The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

The Causal Update Filter – A Novel Biologically Inspired Filter Paradigm for Appearance-Based SLAM

Niko Sünderhauf, Peer Neubert, Peter Protzel
Department of Electrical Engineering and Information Technology
Chemnitz University of Technology
09111 Chemnitz, Germany
{niko.suenderhauf, peer.neubert, peter.protzel}@etit.tu-chemnitz.de

Abstract— **Recently a SLAM algorithm based on biological principles (RatSLAM) has been proposed. It was proven to perform well in large and demanding scenarios. In this paper we establish a comparison of the principles underlying this algorithm with standard probabilistic SLAM approaches and identify the key difference to be an additive update step. Using this insight, we derive the novel, non-Bayesian Causal Update filter that is suitable for application in appearance-based SLAM. We successfully apply this new filter to two demanding vision-only urban SLAM problems of 5 and 66 km length. We show that it can functionally replace the core of RatSLAM, gaining a massive speed-up.**

I. INTRODUCTION

The SLAM problem [4] has been comprehensively investigated by robotics researchers during the last two decades. A huge variety of algorithms has been developed over the years, since the term first appeared in 1986. Today, probabilistic approaches [15] clearly dominate the field.

Recently, however, a novel biologically motivated approach to SLAM, coined RatSLAM, has been developed [9] [8]. This approach differs substantially from the established SLAM algorithms in that it makes no use of Bayesian calculus or probabilities. Instead, it borrows important key ideas from biological systems and mimics the behaviour of different types of brain cells that have been found to be involved in spatial navigation tasks in rodents, primates and humans.

Our contribution explores RatSLAM from a probabilistic point of view and identifies the key difference between RatSLAM and probabilistic approaches which leads us to the formulation of a novel filter scheme. We are going to present the successful application of this filter to two demanding appearance-based SLAM scenarios.

II. RELATED WORK

A. Appearance-Based Localization and SLAM

Appearance-based approaches to localization aim at recognizing already known places using the visual appearance of the currently observed scene to the camera. [12] used a visual vocabulary tree [10] to localize a given query image in a database of 10 million images acquired on 20 km of urban road.

Extended to SLAM, appearance-based methods have to recognize loop closures but also deal with a constantly growing database of images. Furthermore, the algorithm has to incorporate the possibility that the currently observed image originates from an unknown place that has not been visited before.

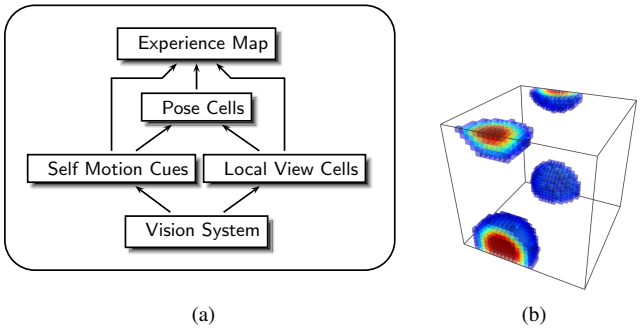
Appearance based place recognition is well suited to be combined with constraint network based map optimization approaches [5] [11] to form a complete SLAM system. Detected loop closures are incorporated into the graph by inserting a new constraint edge.

Depending on the environment and the visual features used for the place recognition, the process is highly prone to false-positives. Two spatially distinct scenes may actually look very similar to the camera or produce very similar image features. This problem, known as *perceptual aliasing* can severely disturb any localization or SLAM process (e.g. by introducing an erroneous edge into a constraint graph).

The incorporation of such catastrophic mismatches into the constraint graph has to be avoided. Different strategies can be applied: The recently proposed FAB-MAP [2] [3] addresses the problem by making the place recognition itself more robust against possible mis-matches caused by the perceptual aliasing. FAB-MAP uses a vocabulary tree based on SURF features [1] and establishes a probabilistic framework that formulates the place recognition as a recursive Bayes estimation problem. This way, false-positive place recognitions are avoided. In contrast, [7] proposes a consistency check based on perspective view geometry to rule out false-positives of the place recognition *after* they occurred.

RatSLAM relies on such post-processing as well: The place recognition uses very simplistic image profile features (a vector of column-wise summed pixels) and is extremely prone to false-positives. The so called *pose cell network* is responsible for filtering the noisy place recognition results before new constraints are introduced into the map.

