Werner, Felix and Sitte, Joaquin and Maire, Frederic D. (2008) Visual Topological Mapping and Localisation using Colour Histograms. In *Proceedings International Conference on Control, Automation, Robotics and Vision (ICARCV08)*, Hanoi, Vietnam.

Visual Topological Mapping and Localisation using Colour Histograms

Felix Werner and Joaquin Sitte and Frederic Maire
Faculty of Information Technology
Queensland University of Technology
and
NICTA, Queensland Lab
Brisbane, Australia
{f.werner, j.sitte, f.maire}@qut.edu.au

***Abstract*—In this paper we present a system for appearance-based topological mapping and localisation using vision data. The algorithms are designed for robots which are equipped with FPGA cameras. Such cameras do not provide the entire image to the robot but simple image features like colour histograms.**

Our mapping approach exploits the continuity of the visual appearance of consecutive images from the robots exploration traversal. For topologically mapping the environment colour histograms are clustered whereby each cluster represents a place.

We use a Monte-Carlo localisation strategy combined with the topological map to localise the robot. For a robot equipped with a panoramic camera the proposed strategy works reasonably well, and is capable of overcoming the challenges of severe perceptual aliasing which occurs because of using simple image features and a sparse environment representation through the topological map.

***Index Terms*—Mobile robot, omni-vision, topological mapping, topological localisation, Monte-Carlo localisation, colour histogram, FPGA**

I. Introduction

In mobile robotics, the problems of *mapping* and *localisation* are considered fundamental in the quest of building truly autonomous robots. The capability of building an environment map which can be used for navigation is crucial for an autonomous robot to fulfil its task [1]. There are two main approaches to mapping and localisation: *metric* and *topologic*. Metric approaches typically aim at building a geometrically accurate map of the environment and estimate the position of a robot with respect to the map's coordinate system [1], [2]. In topological systems the environment is usually mapped to a graph of (possibly connected) vertices which refer to places and the graphs edges map the connectivity of the environment [3]. By *place* we mean a set of physically close positions that share the same appearance. A place is represented through its *fingerprint* and visual *appearance* is represented through global image features such as colour histograms [4] or Fourier coefficients [5].

In this paper we address the problem of topological mapping and localisation with the particular goal of developing a vision-based system which is designed for robots with low computational resources as not every robot can be equipped with expensive and power consuming high performance hardware.

In recent years, cameras have become available with an on board FPGA (Field Programmable Gate Array). These FGPA cameras can perform computer vision algorithms with lower power consumption than standard computing equipment [6]. Instead of acquiring an image from the camera and processing it on the robot's CPU, FPGA cameras perform feature extraction already on the camera module so the robot perceives the image features only [7]–[9]. Mobile robots benefit from such cameras as less power and computational resources are required for robotic vision.

We use an appearance–based method for automatically identifying places for the topological map, which exploits the continuity of the visual appearance when a robot traverses the environment [4], [10]. The visual appearance is represented though colour histograms which can be calculated very efficiently even on small FPGAs [11]. A place is a set of positions with similar appearances, which leads to identifying places using a clustering method.

For localisation, the robot attempts to determine the place which corresponds to the robot's location [2], [5]. While we can assume places to be distinctive [3] the fingerprint of a place may be ambiguous. *Perceptual aliasing* causes the world to appear identical to the robot at different positions and, in turn, the same location can appear different to the robot. Perceptual aliasing occurs due to limited sensor capabilities and noisy data sampled at low frequency or repeated structures in the environment.

Many systems attack the problem of perceptual aliasing and environment ambiguities by integrating additional sensors, such as GPS [12], [13] but not all robots have this equipment. Our system uses images obtained from a panoramic camera and odometry information only. Similar to the work done by Menegatti et al. we use a method called Monte-Carlo localisation to track the robots position and prevent the robot from drifting as it would occur if it were using dead reckoning only [14]–[16]. This technique models the robots state as a multi-modal probability density so it can be localised correctly even when a sensor input matches more than one reference place.

978–1–4244–2287–6/08/$25.00 © 2008 IEEE

The remainder of this paper is organised as follows: In Section II we describe our approach for automatically identifying places from the visual appearance of the environment. Section III explains the Monte-Carlo localisation method and how we incorporate the information of the topological map and the robots perceptions. Experimental results are shown in Section IV which is followed by conclusions (Section V).

