Salient Visual Features to Help Close the Loop in 6D SLAM

Lars Kunze, Kai Lingemann, Andreas Nüchter, and Joachim Hertzberg

University of Osnabrück, Institute of Computer Science
Knowledge Based Systems Research Group
Albrechtstr. 28, D-49069 Osnabrück, Germany
{lkunze|klingema|anuechte|jhertzbe}@uos.de

Abstract. One fundamental problem in mobile robotics research is *Simultaneous Localization and Mapping* (SLAM): A mobile robot has to localize itself in an unknown environment, and at the same time generate a map of the surrounding area. One fundamental part of SLAM algorithms is loop closing: The robot detects whether it has reached an area that has been visited before, and uses this information to improve the pose estimate in the next step. In this work, visual camera features are used to assist closing the loop in an existing 6 degree of freedom SLAM (6D SLAM) architecture. For our robotics application we propose and evaluate several detection methods, including salient region detection and maximally stable extremal region detection. The detected regions are encoded using SIFT descriptors and stored in a database. Loops are detected by matching of the images' descriptors. A comparison of the different feature detection methods shows that the combination of salient and maximally stable extremal regions suggested by [12] performs moderately.

1 Introduction

One application of visual attention is to support recognition of places recorded in digital images. Mobile robotics broaden this need, since it can be used to solve the SLAM problem, i.e., building an environment map and localizing a robot within this map at the same time. Loop closing is the subproblem of recognizing that the robot has reached an already visited place again, enabling the SLAM algorithms to bound accumulated errors and to create a consistent map. More precisely, there are two different problems in the task of loop closing: First, in place recognition, a robot recognizes if it has visited an area under inquiry before. Second, knowledge integration is the task of incorporating this extra information into the localization and mapping data structures. In this work only the problem of place recognition is addressed. To recognize a place, the robot relies only on visual information from camera data. In this paper we re-implement and evaluate the place recognition procedure suggested by [12] that tries to solve loop closing by using using visual features, namely salient features.

The proposed visual feature detector is based on two criteria: saliency and wide-baseline stability. It is compared to other detectors within the framework developed by [10]. In particular, we are interested in the exclusive usage of salient regions or MSERs (maximally stable extremal regions). The evaluation shows that for our robotics application, i.e., the loop detection, encoding the feature

Proceedings of the ICVS Workshop on Computational Attention & Applications - WCAA 2007
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany
This document and other ICVS contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

regions by using SIFT descriptors [6] is justified. In the last step, the procedure is integrated into the existing 6D SLAM architecture [15] on the Kurt3D robot platform to assist the present algorithms regarding loop closing.

This paper is structured as follows: First, we present a brief state of the art. The methods used for visual feature extraction, the feature matching and the application of loop closing are described in Section 2. The used data set, the evaluation criteria and the evaluation results are presented in Section 3. Finally, the robot architecture and the testing results are explained in Section 4. Section 5 discusses the results, addresses open issues and concludes.

Related Work. There is a variety of approaches that tackle the SLAM problem; some methods build 2D maps, while other research groups aim at building 3D maps. To this end, the robot poses are estimated with 6 degrees of freedom taking the x, y and z position and the roll, yaw and pitch angle into account [1,15]. An excellent state of the art of 2D SLAM is given in [16]. Many sensors are used to solve SLAM, e.g., wheel encoders, gyros, GPS, laser range finders or cameras. A few groups combine the latter two sensors [2, 3, 12], e.g., for loop closing.

In order to make loop detection robust, Newman and Ho suggest an approach that does not rely on the robot's pose estimation to decide whether a loop closure is possible [12], since such data is typically erroneous. This approach is followed in this paper, too: Only visual data is considered for the loop detection task. Laser data is purely used for the purpose of map building.

Camera data for loop detection is utilized by extracting features from the captured images and processing them. This is normally done in three steps: detecting the features, encoding them using feature descriptors, and finally matching them against each other. Many region detectors are available, namely, the MSER detector, the Salient region detector, the Harris-Affine detector, the Hessian-Affine detector, the Intensity extrema based detector (IBR), and the Edge based detector (EBR) [5,7,8,13,17]. There are also quite a number of feature descriptors, e.g., SIFT, GLOH, Shape Context, PCA, Moments, Cross correlation and Steerable filters [10]. Important attributes are invariance to scale, rotation, transformation or changes in illumination. A good overview over both detectors and descriptors can be found in [9].

