

Incremental topo-metric SLAM using vision and robot odometry

Stéphane Bazeille & David Filliat
ENSTA ParisTech, Unité Electronique et
Informatique, 32 Boulevard Victor, 75015 Paris,
FRANCE.

stephane.bazeille@ensta-paristech.fr,
david.filliat@ensta-paristech.fr

Abstract— We address the problem of simultaneous localization and mapping by combining visual loop-closure detection with metrical information given by the robot odometry. The proposed algorithm builds in real-time topo-metric maps of an unknown environment, with a monocular or omnidirectional camera and odometry gathered by motors encoders. A dedicated improved version of our previous work on purely appearance-based loop-closure detection [1] is used to extract potential loop-closure locations. Potential locations are then verified and classified using a new validation stage. The main contributions we bring are the generalization of the validation method for the use of monocular and omnidirectional camera with the removal of the camera calibration stage, the inclusion of an odometry-based evolution model in the Bayesian filter which improves accuracy and responsiveness, and the addition of a consistent metric position estimation. This new SLAM method does not require any calibration or learning stage (i.e. no *a priori* information about environment). It is therefore fully incremental and generates maps usable for global localization and planned navigation. This algorithm is moreover well suited for remote processing and can be used on toy robots with very small computational power.

Keywords: SLAM, uncalibrated camera, robot odometry, hybrid topo-metric map.

I. INTRODUCTION

To navigate in an unknown environment a robot requires the ability to build a map and to localize itself using a process named Simultaneous Localization And Mapping (SLAM) [29]. The field of SLAM can be broadly divided into topological and metrical approaches. The topological approach models the environment as a graph of discrete locations and often leads to simple solutions [11], [3]. It is often an easy to build map, suitable for many kinds of environment and for human interactions. Its main drawback is the lack of geometric information that only allows localization in previously mapped areas and local navigation with non optimal path planning. On the contrary, the metrical map is explicitly based on measured distances and positions [5]. The representation of the environment is geometric and clearly corresponds to the real world. The localization can be done continuously and planned navigation may be more precise. The major problem is to ensure geometry consistency between position and perceptions which makes the map hard to build. Number of approaches have attempted to capitalize on the advantages of the two representations (e.g., [19]). For instance, local metrical maps can be embedded into graphs

to enhance scalability [8]. Other graph-based solutions can be used to infer a precise metrical position for the robot, while still allowing for large scale mapping [17].

Without any prior information about the environment, and only using a monocular calibrated camera we have demonstrated that real-time topological SLAM is possible [1]. This method presents many advantages such as its simplicity, its speed, the lack of learning stage, and its efficiency (low false alarm rate), but the lack of metrical information makes the map ill-posed for navigation. In this article, we present an improved version of this method including the metrical information given by the robot odometry. Our new framework is calibration-free, incremental and real-time and allows to build hybrid topological-metrical maps usable for robot guidance. The odometric information is easy to acquire because often provided on robots. It also greatly complements the image data because it remains available notably in case of vision system failure (e.g. device problem, sensor occlusion, strong lighting change, dark areas).

In Section 2, we present related work on visual loop-closure detection, topological mapping and hybrid topo-metric mapping. In Section 3, we recall our previous work on the visual topological SLAM and we present our new framework on calibration free topo-metrical SLAM. In Section 4, we show experimental results and we conclude in Section 5 with a discussion and our future work.

II. RELATED WORK

In Simultaneous Localization And Mapping solutions, the traditional range and bearing sensors are now often replaced by camera (e.g. [2], [5]). It provides indeed many advantages such as smaller size, lighter weight, lower energy consumption, and above all a richer environmental information that is usable as the only environment information. Among all the approaches, we are more particularly interested in vision-based topological SLAM methods (e.g., [1], [4], [18]). The main idea of these approaches is to seek for the past images that look similar to the current one and consider they come from close viewpoints. This matching problem is called loop-closure detection. We have demonstrated in [1] a vision-based loop-closure detection method with a single monocular calibrated camera. The method uses Bayesian filtering, a simple voting scheme to estimate loop-closure likelihood and a multiple-view geometry stage to discard outliers.

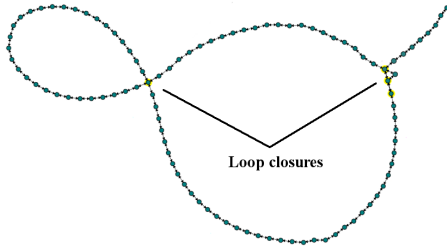


Fig. 1. Topological map (Museum sequence).