II. Appearance-based Place Identification

This section describes our approach for appearance–based place identification from vision data. In this work, visual appearance is represented through colour histograms. The method is based on the assumption that the visual appearance is a continuous function of the robots position. Positions with similar appearance can be associated to a place which, in turn, represents a region of positions that exhibit similar appearances. Abrupt changes in the visual appearance of consecutive positions can indicate a passage between two adjacent places.

The continuity property of the visual appearance leads to realise place identification using a clustering method in the perception space. The centroids of the clusters can be interpreted as automatically determined fingerprints of places. Positions with similar visual appearance should fall into the same cluster and hence represent a cohesive place in physical domain. In contrast to methods employing template shapes [17], [18] this place determination strategy does not rely on any predefined model or previous knowledge about the environment, instead, it relies only on the visual appearance which is represented by the colour histograms.

A. Construction of Colour Histograms

Colour histograms have several attractive features especially for panoramic images. They represent the appearance of an image in a very compact manner and colour histograms of panoramic images are invariant to rotations around the vertical axis. Thus, an image acquired at a particular position represents all images at this position with different orientation. In addition, colour histograms usually vary smoothly as the field of vision sweeps the scene when a robot follows a path through the environment.

Usually a colour histogram is created by calculating a N-bin histogram for each of the R, G and B colour bands [19], [20]. This approach is very fast but the 3D spatial information of the RGB tuples in colour space is lost. To retain this information we use 3D histograms in RGB space where the histogram consists of N^3 equally sized bins. On an FPGA, 2^N-bin histograms are very easy to calculate: the N most significant bits of a pixels colour components form the bin numbers [11].

B. Place Identification

We assume the robot has performed an exploration run and has stored the colour histograms of the sensed images along the travelled path. Storing the histograms is very cheap, for example, a standard colour histogram with 256 bins for each R, G and B colour channel requires only 6,144 bytes. As described above, we exploit the continuity property of the visual appearance when a robot traverses the environment by identifying places using a clustering method. A popular and fast clustering method is k–means [21] which aims to minimise the distortion error function

$$e = \sum_{i=1}^{k} \sum_{x \in C_i} ||x - c_i||^2 \tag{1}$$

where $x \in \mathbb{R}^n$ and c_i are the centroids of k clusters C_i. The clustering associates similar input vectors to the same cluster whereby the centroids can be interpreted as descriptors for places. In general, colour histograms which fall into a cluster can be assumed to originate from images which appear similar to the descriptor represented by the clusters centroid and due to the continuity property of the visual appearance these positions should be physically close.

C. Construction of Fingerprints

Mobile robots gain odometric information such as translation and rotation velocities from their wheel encoders. This information can be integrated over time for estimating the agents position. Odometric information is only locally reliable and subject to drift over time as the errors accumulate, mostly due to wheel slippage and uneven surfaces.

However, the robots odometry can be used to enhance the appearance-based fingerprints with metric information in order to decrease the degree of ambiguity. In our system a fingerprint of a place $\mathbf{F} = \{\mathbf{F}^H, \mathbf{F}^P\}$ combines appearance and metric information. The appearance component consists of a colour histogram $\mathbf{H}$ and the metric component is given by the position $\mathbf{P}$ (integrated from odometry) of the particular colour histogram from the data set from the exploration run which is most similar to the appearance component.

III. Localisation

Colour histograms are not distinctive features as the image content is represented in a very compact and compressed manner. For example, a blue floor and a red curtain will result in similar colour histogram as a blue curtain and a red floor. Hence, the robot can get confused because of the visual appearance of the environment can be very similar at two or more different locations.

In this work we overcome the problem of operating on uncertainty using a Bayesian filter technique for reliably estimating the robot's position.

A. Bayes Filters for Localisation

Bayes filters probabilistically estimate a dynamic system's state from noisy observations. In particular, Bayes filters allow robots to continuously update their most likely position within a coordinate system, based on the most recently acquired sensor data. The state at time t is represented by random variables $\mathbf{s}_t$. At each point of time t, a probability distribution over $\mathbf{s}_t$ represents the uncertainty. Bayes filters aim to sequentially estimate the posterior densities of the state $\mathbf{s}_t$ over the state space conditioned on all information contained in the sensor data.

