Real-Time Visual Loop-Closure Detection

Adrien Angeli, Stéphane Doncieux, Jean-Arcady Meyer, David Filliat

Real-Time Visual Loop-Closure Detection

Adrien Angeli, Stéphane Doncieux,
Jean-Arcady Meyer
Université Pierre et Marie Curie - Paris 6
FRE 2507, ISIR, 4 place Jussieu, F-75005
Paris, France.
firstname.lastname@isir.fr

David Filliat
ENSTA
32, bvd Victor, F-75015 Paris, France.
david.filliat@ensta.fr

***Abstract*— In robotic applications of visual simultaneous localization and mapping, loop-closure detection and global localization are two issues that require the capacity to recognize a previously visited place from current camera measurements. We present an online method that makes it possible to detect when an image comes from an already perceived scene using local shape information. Our approach extends the bag of visual words method used in image recognition to incremental conditions and relies on Bayesian filtering to estimate loop-closure probability. We demonstrate the efficiency of our solution by real-time loop-closure detection under strong perceptual aliasing conditions in an indoor image sequence taken with a handheld camera.**

I. Introduction

Over the last decade, the increase in computing power helped supplementing traditional approaches to simultaneous localization and mapping (SLAM [1], [2]) issues with the qualitative information provided by vision. As a consequence, commonly used range and bearing sensors (i.e. lasers, radars and sonars) tend to be associated with, or replaced by, single cameras or stereo-camera rigs. For example, in a previous work [3], we performed vision-based 2D SLAM for Unmanned Aerial Vehicles (UAV). Likewise, in [4], the authors perform 3D SLAM in real-time at 30Hz using a monocular handheld camera.

However, there are still difficulties to overcome in robotic vision in general and in SLAM applications in particular. Among them, the loop-closure detection issue concerns the difficulty of recognizing already mapped areas, while the global localization issue concerns the difficulty of retrieving the robot's location in an existing map. Also, the *kidnapped robot problem* consists in recovering the robot's position after an arbitrary "blind" displacement (i.e. without knowledge of the displacement), which is what may occur in case of temporary camera dysfunction or occlusion. Those problems can be addressed by detecting when the robot is navigating through a previously visited place from local measurements. The overall goal of the research effort introduced in this article is thus to design a vision-based framework tackling this issue to make it possible for a robot to reinitialize a visual 3D-SLAM algorithm like the one presented in [4] in the situations described above. This comes down to an online image retrieval task that consists in determining if current camera measurements match past ones. Such task bears strong similarities with image classification methods like those described in [5] and [6], but an important difference is our commitment to online processing.

In this paper, we present a real-time scene recognition framework that relies on an incremental version [7] of the bag of visual words method [8] for image comparison. We estimate the probability of loop-closure detection in a Bayesian filtering scheme that helps discarding false recognitions by enforcing temporal coherency. When the probability that the current image and a past one come from close viewpoints is above some threshold, epipolar geometry [9] is used to validate the loop-closure hypothesis.

In section 2, we present a review of related work on visual loop-closure and global localization, section 3 briefly introduces our visual bag of words implementation. The filtering scheme is detailed in section 4 and experimental results are given in section 5. The last two sections are devoted to discussion and conclusion.

II. Related Work

The Monte Carlo Localization (MCL) method [10], recently adapted to vision ([11]), makes global localization possible in a non-incremental perspective (i.e. a map is required). The Rao-Blackwellised particle filter enables loop-closure capabilities in SLAM algorithms (e.g the FastSLAM [12] framework), even when using bearing-only sensors ([13]). However, to perform well, an exponential number of particles is needed (which is intractable in large scale environments) and inaccurate resampling policies lead to degeneration when closing a loop.