Given this framework, our objective is the integration of information to the topological map (Fig. 1) so as to obtain a map with which robot guidance is possible. The most appealing solution to this problem is probably the use of visual odometry, where images coming from neighboring nodes or image sequences taken between nodes are matched to estimate the robot displacement [11], [23], [17], [28]. Instead of estimating node positions, another solution is to use visual servoing, also known as vision-based robot control which uses feedback information extracted from a vision sensor to control the motion of a robot [6]. The robot can then be directly guided to the neighboring nodes without explicitly computing their relative positions. The advantage of these two approaches is to maintain the use of the vision sensor only but they require a lot of processing and are not robust in dark or over-exposed areas for example. So, like several authors [10], [25], we have chosen to use the information given by robot odometry. It adds another sensor but it greatly reduces processing requirement and complements visual sensor in case of perception failure.

Odometry is often used on robots, whether they be legged or wheeled, to estimate their position relative to a starting location. The major problem with robot odometry is a cumulative error due to the integration of noisy measurements that makes the position estimate more and more inaccurate over time. As a consequence, long term use of odometry requires complementary information to enable a correction of this cumulative errors and to produce a consistent topometric map. As loop-closure detection provides a reliable information about position we can apply a relaxation algorithm to estimate the position of nodes that best satisfied loop-closure constraints. Several relaxation methods exist to deal with this problem [16], [7], [12]. Also, recent solutions are very interesting and efficient [24], [15] to solve the particular graph-based formulation of SLAM problem in which the poses of the robots are modeled by nodes in a graph, and constraints between poses resulting from odometry are encoded in the edges.

Visual SLAM approaches have used either standard perspective [5],[4] or omnidirectional ([14], [3]) camera as input. Omnidirectional cameras are interesting for SLAM applications because they give a richer information to characterize scenes as a single image can visually describe the position of the robot regardless of its orientation. For example, omnidirectional camera allow loop-closure detections

when the robot take a previous path in the opposite direction. Many approaches rely on a particular type of camera with its associated calibration [5],[27], while some approaches can be used with uncalibrated camera [26], [13] which greatly simplifies the application to different robots. Our approach is generic as it is calibration-free and as it can use either perspective or omnidirectional camera as input.

III. LIMITATIONS OF LOOP-CLOSURE DETECTION FOR TOPO-METRIC MAPPING

Building globally coherent and locally precise topo-metric maps using robot odometry and graph relaxation requires a robust and reliable loop-closure detection algorithm. In particular, we need a metrically accurate and responsive detection, and the strict absence of false alarm which would lead to globally incoherent maps. Our previous approach on visual loop-closure detection [1] presented some limitations to be used in this framework.

A first limitation is in the definition of loop-closure by itself. In our original approach, as in others [4], loop-closure were validated using multiple-view geometry between the matching images. This policy define loop-closure by the fact that the robot sees the same scene, but does not enforce that the robot positions are very close, and scale ambiguity in multiple-view geometry prevents to recover the real robot displacement. In order to include the loop-closure constraints in a graph relaxation algorithm, we therefore chose to devise a more constrained loop-closure validation stage to only accept loop closure detections with very close robot positions (see section IV-E). While this could appear as a strong constraint, it is in fact quite natural for robots in indoor environments as robots often pass very close to the same positions, for example when crossing doors (e.g., Fig. 7).

A second limitation of our approach is the lack of temporal consistency of loop-closure detection. For example on Fig. 1 the loop-closure detected by the original algorithm were between images 49-80, 9-102, 10-104, 9-105, 9-106 while the ground truth were between images 47-78, 48-79, 49-80, 5-102, 6-103, 7-104. While these loop-closure detection are visually correct according to our previous definition, two major problems can be seen on this example: first the detection are not temporally consistent (10-104, 9-105) and also, successive loop-closure on the same node are detected (9-105, 9-106). This is normally impossible because the images are only processed when the robot has moved for a given distance. Including these inconsistencies, the graph relaxation would lead to large errors in the optimized map. This problem has been solved by the inclusion of an odometry-based evolution model (see section IV-D) that bias loop-closure detection to follow the robot trajectory temporal evolution.

