

Stereo vision SLAM: Near real-time learning of 3D point-landmark and 2D occupancy-grid maps using particle filters.

Pantelis Elinas and James J. Little
University of British Columbia
Laboratory for Computational Intelligence
Vancouver, BC, Canada
{elinas,little}@cs.ubc.ca

Abstract—This paper summarizes our past work on solving the visual Simultaneous Localization and Mapping Problem (SLAM). We focus on robots equipped with stereo vision and develop a SLAM solution based on the theory of the Rao-Blackwellised particle filter (RBPF). We refer to our method as σ SLAM. We construct maps of 3D point-landmarks identified by their visual appearance. Specifically, we use the Scale Invariant Feature Transform (SIFT) for distinguishing among the thousands of landmarks present in the robot’s environment. Our particle filter approach utilizes a mixture proposal distribution and accurately tracks the position of the robot and the landmarks over long trajectories. In addition, we do not depend on robot odometry in the particle filter’s proposal mechanism; instead, we develop a proposal distribution that relies on visual odometry easing the transition towards a SLAM solution for robots performing 3D motion. Most importantly, σ SLAM works in near real-time making it applicable for use with mobile robots today. Moreover, we show that we can construct 2D occupancy-grid maps from stereo vision; these are useful for path planning and obstacle avoidance. Finally, we validate our approach experimentally mapping large indoor environments processing thousands of images.

I. INTRODUCTION

This paper considers the problem of Simultaneous Localization and Mapping (SLAM) using stereo vision. In robotics, SLAM is the problem of estimating concurrently the robot’s position and a map of its surrounding environment. This is an essential skill for a mobile robot but to this day it has eluded solution because noisy robot dynamics and sensors make solving SLAM a difficult task.

We consider the case of visual SLAM for mobile robots equipped with stereo vision. Our solution depends on the identification of 3D landmarks using their appearance in images and more specifically the Scale Invariant Feature Transform (SIFT) [1]. We employ a Bayesian framework based on the Rao-Blackwellised Particle Filter (RBPF) for integrating measurements and control information over time. In addition, our method uses a stream of video images as the only input and yet it achieves results whose accuracy is comparable to other methods that depend on multiple input modalities. We derive an estimate of the robot’s motion from sparse visual measurements using stereo vision and multiple view geometry techniques [2] known in robotics as visual odometry [3], [4], [5]. The maps we learn for SLAM are composed of 3D point-landmarks making them

unsuitable for path planning. Consequently, we fuse over time disparity images computed using correlation-based stereo to construct 2D occupancy-grid maps. We use known mapping methods [6] to compute these separately from our SLAM algorithm given that we have an accurate estimate of the robot’s position.

A large number of SLAM solutions depend on the Extended Kalman Filter (EKF) [7] for the probabilistic fusion of landmark measurements and robot actions. The EKF assumes that the joint probability distribution over robot poses and maps can be adequately represented using a single Gaussian distribution. This assumption is often violated due to the perceptual aliasing problem common in robotics. In addition, the EKF scales quadratically with the number of landmarks in the learned map and handles poorly incorrect data associations. We focus on using the RBPF because it can model arbitrary distributions and handle incorrect data associations more gracefully.

Our SLAM algorithm, σ SLAM [8], differs from other popular approaches both in the type of maps we learn and the formulation of the essential distributions used in the particle filter. Popular SLAM methods learn maps from laser or other time-of-flight sensor measurements [9], [10], [11]. We learn maps of 3D point-landmarks whose location is estimated using correlation-based stereo and identification is performed using their appearance in images. Specifically, we utilize the Scale Invariant Feature Transform (SIFT) that was originally developed for object recognition and whose invariance to changes in image translation, scaling and rotation is suitable for reliable, but not perfect, data association. In this respect, our work is similar to that of Se et al. [12] but differs because we use a particle filter instead of the Kalman filter and also do not require mechanical odometry as input to our algorithm.