2 Loop Closing

The loop closing procedure is basically designed like this: Images are taken from the robot in an incremental fashion. These images are applied to the visual feature extraction pipeline one at a time. The result of the pipeline, the extracted features, are stored in a database. After the first image is processed, the resulting features of each succeeding image are matched against all features that are already stored in the database to detect a loop. The matching of the features is equivalent to the distance between vectors in a high dimensional vector space. A loop closing hypothesis is generated if similar features are found in two images, that is, if their distance is below a certain threshold.

Like in the study [12], three steps are needed in the visual feature extraction pipeline to generate the feature representation that is stored in a database and

Proceedings of the ICVS Workshop on Computational Attention & Applications - WCAA 2007
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany

Fig. 1. Results for the saliency, the MSER and the combined saliency-MSER feature detection for the same image. The detected visual features are highlighted: Salient regions in the left, MSERs in the middle, and the overlapping regions from these both images are shown on the right image.

used for the matching process: the saliency detection (Section 2.1), the MSER detection (Section 2.2) and the SIFT description (Section 2.3). Two criteria are used to detect the features in the image, namely saliency and wide-baseline stability (MSER detection). An example for the detection methods is shown in Figure 1. The detected regions are encoded in the third step with the SIFT description algorithm.

2.1 Salient Region Detection.

Saliency is the first criterion for feature detection. It can be understood best as characteristic for "interesting" regions. Presumably, those regions are relatively sparse in an image. That makes this metric useful for loop detection, because features are more or less unique in an image and, accordingly, for a location.

The scale-saliency algorithm developed by Kadir and Brady [4] was used by Newman and Ho for loop closing [12]. It defines saliency by locally distinct "complexity", i.e., by local changes in entropy H_D of the image region D compared to its environment. A region is a scaled circular window around a center pixel $\boldsymbol{x}$. The window size or scale s is bound to a range between a minimum and a maximum scale value. Pixels and their values within the region are denoted with d_i. The probability density function for region D is $P_D(s, \boldsymbol{x})$, it returns the probability for a certain value of d_i in its corresponding region. The following equation is used to calculate the distinctiveness for each image pixel and for different scales:

$$H_D(s, \boldsymbol{x}) = -\int_{i \in D} P_D(s, \boldsymbol{x}) \log_2 P_D(s, \boldsymbol{x}) d_i. \tag{1}$$

In order to select only those scales which contribute most to the result, the entropy measure is weighted. The weight puts more emphasis on scales where the entropy changes significantly in respect to their next neighbor scales. The rate of change of the probability density function $P_D(s, \boldsymbol{x})$, multiplied with the scale s, meets the needs as weighting factor:

$$\mathcal{W}_D(s, \boldsymbol{x}) = s \int_{i \in D} \left| \frac{\partial}{\partial_s} P_D(s, \boldsymbol{x}) \right| d_i. \tag{2}$$

Thus, the overall metric for salient regions $\mathcal{Y}_D(S, \boldsymbol{x})$ is the described entropy H_D multiplied with the weighting factor $\mathcal{W}_D(s, \boldsymbol{x})$:

$$\mathcal{Y}_D(S, \boldsymbol{x}) = H_D(S, \boldsymbol{x}) \times \mathcal{W}_D(S, \boldsymbol{x}). \tag{3}$$

2.2 Maximally Stable Extremal Region (MSER) Detection.

The second criterion for the feature detection is wide-baseline stability. The benefits of these features are that they are robust against monotonic changes of image intensities as well as continuous transformations of image coordinates. The last property is useful for loop detection, since the robot will barely reach the exact same pose as before. That is, the viewpoint of the robot will have changed at a second encounter of some place.

The detection algorithm that was developed by Matas et al. [7] fits the needs of such wide-baseline stability, since the maximally stable extremal regions have the following properties:

- They are invariant to monotonic changes of image intensities.
- The neighborhood of the regions is preserved under transformation.
- The regions are stable, because they stay unchanged over an interval of thresholds.
- Both very small and very large regions are detected.