Finally, the responsiveness of the algorithm was too low for complex indoor trajectories where loop-closure can only be detected during one or two images. As an example in indoor environments, common trajectories are mostly seen when a door is crossed for a very short distance. In these

cases, the previous approach leads to many missed loop-closure and provides very few constraints for map relaxation and odometry correction. This problem has also been solved through the odometry-based evolution model.

IV. OVERVIEW OF THE APPROACH

In our previous work [1] we have developed a real-time vision-based topological SLAM framework. This method is fully incremental (i.e. the system can be used without any *a priori* information about the environment), and only uses appearance information from a single calibrated camera. The environment model is learned on-line, as the robot discovers its surroundings.

A. The Bayesian filtering using bags of visual words

To solve the image-to-node matching problem based on a similarity measure between the current image and the images of a node previously visited, we choose to use a maximum *a posteriori* scheme which exploits the similarity of image sequences to ensure the temporal consistency and reduce false alarms (e.g. [21]), instead of the more common maximum likelihood which only consider the current image for matching (e.g. [3]). A short overview of the Bayesian filtering framework is provided here for clarity.

The method searches for the node N_i of the map that is the more similar to the current image I_t , in other words, it searches for the node N_i that maximizes the probability of loop-closure with the current image:

$$N_i = \underset{i=0, \dots, n}{\operatorname{argmax}} p(S_t = i | I_t, M) \quad (1)$$

where $S_t = i$ is event “ I_t comes from N_i ” and $M = N_0, \dots, N_n$ is the map. Bayes rule, marginalization and Markov assumption [1] lead to the incremental computation of the *a posteriori* probability as follow:

$$p(S_t | I_t, M) = \eta \cdot \underbrace{p(I_t | S_t, M)}_{\text{likelihood model}} \cdot \underbrace{\sum_{j=0}^n \underbrace{p(S_t | S_{t-1} = j, M)}_{\text{transition model}} \underbrace{p(S_{t-1} = j | I_{t-1}, M)}_{\text{a priori probability}}}_{\text{prediction}}$$

In this equation, the prediction is computed using the *a priori* probability (i.e. the probability at the previous time step) multiplied by an evolution model $p(S_t | S_{t-1} = j, M)$ diffusing the probability of a node to its neighbors to take into account the robot motion since the last localization. Then, the result of this computation called prediction is multiplied by the likelihood (number of correspondences between images through a voting scheme) to obtain the *a posteriori* probability. The likelihood model is computed using a representation of images as a set of unordered SIFT features [20] taken from a dictionary (i.e. the bags of visual words model [9]). An inverted index makes it possible to very efficiently compute this likelihood in time linear with the number of visual words of the current image. To discard outliers, we use a multiple-view geometry stage as in [18]. The posterior probabilities above a threshold are first sorted

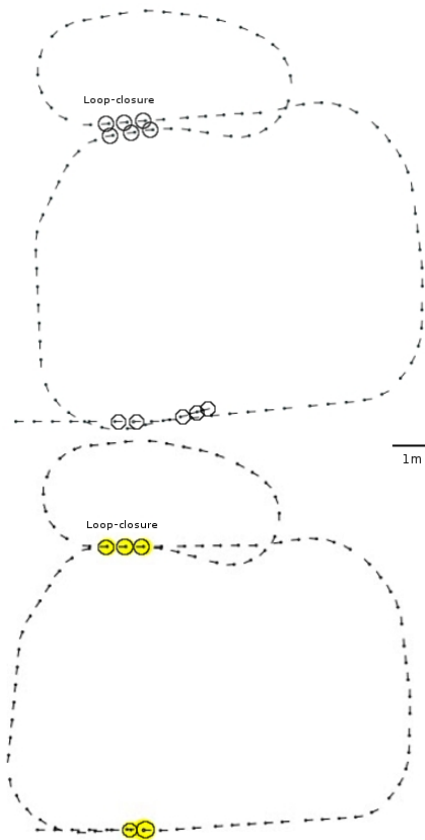


Fig. 2. Topo-metric maps (Museum sequence). (Top) the raw odometry, the loop-closure detections are highlighted with black circles, (Bottom) corrected odometry after relaxation.

and then the essential matrix between the two loop-closure images [22] are computed in the descending order.