Here, the state $\mathbf{s}_t = (x_t, y_t, \theta_t)^T$ describes the position and orientation of the robot. The Bayesian filter technique is used to recursively estimate the robots state $\mathbf{s}_t$ using action $\mathbf{a}_t$ and perception information $\mathbf{z}_t$. The action component $\mathbf{a}_t = (v_t, \omega_t)^T$ is given by the robots odometric information through the translation velocity v_t and the rotation velocity ω_t.

The Bayesian filter recursively calculates the posterior state distribution $P(\mathbf{s}_{t+1}|\mathbf{z}_{1:t+1}, \mathbf{a}_{1:t})$ from the prior state distribution $P(\mathbf{s}_t|\mathbf{z}_{1:t}, \mathbf{a}_{1:t-1})$ and the perception $\mathbf{z}_{t+1}$ using the recursive filter equation [22]

$$\begin{aligned} &P(\mathbf{s}_{t+1}|\mathbf{z}_{1:t+1}, \mathbf{a}_{1:t}) = \\ &\alpha P(\mathbf{z}_{t+1}|\mathbf{s}_{t+1}) \int P(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t, \mathbf{z}_{1:t}) P(\mathbf{s}_t|\mathbf{z}_{1:t}, \mathbf{a}_{1:t-1}) d\mathbf{s}_t \end{aligned} \tag{2}$$

where α is a normalisation factor. To make this computation tractable, Bayes filters assume the dynamics of the system is Markovian – that is, the current state variable $\mathbf{s}_t$ contains all relevant information. For robot localisation the Markov assumption implies that perceptions depend only on the current physical location and that the robot's location at time t depends only on the previous state $\mathbf{s}_{t-1}$ and the last action $\mathbf{a}_t$ executed. Under the Markov assumption Equation 2 can be simplified to

$$\begin{aligned} &P(\mathbf{s}_{t+1}|\mathbf{z}_{1:t+1}, \mathbf{a}_{1:t}) = \\ &\alpha P(\mathbf{z}_{t+1}|\mathbf{s}_{t+1}) \int P(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) P(\mathbf{s}_t|\mathbf{z}_{1:t}, \mathbf{a}_{1:t-1}) d\mathbf{s}_t. \end{aligned} \tag{3}$$

In our system, the estimation of the robots state is propagated using the transition function

$$\mathbf{s}_{t+1} = f_k(\mathbf{s}_t, \mathbf{a}_t) = \mathbf{s}_t + \begin{pmatrix} v_t \Delta t \cos(\theta_t) \\ v_t \Delta t \sin(\theta_t) \\ \omega_t \Delta t \end{pmatrix} \tag{4}$$

and distributed according to a Gaussian distribution $P(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) = \mathcal{N}(\mathbf{s}_t, \mathbf{\Sigma}_\mathbf{s})$. While the state is propagated in a deterministic manner, Gaussian white noise is added artificially after the propagation according to the distribution parameters. This propagation function satisfies the Markov criterion as the new state of the robot at time $t+1$ is only dependent on the previous state at time t.

The recursive propagation of the posterior density via the recurrence equations 2 and 3 is only a conceptual solution which is difficult to determine in an analytical way [22]. A popular and convenient way is to apply Monte-Carlo methods to solve the filtering Equation 3.

B. Monte-Carlo Localisation

The Monte-Carlo localisation method represents the uncertainty in the robot's state estimation by maintaining a set of samples that are randomly drawn from the probability density function of the state space. A collection $\{\mathbf{s}_t^i, w_t^i\}_{i=1}^N$ of weighted *particles* is used to model the robots state distribution $P(\mathbf{s}_t|\mathbf{z}_{1:t})$.