In this paper, we wish to design a simple visual system able to perform loop-closure detection and global localization, within the framework of an online image retrieval task. Following a similar approach, but in a non-incremental perspective, voting methods presented in [14] and [15] call upon maximum likelihood estimation to match the current image with a database of images acquired beforehand. The likelihood depends upon the number of feature correspondences between the images, and leads to a vote assessing the amount of similarity. In [14], the authors also use multiple-view geometry to validate each matching hypothesis, while in [15] the accuracy of the likelihood is qualitatively evaluated in order to reject outliers. Even though they are easy to implement, voting methods rely on an offline construction of the image database and need expensive one-to-one image

comparisons when searching for the most likely hypotheses. Moreover, the maximum likelihood criterion does not provide a suitable framework to manage multiple hypotheses over time and is thus prone to transient detection errors.

In [16] and [17], bag of words methods are used to perform global localization and loop-closure detection in an image classification scheme. Bag of words methods ([5], [6], [8]) rely on a representation of images as a set of unordered elementary features (the visual words) taken from a dictionary (or codebook). The dictionary is built by clustering similar visual descriptors extracted from the images into visual words. Using a given dictionary, image classification is based on the frequencies of the words in an image for example to infer its class ([8]). In [16] and [17], images are represented as vectors of visual words' statistics taken from an offline-built visual vocabulary. The size of the vectors is equal to the number of words in the dictionary. Matching between current and past images is defined as a Nearest Neighbor (NN) search among the cosine distances separating the corresponding vectors. In [16], a simple voting scheme selects the *n* best candidates from the NN search and multiple-view geometry discards outliers. In [17], the NN search results fill a *similarity matrix* whose off-diagonal elements correspond to loop-closure events, providing a powerful way to manage multiple hypotheses over time. In both approaches, the use of a dictionary enhances the robustness of the matching, enabling a good tolerance to image noise, but the NN search involved, relying on exhaustive one-to-one vector comparisons, is computationally expensive.

More recently, the authors of [18] proposed a vision-based probabilistic framework for the estimation of the probability that two observations originate from the same location. This approach, based on the bag of words scheme, is very robust to perceptual aliasing: a generative model of appearance is learned in an offline process, approximating the probabilities of co-occurrences of the words contained in the offline-built dictionary. The main asset of this model is its ability to evaluate the distinctiveness of each word, thus accounting for perceptual aliasing at the word level, while its principal drawback lies in the offline process needed for model learning and dictionary computation.

In the majority of the methods presented above, SIFT (Scale Invariant Feature Transform [19]) keypoints are the preferred input information, notably for their robustness to affine transformations.

III. Visual Dictionary

In the implementation of the bag of words method [7] used here, dictionary construction is performed online, in an incremental fashion, using a tree structure to allow logarithmic-time complexity in the number of words during the matching step (the description of this structure is beyond the scope of this paper). In the work reported here, images are described using SIFT keypoints [19]: interest points are detected as maxima over scale and space in differences of Gaussians convolutions. The keypoints are memorized as histograms of gradient orientations around the detected point at the detected scale. The corresponding descriptors are of dimension 128 and are compared using L2 distance.

IV. Bayesian Loop-Closure Detection

In this paper, we address the problem of loop-closure detection as an image retrieval task, using Bayesian filtering to ensure temporal coherency and reduce the effects of transient detection errors. Let S_t be the random variable representing loop-closure hypotheses at time t: $S_t = i$ is the event that current image I_t "closes the loop" with past image I_i. This implies that the corresponding viewpoints x_t and x_i are close, and that I_t and I_i share some similarities. The event $S_t = -1$ is the event that no loop-closure occurred at time t. In a probabilisitc Bayesian framework, the loop-closure detection problem can hence be formulated as searching for the past image I_j whose index satisfies:

$$j = \text{argmax}_{i=-1,\ldots,t-p} p(S_t = i | I^t) \tag{1}$$

where $I^t = I_0, \ldots, I_t$, with $j = -1$ if no loop-closure has been detected. This search is not performed over the last p images because I_t always looks similar to its neighbors in time (since they come from close locations), and doing so would result in loop-closure detections between I_t and recently seen images (i.e. $I_{t-1}, I_{t-2}, \ldots, I_{t-(p+1)}$). This parameter, set to 10 in our experiments, is adjusted depending on the frame rate and on the velocity of camera motion.

We therefore need to estimate the *full posterior*, $p(S_t|I^t)$ for all $i = -1, \ldots, t-p$, which is a probability density function (i.e. $p(S_t \geq -1|I^t) = 1$), in order to find, if a loop-closure occurred, the corresponding past image.