The use of the RBPF in SLAM was first suggested by Murphy [13] who used it to learn occupancy grid maps. Later, it was popularized by Montemerlo et al. [9] who developed the FastSLAM algorithm for mapping using a laser sensor. Barfoot [14] extended FastSLAM for use with a stereo camera but only published results using a single particle and for short trajectories. Other visual SLAM methods operate using a single camera either using the EKF [15] or RBPF [16].

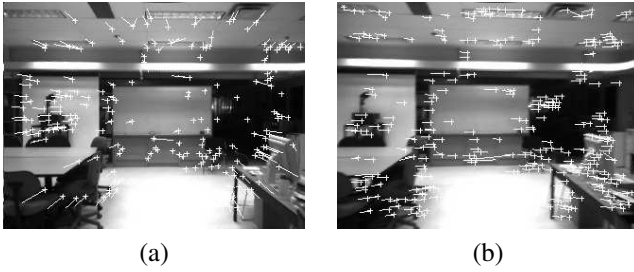


Fig. 1. Examples of point correspondences between two consecutive image frames obtained by comparing SIFT vectors for (a) forward motion and (b) a rotation to the right. Crosses mark the points at time $t - 1$ and lines point to their location at time t .

The rest of this paper is structured as follows. First, we briefly describe our notation and RBPF-based SLAM. Next, we present our visual odometry algorithm and mixture proposal distribution, essential component of the RBPF. We present results using σ SLAM to map a large, office-like environment and outdoors using data provided by SRI International. We conclude with suggestions for future work on visual SLAM extending our algorithm for mapping larger environments with higher accuracy.

II. SLAM USING THE RAO-BLACKWELLISED PARTICLE FILTER

Formally, let s_t denote the robot's pose at time t , m_t the map learned thus far and $x_t = \{s_t, m_t\}$ be the complete state. Also, let u_t denote a control signal or a measurement of the robot's motion from time $t - 1$ to time t and z_t be the current observation. The set of observations and controls from time 0 to t are denoted by z^t and u^t respectively.

SLAM is the problem of estimating the joint distribution over the complete state x_t at time t . Because the state includes all the landmarks in the map which could be a very large number, the standard particle filter (PF) requires a large number of particles to adequately approximate the joint posterior distribution. The method of Rao-Blackwellization can be used to reduce the complexity of the PF by factoring the posterior [13], [17]

$$p(s^t, m_t | z^t, u^t) = p(s^t | z^t, u^t) \prod_k p(m_t(k) | s^t, z^t, u^t) \quad (1)$$

where $m_t(k)$ denotes the k -th landmark in the map. We use a standard PF to estimate $p(s^t | z^t, u^t)$ and an EKF for each of the k landmarks. Notice also, that this new SLAM posterior is over robot trajectories as described in [9].

III. ESTIMATING CAMERA MOTION

We said earlier that one of the main differences between σ SLAM and other algorithms is its lack of dependence on mechanical odometry measurements or other sensors in addition to the stereo camera. We use structure from motion techniques to estimate camera motion directly from the stream of stereo images.

Given correspondences between 3D points in two consecutive image frames, we use the non-linear least squares

Odometry			Vision-based Estimate					
x	y	θ	x	y	z	α	θ	β
0.00	0.00	-1.76	0.01	0.00	-0.01	0.22	-1.83	0.05
0.00	0.00	-2.11	-0.01	-0.00	0.01	-0.01	-1.97	-0.02
0.00	0.00	2.11	0.01	0.02	0.00	0.01	1.63	0.00
0.12	0.00	0.00	0.18	-0.02	0.01	-0.01	-0.13	0.18
0.11	0.01	0.00	0.09	0.01	0.01	-0.26	-0.52	0.04

TABLE I

COMPARING OUR LEAST-SQUARES ESTIMATE OF CAMERA MOTION WITH ODOMETRY. POSITION IS SHOWN IN *cm* AND ORIENTATION IS DEGREES.

Levenberg-Marquardt algorithm to estimate the camera motion [18], [19]. Our formulation focuses on minimizing the re-projection error of the 3D coordinates of matched landmarks. We establish correspondences between landmarks by comparing their SIFT keys. We use tight thresholds for similarity in order to minimize the number of bad data associations and improve the accuracy of our visual odometry estimates.