Despite its relative simplicity, the biologically motivated pose cell network performed so well that the overall system was able to perform SLAM in a 66 km long demanding urban scenario [9]. It is therefore absolutely promising to analyse the concepts behind the algorithm, compare it to established approaches and derive new insights from it. Our contribution will be able to replace this pose cell network with a more efficient approach while maintaining its desirable properties for robust filtering.



(a) (b)

Fig. 1. a) The general structure of RatSLAM. The vision system provides the input for odometry information (self motion cues) and place recognition (local view cells). The pose cells are inspired by the rodent head direction and place cells, while the experience map is responsible for loop closing and maintaining a topologically consistent map. b) A screenshot of our C++ implementation of the pose cell network visualized using OpenGL. The network contains a single packet of activity. Notice how the activity wraps around the network borders.

B. RatSLAM - Biologically Inspired SLAM

The *pose cell network* in RatSLAM [9] [8] is the biologically inspired part of the system and mimics the behavior of so called *place cells* and *head direction cells*. These brain cells have been found to be involved in spatial navigation in rodents, primates and humans. RatSLAM borrows important key ideas from the models of these cells developed in the neuroscience community and adopts and simplifies these models in order to gain computational feasibility.

Fig.1(a) sketches the RatSLAM system. The sole input is provided by the vision system. It provides coarse *self motion cues* by means of visual odometry and is furthermore able to recognize known places the robot already visited. Both the calculation of visual odometry and the place recognition are performed using very simplistic algorithms.

The core of the system is a 3 dimensional *continuous attractor network*, called the *pose cell network* (PCN). Due to the attractor dynamics, self-preserving packets of local activity form in the network of pose cells. These local packets compete, trying to annihilate one another until a stable state is reached. The pose cell network is used to maintain an estimate of the system's current pose in (x, y, θ)-space. Fig. 1(b) illustrates the network and shows how the activity can wrap around the network borders.

Each cell in the PCN can receive additional stimuli from the *local view cells* which inject energy into the pose cell network. These local view cells are driven by the vision system's place recognition.

On the top of the system we find the *experience map* which is responsible for managing a topologically and (to some extend) metrically consistent global map of the environment. It is a graph structure and consists of single *experiences*, each bound to a particular position in the state space and connected to previous and successive other experiences.

III. RatSLAM from a Probabilistic Perspective

In probabilistic SLAM we usually have a multidimensional state space and a time variant random variable X_t that denotes the state of the system at time t. The state space in the RatSLAM approach is 3 dimensional and covers the usual 3 degrees of freedom (x, y, θ) encountered by a ground operating robot in a flat world.

This continuous and infinite state space X_t is mapped into a 3 dimensional discrete and finite structure, the network of pose cells. Each cell P_i in this network has an associated activity level and covers a certain portion of space (e.g. $10m \times 10m \times 10^\circ$ for the city-scale environment in [9]). The activity of each cell can be understood as being proportional to the likelihood that this cell represents the true system state (vehicle pose) according to the system's current knowledge.

This discrete network of cells that represents the system's state estimate, bears a close resemblance to histogram filters, that are discrete versions of the general Bayes filter. Like the pose cell network, histogram filters express the belief distribution $\text{bel}(x_t)$ in a multi-dimensional discretized form.

We now want to shortly review the equations for the discrete histogram filter and compare them to the calculations implemented in the RatSLAM framework.

A. State Prediction

The general equation for the Bayesian state prediction in its discrete form is given by:

$$\overline{p}_{k,t} = \sum_i p(X_t = x_k \mid u_t, X_{t-1} = x_i) \cdot p_{i,t-1} \tag{1}$$

The prediction step incorporates the state transition or motion model that transfers a system state x_{t-1} into a new state x_t given the control input u_t. In RatSLAM, the motion model is very simple, because no uncertainty is added to the system's belief under motion. The activity pattern in the pose cell network P is merely shifted directly according to the control input u_t. This has been pointed out explicitly in [9]. Thus the prediction step in RatSLAM can be expressed by

$$\overline{P}_{k,t} = P_{k',t-1} \tag{2}$$

where $P_{k,t}$ is the activity of the pose cell with coordinates $k = (x, y, \theta)$ at time t, according to the path integration. Given the control input $u = (v, \Delta\theta)$, which is generated by the vision system, the discretized motion model g maps a cell k' to k. This function g of course has to consider the wrap-around structure of the pose cell network.

