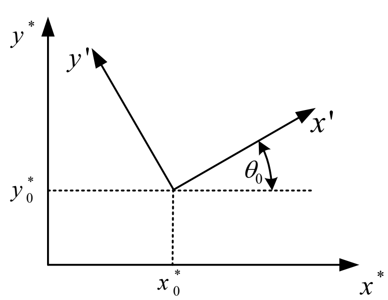

Figure 4: Transformation between the camera coordinate system (x^*, y^*) and the AGV's inertial coordinate system (x', y'). [7]

The following equation calculates the projection of a point of the inertial coordinate system into the camera coordinate system using the mentioned transformation parameters:

$$\begin{pmatrix}x^*\\ y^*\end{pmatrix}=\begin{pmatrix}\cos(\theta_0) & -\sin(\theta_0)\\ \sin(\theta_0) & \cos(\theta_0)\end{pmatrix}\begin{pmatrix}x'\\ y'\end{pmatrix}p_0+\begin{pmatrix}x_0^*\\ y_0^*\end{pmatrix} \tag{1}$$

Using the substitutions

$$a_0=\cos\theta_0\cdot p_0 \tag{2}$$

$$b_0=\sin\theta_0\cdot p_0 \tag{3}$$

simplifies (1) to:

$$\begin{pmatrix}x^*\\ y^*\end{pmatrix}=\begin{pmatrix}a_0 & -b_0\\ b_0 & a_0\end{pmatrix}\begin{pmatrix}x'\\ y'\end{pmatrix}+\begin{pmatrix}x_0^*\\ y_0^*\end{pmatrix}=\begin{pmatrix}x' & -y' & 1 & 0\\ y' & x' & 0 & 1\end{pmatrix}\begin{pmatrix}a_0\\ b_0\\ x_0^*\\ y_0^*\end{pmatrix} \tag{4}$$

The scalars x^*, y^*, x' and y' in (4) can be extended to vectors X^*, Y^*, X' and Y' of dimension q×1, containing the q waypoints of the trajectories measured by internal odometry sensors and by camera image processing. The elements 1 and 0 are furthermore replaced by q×1 vectors O and Z containing q ones respectively q zeros. Thereby (4) changes to the following equation:

$$\begin{pmatrix}X^*\\ Y^*\end{pmatrix}=\begin{pmatrix}X' & -Y' & O & Z\\ Y' & X' & Z & O\end{pmatrix}\begin{pmatrix}a_0\\ b_0\\ x_0^*\\ y_0^*\end{pmatrix} \tag{5}$$

To calculate a_0, b_0, x_0^* and y_0^* the pseudo inverse matrix (pinv()) needs to be determined.

$$\begin{pmatrix}a_0\\ b_0\\ x_0^*\\ y_0^*\end{pmatrix}=\mathrm{pinv}\begin{pmatrix}X' & -Y' & O & Z\\ Y' & X' & Z & O\end{pmatrix}\begin{pmatrix}X^*\\ Y^*\end{pmatrix} \tag{6}$$

Using the calculated values and equation (5), the inertial trajectory waypoints can be projected into the camera coordinate system. To compare two trajectories an error value e is calculated as follows:

$$e=\frac{1}{q}\sum_{i=1}^{q}\begin{pmatrix}x_i^p-x_i^*\\ y_i^p-y_i^*\end{pmatrix} \tag{7}$$

where (x_i^p, y_i^p) is a waypoint of the projected trajectory and (x_i^*, y_i^*) is a waypoint of an object's trajectory detected by the camera. By comparing the error value with a predefined threshold each detected object can be defined as the considered vehicle, if the error e is below the threshold. To determine each AGV's pose the scaling factor p_0 and the AGV's orientation need to be calculated as follows:

$$p_0=\sqrt{a_0^2+b_0^2} \tag{8}$$

$$\theta_0=\mathrm{atan2}(b_0,a_0) \tag{9}$$

V. Use of Low-cost Laser Scanner Data

The laser scanner integrated in our approach is a low-cost two dimensional LIDAR system. It is distributed for 400 US$, whereas other laser scanners used in AGVs mentioned in chapter II are usually way more expensive. The laser scanner detects obstacle within a range of 6 m at a sampling rate of 2000 samples/s and a scanning rate of 5.5 Hz.