Figure 1 shows an example of the point correspondences between two consecutive frames for two different robot motions: moving forward and rotating to the right. Table I compares our vision-based estimate of camera motion with that of the robot's mechanical odometry; the data shows that visual odometry is an accurate estimate of the camera's motion.

IV. OBSERVATION FUNCTION

In order to implement the particle filter, we must first specify the distribution $p(z_t | s_t, m_t)$ that is used to compute the importance weights.

A pose of the camera, s_t , defines a transformation $[R, T]_{s_t}$ from the camera to the global coordinate frame. Specifically, R is a 3×3 rotation matrix and T is a 3×1 translation vector. Each landmark in the current view with 3D coordinates, P , can be transformed to global coordinates using the well known equation

$$P_{o_j}^G = R_{s_t} P_{o_j} + T_{s_t} \quad (2)$$

where G denotes the global coordinate frame and o_j is the j th landmark correspondence between the map and most recent image frame. Using equation 2 and the Mahalanobis distance metric (in order to take into account the map's uncertainty), we can define the observation density by:

$$p(z_t | s_t, m_t) = e^{-0.5 \sum_{b=1}^k (P_{o_j}^{G,b} - P_i^{G,b})^T C^{-1} (P_{o_j}^{G,b} - P_i^{G,b})} \quad (3)$$

where C is given by:

$$C = R_{s_t} C_{o_j} R_{s_t}^T + C_i \quad (4)$$

V. THE PROPOSAL DISTRIBUTION

The correct performance of a particle filter highly depends on the selection of proposal distribution. Most SLAM approaches using the RBPF depend on a proposal that is derived from the motion model using the observations to offset the difference between the proposal and target distribution.

The FastSLAM 2.0 [20] algorithm employs a proposal that also takes into account the most recent observation via the linearization of the motion and observation models allowing them to compute the proposal in closed form. In contrast, [21] uses a mixture proposal that combines hypotheses from the motion and observation models weighted according to either the motion or observation models (such a mixture proposal has been previously used for global localization and tracking [22], [23]). However, computing the particle weights according to two different models generates an inconsistent set of hypotheses that in [21] is not treated in a principled way.

We also use a mixture proposal generating hypotheses from both the motion model and a second distribution that depends on the latest observation and learned map thus far. We compute the weights correctly avoiding the problem described in [21].

Formally, our proposal distribution is given by:

$$q(s_t | s_{t-1}, z_t, u_t) = \phi q_{global}(s_t | z_t, m_{t-1}) + (1 - \phi) q_m(s_t | s_{t-1}, u_t) \quad (5)$$

where ϕ is the mixing ratio. Sampling from the motion model, $p(s_t | s_{t-1}, u_t)$, is straightforward as all particles from time $t - 1$ are updated using our estimate of the camera's motion, u_t , with noise added drawn from our confidence on this estimate given by $(J^T J)^{-1}$ with J representing the Jacobian matrix. In order to sample from q_{global} , we generate it by taking advantage of the 3D geometry of our map and camera configuration.

Given correspondences of 3D points between those in the map and the most recent stereo image pair, we can estimate the camera pose using weighted-squares similar to estimating camera motion between two consecutive frames described in Section III. This method is sensitive to outliers that are present in our data association. To overcome this problem, we employ a RANSAC-type approach where we select subsets of the point correspondences and compute a candidate pose for each. We have found that generating as few as 200 candidate poses, $s_{t_1 \dots t_{200}}$, is sufficient for good localization given our noisy observations. For computational efficiency, we only compute the candidate poses with respect to the map of the most likely particle at time $t - 1$.