B. The addition of robot odometry information

We did four main modifications in order to incorporate the odometry information in the map (a complete processing diagram of the new algorithm is shown Fig. 3):

- images are now acquired with the odometry when the robot has moved for a given distance or turned of a given angle. This image acquisition policy enforces a more regular sampling of positions in the environment, independent of the robot velocity and also reduces the computational burden of the algorithm when the robot is not moving.
- the Gaussian evolution model was not precise enough and has been replaced by an odometry-based evolution model. Through a probabilistic model of odometry, the evolution model can take into account not only the nodes topological proximity, but also their relative position.
- the acceptance step verifying epipolar geometry of the most probable loop-closure has been replaced by a validation stage which select the loop-closure that present the smallest 2D motion in images among all the hypothesis whose probability is above a threshold.

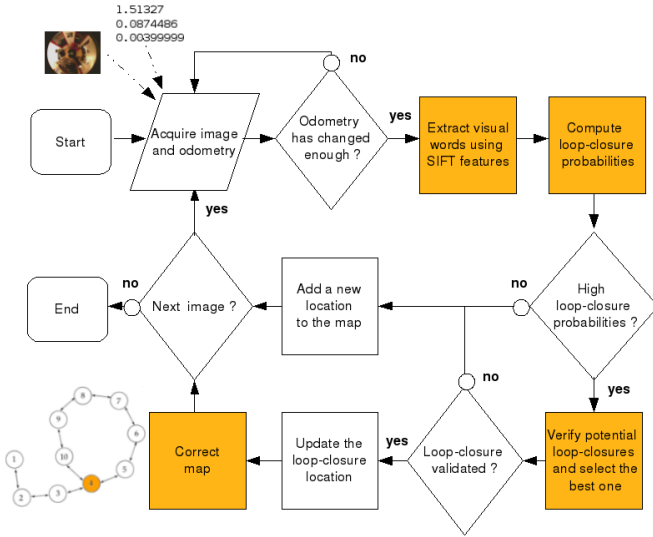


Fig. 3. Processing diagram of the topo-metric map building.

- the relative position between nodes is saved on each link of the graph and a fast relaxation algorithm is applied each time a loop-closure is detected.

Beside the capacity to guide the robot because the map is more accurate and incorporate consistent geometric information, the addition of this second sensor makes also the system more robust notably in case of vision system failure (i.e. device problem, sensor occlusion, strong lighting change, dark areas).

C. A new map and a relaxation algorithm

The topological map is constituted of a set of nodes associated with an image and linked by edges. We have integrated metrical information in two forms in order to produce a topo-metric map. First, each node is associated with an absolute pose in the map (x, y, θ) , where x and y are the 2D position coordinates and θ an angle representing the direction of the robot when the image was taken. Secondly, the edges are associated with a relative position between two nodes defined by (d, α, ϕ) , where d and α are the polar coordinates of the second node in the coordinate space of the first, and ϕ is the difference angle between the two nodes direction.

As shown on Fig. 2 and 7 (Top) the geometric consistency of the map deteriorates over time. When a loop-closure is detected the robot is assumed to have returned at the position of a previous passing. By constraining two nodes to have the same position (Fig. 2, Bottom) we can correct the odometry drift. To do so, we choose to apply a relaxation algorithm called Tree-based network optimizer (TORO) [15], because of its speed and its high efficiency. It is called when loop-closure is found to estimate the consistent node configuration which maximally satisfy the odometry constraints between nodes. This algorithm is really fast to optimize the kind of map we use which are very simple in regards to constraints. As an example it takes less than 3 seconds for 10000 nodes, 800 constraints and 100 iterations.

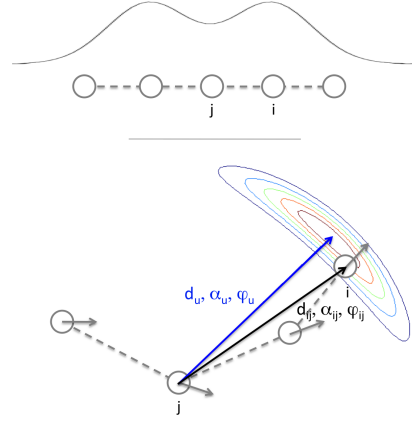


Fig. 4. Top: the original evolution model, assuming only temporal consistency of positions. Bottom: the new evolution model including odometry and relative node positions (see text for details).