B. Measurement Update

The second step of the Bayes filter algorithm incorporates the sensor measurements and is called the measurement update step. This step yields the resultant probability that the true state at time t is x_k. In its discrete form it is:

$$p_{k,t} = \eta \cdot \hat{p}_{k,t} \cdot \overline{p}_{k,t} \tag{3}$$

where

$$\hat{p}_{k,t} = p(z_t \mid X_t = x_k) \tag{4}$$

As usual, η is merely a constant that scales the probabilities so that they sum up to 1. $\overline{p}_{k,t}$ is the state prediction that has been calculated in (1), while the conditional probability $p(z_t \mid X_t = x_k)$ represents a connection between the system

state and the sensor measurements and expresses how probable it is to measure the given z_t from the state x_k.

In RatSLAM, a very similar idea is expressed with the help of the local view cells V_i and the connection weight matrix β. The activity of each local view cell V_i expresses the vision system's belief that the currently observed scene is either an already known place the robot visited in the past ($i < t$) or a new, previously unobserved place, in which case a new local view cell V_i with $i = t$ has just been created.

The matrix β contains the learned connection weights between local view cells and pose cells: $\beta_{i,x,y,\theta}$ is large when the i-th local view cell is highly active whenever the pose cell at (x, y, θ) is highly active and vice versa. These connections are continuously updated by Hebbian learning.

Using the weight matrix β, the local view cells can inject energy into the pose cell network. We introduce

$$\hat{P}_{k,t} = \sum_i (\beta_{i,k} \cdot V_i) \tag{5}$$

This data structure $\hat{P}_t = (\hat{P}_{1,t}, \hat{P}_{2,t}, \ldots, \hat{P}_{n,t})$ is a structure analogous to the pose cell network: a three dimensional array of cells, with a size equal to the pose cell network. Each cell in $\hat{P}_t$ contains the energy the local view cells inject into the corresponding pose cell. $\hat{P}_t$ can be understood to be the analogon of the discrete distribution $p(z_t|x_{k,t})$, the probability of the measurement z_t conditioned on $x_{k,t}$.

The injection of energy into the network of pose cells corresponds to the measurement update step of the general Bayes filter. Here, the most important difference between the classical Bayesian approaches and RatSLAM becomes apparent: As we have seen in the above equation (3), the measurement update step is *multiplicative* in the Bayes filter framework. In RatSLAM, however, it is done *additively*:

$$P_{k,t} = \eta(\alpha \hat{P}_{k,t} + \overline{P}_{k,t}) \tag{6}$$

Again, η is a constant that scales the resulting distribution to sum up to 1. α is a constant that controls the strength of the evidence's influence on the posterior.

In RatSLAM, the attractor dynamics of the pose cell network would transform and rescale the shapes of the activity packets. Due to the dynamics, different packets of activity compete, trying to annihilate each other until a stable state is reached. For the further argumentation, we can safely ignore the exact implications of the network dynamics.

Given the assumptions made before, the *additive update step* (6) in the pose cell update algorithm can be generalized to

$$p(x_t|z_t) = \eta\,(\alpha p(z_t|x_t) + p(x_t)) \tag{7}$$

We observe that $p(x_t|z_t)$ now is the weighted average of $p(z_t|x_t)$ and $p(x_t)$ with the weight constant α.

If we combine this equation with the normal state prediction from the Bayesian filter framework we gain a new type of filter, which we call the *Additive Update Filter*:

$$\overline{\text{bel}}(x_t) = \int p(x_t|u_t, x_{t-1})\,\text{bel}(x_{t-1})\,dx_{t-1} \tag{8}$$

$$\text{bel}(x_t) = \eta\left(\alpha p(z_t|x_t) + \overline{\text{bel}}(x_t)\right) \tag{9}$$

C. The Additive Update Filter

Notice the fundamental difference between (9) and (3) is the summation of the prior and evidence distributions instead of the Bayesian multiplication.