In order to sample from q_{global} , we evaluate, using Equation 3, the probability of our latest observation given each candidate pose. We then fit a Gaussian distribution to these data points such that:

$$q_{global} = N_{global}(\mu, \Sigma)_{s_{t_1 \dots t_{200}}} \quad (6)$$

We compute the weight for the i th particle correctly by evaluating the ratio of the target and proposal distributions,

$$w_i = \frac{p(z_t | s_t^{(i)}, m_{t-1}^{(i)}) p(s_t^{(i)} | s_{t-1}^{(i)}, u_t)}{\phi q_{global}(s_t^{(i)} | z_t, m_{t-1}^{(i)}) + (1 - \phi) p(s_t^{(i)} | s_{t-1}^{(i)}, u_t)} \quad (7)$$

Each of the distributions involved is a Gaussian that we have already described how to compute. One should notice that

the weights for the particles are equal to the observation likelihood scaled by the ratio of the probability of the pose under the motion model and the weighted sum of the probability under the motion model and the *global* distribution. That is those particles that are supported by both models are given weights that are mostly proportional to the observation probability while those that disagree are penalized.

Finally, our empirical results show that using a constant mixing ratio tends to generate noisy robot trajectories. This is the result of the bad data associations in our observations. Also, the observations are often dominated by landmarks that were seen most recently, biasing the *global* distribution towards the most recently added landmarks. This potentially prohibits the closure of large loops. To correct this, we are varying the mixing ratio as a function of the ratio of old landmarks to total landmarks observed at time t . An old landmark in our case is defined as one that has not been observed for longer than 2 minutes. Using a variable mixing ratio, we rely on the standard proposal in the short term and the mixture proposal for closing loops. In the next section, we give the complete algorithm for our particle filter.

VI. THE σ SLAM ALGORITHM

Algorithm 1 gives the filter update steps for σ SLAM. The input to the algorithm is the set of particles from the previous time step, the current observation and the visual odometry estimate. It starts by computing the ratio of old to total number of landmarks. If this number is larger than 30%, poses are sampled using the mixture proposal otherwise all poses are sampled from the motion model. In the latter case the particle weights are just proportional to the likelihood of the observation given by Equation 3. If the mixture proposal is used, then the weights are computed according to Equation 7. As we mentioned earlier, for efficiency, we only compute q_{global} with respect to the most likely particle from the previous time. If we do not wish to make this approximation then we must move the code that estimates q_{global} inside the last **for** loop, i.e., compute it for every particle.

VII. EXPERIMENTAL RESULTS

We have implemented the σ SLAM algorithm for our RWI-B14 mobile robot. The robot is equipped with a Point-Grey Research Bumblebee stereo vision camera mounted on a pan-tilt unit. The robot also has an on-board Pentium-based computer and no other functioning sensors.

For our experiments, we manually drove the robot inside two adjacent rooms, 10×16 meters, in our laboratory. We collected the stereo images for offline processing. During data collection, we followed a star-like exploration pattern starting from the middle of one room, exploring one corner at a time and returning back to the center to close the loop. We collected a total of 8500 frames and we processed all of them. The total length of the trajectory is 110 meters. The collected data was challenging from a computer vision point of view because they include many frames with blurring caused by the uneven floor surface and people walking in front of the