Using such a low-cost, yet efficient sensor helps to keep the costs per AGV low and extends the usable area of camera monitored regions. The laser scanners in this work serve several purposes. First of all, their data are used to create a static map of the environment. Second, based on this map, localization is performed using a particle filter and finally their current data are deployed during path planning to detect obstacles and to avoid collisions. To fulfill these requirements the Robot Operating System (ROS) is used. Especially the navigation metapackage contains the main packages applied.

A. Static Environment Mapping

A static map of the application environment is created once by use of a Rao-Blackwellized particle filter, as described in [10] and implemented in the gmapping package. The algorithm is a solution to the SLAM (Simultaneous Localization and Mapping) problem and fusions current laser scanner data with odometry measurements to create a two dimensional occupancy grid map. This map can be described as an image containing white pixels as free space, black pixels as obstacles and grey pixels as unknown areas. Depending on the real size of the map and its resolution the size of the map can be large.

B. Localization of the Automated Guided Vehicles

To realize the moving robots' localization their laser scanner data and the adaptive Monte Carlo localization (amcl) are used. Amcl is a dynamic implementation of a particle filter, introduced in [11]. The advantage of a dynamic implementation of the particle filter comes from the different tasks during localization. Whereas up to several thousand particles spread over the whole map are used for the initial localization, the particle filter needs only several hundred particles once the robot is located in the right pose. The adaption of the particle number helps to run the localization with an optimal usage of computational power.

C. Data Usage during Path Planning

For collision free path planning two different kind of occupancy grid maps, the static global map and dynamic local maps are used, as described in [12] and provided by the costmap_2d package. Both map types contain obstacles, which get inflated due to the size of the AGVs and measurement inaccuracies, and serve as so-called costmaps. The static map is based on a previously created occupancy grid map that we recorded once as described above. The inflated global costmap is used by the global path planner to create a path from an AGV's current pose to a transmitted goal pose. As path planning is not focused in this work, an implemented path planner from the navigation package, based on the potential field method introduced in [13], is used. The dynamic local costmaps correspond to an individual AGV, have a fixed size and are always centered at their current pose. They only consider the present data provided by onboard sensors, as the low-cost laser scanner, simple range sensors or more sophisticated depth cameras. Currently recognized surrounding obstacles are integrated into these local costmaps, that serve for local path planning after inflating the hereby occupied areas. The local path planner used in this work is the Dynamic Window Approach [14]. It creates the actual speed commands sent to the AGV's drive by trying to follow the global path and avoiding obstacles mapped in its local costmap. Thus, even in the case of a false localization or newly detected obstacles, collisions between the AGVs and obstacles are avoided.

Nevertheless, the single use of this low-cost LIDAR comes with several restrictions. Especially compared to more expensive laser scanners the range of the sensor is quite short. This may lead during the initial global localization to a quite long lasting, computational expensive procedure until the AGV is located in its correct pose. Another disadvantage rises from the use of a global static map for path planning. If there are changes in the areas like small hallways or other areas between the current pose and the goal pose, they can't be recognized and may let the AGV trap into a dead end.

VI. Fusion of Camera and Laser Scanner Data

To overcome these issues the localization and mapping via external ceiling cameras are included in our approach. Therefore, the transformation between the used coordinate systems need to be implemented. The global coordinate system called map, in which the adaptive Monte Carlo localization takes place, is the coordinate system of the static occupancy grid map. The AGVs' inertial coordinate systems are called base_link. As the odometry measurement's initial coordinate systems are odom, the transformations between the individual odom and the base_link coordinate systems are based on the movements of the AGVs and calculated from the wheel rotations reported by the incremental sensors. To include the camera data, the transformation between the map and the camera coordinate system (cam) needs to be determined by a one-time calibration.