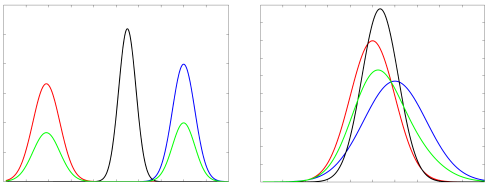
The resulting additive posterior $\text{bel}(x_t)$ clearly is not Bayesian, as it contradicts (3), which in turn is directly derived from Bayes' law of conditional probability. Which consequences arise from this non-Bayesianity? In the Bayesian world, the update step performs an *and*-like operation due to the multiplication of both distributions. The additive scheme resembles an *or* operation. So while in the Bayes world the predicted $\bar{p}(x) = 0$ enforces a posterior $p(x) = 0$ regardless of the evidence $\hat{p}(x)$, this is clearly not the case in the additive world due to the summation of both distributions. This behaviour violates another fundamental law of Bayesian calculus, because $p(\bar{x}) \neq 1 - p(x)$.

The proposed additive scheme is closer to alternative formulations of probability like Dempster-Shafer [13], the possibility theory of Zadeh [16] or the transferable belief model [14]. These formulations distinguish between *uncertainty* and *ignorance*, a distinction that is not possible in the world of Bayesian probabilities. However, the exact connections between the proposed Additive Update filter and these alternative theories of probability are still open for future work.

To conclude this section, we underline our finding that RatSLAM's pose cell network performs an additive update step that contradicts the Bayesian multiplicative update. Fig. 2 compares the results of both update strategies with Gaussian priors and evidence distributions. It is important to understand that although the example is illustrated using Gaussians, neither the Bayesian nor the additive formulation require the involved distributions to be Gaussian.

The situation depicted in Fig. 2(a) corresponds to a large prediction error that might have been caused in the event of a loop-closure after the odometry system accumulated a lot of under-estimated errors due to a wrong estimation of the system noise or measurement noise (over-confidence). Other possible causes (especially in appearance-based SLAM) are false-positive place recognitions or, in the general case, outliers in the sensor data or an erroneous data association. Whatever the causes were, the Bayesian posterior distribution (black) postulates a high probability in an area where neither the prior nor the evidence distribution had a significant proportion of their probability masses.

The solution from the additive update (green) in Fig. 2(a) appears to be optimal in this situation and also more intuitive: When the prediction and evidence distribution strongly disagree, keeping both hypotheses instead of forcing a fusion of the two contradicting distributions expresses the available information more adequately.



(a) (b)

Fig. 2. (a) Multiplicative (Bayesian) and additive update of dissenting Gaussian prior (blue) and evidence (red) distributions along with the respective Bayesian posterior (black) and additive posterior (green). Notice how the additive update splits the probability mass and keeps two independent hypotheses. (b) The same comparison for the standard textbook case of consenting prior and evidence. Notice that the additive result (green) in both cases is not a Gaussian anymore.

IV. The Causal Update Filter

The additive update scheme we derived from RatSLAM's pose cell network bears a significant drawback. To calculate the filter response, one has to calculate the value of the prior, evidence, and posterior distribution for every point in the state and measurement spaces.

As the involved spaces are usually infinite and continuous, the additive filter scheme can not be applied without further precautions: The pose cell network in RatSLAM maps the infinite, continuous state space into a finite, but unbounded, discrete structure. The additive scheme can be easily applied to a structure like this. One then needs a mechanism to map the information contained in this finite discrete structure back into the original, infinite continuous space. In RatSLAM, this is performed by the experience mapping algorithm.

Seeking an efficient implementation of the additive update scheme, we propose the *Causal Update filter* (CUF) whose response is a mixture of the additive and the Bayesian multiplicative update.

In contrast to the pure additive scheme, the CUF can be calculated very efficiently directly in the involved state space without the need of remapping into a finite or discrete space. For performance reasons, we use the established Bayesian calculus for Gaussian distributions. Thus the prior, evidence and posterior distributions in the Causal Update Filter are modelled as multimodal Gaussian distributions.

The core idea of the fusion between prior and evidence distribution is, in short, to use the multiplicative update where the two distributions agree, and the additive update when they disagree.

To achieve this, the algorithm (see Algorithm IV.1 for pseudo-code) determines the pairwise consent between the involved Gaussians using a Mahalanobis-like measure:

$$d_{(N_1,N_2)} = \sqrt{(\mu_1-\mu_2)^T(\Sigma_1+\Sigma_2)^{-1}(\mu_1-\mu_2)} \qquad (10)$$

A threshold on this consent measure decides whether the Gaussians are incorporated into the resulting posterior in a multiplicative or additive way.

The algorithm ensures that *every* Gaussian from the prior and evidence distribution is incorporated into the posterior. Either they are used for a Bayesian update step or they are incorporated additively.