In this implementation, gray-scale images with pixel values from the range $[0, \ldots, 255]$ are considered. First all pixels of the image are sorted in $O(n)$ into bins with regard to their intensities using the BINSORT algorithm. For each intensity from 0 to 255, a list of connected pixels (or extremal regions) below the current intensity is maintained. The union-find algorithm determines the extremal regions efficiently; they are then stored in a data structure with their corresponding intensity levels and sizes. If two regions merge, the smaller is subsumed by the larger one. In a last step, those intensities of the regions are chosen from the data structure where the rate of change of the region size has a local minimum. The regions with these selected intensities form the maximally stable extremal regions. The algorithm that is described here detects the minimum intensity regions. Maximal intensity regions can be found analogously, only the input image needs to be inverted. In this work both types of regions are detected.

2.3 SIFT Description.

SIFT descriptors as developed by Lowe [6] are compact, highly distinct and invariant to image rotation. These aspects make them very attractive and popular in recent work [14].

In this work, SIFT descriptors are used to encode the detected feature regions and store them in a database. Besides the SIFT description method other approaches are tested on the regions detected by the Salient-MSER detector in the publicly available evaluation framework [10]. SIFT descriptors perform best on nearly all tested scenarios.

Proceedings of the ICVS Workshop on Computational Attention & Applications - WCAA 2007
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany
This document and other ICVS contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

2.4 Application of Loop Closing

Loop closing uses visual features in the following way for detecting a loop in the path of the robot. This algorithm detects a loop if some descriptors of two images are similar, taking the norm of two descriptors as a similarity measure. For each image pair the similar descriptors are counted. If this number exceeds a certain threshold, a loop hypothesis is generated. In our experiments we found that a sensible value of the threshold on the number of similar descriptors is 2 or 3. More precisely, for a query image I_q:

1. Generate n_q feature descriptors V_q from the image I_q.
2. Store feature descriptors and capture time of the image in the database.
3. For each candidate image I_c in the database:
   (a) Retrieve all n_c candidate feature descriptors V_c from the database.
   (b) Build a $n_q \times n_c$ matrix $M_{q,c}$ where the (i,j)-th entry $M_{q,c}(i,j)$ is the Euclidean norm $d_{ij} = \| V_q(i) - V_c(j) \|$.
   (c) Thresholding the distances results in n_{qc} matched descriptors.
4. After all candidate images are processed, the candidate images with the largest number of n_{qc} matched descriptors are selected if the number is higher than a certain threshold.
5. The capture times of the selected images are compared with a separate journal of temporal and spatial information in order to determine the location where the candidate image was made. Finally, a loop hypothesis for the assumed location is generated.

3 Evaluation

The presented algorithms for feature detection in Sections 2.1 and 2.2, and the description in Section 2.3 are compared with other methods within the framework developed by Mikolajczyk and Schmid [10]. The used test data and evaluation criteria are described in Section 3.1. The results are presented in detail in Section 3.2.

3.1 Data Set and Evaluation Criteria

To produce comparable results, the described detectors and descriptors were tested on the same data set. It is publicly available on the website `www.robots.ox.ac.uk/~vgg/research/affine/`. This test set contains images where different image transformations were applied: image blur, viewpoint change, zoom and rotation, light change, and JPEG compression. In addition to these transformations, the images feature either structured or textured scenes.

The evaluation framework provides the tester with two measures that can be used for analyzing the feature detectors. These measures are the repeatability and the matching score. Both measures are calculated for a given image pair.

The repeatability score is the ratio between the number of region-to-region correspondences and the smaller number of detected regions in the image pair:

$$repeatability\ score = \frac{\#\ corresponding\ regions}{\#\ detected\ regions}. \tag{4}$$

Proceedings of the ICVS Workshop on Computational Attention & Applications - WCAA 2007
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany
This document and other ICVS contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

The second measure, the matching score, is the relative number of correctly matched regions compared with the smaller number of regions in the pair of images:

$$matching\ score = \frac{\#\ correct\ matches}{\#\ detected\ regions}. \tag{5}$$

3.2 Evaluation Results

The following feature detectors are tested within the presented framework: the Salient-MSER detector, the MSER detector, the Salient region detector, the Harris-Affine detector, the Hessian-Affine detector, the Intensity extrema based detector (IBR), and the Edge based detector (EBR). Recall that the study [12] has used only the combined Salient-MSER detector.

