

Improvements for an appearance-based SLAM-Approach for large-scale environments

Alexander Koenig* Jens Kessler* Horst-Michael Gross*

**Neuroinformatics and Cognitive Robotics Lab, Ilmenau University of Technology, Ilmenau, Germany*

Abstract—In continuation of our previous work on visual, appearance-based localization and mapping, we presented in [5] a novel appearance-based, visual SLAM approach. The essential contribution of this work was an adaptive sensor model, which is estimated online, and a graph matching scheme to evaluate the likelihood of a given topological map. Both methods enable the combination of an appearance-based, visual localization and mapping concept with a Rao-Blackwellized Particle Filter (RBPF) as state estimator to a real-world suitable, online SLAM approach. In our system, each RBPF particle incrementally constructs its own graph-based environment model which is labeled with visual appearance features (extracted from panoramic 360° snapshots of the environment) and the estimated poses of the places where the snapshots were captured. The essential advantage of this appearance-based SLAM approach is its low memory and computing-time requirements. Therefore, the algorithm is able to perform in real-time. In this paper we improve our algorithm to deal with dynamic changes in the environment which is typical in real-world environments. Furthermore, we describe a method to limit the memory consumption of the environment model that is needed for large maps. Finally, we present the results of SLAM experiments in a dynamical and large environment that investigates the stability and localization accuracy of this SLAM technique.

Index Terms—Visual SLAM, panoramic vision, appearance-based localization, view-based approach

I. INTRODUCTION

Robust self-localization and map building plays a central role in our long-term research project PERSES (PERsonal SERVICE System) which aims to develop an interactive mobile shopping assistant that can autonomously guide its user within a home improvement store [4]. In everyday life and in mobile robotics, two main types of visual self-localization and mapping methods are typically used: landmark-based methods and appearance- or view-based approaches. While landmark based methods require the extraction and reassignment of distinct visual landmarks, appearance-based methods use a description of the view at a certain point, leading to a more global impression of a scene. Appearance-based approaches compare the appearance of the current view with those of the reference images to estimate the robot's pose [18, 20]. That makes these methods so interesting for robust localization and map building in mobile robotics.

One objective of our ongoing research is to investigate whether appearance-based SLAM approaches are generally suited for large-scale and uniformly structured indoor environments, like the aforementioned home improvement store, and if so, how they can be made capable of working online and in real-time. In our research on vision-based

robot navigation we are preferring these approach for the following reasons: i) In a highly dynamic, populated maze-like environment, a robust recognition of natural landmarks cannot be guaranteed in any case. ii) Furthermore, the need for a robust and invariant detection of visual landmarks often results in high computational costs and, therefore, map building is often performed off-line by these approaches. In our previous approaches [1, 3] on appearance-based Monte Carlo Localization, a static, graph-based model of the environment was developed (see Fig. 1). The nodes (poses of the robot) of this environment model are labeled with appearance-based observations extracted from an omnidirectional image. Based on this environment model, we developed an appearance-based visual SLAM approach [5] that is using the Rao-Blackwellized Particle Filter (RBPF) concept [7] to estimate a map. Montemerlo et. al. A more detailed overview over related work is given in Section II.

The essential contribution of our approach presented in [5] was the combination of the appearance-based, visual localization concept with a Rao-Blackwellized Particle Filter as state estimator to a real-world suitable, online SLAM approach. Instead of single observations, typically used in the field of appearance-based localization and mapping, another key idea of our approach was to utilize local graphs to perform the evaluation step, representing a kind of short-term memory or time window of the current and the most recent observations and pose estimations. Based on this, we introduced a graph-matching technique to compare the local graph of each particle with its particular global graph to determine the best matching map. Another novel idea consisted in online estimation of an environment dependent sensor model. In continuation of this work mentioned above, we developed on a method to further reduce the demand on memory of the environment model. This is a prerequisite for building maps of larger environments which are represented in this paper. Furthermore, we are introducing a method to deal with dynamically changing environments. In real-world scenarios changes occur frequently (e.g. by moving objects, changing light conditions, walking people). To deal with these changes, most algorithms try to make the image comparison as robust as possible. Our way to handle this problem is to consider dynamic changes of appearance and model these changes in the map directly. The necessary extension of the environment model will be explained in detail later.