Algorithm IV.1 Causal Update Filter (evidences, priors)

```
1:  for all p_i in priors do
2:    for all e_j in evidences do
3:      if CONSENT(p_i, e_j) then
4:        new_posterior ← BAYESIAN_UPDATE(p_i, e_j)
5:        INCORPORATE(new_posterior, all_posteriors)
6:        p_i.used ← True ; e_j.used ← True
7:      end if
8:    end for
9:    if p_i.used = False then
10:     INCORPORATE(p_i, all_posteriors)
11:   end if
12: end for
13:
14: for all e_j in evidences do
15:   if e_j.used = False then
16:     INCORPORATE(e_j, all_posteriors)
17:   end if
18: end for
19:
20: RESCALE_AND_PRUNE( all_posteriors )
21: best_posterior ← FIND_POSTERIOR_PEAK(all_posteriors)
22: return best_posterior, all_posteriors
```

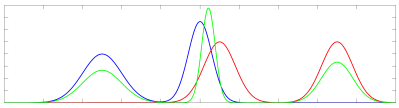


Fig. 3. 1-dimensional example of the CUF. The prior (blue) and evidence (red) distribution are multimodal, each consisting of two peaks. The resulting posterior (green) consists of three Gaussians. Notice how the two hypotheses in consent have been incorporated multiplicatively and the remaining ones have been incorporated additively.

When one of the Gaussians is to be incorporated additively or before a new posterior Gaussian is incorporated into the joint posterior, the filter checks that no other hypothesis exists at the regarded position in the state space. Otherwise, a Bayesian update step is used to fuse the two Gaussians that would otherwise overlap. Again, we use the Mahalanobis distance as a measure of consent.

Fig. 3 demonstrates the concept. We see how the prior and evidence distributions are fused according to the rules given above: The two central peaks of both distributions that are in consent, are fused multiplicatively, the others are incorporated according to the additive scheme.

Each Gaussian is assigned a weight, that is not changed when the Gaussian is incorporated additively. When two Gaussians are fused because they are in consent, the resulting Gaussian is assigned the sum of the weights of the two parent Gaussians. The weights are used in a pruning and rescale step at the end of the algorithm where posterior Gaussians whose weights are too small are removed from the joint posterior. This way, we achieve a kind of voting scheme where posterior Gaussians that represent consent prior and evidence hypotheses are considered more important.

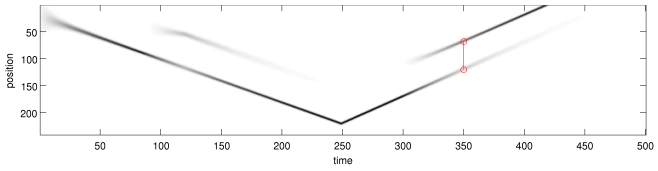


Fig. 4. Simple 1-dimensional example of appearance-based localization using the proposed filter scheme. See the text in section V-A for discussion.

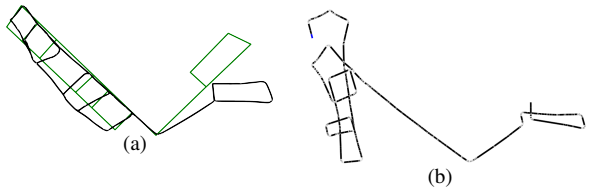


Fig. 5. (a) Result of our vision-only SLAM algorithm (black) for the 5 km dataset using the CUF and landmarks based on visual saliency. Ground truth data (green), derived manually from Google Earth. (b) map created from raw visual odometry input.

V. Applying the Causal Update Filter to Appearance Based SLAM

Before we present results from two real-world experiments, we want to illustrate the application of the Causal Update filter (CUF) in a simple 1-dimensional example.

A. Simulation - Proof of Concept

Fig. 4 sketches a simple experiment where a robot moves in a 1-dimensional world. The gray level represents the estimated bel(x_t), where a darker gray represents a higher probability. As time progresses from left to right, the robot slowly moves through the unknown environment. At time 250, it turns around and heads back, passing through now known terrain.

An appearance-based approach to place recognition is used to recognize possible loop closures. During the first 100 time steps this place recognition behaves correctly. Around time 100 however, the place recognition fails due to perceptual aliasing and erroneously assumes the robot returned to a previously visited position. We can see how a second peak of belief mass (a second hypothesis) is introduced, and how a part of the belief is transferred from the original hypothesis to the new one. As this new hypothesis is not supported by the place recognition for a long time, it quickly weakens and vanishes again.