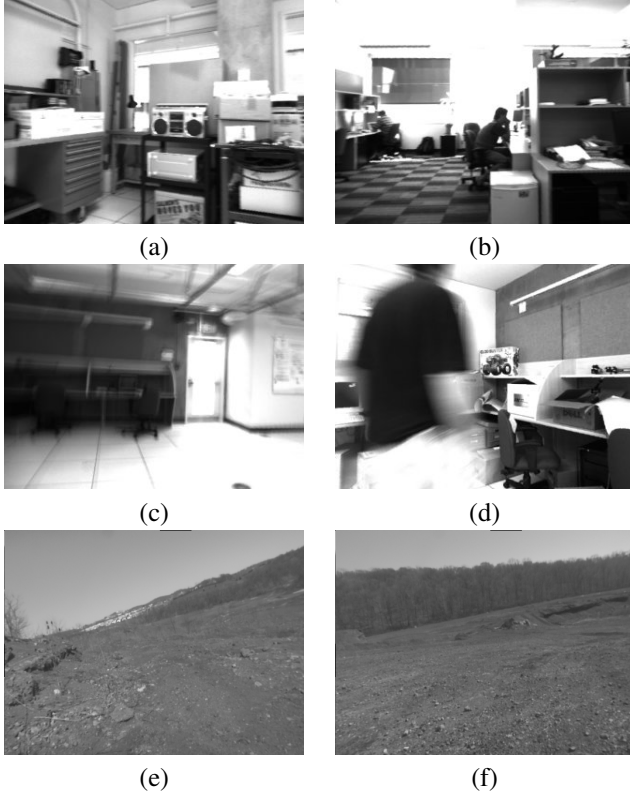


Fig. 2. Sample frames from our test sequences. From our indoor dataset, a scene with (a) lots of texture that is good for localization; (b) excessive brightness and blurring caused by (c) uneven floor surface and (d) a person walking in front of the camera. Also, (e) and (f) are two examples of typical images from the outdoors dataset.

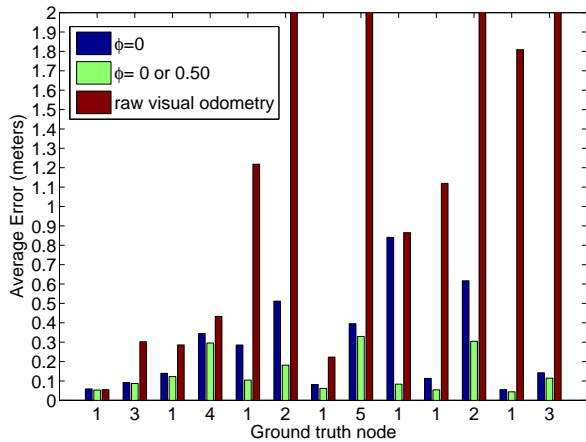


Fig. 3. The localization error for the filter using just the standard proposal, mixture proposal (with a variable mixing ratio) and the raw visual odometry estimates. Note that the robot starts at the left-most node in the plot and finishes at the right-most node.

Algorithm 1 $update(P_{t-1} = \{x^{(i)}, w^{(i)}\}_{(t-1)}^{i=1, \dots, M}, z_t, u_t)$

```

if  $\frac{old\ landmarks}{total\ landmarks} > 30\%$  then
   $use\_mixture = true$ 
   $P_{t-1}^{ML} = \text{most likely particle from } P_{t-1}$ 
   $\{h^{(j)}, h_w^{(j)}\}_{j=1, \dots, N} = generatePoses(P_{t-1}^{ML}, z_t)$ 
  for  $i = 1$  to  $i = N$  do
    compute weight  $h_w^{(i)} = p(h^{(i)} | z_t, m_{t-1}^{ML})$ 
  end for
  Estimate  $N_{global}(\mu, \Sigma)$  from  $\{h^{(j)}, h_w^{(j)}\}$ 
end if
{Compute the new set of particles}
if  $use\_mixture == false$  then
  for  $i = 1$  to  $i = M$  do
    sample  $s_t^{(i)}$  from  $p(s_t^{(i)} | s_{t-1}^{(i)}, u_t)$ 
    compute  $w_t^{(i)} = p(z_t | s_t^{(i)}, m_{t-1}^{(i)})$ 
  end for
else
  for  $i = 1$  to  $i = M$  do
    Draw a random number,  $rand$ , from a uniform distribution
    if  $rand < \phi$  then
      sample  $s_t^{(i)}$  from  $p(s_t^{(i)} | s_{t-1}^{(i)}, u_t)$ 
    else
      sample  $s_t^{(i)}$  from  $N_{global}(\mu, \Sigma)$ 
    end if
    Compute  $w_t^{(i)}$  using Equation 7
  end for
end if
  Normalize particle weights
  Resample
  Update particle maps using  $z_t$ 

```

camera. Also the existence of large windows on the North, East and South walls of the rooms cause large portions of some image frames to be saturated with bright light. Figure 2 shows a small sample of good and bad frames in our test sequence.