The article is structured as follows: The next section gives an overview of related work. After that, the important methods

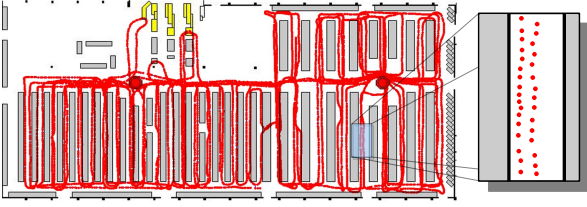


Fig. 1. A map of the operation area, a regularly structured, maze-like home improvement store with a size of $100 \times 50 m^2$ (taken from [3]). The red dots show the positions of stored reference observations.

of our appearance-based SLAM approach described in [5] are recapitulated. This is needed for a better understanding of the extensions of our algorithm to deal with dynamic changes which are introduced in the section following. After that, experimental results achieved with our SLAM approach in a public large environment are presented and discussed. A conclusion and suggestions for future work are given at the end of the paper.

II. RELATED WORK

Many solutions have been presented in the past to realize a robot self-localization in more or less complex environments including methods based on feature or landmark extraction and tracking and those based on appearance models of the environment. A short overview of the most relevant research is given in the following paragraphs:

Feature/Landmark-based approaches: In many SLAM approaches, the map representation is assumed to be a vector of point-like feature positions (landmarks) [10]. The attractiveness of feature/landmark-based representations for SLAM lies in their compactness. However, they rely on *a priori* knowledge about the structure of the environment to identify and distinguish potential features or landmarks. Furthermore, a data association problem arises from the need to robustly recognize the landmarks not only in local vicinities, but also when returning to a position from an extended round-trip. In the field of visual landmark-based SLAM algorithms, Lowe's SIFT-approach [11, 12] has often been used so far. Further important feature/landmark-based approaches are those by Davison using Stereo vision [13] or monocular vision [14]. To estimate the landmark positions, popular methods like the Extended Kalmanfilter (EKF) [14], Rao-Blackwellized Particle Filters (RBPF) [15] (e.g. FastSLAM [16]) are applied. Using such complex visual features often results in high computational costs and, therefore, map building is often performed off-line by these approaches. **Appearance-based SLAM/CML approaches:** The Concurrent Map-building and Localization (CML) approach of Porta and Kroese proposed in [22] was one of the first techniques to simultaneously build an appearance-map of the environment and to use this map, still under construction, to improve the localization of the robot. Another way to solve the SLAM-problem was proposed by Andreasson et. al. [23]. Here, a topological map stores nodes with appearance-based features and edges which contain relations between nodes and their poses. Essential drawbacks of this approach are, however, the required offline

relaxation phase and the computational costs for calculation of the SIFT features on high resolution images. To avoid these requirements, our approach uses an RBPF [7] to avoid off-line relaxation methods. Moreover, the method to estimate the pose difference between images applying the image similarity introduced by Andreasson [23] has been picked up and extended in our SLAM approach. Further approaches that use a topological map representation are described in [25], where a Bayesian inference scheme is used for map building, and in [26], where a fast image collection database is combined with topological maps that allows an online mapping, too.

III. APPEARANCE-BASED SLAM APPROACH WITH RBPF

In this section, the basic idea of our algorithm presented in [5] is explained briefly. The graph matching process to determine the likelihood of the map to be correct will be described more precisely. Furthermore, the adaptive sensor model is investigated and discussed. With that background subsection the section IV introduces an extension to model dynamic changes in the algorithm.