A little after the robot turned around and started heading upwards again, at time 300, the place recognition correctly recognizes a loop closure that was not captured by odometry. The robot returned to a previously visited location. Again, a new hypothesis is introduced. This time however, it is repeatedly supported by the place recognition, therefore grows stronger and finally outweights the original position estimate. At this time a new constraint link between both position estimates can be introduced (indicated by the red line) if a constraint graph is used to maintain a map.

B. Real-World Results

What was demonstrated in the simple example above was also applied to two large appearance-based SLAM scenarios: The first is the RatSLAM scenario that was originally presented in [9]. It consists of video footage taken on a 66 km long course along the roads of St. Lucia, an Australian suburb. The second dataset is shorter and covers 5 km around our university's campus. For both datasets visual odometry was calculated by very simple means, similar to the technique presented in [9]. A TORO pose graph [5] was used to maintain the map in both scenarios.

Two different techniques to recognize possible loop closures were applied. We tested the simple image profiles proposed by [9] on the larger 66 km dataset. For the smaller scenario we applied a scene descriptor built from saliency based image features [6]. Both techniques produced a high amount of false-positive place recognitions and putative loop closures.

Fig. 6 gives an idea of how often old places were erroneously recognized in the smaller 5 km dataset. Although one can identify the real loop closures as short dark secondary diagonals, the plot demonstrates the high amount of noise that arises from the low descriptiveness of the used image features. On average, 4.8 matchings to other scenes were found for the currently observed scene. The most ambiguous scene was incorrectly recognized at 617 other places while on average, each scene was recognized by 11 other places. Despite the bad quality of the scene matchings and the high rate of false positives, the Causal Update filter was able to maintain a stable estimate of the system's state and filters out all of the erroneous loop closings indicated by the place recognition.

The results of the complete appearance-based SLAM algorithm are displayed in Figures 5 and 7. As in [9] we can only judge the map of St. Lucia (66 km dataset) in Fig. 7 qualitatively, as no ground truth information like a GPS track is available. Compared with the map given in [9] one can identify some flaws (e.g. in the very top left or at the bottom) but in general the road network was successfully captured and the resulting map is topologically correct. Although it is of course not metrically correct, it can be considered semi-metric as the general proportions of the environment are represented adequately.

Compared to the original pose cell network, the CUF is much more efficient: It can be calculated in well under 0.1 ms while our C++ implementation of the network took 600 ms to incorporate odometry information and calculate the network dynamics, which is very expensive. This massive speed-up comes at no costs or disadvantages, as the CUF is able functionally replace the pose cell network and to reproduce its results, as is shown by Fig. 7.

VI. Conclusions

The most important contribution of our paper is the establishment of the Causal Update filter. This novel filter scheme is inspired by the fundamental principles underlying RatSLAM's pose cell network, which in turn was inspired

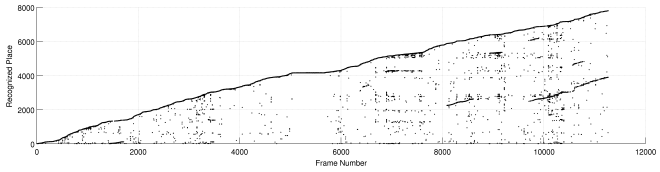


Fig. 6. This plot shows the recognized places for each image frame according to the saliency based place recognition. Several loop closings are visible around frames 1500, 8500, 10000, 11000. Near frame 5000 the vehicle stopped for a while at a red traffic light. Notice that for many frames, several putative loop closures are erroneously recognized by the place recognition system, resulting in an noisy, outlier-perturbed and often multimodal evidence distribution. Without proper filtering, many wrong constraints would be introduced into the TORO pose graph and the mapping algorithm would fail.