D. An evolution model using odometry information

In the original framework, the evolution model used to obtain the prediction given the *a priori* probability applied a diffusion of the probability over the neighboring locations in the graph. The weight was defined as a sum of Gaussian centered on the current location (Fig. 4, Top). This diffusion was done in all directions without preference, because it only assumes that the neighboring images in time are close together, without any information about the real robot motions. Assuming such a model, the loop-closure probability tends to spread out, the system loose some responsiveness and results are not temporally consistent. Because a reliable metrical information is now available, we integrated odometry in the evolution model to predict more precisely the evolution of the probability and therefore to enhance the reactivity and above all the accuracy of loop-closure detection. Thus, starting from a given node, we distribute the probability to each neighboring location in the map depending on the deviation of these nodes relative positions with the robot displacement since the last update d_u, α_u, ϕ_u measured by odometry (Fig. 2, Bottom).

We used the standard motion model for robot odometry [29], assuming Gaussian noise on the robot displacement measured in polar coordinates:

$$p(d, \alpha, \phi | d_u, \alpha_u, \phi_u) = G_{\mu_d, \sigma_d}(d - d_u) G_{\mu_\theta, \sigma_\theta}(\alpha - \alpha_u) G_{\mu_\phi, \sigma_\phi}(\phi - \phi_u)$$

where d, α gives the odometry displacement in polar coordinates in the frame of the previous robot position and ϕ is the variation of robot direction during movement. $G_{\mu, \sigma}(X)$ is the Gaussian distribution of mean μ and variance σ^2 .

The evolution model becomes: $p(S_i | S_j, u_t, M) =$

$$G_{\mu_d, \sigma_d}(d_{ij} - d_u) G_{\mu_\theta, \sigma_\theta}(\theta_{ij} - \theta_u) G_{\mu_\phi, \sigma_\phi}(\phi_{ij} - \phi_u)$$

where $u_t = d_u, \theta_u, \phi_u$ gives the odometry displacement and $d_{ij}, \theta_{ij}, \phi_{ij}$ is the relative position between nodes i

and j . The substitution makes the prediction of the a *posteriori* probability more accurate, improving robustness and responsiveness of the algorithm. The original algorithm required several corresponding frames before detection, and sometimes gives some bad results (consecutive loop-closure on the same node). By including the use of odometry two less images are required to detect, and the temporal consistency of the detection is well improved.

E. Loop-closure validation

In our previous work we verified the epipolar constraint by computing the essential matrix for the loop-closure candidate showing the higher probability. If the loop-closure was not validated, verification follows in the descending order of loop-closure probability. This strategy gives confidence in probability results and is not robust enough for our use: often the loop-closure which shows the highest probability is visually correct but is not the one corresponding to the smallest robot displacement. To find the previous position the closest to the current one we choose to use the results of the Bayesian filter as a first step to select potential loop-closure locations. On a next step, we verify all the hypothesis above a threshold with a 2D motion computation based on the SIFT keypoints and we select the loop-closure which shows the smallest translation. In order to discard outliers, the 2D motion (translation and rotation in image plane) is computed using RANSAC, accepting the result only if the number of matching points is above a threshold (30).

This new validation stage makes the method calibration free, and can be used with perspective or omnidirectional camera. According to the model of camera, we can distinguish two different cases for the graph constraints:

- Perspective camera: we assume the robot has returned to the same position in the same direction so we constrain the two positions to be the same.
- Omnidirectional camera: the robot has returned to the same position but with a different direction. In the relaxation algorithm we constrain the nodes to have the same position with a difference between the two directions that is equal to the angle between the two omnidirectional views.

This validation is more accurate but also more restrictive compared to the previous one because it does not accept images if the motion between the two is too important. This method therefore requires that the robot come back close to a previously visited locations, which is well suited for indoor environments where doors for example are a required route to go from one room to an other. Outdoor environment have also been tested successfully but usually provide less loop-closure detection as the path are often less constrained.

V. RESULTS AND DISCUSSION

To demonstrate the quality of the approach we have used data acquired with a Pioneer 3 DX mobile robot and some data set used by the SLAM community (omnidirectional



Fig. 5. Examples of loop-closure accepted at a door crossing in Gostai sequence. On the left, two matching images accepted by the epipolar validation module (more than 1 meter between position), on the middle and on the right, images accepted using the new validation stage. The new method is more restrictive but ensure accuracy of detection and therefore a quality of the resulting topo-metric map.