For each image transformation and each scene type, a set of six images is applied to the detectors. One image is the reference image, the others show the same scene under increasing image transformations. For the evaluation, the reference image is pairwise processed with each of the other five images. The repeatability and the matching score are reported exemplarily for some transformations in Fig. 2, 3, 4, and 5.

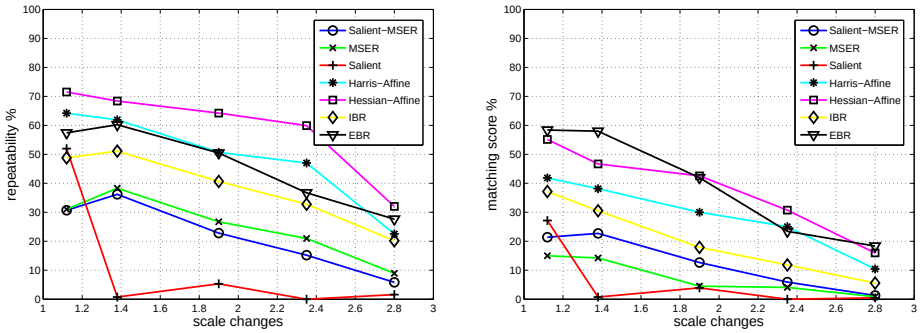

Fig. 2. Scale change transformations for the structured boat scene, cf. website in Section 3.1. Left: Repeatability score. Right: Matching score.

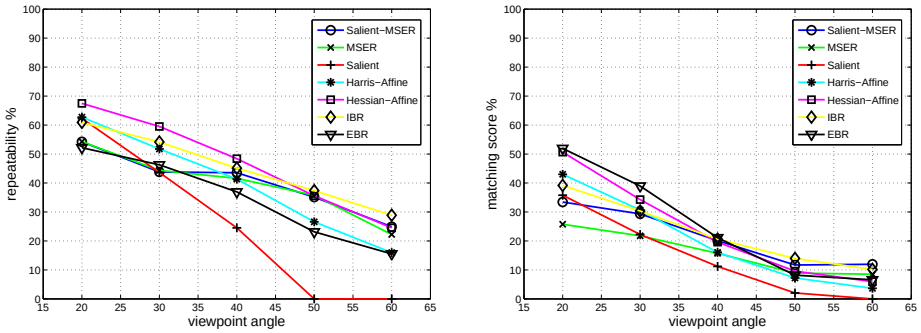

Fig. 3. Viewpoint change transformations for the structured graffiti scene, cf. website in Section 3.1. Left: Repeatability score. Right: Matching score.

Proceedings of the ICVS Workshop on Computational Attention & Applications - WCAA 2007
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany
This document and other ICVS contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

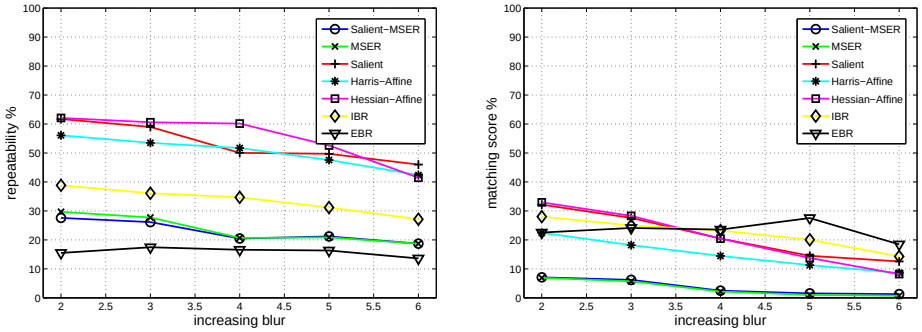

Fig. 4. Blur transformations for the textured tree scene, cf. website in Section 3.1. Left: Repeatability score. Right: Matching score.