This sample based representation of the robot's state can be used to solve the filtering equation 3 in a Monte-Carlo fashion. Particles are denoted by $\mathbf{s}_t^i$ and the w_t^i are non-negative weights, called *importance factors* which are normalised such that

$$\sum_{i=1}^N w_t^i = 1. \tag{5}$$

After the state is propagated using Equation 4 the measurement $\mathbf{z}_t$ is used to update the prior state of the robot. In Monte-Carlo approaches this is done by weighting the samples

$$w_t^i = P(\mathbf{z}_t|\mathbf{s}_t^i). \tag{6}$$

In our case, the high level of ambiguity in the appearance of the environment makes it quite difficult to relate the robots state with a measurement. Moreover, due to the sparse representation of the environment through the topological map the robot visits a place usually only for a short time. Also, when the robot travels between two places the measurement might match the previous and as well the place the robot approaches next.

For each particle $\mathbf{s}_t^i$ the geometrically closest place is determined using the position component of the place descriptors of the memorised topological map:

$$c_t^i = \underset{j}{\operatorname{argmin}} ||\mathbf{s}_t^i - \mathbf{F}_j^P||, \qquad j = 1...|\mathbf{F}| \tag{7}$$

The physically closest place c_t^i of a particle is used for weighting its likelihood given the measurement $\mathbf{z}_t$

$$\begin{aligned} &w_t^i = P(\mathbf{z}_t|\mathbf{s}_t^i) \propto \\ &e^{-\frac{1}{2}((\mathbf{z}_t - \mathbf{F}^P_{c_t^i})\mathbf{\Sigma_H}^{-1}(\mathbf{z}_t - \mathbf{F}^P_{c_t^i})^T + (\mathbf{s}_t - \mathbf{F}^H_{c_t^i})\mathbf{\Sigma_s}^{-1}(\mathbf{s}_t - \mathbf{F}^H_{c_t^i})^T)} \end{aligned} \tag{8}$$

using a Gaussian measurement model $P(\mathbf{z}_t|\mathbf{s}_t) = \mathcal{N}(\mathbf{z}_t, \mathbf{\Sigma_z})$. The noise covariances in histogram space $\mathbf{\Sigma_H}^{-1}$ and the robots state $\mathbf{\Sigma_s}^{-1}$ influence the contribution of the appearance and odometric component to the samples weight. The weight associated with every sample is proportional to the likelihood that the robot is occupying that position.

The position estimation is given by the expectation of the posterior state estimation $P(\mathbf{s}_{t+1}|\mathbf{z}_{1:t+1}, \mathbf{a}_{1:t-1})$, that is

$$\mathbf{s}_{t+1} = \sum_1^N w_t^i \mathbf{s}_t^i. \tag{9}$$

The samples are updated recursively using a procedure called *sampling-importance-resampling* [23]. During resampling, the samples are drawn with a probability proportional to their weights. As a result, unlikely samples die out while samples with high probability are replicated.

This localisation strategy also relies on the assumption that the visual appearance of neighbouring positions is similar so the new samples are concentrated in a region around the correct place. Thus, it is even possible to track multiple hypotheses when a robot is kidnapped which eventually will condensate at the true position of the robot. In case of perceptual aliasing, it is a transient situation lasting for a few steps that will not effectively disturb the estimation of the robot's position.

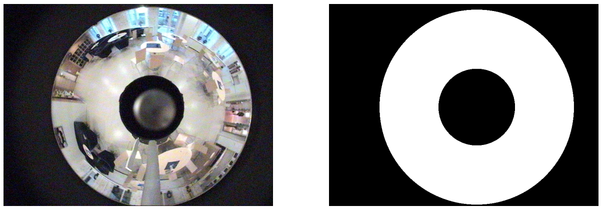

Fig. 1. left: Example panoramic image. right: Pre-defined mask to cut out the used pixel-ring.

IV. Results from Experiments

In this section we show that incorporating a topological map, which may contain ambiguous fingerprints, helps the robot to localise reasonably well and not to get lost or drift as it would happen using dead reckoning only.

A. The Experimental Setup

We evaluate our approach on a data set which was recorded while driving a remote controlled mobile robot through several corridors and office rooms at the University of Örebro, Sweden [24]. From an area covering approximately 60×55 meters a total of 603 images were collected. New images were taken when the robots translation exceeded 50 cm from the previously taken image or, the rotation exceeded 15°. Our system uses only a predefined ring of pixels in the panoramic images (see Figure 1, top right). For evaluation purposes odometry measurements and laser range scans were processed with a SLAM algorithm to provide reliable *ground truth* locations in the real world domain and a 2D environment map (see Figure 2).