A. Trajectory Comparison as Initial Localization

To include the localization based on the ceiling cameras the poses of all AGVs have to be calculated in the global map coordinate system. Therefore, the robots need to be in a camera monitored area. To start the initial localization with the described algorithm, the AGVs need to move and send their calculated trajectory to the embedded processing unit(s). There the trajectories of the detected moving objects are calculated and compared with the AGVs'. If there are matching trajectories, the positions of the corresponding robots in the camera images are converted from pixels into meters, using the defined resolution of the camera image. Afterwards, the pose is projected from the individual cam coordinate system into the global map coordinate system. The amount of particles at the initial localization is set to a predefined maximum value. In case of a correct localization the number decreases quickly.

B. Static Map Updating by the Generated Camera Map

Each camera's calculated occupancy grid map, based on the grayscale image of the detected obstacles, is integrated into the navigation workflow by updating the global static costmap. Therefore the individual camera's occupancy grid is provided together with its real world size and the position and orientation of the cam coordinate system in the map coordinate system. Thereby the static costmap, originally based on one-time generated laser scanner data, gets regularly updated in areas that are observed by the fixed, ceiling mounted cameras, while the rest of the map stays unchanged.

VII. Experiments

To evaluate the initial and the permanent localization methods based on the camera and laser scanner data two experiments are performed. In the test setup one vehicle and one fixed camera are used.

A. Evaluation of the Localization Accuracy Based on Camera Data

The aim of the first experiment is to evaluate the accuracy of the trajectory comparison described in chapter IV. Therefore, a webcam is placed at the ceiling of our laboratory facing towards the floor. The resolution of the camera is 320 x 240 pixels and the framerate is 10 fps. Mounted at a height of four meters the camera monitors an area of 4 x 3 m on the floor. The resolution is 1.25 cm/pixel. The vehicle's trajectory is sent via WLAN to the embedded system unit, where the described algorithm calculates the vehicle's pose. In a previous calibration step the transformation between the camera coordinate system and the map coordinate system is determined.

To start localization, one vehicle is placed in the camera monitored area and steered manually. After an successful comparison between the AGV's trajectory and the object's trajectory, the current pose in the map coordinate system gets published and the AGV is forced to stop to measure its current real world pose. This procedure is repeated ten times and each result gets documented.

In each of the ten runs the AGV was localized. Fig. 5 shows the translational error of each measurement as the Euclidian distance in meters between the calculated and measured position. It shows that in nine runs the error was between 0.05 and 0.10 m but in one run the AGV was located in a wrong pose. Thus the error is larger than 2.5 m. As the initial localization uses the maximum amount of particles, which is set in this work to 5000, the particles are spread to a relatively wide area and the algorithm calculates the correct pose within short time. By using unique trajectories for each vehicle the probability of mislocalization could be reduced.

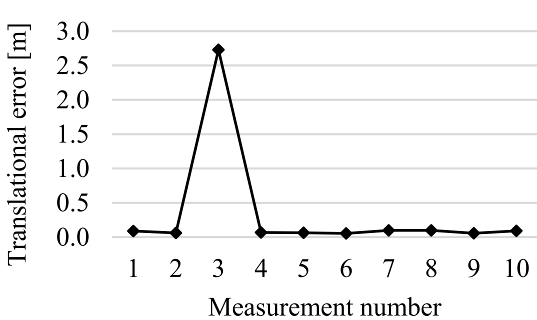

Figure 5: Translational error as Euclidian distance in meters between the manually measured and the calculated position. The pose is calculated using the trajectory comparison method.

B. Evaluation of the Localization Accuracy Based on Laser Scanner Data

In a second experiment the localization via the adaptive Monte Carlo localization is evaluated using the low-cost laser scanner. Therefore, a part of the laboratory is mapped as described above.