Part (a) of Figure 4 shows a top-down view of the 3D landmark map, learned using $\sigma SLAM$ with 500 particles for our indoor data set. We used a mixing ratio that varied between 0 and 50% according to the age of the observed landmarks. The map shown is that of the most likely particle at the end of the trajectory. The map has a total of 80,000 landmarks. Part (b) of Figure 4 also shows the occupancy grid associated with this particle, constructed using the method given in [6], and the dense stereo images returned by the stereo camera. The resolution of the grid cells is $15 \times 15cm$.

In order to measure the accuracy of $\sigma SLAM$, we have measured ground truth pose data for 5 different locations. The starting location is node 1 and each of the corners in the first room are labeled nodes 2, 3, 4 and 5. We have labeled these locations on the learned map shown in part (b) of Figure 4. The plot in Figure 3 shows the average position

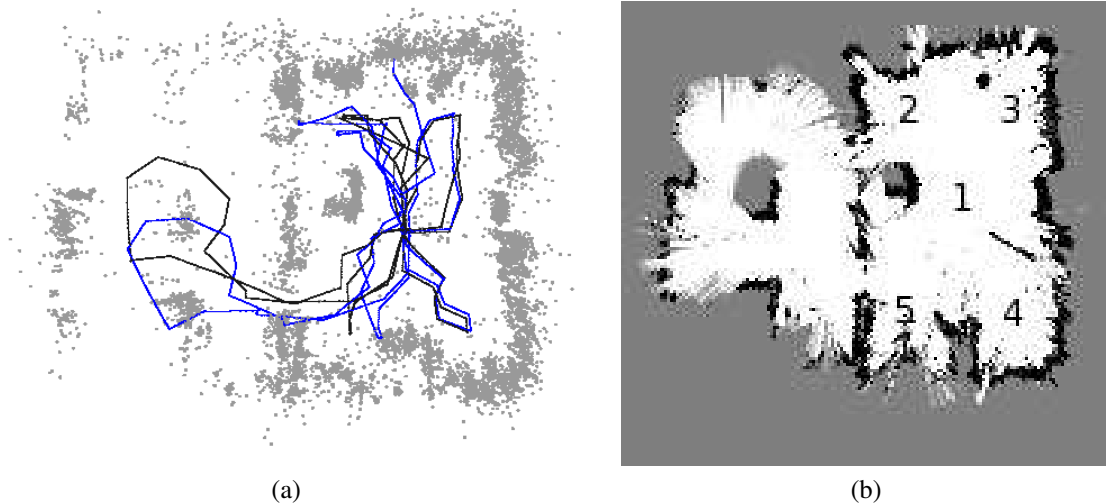


Fig. 4. Examples of the maps learned with σ SLAM for two adjacent rooms size 16×10 meters. Part (a) shows a top-down view of the map of 3D landmarks used for localization. Shown in black is the filter trajectory and in blue the raw visual odometry trajectory. Part (b) shows the occupancy grid constructed using the camera location estimated with σ SLAM. In the grid, white pixels represent empty space while black pixels represent occupied space. Gray pixels denote unobserved areas. The grid's resolution is 15×15 cm.

error at each node for the raw visual odometry, the particle filter using the standard proposal and σ SLAM using the mixture proposal. One can see that the latter outperformed the other methods. In fact, the average error, over all nodes, using the mixture proposal is approximately 15 cm; using the standard proposal it is 25 cm and using just the raw visual odometry measurements it is 159 cm. Also, using σ SLAM the robot closed a large loop (25 meters long), going from node 1 to the room on the left and back to node 1 achieving an accuracy of approximately a factor of two better than using the standard proposal.

Figure 5 shows the processing time for each frame of the two room image sequence running on a Pentium Xeon 3.2 GHz computer. The average time over all frames is 1.5 seconds. A large part of this, 0.35 seconds, is spend on SIFT extraction and matching. The time increases as the number of landmarks in the map increases because it takes longer to compute data associations, update the map for each particle and copy maps between particles. For efficiency, we are organizing all SIFT keys using a kd-tree data structure as originally suggested in [1]. In addition, in order to perform efficient copying of maps between landmarks, we have implemented the tree data structure introduced in [9]. There are frames when the processing time increases well above the average time; this is due to instances when a large number of particles and in effect their maps must be pruned during the resampling step because of their low likelihood. The reader can find our more careful study of the properties of RBPF-based SLAM for both monocular and stereo vision in [24].