A. RBPF with local and global graph models

Our appearance-based SLAM approach also utilizes the standard Rao-Blackwellized Particle Filter approach to solve the SLAM problem, where each particle contains a pose estimate \mathbf{x}_i (position x, y and heading direction φ) as well as a map estimate (see Fig. 2). The environment model (map) used in our appearance-based approach is a graph representation, where each node i , representing a place in the environment, is labeled with appearance features of one or more omnidirectional impressions at that node. The observation \mathbf{z}_i is extracted from the panoramic view captured at that place and the estimated pose \mathbf{x}_i . To solve the SLAM problem, the RBPF has to determine the likelihood of the graph-based maps in the particles to be correct. Hence, the likelihood of a given map \mathbf{m} has to be calculated by comparing the current observation \mathbf{z} and the map \mathbf{m} itself. Therefore, our approach uses two different types of maps: a *global map* $\mathbf{m}^G = \langle \mathbf{x}_{1:(l-1)}, \mathbf{z}_{1:(l-1)} \rangle$, which represents the already known environment model learned so far and a *local map* $\mathbf{m}^L = \langle \mathbf{x}_{l:t}, \mathbf{z}_{l:t} \rangle$ representing the current and the n latest observations and the local path between them (e.g. the last two meters of the robot's trajectory [2]). The global map stores all observations before the time-step $l = t - n$. Thus, in extension of known RBPF techniques, in our approach each particle (see Fig. 2) estimates and stores a local map, a global map, and the currently estimated pose. Figure 2 schematically shows the local and global map in more detail. Serving as a short-term time-window of observations, the local map is used to compute the likelihood of each global map to be correct. This has some advantages. First, the likelihood of a given global map can be evaluated in a simple way by comparing the local and the global map directly. Based on this comparison, the RBPF can determine for each particle the likelihood of the trajectory and the global map estimated by that particle. Second, the still more relevant advantage is, that the local map provides geometric and visual information about

the most recently observed places. So correct comparisons can be made by taking spatial relations *and* visual observations into consideration. Finally, this approach is more robust against single disturbances in images (e.g. people walking by) because not a single observation determines the likelihood of the map but the n latest observations and their geometric relations. That is a big difference to other approaches where a single observation is used to detect a loop closure.

The approach assumes that a short part of the robot's trajectory is measured with sufficient accuracy. Hence, the length of the local map has to be as short as possible. On the other hand, for reliable comparison results, the graph of a local map has to contain enough nodes for the matching process. We achieved the best results with a local trajectory length of about 5 meters and average distances between the nodes of 0.5 meters. Note that these values are depending on the accuracy of the robot's odometry sensors. The chosen length of the map corresponds to the distinctiveness of the given appearance description \mathbf{z} . While using unspecific descriptions, the length of the local map has to be larger than on distinctive appearance descriptions. In the first case, distinctiveness has to be achieved by a greater number of observations. In the shown results we use a global scene description based on a set of SIFT features and HSV features. In our experience the accuracy of the map depends more on the used appearance description \mathbf{z} with its distinctiveness than on the size of the local map. In our experiments we found values of the length of the local map between 4 and 10 meters useful as well as distances between the nodes of 0.2 to 0.5 meters.

B. Graph matching

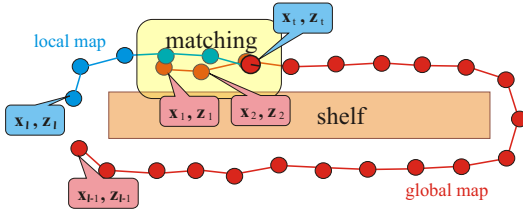


Fig. 2. Graph-based environment representation of our appearance-based approach: The red nodes show the global map of a single particle with respect to the path estimated by this particle. The blue nodes represent the local map, whose data creates a short-term time-window of observations (including the current pose and the n latest observations) used for map matching to determine the likelihood of the respective global map. The idea of our appearance-based RBPF is, that only particles with correctly estimated trajectories are able to build correct maps, so that the matching between local and global map provides a higher matching value than wrongly estimated trajectories.