The data set we use for experiments is particularly challenging because it contains a high degree of perceptual aliasing as the representation of an image by a colour histogram is highly ambiguous. Figure 2 shows the severity of perceptual aliasing. The redness and size of the dots are proportional to the similarity with respect to the sample taken at the marked position of the robot. It is apparent that there is one large region and smaller region in the environment which share similar visual appearance. Furthermore, images with similar histograms which were recorded in different regions of the environment (A and B) illustrate the fact that different sceneries can induce similar colour histograms.

The experiments were performed off-line only. The input to the system are colour histograms and the corresponding reference locations from integrated odometry measurements. For the fingerprints of the places contained in the topological map, however, the metric component is given by the exact ground truth position for the ease of displaying the results.

Note, we use the same data for place identification and localisation. We do this as we experienced that colour histograms are strongly affected by changes in illumination. However, there is no perfect match between the topological map. That means the fingerprints and the input data. The fingerprints are identified using a clustering method in feature space, so, due to the nature of the clustering method, none of the place colour

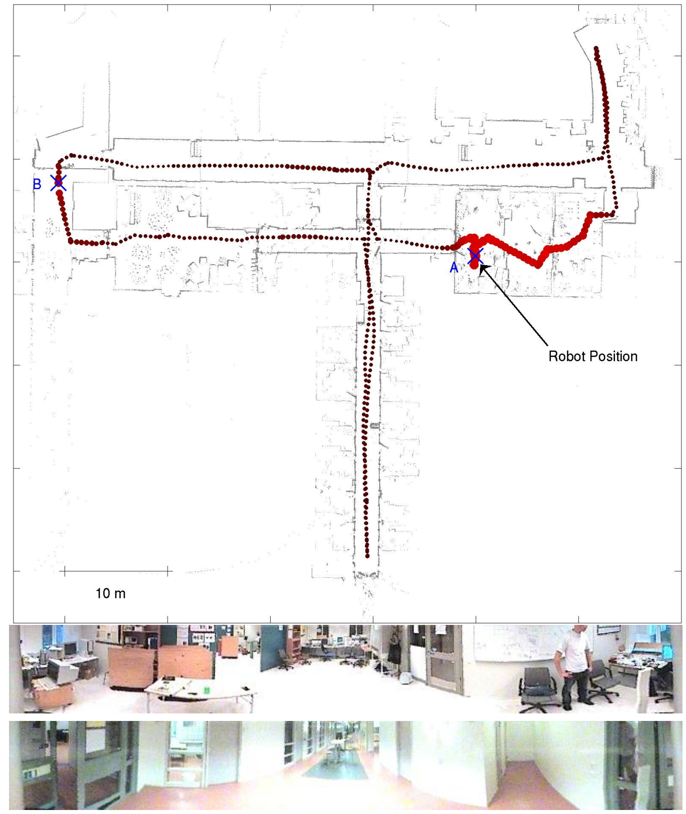

Fig. 2. This figure shows the severity of the aliasing in our experimental environment using colour histograms as image features. top: A big red dot reflects high similarity to the histogram of the robots position. It reveals, that not only the neighbourhood of the robots position but also different regions in the environment have similar appearance in histogram space. middle and bottom: Two unwrapped images (A and B) with similar colour histograms taken in different regions in the environment as indicated in the top image.

histogram components of the descriptors is identical to any measured colour histogram any more. Moreover, as mentioned before, instead of the position gained from odometry, the corresponding ground truth position was incorporated as geometric components of the fingerprints for the ease of displaying the results.

B. Place Identification

The images that were recorded during the exploration of the environment were compressed to standard colour histograms with one histogram for each colour band and 3D colour histograms with various numbers of bins. This paper is focused on the applicability of colour histogram based fingerprints for robot localisation so we do not give an analysis of the effects of varying number of histogram bins or the comparison of standard and 3D colour histograms. In general, keeping the 3D structure of the RGB-tuples allows to use histograms with fewer bins for similar results as standard colour histograms with many bins. For the following evaluations, we use 3D colour histograms with $2^{2^3} = 64$ bins.