Fig. 6 shows the resulting occupancy grid. Afterwards four different poses are defined and their transformations to the map coordinate system are determined. The four poses are also marked in Fig. 6. Pose 0 and 4 are identical and also represent the map coordinate system. By initializing the particle filter, while the robot is placed at pose 0, the initial localization is correct. The AGV gets steered manually to each pose and stopped to document the estimated pose. Subsequently the differences to the real poses can be calculated.

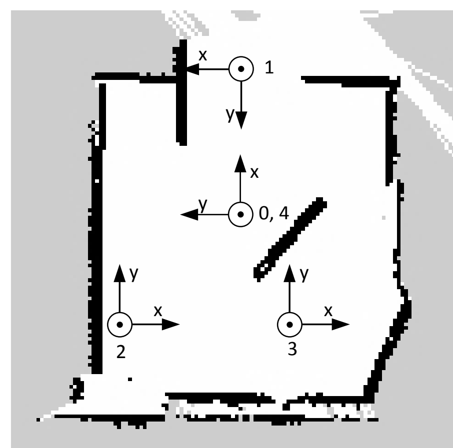

Figure 6: Occupancy grid map, created using the Rao-Blackwellized particle filter with low-cost laser scanner data, and test poses 0 to 4.

Fig. 7 shows the translational error as Euclidian distance between the real position and the estimated position. The result shows that the maximum error in this experiment appears at pose 2 with an Euclidian distance between the estimated and the real pose of 0.17 m. Fig. 8 shows the number of particles needed to represent the state estimation during the AGV's movement. The initial localization is done with the maximum amount of 5000 particles. As the initial localization is correct in this experiment the amount of particles reduces quickly to a value of around 1000 particles. In the case of a wrong initial localization the number of particles wouldn't shrink that quickly because the area, the particles are spread to, would increase to sample particles at the correct current pose.

VIII. Conclusion

In this paper we present a concept to fusion low-cost external camera and internal laser scanner data to navigate

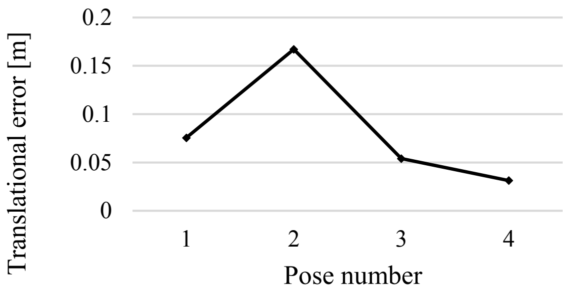

Figure 7: Translational error as Euclidian distance at the different poses shown in Fig. 6.

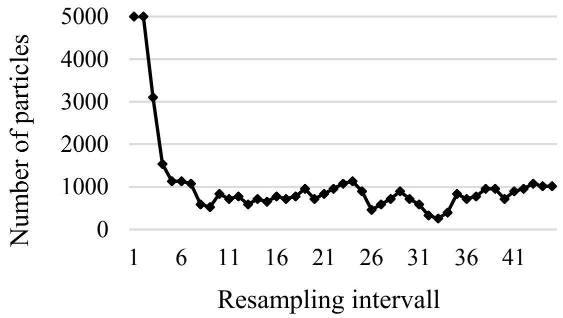

Figure 8: Number of particles during the movement of the AGV from pose 0 to pose 4 in Fig. 6.

AGVs. The current test setup uses one AGV and one ceiling mounted camera. The camera's processed data are used to provide an initial localization and to update the observed part of the global static map. By decentralizing the system setup, using small processing units at each camera, the required data can be directly provided to the corresponding AGVs, depending on their current positions and paths. This allows to reduce the amount of data sent by WLAN and makes the system adaptable to different kinds of usecases.

Depending on the applied scenario the AGVs can be adapted by integrating additional sensors. A WLAN based indoor localization can be applied, to reduce the number of objects that have to be considered in each camera surveillance area and thereby speeding up the correlation process between detected objects and known vehicles by the Hungarian Method. Currently the already mounted range sensors and the depth camera aren't included in the navigation process. Nevertheless, there is already the possibility to integrate them into the dynamic local costmap. To avoid harmful collisions, especially if larger AGVs are used, additional near field or tactile sensors would be necessary.