To determine the likelihood of the estimated map our approach requires loops in robot's trajectory, like other RBPF-based approaches too. Instead of calculating the probability distribution $p(\mathbf{z}|\mathbf{x}, \mathbf{m})$ directly, here the likelihood of a given map to be correct is estimated by comparing the local and the global map. In the context of RBPF, this distribution determines directly the importance weight $w = p(\mathbf{z}|\mathbf{x}, \mathbf{m})$ of a particle. For this purpose, corresponding pairs of nodes in both maps are selected by a simple nearest neighbor search in the position space. The relation between each selected pair of corresponding nodes e_i^L (of the local map \mathbf{m}^L) and e_j^G (from

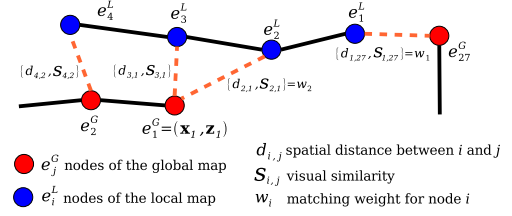


Fig. 3. Basic idea of our map matching algorithm: the likelihood of a given global map (particle-specific) is determined by comparison of the spatial distances d_{ij} and visual similarities S_{ij} between each pair of nodes i (in the local graph) and j (in the global graph). Corresponding nodes e_i^L and e_j^G are defined by the minimum spatial distances d_{ij} . The matching weights w_i per node in the local map are calculated in dependence of the spatial distance d_{ij} and the visual similarity S_{ij} (for more details see Section III-C).

the global map \mathbf{m}^G) provides two pieces of information, a geometric one (spatial distance d_{ij}) and a visual one (visual similarity S_{ij}), depicted in Fig. 3. Both aspects of each relation ij are used to determine a matching weight w_i for the respective node i of the local map. Assuming an independence between the node weights of the local map, the total matching weight $w^{[k]}$ between the local and the global graph of particle k is simply calculated as follows:

$$w^{[k]} = \prod_{i=1}^n w_i^{[k]} \quad (1)$$

with n describing the number of nodes in the local map.

To evaluate the matching weight w_i we have to compute the probability that two observations \mathbf{z}_i and \mathbf{z}_j show a similarity S_{ij} at a given distance d_{ij} . To solve this problem, we use an adaptive sensor model, which is described in the next section. The method described up to now can be summarized as follows:

Estimating correct maps with RBPF approaches requires loops in the trajectory. When the robot closes a loop, the local and global maps overlap and the graph-matching algorithm can estimate the likelihood of a map. In the case of non-overlapping maps no weights are determined by graph-matching, but set to the a priori weight computed by the average weight of all particles with overlapping maps. This prevents particle depletion before a loop is closed. If there are no particles with overlapping maps (e.g. at the start) the uniform distribution is used for particle weighting. Note, that in the context of RBPF no explicit loop closing detection is performed and, therefore, no explicit trajectory (or map) correction can be done.

C. Adaptive sensor model

To compute the matching weights between corresponding nodes, an adaptive sensor model has been developed. In the context of appearance-based observations, the visual similarity between observations is not only depending on the difference in position but also on the environment itself. If the robot, for example, moves in a spacious environment with much free-space, the similarity between observations from slightly different positions will be very high. In a narrow environment with many obstacles, however, observations at positions with low spatial distance are already drastically influenced, which leads to a low visual similarity. Therefore, our sensor model

estimates online the dependency between surrounding-specific visual similarities \hat{S}_{ij} of the observations \mathbf{z}_i and \mathbf{z}_j and their spatial distance \hat{d}_{ij} . An example for this dependency is shown in Fig. 4. The samples (black dots) to build such a model are taken from the nodes of the *local map* where each node is compared to each other. This results in $n^2/2$ pairs of \hat{S}_{ij} and \hat{d}_{ij} representing samples describing the appearance variability in the local environment. In [5] different non-parametric and parametric approaches to approximate the model were investigated, e.g. the Gaussian Process Regression (GPR) of Rasmussen [27], while we used a parametric polynomial description of the sensor model and its variance. In this paper, our approach uses an non-parametric histogram-based sensor model. For that model the distance d is divided in bins of different size and for each bin the mean similarity and its variance are computed (see Fig. 4). The bin size s is computed by a parametric exponential function $s(d) = a \cdot (d+b)^c - 1$. The non-linear size of the bins is required to get a higher resolution for small distances. That results in a higher accuracy of the matching process. This model works like the model presented in [5] but has the advantage that no least square optimization needs to be done. The likelihood that two nodes i and j of particle k are matching is computed as follows:

$$w_i^{[k]} = p(S_{ij}|d_{ij}) \approx \exp - \frac{(S_{ij} - \hat{S}(d_{ij}))^2}{\hat{\sigma}(d_{ij})^2} \quad (2)$$

With that adaptive sensor model and the aforementioned graph-matching algorithm, the importance weight of each particle (the likelihood of its map) can be determined. As mentioned in [5] a variety of image features can be used in our approach. For the reasons discussed in Section I, we prefer an appearance-based approach utilizing holistic appearance features. These features are extracted from panoramic snapshots obtained from the omnidirectional camera located on the top of our experimental robot platforms. The environment observation for a given position of the robot is described by one feature vector \mathbf{z} of the 360° view. Possible feature vectors are gradient histograms, color-histograms, SIFT feature sets, or simple RGB mean values. For the experiments in this paper, we chose SIFT feature sets [11, 23] as appearance-based image description, because of their ability to determine the similarity of observation with high accuracy.

IV. DYNAMIC CHANGES

To suit the problems of the dynamic environments we face two main problems. At first, the visual impression at the same place can change due to varying lighting conditions, occlusions and moving objects. Second, the map grows infinitely while observing the environment and including every estimated position and observation. So we have to select which observations or new position needs to be included in the map. Therefore we extend the nodes to collect more than one observation. The problem of including new nodes is solved strait forward. We simply insert new nodes, when a distance d_{min} to the nearest node is exceeded. In each new node the actual observation is also included. The problem of including new observations

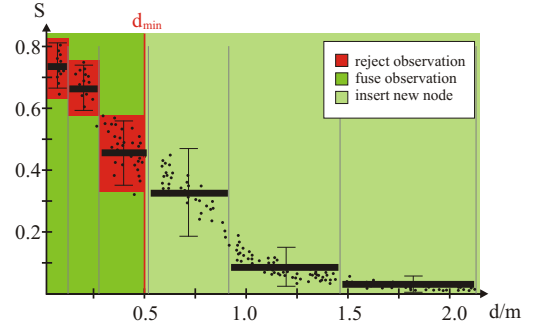


Fig. 4. The histogram model with mean value and σ per bin. Points in the light green area (with distance to the nearest neighbor $> d_{min}$) are always included as new nodes. All other points are tested by the sensor model. If they are within the model expectation they need not be included (red area), else they extend the local model by adding a new appearance variant to the node.



Fig. 5. Results of the node-fusion process. While in a) all observations and places are stored in b) they are fused together. It is clear that version b) requires less memory than a).

in existing nodes is slightly more complicated. We use the extended nodes to collect observations of different appearance states by taking the similarity model $\hat{S}(d_{ij}), \hat{\sigma}(d_{ij})$ to decide to include new observations to an existing node or not. For each particle, we assume to be in a certain position \mathbf{x} with a certain map \mathbf{m} . We compare (like it is done in the weight calculation) the similarity S and distance d from position \mathbf{x} to the nearest neighbor in the map \mathbf{m} . Since more than one appearance description can reside inside a node, we look for the best matching similarity ($S_{max} = \max_i(S_i)$). According to equation (2) we now calculate the probability $p(S_{max}|d)$. If this probability is above a certain threshold ξ , the observation matches the expected similarity (we have seen the observation in this configuration before) and can be ignored. If the observation does not match the model, it is included to the current node at \mathbf{x} .

Now we have a mechanism to include unknown variations of the environment regarding the appearance. Because of the maximum selection on similarities occupied by the node, we are sure to find any known appearances when coming to those nodes again. The sensor model does not have to be adjusted, since we do not care inside this model from which observation the similarity comes, and so the model represents a function of the distance to the *best possible* similarity.