Figure 3 illustrates the results of the place identification process using k–means clustering to five clusters. Five clusters were chosen for the ease of visualising the results, however,

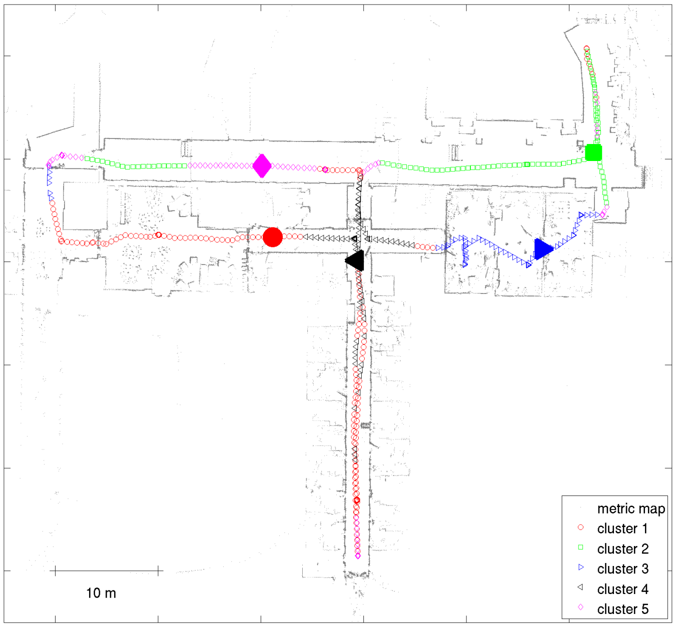

Fig. 3. Results of the place determination process to five places. It illustrates that if the number of fingerprints is too sparse for mapping the environment there are several cohesive regions which associated to single fingerprint and thus form an alias for a place.

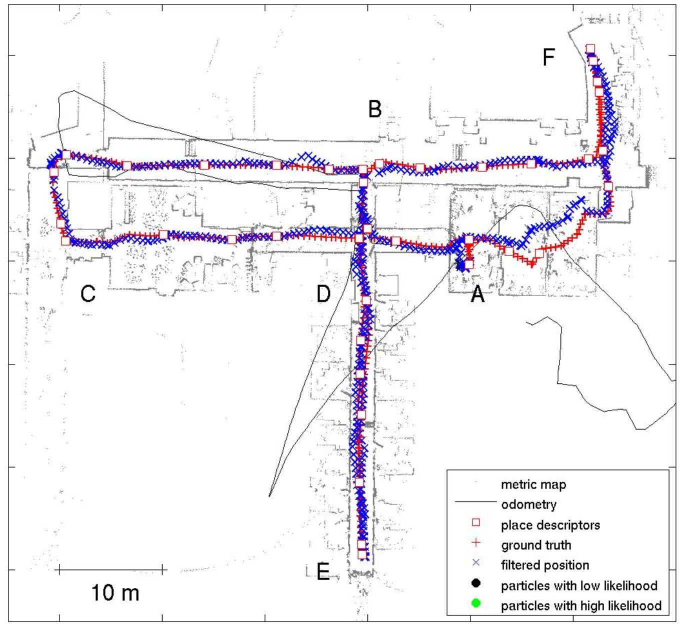

Fig. 4. Result of the Monte-Carlo localisation after 602 steps. The trajectory of the robot is given through `A-B-C-D-E-D-B-F-A` with respect to the labels. It is demonstrated that the Monte-Carlo localisation prevents the robot from getting lost and is capable of keeping track of the robot fairly well. Furthermore, it is to see that close to a fingerprint, the likelihood of the particles and hence their weight increases as it is displayed through the size and level of green of the particles. When the robot starts, it occupies the same position as indicated in Figure 2 and after the first resampling, the particles distribute close to the fingerprints which represent the regions with similar appearance to the robots perception. Note the drift of the robots odometry (–).

the general result is demonstrated. Positions whose histograms fall into a particular cluster are marked as ◁, ▷, ◇, ○, □. The enlarged markers display the geometric parts of the fingerprints. It is apparent that the clustering is capable of finding homogenous places with similar appearance. This justifies the assumption of continuity in visual appearance when a robot moves through the environment. Furthermore, Figure 3 illustrates that despite colour histograms can saliently represent the appearance of the environment, they often do not contain enough information to characterise a place distinctively. So it occurs that there are cohesive regions which are not represented by a fingerprint but are associated with a fingerprint from a place which is physically different. In this case, the robot might get misled as the memorised map does not contain enough places to represent the environment sufficiently for the applied localisation strategy to work successfully.