	75	76	77	78	79	80	81	82
44								
45								
46					1			
47				2				
48					2	1	1	
49						2		
50								
51								

	91	92	93	94	95	96	97	98
44								
45	2							
46		2						
47			1	2	1			
48				2				
49					1	2	2	
50						1	2	
51								1
52								2

Fig. 6. Comparing loop-closure detection results with the two method (Gostai sequence). In red (1) using the previous approach with diffusion transition model and epipolar check without threshold. In Yellow (2) using the proposed approach with odometry transition model and the new validation check (see text for details).

home sequence¹, and Oxford city center sequence²). Concerning our data set, the robot was guided to do some loops in indoor environments showing strong perceptual aliasing conditions. The images and the odometry information were taken each time the robot moves at least 25 cm or turn of at least 10 degrees. This sampling rate is largely enough to describe the environment without saving too much redundant information, it corresponds to an average acquisition time of one image each 0.7s and an average speed of the robot about 0.4 m/s. The computer used for experimentation was an Intel Xeon 3Gh, and the images size 320x240. The path of the experiments (for the sequences providing odometry) and the resulting corrected maps are shown on Fig. 2, 7, 8, and 9.

Table I shows the improvements obtained with the new approach (LCD New) over the previous one (LCD Old). We can see that the detection rate is improved and that the false alarms are nearly suppressed. Also, our improvements produce precise loop-closure detection that always lead to maps consistent with the real world. Even if some false alarms exist, they are produced by very close location, leading to maps with the correct global topology but with local drift in some situations such as very long corridors (Fig. 9). The last sequence “city” which was outdoor presents

¹<http://www2.science.uva.nl/sites/cogniron/fs2hsc/Data/Home2/run1> [30]

²Oxford Mobile Robotics Group. City center [4]

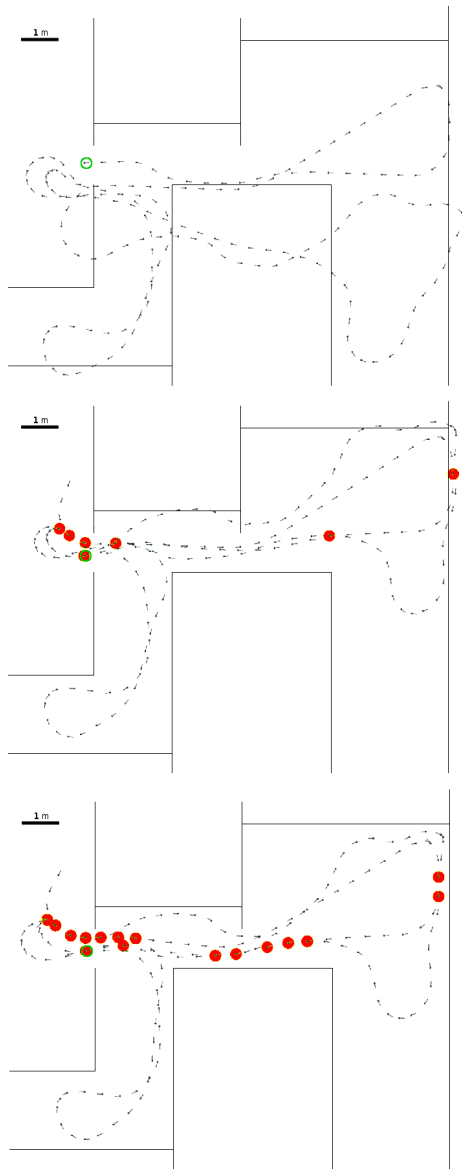


Fig. 7. Example of topo-metric mapping result (sequence Office). (a) raw odometry, (b) map using the Gaussian sum transition model and the epipolar geometry. (c) results using the odometry transition model and the new validation step. The landmark is the starting location just in front of the first door.

TABLE I

SUMMARY OF LOOP-CLOSURE DETECTION ON DIFFERENT SEQUENCES.