V. EXPERIMENTS AND RESULTS

To investigate the extensions of our approach, we used our default test environment [3], the home improvement store already introduced in Section I and shown in Fig. 7, where many



Fig. 6. A few examples of dynamic aspects of our recorded map. In most cases customers represent the dynamic changes in our mapping experiment.

dynamic effects occur. All data for the analysis were recorded under realistic conditions, i.e. people walked through the operation area (see fig. 6), shelves were rearranged, illumination changes, and occlusion. Additionally, laser data were captured to generate a reference to evaluate the results of our approach (details, see below). Note, that the laser scanner in all cases was used for comparison and visualization only but not for mapping or localization.

The main challenge of the environment are a) the big loop of approx. 340m the approach has to close correctly and b) a lot of hallways with very similar appearances. The robot was moved several times through the home store along a path of about 2400 meters length. The problem here is, if the robot travels in unmapped areas the particles should not match to already mapped hallways left or right of the correct trajectory. For a visual SLAM approach this is really challenging, especially if the hallways are very similar. The resulting graph (Fig. 7, red line) covers an area of 120 x 50 meters and was generated by a mean count of 250 particles (max. 2000) in the RBPF. We use a varying count of particles to approximate the probability density. Thus, max. 2000 global graph maps had to be built, whereas Fig. 7 only shows the most likely final trajectory and a superimposed occupancy map for visualization. To evaluate the visually estimated path shown as red trajectory in Fig. 7, in addition a reference path and map built by means of a Laser-SLAM algorithm were calculated (GMapping of G. Grisetti et al. [6] taken from www.openslam.org). The Laser-SLAM estimated path (ground truth) is used to determine the mean and the variance of the position error of our algorithm. A first result was, that the trajectories estimated by both SLAM approaches are very similar. This is also expressed by a mean localization error of 0.47 meters (with a variance $\sigma \approx 0.24m$) compared to the laser-based SLAM results. The maximum position error in this experiment was about 1.21 meters. These experimental results demonstrate, that our approach is able to create a consistent trajectory and based on this, a consistent graph representation, too. Furthermore, in contrast to grid map approaches (up to 4GB for our test environment), topological maps require less memory (up to 1.5GB for 4000 obs.) because of the efficient observation storage, where each observation's features of each snapshot are linked to all global maps. That means the memory requirements are nearly independent from the used number of particles. The particles require memory to store their own trajectories only.

TABLE I
OVERVIEW OF THE ACHIEVED RESULTS IN ALL EXPERIMENTS AND ENVIRONMENTS.

Experiment	Home store
Size of area	50x120m
Total path length	2400m
# of particles	250 (mean) 2000 (max)
Error Mean/Var/Max	0.47/0.24/1.21 m
Time per map update	0.250 s
Building sensor model	0.300 s

To quantify the dynamic aspects of our operation area is quite hard. We can't measure the quality amount of dynamic changes occurring in the environment. The only value we can count are the numbers of multiple observation entries into each map node. The experiments show, that a lot of nodes store more than one observation. The other improvement of fusing nodes during the mapping process can be shown easier. With fusing of nodes the map contains ≈ 2000 nodes, compared to ≈ 4600 without. A final overview of the achieved results and the computational costs of our proposed approach is given in Table I. It becomes evident, that in such environments, like the home improvement store, where the robot has to close large loops, the approach is working correctly and with a high accuracy for a visual SLAM technique.

The computational and the memory costs of the algorithm depend linearly on the number of particles and quadratically on the number of nodes in the local maps required for graph matching. At a single-core CPU with 2.4GHz, the computation of the adaptive sensor model is done in approximately 300 ms (independent from map size and particle count), whereby 10 nodes of the local map were used for the estimation of the similarity model. The rest of the computational costs (see Table I) is spent for map updates, weights determination, and the resampling process. For small numbers of particles, our approach is running in real-time. With higher numbers of particles, the SLAM algorithm nearly works in real-time. In this case, the robot only has to be moved a bit slower through the environment (motion speed is limited).