For our system, determining the number of places is crucial and must be adapted to fit the requirements of the localisation strategy. As a travelling robot usually visits the physical location of a fingerprint only for a short time, the distance between the fingerprints should not be too far to avoid the robot to get lost. We can simply increase the number of possible places, which results in several very similar fingerprints which represent an appearance but are located at different positions. This is demonstrated in Figure 4 where the colour histograms from the exploration run were clustered to 30 fingerprints. In this case, the maximal physical distance between two fingerprints is less than 7 meters.

C. Localisation

Figure 4 shows the result of the localisation strategy using a topological map with 30 places (□). The estimated position of the robot is shown as blue crosses (×) and the ground truth as red pluses (+). The black line (–) displays the trajectory calculated from the robots odometry. The position of the robot is calculated as the weighted mean of the particles according to their weighs (see Equation 8). It is shown that our approach works fairly well and the robot can avoid drift as it would happen when using odometry only and it does not get lost despite the very sparse representation of the environment through the topological map. The average localisation error with respect to the ground truth is 0.6105 meter with a variance of 0.1749.

One of the main advantages of the Monte-Carlo localisation strategy is the capability to represent multi-modal probability distributions. This is especially helpful to localise the robot from scratch. In our implementation, we use 100 particles to model the distribution of the robots position, which are randomly distributed over the reference location when the robot needs to localise at the beginning of its journey. Usually, our system is capable of localising the robot correctly after it has passed at least two fingerprints. Note, the descriptors do only provide a location but no orientation so after visiting one place the robot is not able to determine its orientation and might get lost.

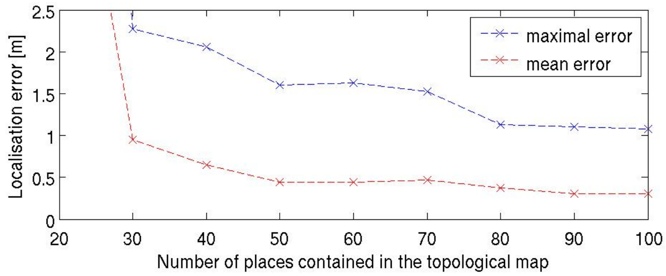

Fig. 5. Results of the place determination process to five places. It illustrates that if the number of fingerprints is too sparse for mapping the environment there are several cohesive regions which associated to single fingerprint and thus form an alias for a place.

The graphs in Figure 5 show the mean and maximum localisation error for different sizes of topological maps. It is illustrated that the mean localisation error decreases with increasing number places contained in the topological map. The maximum localisation error points out whether the robot might get mislead or the position estimation fails for a short while. According to Figure 5 we can assert that the robot does not get totally lost and, as expected the maximal localisation error also decreases with more places represented in the topological map.

V. Conclusion

In this paper, we presented an approach for appearance-based visual topological mapping and localisation. The system is particularly designed to make it usable on robots with limited computational power which are equipped with an FPGA camera. FPGA cameras can perform feature extraction already on the camera module and are known to be low in power consumption in comparison to standard computing equipment. We use colour histograms as image features as they can be perceived directly from an FPGA camera so the robot does not need to perform any image processing routines. For topologically mapping the environment we exploit the property of the colour histograms varying smoothly when the field of vision sweeps through the scene. The places are determined using a clustering methods where the centroids of the cluster contribute the appearance component to the fingerprints.

We overcome the problems of the perceptual aliasing which is mainly caused by the limitation of colour histograms to describe places distinctively and the sparse representation of the environment through the topological map, by applying a Monte-Carlo localisation technique. It is shown that our system is able to track the position of the robot well and is capable to avoid the robot drifting or getting lost. Using only 100 particles for the Monte-Carlo localisation, keeps the system low on computational cost and memory requirements.

Acknowledgment

The authors would like to thank Peter Biber (Tuebingen University) and Henrik Andreasson (University of Örebro) for providing the dataset and also Christian Weiss (Tuebingen University) for the calculation of the reference positions and the environment map.