This bundle of sensors, which is capable of being integrated to localize the AGVs and digitize their workspace, allows us to transform the results of this work to different usecases. On the one hand side, the possibility to use decentralized ceiling cameras with an integrated image processing together with low cost on-board sensors for collision avoidance is a perfect combination for a small area to observe. In this case, a big amount of AGVs transports small workpieces, for example printed circuit boards, in a workshop. Therefore, the stationary sensor concept leads to the best tradeoff regarding the overall sensor cost. On the other hand, it could be possible, that a small amount of vehicles has to operate in a large workspace, for example a whole factory building. From the cost point of view, the best scenario is to equip the vehicles with localization sensors and reduce the number of ceiling cameras to a minimum. However, stationary sensors make sense even in this case, especially at intersections. An industrial field of application for AGVs will be usually between those two scenarios. Therefore, the fusion of low-cost on-board and stationary sensors gives the possibility to adapt the sensor concept to each possible intralogistics scenario.

REFERENCES

[1] Swisslog, Standard Automated Guided Vehicle Systems. Available: http://www.swisslog.com/en/Products/WDS/Automated-Guided-Vehicles/AGV-Standard (2015, May. 10).

[2] Adept Technology Inc, Lynx - Autonomous Indoor Vehicle. Available: http://www.adept.com/products/mobile-robots/mobile-platforms/lynx/general (2015, May. 11).

[3] Vorwerk, Kobold VR200 Saugroboter. Available: http://kobold.vorwerk.de/de/produkte/kobold-vr200-saugroboter/ (2015, May. 11).

[4] S. Zickler, T. Laue, O. Birbach, M. Wongphati, and M. Veloso, "SSL-Vision: The Shared Vision System for the RoboCup Small Size League," in Lecture Notes in Computer Science, RoboCup 2009: Robot Soccer World Cup XIII, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 425–436.

[5] E. Kruse and F. M. Wahl, "Camera-based monitoring system for mobile robot guidance," in Proceedings. 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems. Innovations in Theory, Practice and Applications, pp. 1248–1253.

[6] J. Franke and F. Lütteke, "Versatile autonomous transportation vehicle for highly flexible use in industrial applications," CIRP Annals - Manufacturing Technology, vol. 61, no. 1, pp. 407–410, 2012.

[7] Y.-H. Chen, F. Lütteke, Q.-J. Chen, and J. Franke, "Indoor Moving Vehicle Detection by Trajectory Matching," Journal of Central South University of Technology, vol. 42, pp. 561–566, 2011.

[8] F. Lütteke, X. Zhang, and J. Franke, "Implementation of the Hungarian Method for object tracking on a camera monitored transportation system," in Proceedings for the conference of ROBOTIK 2012, 7th German Conference on Robotics: 21 - 22 May 2012, International Congress Center Munich (ICM) in conjunction with AUTOMATICA, Berlin, Offenbach: VDE-Verlag, 2012, pp. 343–348.

[9] Y.-H. Chen, F. Lütteke, and Q.-J. Chen, "Indoor moving vehicle detection by trajectory matching," in Journal of Central South University of Technology48th ed, Changsha, Hunan: Editorial Office of Journal of Central South University, 2011, pp. 561–566.

[10] G. Grisetti, C. Stachniss, and W. Burgard, "Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters," IEEE Transactions on Robotics, vol. 23, no. 1, pp. 34–46, 2007.

[11] D. Fox, "Adapting the Sample Size in Particle Filters Through KLD-Sampling," The International Journal of Robotics Research, vol. 22, no. 12, pp. 985–1003, 2003.

[12] D. V. Lu, D. Hershberger, and W. D. Smart, "Layered costmaps for context-sensitive navigation," in 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), pp. 709–715.

[13] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in 1991 IEEE International Conference on Robotics and Automation, pp. 1398–1404.

[14] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," IEEE Robot. Automat. Mag, vol. 4, no. 1, pp. 23–33, 1997.