VI. CONCLUSION AND FUTURE WORK

We presented some extensions of our appearance-based SLAM approach to limit the memory consumption and to deal with dynamic changes that occur in typical real-world environments. These improvements allow an appearance-based on-line SLAM in large-scale and dynamic real-world environments. The key ideas of our approach are the use of global and local graph models per particle, the introduction of an adaptive sensor model, and the sophisticated graph-matching technique to compare the local graph of the particle with the respective global graph to determine the best matching map, and with that the best particles for the resampling step. The essential advantages of our appearance-based SLAM approach are its low memory and computing-time requirements. Therefore, our algorithm is able to perform in real-time, which is a prerequisite for all on-line working map builders or mapping assistants.

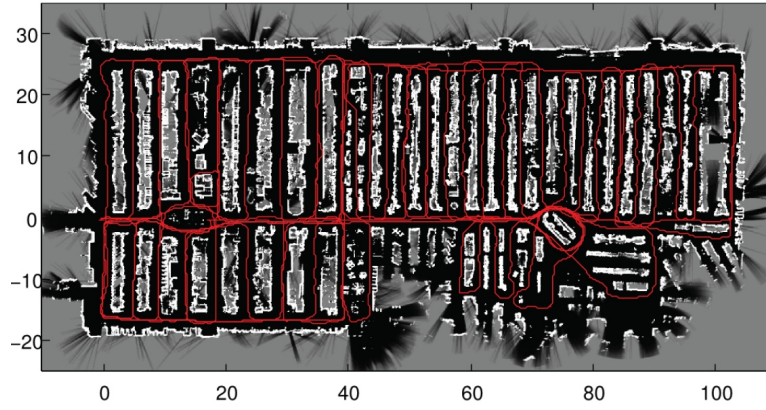


Fig. 7. In the home improvement store: The red path shows the robot's movement trajectory only estimated by our appearance-based SLAM approach. The map shows a high accuracy with only small alignment errors. For visualization of the localization accuracy, a laser-based occupancy map was built in parallel and superimposed to the estimated movement trajectory (visual SLAM).

For the near future we plan to implement a visual scan-matching technique to limit the number of required particles to close loops correctly and to increase the accuracy of estimated maps. This technique is used by many other successful SLAM approaches (e.g. [6]). Furthermore, we want to investigate the influence of dynamic changes in the environment (changed filling of the goods shelves, re-arrangements in the hallways, occlusions by people, changing illumination) on the robustness and long-term stability of our approach.