	Museum	Gostai	Lab	Home	City
Images	112	169	350	1400	1237
Odometry	Yes	Yes	Yes	Yes	No
Omni	No	No	No	Yes	No
CPU Time	42s	70s	208s	22min	18min
CPU Time/image	0.37s	0.41s	0.59s	0.94s	0.87s
Figure	1-2	7	9	-	-
LCD Truth	14	25	9	≈300	≈500
LCD Old [1]	2	7	2	229	-
Missed	64 %	52 %	55 %	23 %	-
False alarm	0 %	28 %	22 %	-	-
LCD New	14	18	7	256	84
Missed	0 %	12 %	20 %	14 %	79 %
False alarm	0 %	5 %	0 %	-	-

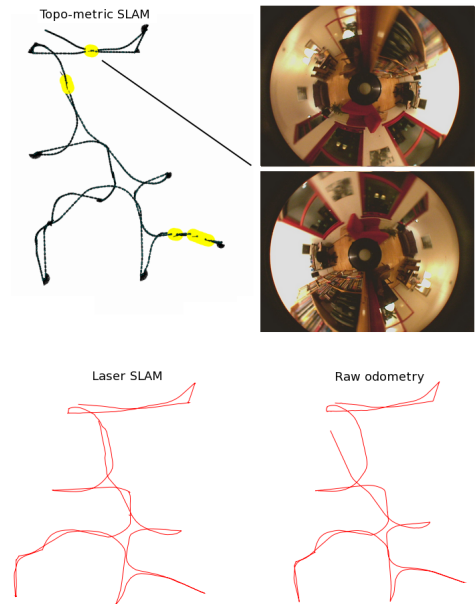


Fig. 8. Example of results of topo-metric mapping using an omnidirectional camera (sequence Home).

lower detection results mainly because the odometry was not available. Figure 6 shows some loop-closure details on the Office sequence to highlight differences between the two models. Notably, we can see that the detection rate is more important: only two consecutive similar frames are now required before effective loop-closure detection and the first validated loop-closure with the new model comes two images earlier. This figure also illustrates that the temporal consistency of the detections is recovered; there is no more gap between images and multiple loop-closure detection on the same node are discarded.

Concerning computation times, the duration of the mission “Lab” was 241 seconds and the overall processing time was 208 seconds. Extrapolating these data, we can stay on real time processing for environments up to 1400 images. It has to be noted that SURF could be used in order to replace SIFT which will divide by 2 the keypoints extraction time (SIFT 190ms, SURF 85ms for our 320x240 images). Using SURF would therefore lead to a 15% computation time reduction as the time for SIFT extractions is about 30% of the total time.

VI. CONCLUSION AND FUTURE WORK

We have introduced in this paper a system that is able to build an hybrid topo-metric map in real-time without any camera calibration or learning stage. The developed framework combines vision-based loop-closure detection with metrical information given by the robot odometry. The odometry used to give a metrical position to each location is also used in the evolution model of the Bayesian filter to make the detection more accurate and responsive. The proposed solution is fast and robust with the use of our new validation stage which allows the use of any kind of camera (perspective or omnidirectional). The geometrical consistency between robot motion and visual perception is



Fig. 9. Example of a topo-metric map included in a laser map for reference (sequence Lab). (a) raw odometry, (b) trajectory corrected using our algorithm. The length of the trajectory was about 96 meters.

regain each time a loop-closure is detected with the use of a fast optimization algorithm. The generated hybrid topo-metric map is usable for robot guidance.

Our future work will be to optimize visual processing to further reduce computational cost and to apply this framework on mobile toy robots using wireless remote processing. The approach is well suited for this as robot guidance can be performed on-board using odometry, while remotely and asynchronously detecting loop-closure and optimizing maps for odometry correction. We are also planning to develop autonomous exploration algorithms taking into account the necessity to detect loop closure in order to be able to map large-scale environments.