We have also tested σ SLAM using data provided by SRI International collected with an outdoors rover (see Figure 2 for example frames.) Ground truth data available with this data set were obtained using RTK GPS. Figure 6 shows the map learned using σ SLAM on this data set for 1182 frames. For this experiment, we used 5000 particles and the robot

traveled a distance of 304 meters. The final map has a total of 5277 landmarks. Comparing the filter estimate with the ground truth for each frame the average error over all frames is approximately 40 meters. This error is the result of two factors. First, even though our visual odometry estimate is in 3D, our particle filter is assuming planar motion; this assumption is violated in this set of data and as a result, the filter estimate is inherently inaccurate. Second, most of the error in the filter is the result of very bad visual odometry estimates on two occasions marked with arrows on Figure 6. The mistakes in visual odometry were the result of very noisy data associations for some frames due to the lack of nearby landmarks and sufficient texture for SIFT matching to work well.

We believe, that if we relax the planar motion assumption in the particle filter and better optimize our visual odometry algorithm, σ SLAM can be an effective algorithm for SLAM in outdoor environments.

VIII. CONCLUSIONS

In this paper we have summarized our work on solving the visual Simultaneous Localization and Mapping problem for mobile robots equipped with stereo vision. We have presented our particle filter-based σ SLAM algorithm that depends on visual odometry and a mixture proposal distribution to map large cyclic, indoor environments. We have also shown that we can construct high-resolution, 2D occupancy-grid maps from correlation-based stereo; these maps are useful for path planning and navigation. Finally, we evaluated the accuracy of σ SLAM using data collected with mobile robots operating in indoor and outdoor environments.

In future work, we will address occasional issues with inaccuracies in visual odometry due to frames with excessive noise making it difficult to robustly identify a sufficient number of landmarks. In addition, we plan to complement

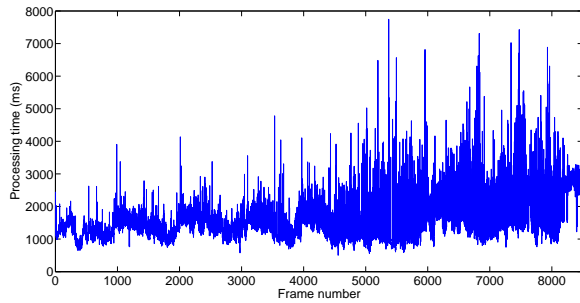


Fig. 5. Plot of the average processing time vs frame number for 500 particles using σ SLAM.

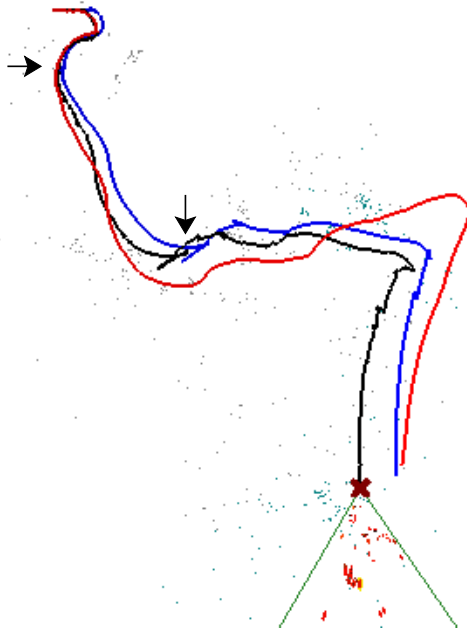


Fig. 6. Results from the outdoors image sequence with 5000 particles using σ SLAM. The visual odometry trajectory is shown in blue. The GPS ground truth data is shown in red and the maximum likelihood particle is shown in black. Not all landmarks are visible in the image. The map has 5277 landmarks. The arrows point to the locations where the visual odometry was very unreliable.

our method with a topological mapper in order to handle even larger environments. Finally, we plan to relax our assumption that the robot is executing planar motion; extending σ SLAM to 3D motion will certainly improve its performance in outdoor environments.