REFERENCES

- [1] H.-M. Gross, A. Koenig, H.-J. Boehme, and Chr. Schroeter. Vision-Based Monte Carlo Self-localization for a Mobile Service Robot Acting as Shopping Assistant in a Home Store. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Systems (IROS'02)*, 2002, pp. 256–262.
- [2] Chr. Schroeter, H.-J. Boehme, and H.-M. Gross. Memory-Efficient Gridmaps in Rao-Blackwellized Particle Filters for SLAM using Sonar Range Sensors. In *Proc. of the 3rd European Conference on Mobile Robots (ECMR'07)*, 2007, pp. 138–143.
- [3] H.-M. Gross, A. Koenig, Chr. Schroeter and H.-J. Boehme. Omnivision-based Probabilistic Self-localization for a Mobile Shopping Assistant Continued. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'03)*, 2003, pp. 1505–1511.
- [4] H.-M. Gross and H.-J. Boehme. PERSES - a Vision-based Interactive Mobile Shopping Assistant. In *Proc. IEEE Int. Conf. on Systems, Man and Cybernetics (SMC'00)*, 2000, pp. 80–85.
- [5] A. Koenig, J. Kessler, H.-M. Gross A Graph Matching Technique for an Appearance-based, visual SLAM-Approach using Rao-Blackwellized Particle Filters. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'08)*, 2008, pp. 1576–1581.
- [6] G. Grisetti, C. Stachniss, W. Burgard Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters. In *IEEE Transactions on Robotics*, 2006, pp. 34–46.
- [7] K. P. Murphy. Bayesian map learning in dynamic environments. In *Advances in Neural Information Processing Systems 12 (NIPS'99)*, 1999, pp. 1015–1021.
- [8] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proc. of the AAAI Nat. Conf. on Artificial Intelligence*, 2002, pp. 593–598.
- [9] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. Fastslam 2.0: An improved particle filtering for simultaneous localization and mapping that provably converges. In *Proc. of Int. Joint Conf. on Artificial Intelligence (IJCAI'03)*, 2003, pp. 1151–1156.
- [10] R. Smith, M. Self, and P. Cheeseman. A stochastic map for uncertain spatial relationships. In *Robotics Research, 4th Int. Symposium*, MIT Press, 1988, pp. 467–474.
- [11] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. Int. Conf. on Computer Vision ICCV'99*, 1999, pp. 1150–1157.
- [12] S. Se, D. Lowe, and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, vol. 21, no. 8, pp. 735–758, 2002.
- [13] A. Davison and D. Murray. Simultaneous localization and map-building using active vision. *IEEE Trans. on PAMI*, vol. 24, no. 7, pp. 865–880, 2002.
- [14] A. Davison. Real-time simultaneous localisation and mapping with a single camera. In *Proc. Int. Conf. on Computer Vision (ICCV'03)*, 2003, pp. 1403–1410.
- [15] P. Elinas, R. Sim, and J. J. Little. σ SLAM: Stereo vision SLAM using the Rao-Blackwellised particle filter and a novel mixture proposal distribution. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA'06)*, 2006, pp. 1564–1570.
- [16] T D. Barfoot. Online visual motion estimation using fastslam with sift features. In *Proc. of 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and System (IROS'05)*, 2005, pp. 579–585.
- [17] F. Pourraz and J.L. Crowley. Continuity properties of the appearance manifold for mobile robot position estimation. In *Proc. Symp. Intelligent Robotics Systems (SIRS)*, 1998, pp. 251–260.
- [18] I. Ulrich and I. Nourbakhsh. Appearance-based Place Recognition for Topological Localization. In *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'00)*, 2000, pp. 1023–1029.
- [19] P. Blaer and P. Allen. Topological mobile robot localization using fast vision techniques. In *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'02)*, 2002, pp. 1031–1036.
- [20] E. Menegatti, M. Zoccarato, E. Pagello, and H. Ishiguro. Hierarchical Image-based Localisation for Mobile Robots with Monte-Carlo Localisation. in: *Proc. 1st European Conf. on Mobile Robots (ECMR'03)*, 2003, pp. 13–20.
- [21] F. Dellaert, W. Burgard, D. Fox, and S. Thrun. Using the Condensation Algorithm for Robust Vision-based Mobile Robot Localization. In: *Proc. IEEE Int. Conf. on Comp. Vision and Pattern Recogn.*, 1999.
- [22] J. M. Porta and B. J.A. Kroese. Appearance-based Concurrent Map Building and Localization using a Multi-Hypotheses Tracker. in: *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'04)*, 2004, pp. 3424–3429.
- [23] H. Andersson, T. Duckett, and A. Lilienthal. Mini-slam: Minimalistic visual slam in large-scale environments based on a new interpretation of image similarity. In *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'07)*, 2007, pp. 4096–4101.
- [24] D. Haehnel, W. Burgard, D. Fox, and S. Thrun. An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'03)*, 2003, pp. 206–211.
- [25] A. Ranganathan, E. Menegatti, F. Dellaert. Bayesian inference in the space of topological maps. In *Robotics, IEEE Transactions on*, vol. 22, no. 1, pp. 92–107, Feb. 2006.
- [26] F. Fraundorfer, C. Engels, D. Nister. Topological mapping, localization and navigation using image collections. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'07)*, 2007, pp. 3872–3877.
- [27] C. E. Rasmussen, C. Williams. Gaussian Processes for Machine Learning, the MIT Press, 2006.