REFERENCES

- [1] A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer, "A fast and incremental method for loop-closure detection using bags of visual words," *IEEE Transactions On Robotics, Special Issue on Visual SLAM*, vol. 24, no. 5, pp. 1027–1037, October 2008.
- [2] T. Bailey and H. Durrant-Whyte, "Simultaneous localisation and mapping (slam): Part ii," *IEEE Robotics and Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.
- [3] O. Booij, B. Terwijn, Z. Zivkovic, and B. Kröse, "Navigation using an appearance based topological map," in *IEEE International Conference on Robotics and Automation*, 2007.
- [4] M. Cummins and P. Newman, "Accelerated appearance-only slam," in *IEEE Conference on Robotics and Automation*, 2008.
- [5] A. Davison, I. Reid, N. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, June 2007.
- [6] A. Diosi, A. Remazeilles, S. Segvic, and F. Chaumette, "Outdoor visual path following experiments," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'07*, 2007.
- [7] T. Duckett, S. Marsland, and J. Shapiro, "Fast, on-line learning of globally consistent maps," *Autonomous Robots*, vol. 12, no. 3, pp. 287–300, 2002.
- [8] E. Eade and T. Drummond, "Monocular slam as a graph of coalesced observations," in *International Conference on Computer Vision*, 2007.
- [9] D. Filliat, "A visual bag of words method for interactive qualitative localization and mapping," in *IEEE International Conference on Robotics and Automation*, 2007.
- [10] D. Filliat and J. A. Meyer, "Global localization and topological map learning for robot navigation," in *From Animals to Animats 7. The Seventh International Conference on simulation of adaptive behavior (SAB02)*, 2002.
- [11] F. Fraundorfer, C. Engels, and D. Nistér, "Topological mapping, localization and navigation using image collections," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007.
- [12] U. Frese, P. Larsson, and T. Duckett, "A multilevel relaxation algorithm for simultaneous localization and mapping," *IEEE Transactions on Robotics and Automation*, vol. 21, no. 2, pp. 196–207, 2005.
- [13] S. Fu and G. Yang, "Uncalibrated monocular based simultaneous localization and mapping for indoor autonomous mobile robot navigation," in *Networking, Sensing and Control, 2009. ICNSC '09. International Conference on*, 2009, pp. 663 – 668.
- [14] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. V. Gool, "Omnidirectional vision based topological navigation," *International Journal of Computer Vision*, vol. 74, no. 3, pp. 219–236, 2007.
- [15] G. Grisetti, C. Stachniss, S. Grzonka, and W. Burgard, "A tree parameterization for efficiently computing maximum likelihood maps using gradient descent," in *Proceedings of Robotics: Science and Systems*, Atlanta, GA, USA, June 2007.
- [16] A. Howard, M. J. Mataric, and G. Sukhatme, "Relaxation on a mesh: a formalism for generalized localization," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2001, pp. 1055–1060.
- [17] K. Konolige and M. Agrawal, "Frameslam: From bundle adjustment to real-time visual mapping," *IEEE Transaction on Robotics*, vol. 24, no. 5, pp. 1066–1077, 2008.
- [18] J. Kosecká, F. Li, and X. Yang, "Global localization and relative positioning based on scale-invariant keypoints," *Robotics and Autonomous Systems*, vol. 52, pp. 209–228, 2005.
- [19] K. Kouzoubov and D. Austin, "Hybrid topologicavmetric approach to slam," in *In proceedings of the 2004 IEEE International Conference on Robotics and Automation*, April 2004.
- [20] D. Lowe, "Distinctive image feature from scale-invariant keypoint," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [21] E. Menegatti, M. Zoccarato, E. Pagello, and H. Ishiguro, "Image-based monte-carlo localisation with omnidirectional images," *Robotics and Autonomous Systems*, vol. 48, no. 1, pp. 17–30, 2004.
- [22] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 756–777, 2004.
- [23] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry for ground vehicle applications," *Journal of Field Robotics*, vol. 23, no. 1, pp. –, 2006.
- [24] E. Olson, J. Leonard, and S. Teller, "Fast iterative alignment of pose graphs with poor initial estimates," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, 2006, pp. 2262–2269.
- [25] P. Rybski, F. Zacharias, J. Lett, O. Masoud, M. Gini, and N. Papanikolopoulos, "Using visual features to build topological maps of indoor environments," in *IEEE International Conference on Robotics and Automation*, 2003.
- [26] M. Saedan, L. Chee Wang, and M. Ang, "Appearance-based slam with map loop closing using an omnidirectional camera," in *Advanced intelligent mechatronics, 2007 IEEE/ASME international conference on*, 2007, pp. 1 – 6.
- [27] D. Scaramuzza and R. Siegwart, "Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles," *IEEE transactions on robotics*, vol. 24, 2008.
- [28] G. Sibley, C. Mei, I. Reid, and P. Newman, "Adaptive relative bundle adjustment," in *Robotics Science and Systems (RSS)*, Seattle, USA, June 2009.
- [29] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. The MIT Press, 2005.
- [30] Z. Zivkovic, O. Booij, and B. Kröse, "From images to rooms," *Robotic and Autonomous Systems*, vol. 55, pp. 411–418, 2007.