IX. ACKNOWLEDGMENTS

The authors would like to thank Robert Sim for his help developing the software and ideas presented in this paper.

REFERENCES

- [1] D. G. Lowe, "Object recognition from local scale-invariant features," in *Int. Conf. on Computer Vision*, Corfu, Greece, September 1999, pp. 1150–1157.
- [2] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, UK: Cambridge Univ. Pr., 2000.

- [3] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, 2004, pp. 652–659.
- [4] M. Agrawal and K. Konolige, "Rough terrain visual odometry," *Proceedings of the International Conference on Advanced Robotics (ICAR)*, August 2007.
- [5] K. Konolige and M. Agrawal, "Frame-frame matching for realtime consistent visual mapping," *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, April 2007.
- [6] V. Tucakov, M. Sahota, D. Murray, A. Mackworth, J. Little, S. Kingdon, C. Jennings, and R. Barman, "A stereoscopic visually guided mobile robot," in *Proc. of Hawaii International Conference on Systems Sciences*, 1997.
- [7] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous Robot Vehicles*, I. Cox and G. T. Wilfong, Eds. Springer-Verlag, 1990, pp. 167–193.
- [8] 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)*, Florida, USA, 2006.
- [9] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proceedings of the AAAI National Conf. on Artificial Intelligence*. Edmonton, Canada: AAAI, 2002, pp. 593–598.
- [10] D. Hähnel, D. Fox, W. Burgard, and S. Thrun, "A highly efficient FastSLAM algorithm for generating cyclic maps of large-scale environments from raw laser range measurements," in *Proc. of the Conference on Intelligent Robots and Systems (IROS)*, 2003.
- [11] A. I. Eliazar and R. Parr, "DP-slam 2.0," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*. New Orleans, LA: IEEE Press, 2004.
- [12] S. Se, D. G. Lowe, and J. J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *Int. J. Robotics Research*, vol. 21, no. 8, pp. 735–758, 2002.
- [13] K. Murphy, "Bayesian map learning in dynamic environments," in *Neural Information Processing Systems (NIPS)*, 1999, pp. 1015–1021.
- [14] T. D. Barfoot, "Online visual motion estimation using FastSLAM with SIFT features," Edmonton, Alberta, August 2005, pp. 3076–3082.
- [15] A. J. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *Proc. International Conference on Computer Vision, Nice*, Oct. 2003.
- [16] M. Tomono, "Monocular slam using a Rao-Blackwellised particle filter with exhaustive pose space search," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA '07)*, Rome, Italy, April 2007.
- [17] A. Doucet, N. de Freitas, K. Murphy, and S. Russell, "Rao-Blackwellised particle filtering for Dynamic Bayesian Networks," in *Uncertainty in Artificial Intelligence*, 2000.
- [18] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 5, pp. 698–700, 1987.
- [19] D. Lowe, "Fitting parameterized three-dimensional models to images," *IEEE Trans. Pattern Analysis Mach. Intell. (PAMI)*, vol. 13, no. 5, pp. 441–450, May 1991.
- [20] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *Proceedings of the Eighteenth Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*. San Francisco, CA: Morgan Kaufmann Publishers, 2003, pp. 1151–1156.
- [21] N. Karlsson, E. D. Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. E. Munich, "The vSLAM algorithm for robust localization and mapping," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, Barcelona, Spain, April 2005, pp. 24–29.
- [22] P. Elinas and J. Little, " σ MCL: Monte-Carlo localization for mobile robots with stereo vision," in *Proceedings of Robotics: Science and Systems*, Cambridge, MA, USA, 2005.
- [23] S. Thrun and D. Fox, "Monte-Carlo localization with mixture proposal distribution," in *Proceedings of the AAAI National Conference on Artificial Intelligence*, 2000.
- [24] R. Sim, P. Elinas, and J. J. Little, "A study of the Rao-Blackwellised particle filter for efficient and accurate vision-based SLAM," *Int. J. Comput. Vision*, vol. 74, no. 3, pp. 303–318, 2007.