NICTA is funded by the Australian Government's Backing Australia's Ability initiative, in part through the Australian Research Council and the Queensland State Government.

References

[1] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann Publishers Inc., 2002.

[2] D. Filliat and J.-A. Meyer, "Map-based navigation in mobile robots: I. a review of localization strategies," *Cog. Sys. Res.*, vol. 4, no. 4, pp. 243–282, December 2003.

[3] J.-A. Meyer and D. Filliat, "Map-based navigation in mobile robots: II. a review of map-learning and path-planning strategies," *Cog. Sys. Res.*, vol. 4, no. 4, pp. 283–317, December 2003.

[4] F. Werner, J. Sitte, and F. Maire, "Automatic place determination using colour histograms and self-organising maps," in *ICAR*, Jeju, Korea Republic, August 2007, pp. 111–115.

[5] E. Menegatti, M. Zoccarato, E. Pagello, and H. Ishiguro, "Image-based monte-carlo localisation with omnidirectional images," *Rob. Auton. Syst.*, vol. 1, pp. 17–30, 2004.

[6] S. Choi, R. Scrofano, V. K. Prasanna, and J.-W. Jang, "Energy-efficient signal processing using fpgas," in *ACM/SIGDA Int. Symp. on FPGAs*. New York, NY, USA: ACM, 2003, pp. 225–234.

[7] J. D. Anderson, D. J. Lee, and J. K. Archibald, "Fpga implementation of vision algorithms for small autonomous robots," in *SPIE*, 2005.

[8] T. Chinapirom, U. Witkowski, and U. Rückert, "Universal fpga-microcontroller module for autonomous minirobots," in *AMiRE*, 2005.

[9] R. LeGrand, K. Machulis, D. P. Miller, R. Sargent, and A. Wright, "The XBC: a modern low-cost mobile robot controller," in *IROS*. IEEE, August 2005, pp. 3896–3900.

[10] F. Werner, J. Sitte, and F. Maire, "On the induction of topological maps from sequences of colour histograms," in *DICTA*. Adelaide, Australia: IEEE, December 2007, pp. 167–174.

[11] M. Ali, J. Sitte, and U. Witkowski, "Parallel early vision algorithms for mobile robots," in *AMiRE*, 2007.

[12] J. Wang, L. Dai, T. Tsujii, C. Rizos, D. Grejner-Brzezinska, and C. Toth, "GPS/INS/Pseudolite integration: Concepts, simulation an testing," in *ION GPS*, Salt Lake City, UT, USA, September 2003, pp. 2708–2715.

[13] D. Berdjag and D. Pomorski, "DGPS/INS data fusion for land navigation," in *FUSION*, 2004.

[14] D. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello, "Bayesian filtering for location estimation," *Perv. Comp.*, vol. 2, no. 3, pp. 24–33, 2003.

[15] S. Thrun, "Particle filters in robotics," in *UAI*, 2002, pp. 511–518.

[16] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *ICRA*, vol. 2. IEEE, May 1999, pp. 1322–1328.

[17] E. Krotkov, "Mobile robot localization using a single image," in *ICRA*. IEEE, May 1989, pp. 978–983.

[18] G. Dedeoglu, M. J. Matarić, and G. S. Sukhatme, "Incremental, on-line topological map building with a mobile robot," in *DARS*, vol. 15. Boston, MA, USA: Springer, 1999, pp. 129–139.

[19] I. Ulrich and I. R. Nourbakhsh, "Appearance-based place recognition for topological localization," in *ICRA*, 2000, pp. 1023–1029.

[20] I. Rañó, E. Lazkano, and B. Sierra, "On the application of colour histograms for mobile robot localisation," in *ECMR*, vol. 1, 2005, pp. 189–193.

[21] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience, 2000.

[22] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, February 2002.

[23] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," in *Radar and Signal Processing*, vol. 140, no. 2, Apr 1993, pp. 107–113.

[24] H. Andreasson, A. Treptow, and T. Duckett, "Localization for mobile robots using panoramic vision, local features and particle filter," in *ICRA*. Barcelona, Spain: IEEE, April 2005, pp. 3348 – 3353.