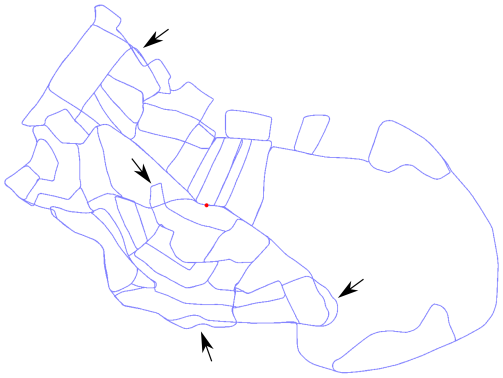


Fig. 7. Final map of St. Lucia (path length 66 km, dimensions 1.8 by 3 km). The Causal Update filter was able to replace the pose cell network and managed to filter the erroneous data from the simplistic scene descriptors. The map has a few flaws (marked by the arrows) but covers the topologic layout and the general metric proportions of the environment.

by biological principles and mimics place cells and head direction cells found in the mammalian brain. While the pose cell network captured the underlying principle (the additive incorporation of evidence and prior information) by sticking closely to the neural nature of the biological archetypes, our CUF is a higher abstraction of this additive incorporation principle, leading to a higher efficiency while maintaining the desirable robustness.

Applied to the same large scale dataset (see the supplementary material and our website[1] for a video), our novel algorithm was able to functionally replace the pose cell network, performing equally well in this very demanding scenario of vision-only SLAM. The main advantage of the CUF over the pose cell network is its efficiency: We measured a speed-up of 600, as the CUF does not have to calculate the expensive network dynamics of a 3-dimensional attractor network but can be calculated with only a small number of operations.

Furthermore, we successfully applied the filter to another appearance-based SLAM scenario, using a different and more sophisticated (and biologically inspired) scene descriptor.

[1] `http://www.tu-chemnitz.de/etit/proaut`

The proposed filter scheme therefore was found to be feasible to be applied as a post-processing step after the place recognition in appearance-based SLAM. The filter is robust against erroneous place recognitions and prevents putative loop closures to be introduced in the constraint graph which would result in a failure of the whole mapping process. No other methods like consistency checks as in [7] or sophisticated place recognition techniques as in [3] were necessary.

In future work we will have to explore the possibilities of the novel CUF filter scheme and apply it to different problems where Bayesian filter techniques are the state of the art today. Seeing if the CUF can be applied to other problems and how well it performs there, remains an open but exciting question.

Acknowledgements

We would like to thank Michael Milford for providing the St. Lucia video footage that was presented in his paper [9]. Further material on RatSLAM is available at the RatSLAM website (`http://ratslam.itee.uq.edu.au`). Furthermore we would like to thank Holger Lietz from the CARAI-Team (`http://www.carai.eu`) for his help during the collection of images for the small SLAM dataset.

References

[1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Proceedings of the ninth European Conference on Computer Vision*, May 2006.

[2] Mark Cummins and Paul Newman. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008.

[3] Mark Cummins and Paul Newman. Highly Scalable Appearance-Only SLAM – FAB-MAP 2.0. In *Robotics Science and Systems*, 2009.

[4] Hugh Durrant-Whyte and Tim Bailey. Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms. *IEEE Robotics and Automation Magazine*, 13(2):99–110, 2006.

[5] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. Non-linear constraint network optimization for efficient map learning. *IEEE Transactions on Intelligent Transportation Systems*, 10(3):428–439, 2009.

[6] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20, November 1998.

[7] K. Konolige, J. Bowman, J. D. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua. View-based maps. *International Journal of Robotics Research (IJRR)*, 29(10), 2010.

[8] Micheal J. Milford. *Robot Navigation from Nature*. Springer Verlag, March 2008.

[9] Micheal J. Milford and Gordon F. Wyeth. Mapping a Suburb with a Single Camera using a Biologically Inspired SLAM System. *IEEE Transactions on Robotics*, 24(5), October 2008.

[10] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *Proceedings of the IIEEE Conference on Computer Vision and Pattern Recognition*, pages 2161–2168. IEEE Computer Society, 2006.

[11] Edwin Olson, John Leonard, and Seth Teller. Fast iterative optimization of pose graphs with poor initial estimates. pages 2262–2269, 2006.

[12] Grant Schindler, Matthew Brown, and Richard Szeliski. City-scale location recognition. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:1–7, 2007.

[13] Glenn Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, 1976.

[14] Philippe Smets. The combination of evidence in the transferable belief model. *IEEE Pattern Analysis and Machine Intelligence*, 12:447–458, 1990.

[15] Thrun, Burgard, and Fox. *Probabilistic Robotics*. The MIT Press, Cambridge, Massachusetts, London, England, 2005.

[16] Lotfi Zadeh. Fuzzy sets as the basis for a theory of possibility. *Fuzzy Sets and Systems*, 1:3–28, 1978.