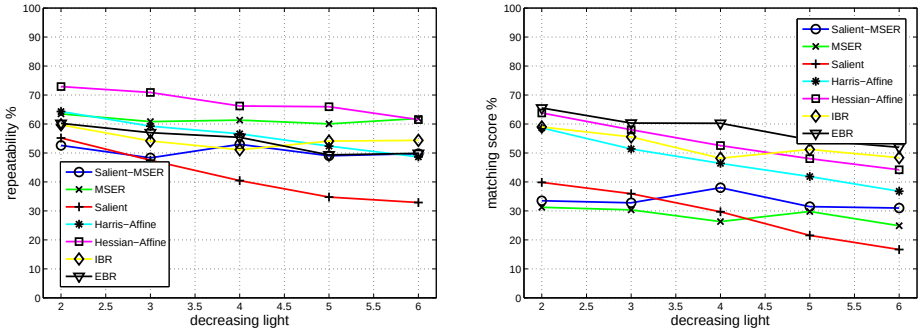

Fig. 5. Light change transformations for the outdoor scene, cf. website in Section 3.1. Left: Repeatability score. Right: Matching score.

In general, the change of the viewpoint seems to be the most difficult setting for all detectors, followed by the change of the scale. For increasing blur and decreasing light changes, nearly all detectors are relatively robust and show almost horizontal curves. The matching of feature region is better in structured than in textured scenes. The Hessian-Affine detector showed the best performance for most scenarios. Differences in the results to those obtained in [11] are due to distinctive parameters used in this implementation.

To assess the proposed methods by [12] the focus of this evaluation is on the Salient-MSER detector and how it performs different from the MSER and the Salient region detector. The Salient region detector performs better on textured scenes than on structured scenes. For the MSER detector the opposite is true, its performance is better on structured scenes. These two results sound promising for the combined Salient-MSER detector. The Salient-MSER detector obtains slightly higher scores than MSER detector for structured scenes. But for the textured the performance is similar. In total the performance of the Salient-MSER detector is not significantly different from the MSER detector. Therefore the results do not show a substantial advantage of the combination of the two detectors.

In the descriptor evaluation, the SIFT descriptor is compared with other descriptors: GLOH, Shape Context, PCA, Moments, Cross correlation and Steerable filters. All descriptors are calculated from the feature regions that are detected by the proposed Salient-MSER detector [12]. The performance of all descriptors are relatively similar but nevertheless imply a ranking of the description methods. The best results are obtained by the SIFT description method. Results for the changing viewpoint scenario are shown exemplarily in Figure 6; the results in the other scenarios are qualitatively similar.

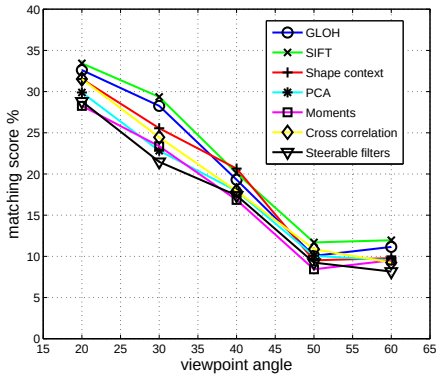

Fig. 6: The matching score for viewpoint change transformations for a structured graffiti scene. The descriptors were calculated from Salient-MSER regions.

4 Robot Experiments and Results

The loop closing application that is shortly described in Section 2.4 is integrated and tested on the existing 6D SLAM robot platform Kurt3D. The robot is described in Section 4.1, followed by the experimental setup and results.

4.1 The 6D SLAM Robot Platform

Kurt3D (Fig. 7) is a mobile robot, 45 cm (l) × 33 cm (w) × 29 cm (h) in size and a weight of 22.6 kg. Two 90 W motors are used to power the 6 skid-steered wheels. The core is a Linux P1400 with 768 MB RAM. Kurt3D is equipped with a 3D laser range finder based on a SICK 2D range finder, extended with a mount and servomotor. Through a controlled pitch motion, the area in front of the robot is scanned in 3D, in a stop-scan-go fashion. The acquired data serves as input for solving the 6D SLAM problem [15].

Fig. 7: The mobile robot Kurt3D.

4.2 Loop Closing Experiments

In the experiments, the robot was driven twice around a loop in our lab. In the matching process, the minimum number of similar feature descriptors was varied between 2 and 3. For each number, different thresholds for the Euclidean distance measure were tested. The number of generated loop hypotheses are reported for different thresholds in terms of true and false positives in Tables 1 and 2. A successful loop detection counts as true positive, whereas a wrong hypothesis counts as false positive. The ground truth information was provided manually, that means, an operator decided whether two images showed the same scene. Figure 8 shows two examples of successful loop detections.

Table 1. Results with at least 3 matched feature descriptors. The threshold for the distance is denoted with d, cf. step 3(c) in Section 2.4.

	$d = 250$		$d = 220$		$d = 200$	
# Images	True Pos.	False Pos.	True Pos.	False Pos.	True Pos.	False Pos.
53	41	67	27	21	18	5

Table 2. Results with at least 2 matched feature descriptors. The threshold for the distance is denoted with d, cf. step 3(c) in Section 2.4.

	$d = 200$		$d = 170$		$d = 150$	
# Images	True Pos.	False Pos.	True Pos.	False Pos.	True Pos.	False Pos.
53	30	30	17	3	8	1

Table 1 and 2 show the results for at least 3 and 2 similar matched feature descriptors, respectively. The minimal distances between the feature descriptors were varied in both runs. The number of processed images was 53. The ratio between true and false positives changes significantly for the tested thresholds. In general the characteristic of the numbers for the tested distances are similar. As the ratios between the true positives and false positives suggest the best performance is achieved for smaller distances – which is an expected effect.

5 Discussion and Conclusion

This paper has presented and evaluated the use of saliency for place recognition. A mobile robot was used to acquire images from its surroundings and to extract

Fig. 8. Examples for two true positive loop hypotheses.

visual features that have been encoded and matched. The results of the evaluation show that salient features alone work moderately and that the proposed Salient-MSER detector by [12] performs generally much like the MSER detector. Saliency and wide-baseline stability does not lead to a significant performance improvement.

Future work on loop closing will address the exclusive usage of maximally stable extremal regions or a combination of theses regions with other feature detectors, e.g., the Hessian-Affine. The implementation of loop closing proposed here is based on single visual features only. It is also possible to incorporate other methods such as object or landmark recognition to achieve a more robust performance.

References

1. D. M. Cole, A. R. Harrison, and P. M. Newman. Using Naturally Salient Regions for SLAM with 3D Laser Data. *IEEE ICRA*, 2005.
2. S. Frintrop, P. Jensfelt, and H. Christensen. Attentional Landmark Selection for Visual SLAM. In *Proc. IEEE/RSJ IROS*, October 2006.
3. K. Ho and P. Newman. Combining Visual and Spatial Appearance for Loop Closure Detection. *Proc. ECMR*, 2005.
4. T. Kadir and M. Brady. Saliency, Scale and Image Description. *Int. J. Comput. Vision*, 45(2), 2001.
5. T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *Proc. ECCV*, 2004.
6. D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Computer Vision*, 60(2):91–110, 2004.
7. J. Matas, O. Chum, U. Martin, and T. Pajdla. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In *Proc. BMVC*, 2002.
8. K. Mikolajczyk and C. Schmid. An Affine Invariant Interest Point Detector. In *Proc. ECCV*, 2002.
9. K. Mikolajczyk and C. Schmid. Comparison of affine-invariant local detectors and descriptors. In *Proc. 12th Europ. Signal Processing Conf.*, 2004.
10. K. Mikolajczyk and C. Schmid. A Performance Evaluation of Local Descriptors. *IEEE Transaction on PAMI*, 27(10):1615–1630, 2005.
11. K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A Comparison of Affine Region Detectors. *Int. J. Computer Vision*, 65(1-2):43–72, 2005.
12. P. M. Newman and K. L. Ho. SLAM - Loop Closing with Visually Salient Features. *IEEE ICRA*, 2005.
13. F. Schaffalitzky and A. Zisserman. Multi-view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?". In *Proc. ECCV*, 2002.
14. R. Sim, P. Elinas, M. Griffin, A. Shyr, and J. J. Little. Design and analysis of a framework for real-time vision-based SLAM using Rao-Blackwellised particle filters. In *Proc. of Canadian Conf. on Computer and Robotic Vision*, 2006.
15. H. Surmann, A. Nüchter, K. Lingemann, and J. Hertzberg. 6D SLAM - Preliminary Report on Closing The Loop in Six Dimensions. In *Proc. IAV*, 2004.
16. S. Thrun. Robotic Mapping: A Survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002.
17. Tinne Tuytelaars and Luc Van Gool. Matching Widely Separated Views Based on Affine Invariant Regions. *Int. J. Comput. Vision*, 59(1):61–85, 2004.