Following Bayes' rule and under the Markov assumption the posterior can be decomposed into:

$$p\big(S_t|I^t\big) = \eta p\big(I_t|S_t\big) p\big(S_t|I^{t-1}\big) \tag{2}$$

where η is the normalization term. Let Z_i be the state of the dictionary at time index i. The time subscript i is inherent to the incremental aspect of the vocabulary construction: $Z_0 \subseteq Z_1 \subseteq \ldots \subseteq Z_i$, with $Z_0 = \emptyset$ (SIFT features extracted in I_i are used to build Z_{i+1}). Also, let the subset z_i of words taken from Z_i and found in image I_i denote the representation of this image in the SIFT feature space. The sequence of images I^t can therefore be represented by the sequence $z^t = z_0, \ldots, z_t$.

So, the full posterior, now rewritten $p\big(S_t|z^t\big)$, can be expressed as follows:

$$p\big(S_t|z^t\big) = \eta p\big(z_t|S_t\big) p\big(S_t|z^{t-1}\big) \tag{3}$$

where $p\big(z_t|S_t\big)$ is the likelihood $\mathcal{L}\left(S_t|z_t\right)$ of S_t given the words z_t (see section IV-B). Finally, by marginalizing the right hand side of equation 3 we obtain:

$$p\big(S_t|z^t\big) = \eta p\big(z_t|S_t\big) \sum_{j=-1}^{t-p} p\big(S_t|S_{t-1} = j\big) p\big(S_{t-1} = j|z^{t-1}\big) \tag{4}$$

where $p(S_t|S_{t-1})$ is a time evolution model of the pdf (see section IV-A). Following equation 4, the full posterior can be obtained as a product of the likelihood with the full posterior calculated one step before and summed over all possible transitions between time $t-1$ and t. Note that in our framework, the sequence of words z^t evolve in time with the acquisition of new images, diverging from the classical Bayesian framework where such sequences would be fixed.

A. Transition from $t-1$ to t

Between $t-1$ and t, the full posterior is updated according to the time evolution model of the pdf, $p(S_t|S_{t-1}=j)$, which gives the probability of transition from one state j at time $t-1$ to every possible state at time t. This enforces the temporal coherency of the estimation, limiting transient detection errors. Depending on the respective values of S_t and S_{t-1}, this probability takes one of the following values:

- $p(S_t=-1|S_{t-1}=-1)=0.9$, the probability that no loop-closure event will occure at time t is high given that none occurred at time $t-1$.
- $p(S_t=i|S_{t-1}=-1)=\frac{0.1}{(t-p)+1}$ with $i\in[0;t-p]$, the probability of a loop-closure event at time t is low given that none occurred at time $t-1$.
- $p(S_t=-1|S_{t-1}=j)=0.1$ with $j\in[0;t-p]$, the probability of the event "no loop-closure at time t" is low given that a loop-closure occurred at time $t-1$.
- $p(S_t=i|S_{t-1}=j)$, with $i,j\in[0;t-p]$, it is a Gaussian on the distance between i and j whose sigma value is chosen so that it is non zero for exactly 4 neighbors (i.e. $i=j-2\dots j+2$). The size of this neighborhood is adjusted depending on the frame rate and on the velocity of camera motion. This corresponds to a diffusion of the posterior in order to account for the similarities between neighboring images.

Note that in order to have $p(S_t>=-1|S_{t-1}=j)=1$ when $j\in[0;t-p]$, the coefficients of the Gaussian used in the last case have to sum to 0.9.

B. Likelihood in a Voting Scheme

During the computation of the likelihood, we wish to avoid an exhaustive image-to-image comparison of the visual features, as implemented in most of the voting and bag of words methods cited in section II. In order to efficiently find the most likely past image I_i that closes the loop with the current one, we take advantage of the *inverted index* associated with the dictionary. The inverted index lists the images in which each word has been seen in the past. Then, during the quantization of the current image I_t with the words z_t it contains, each time a word is found, we retrieve the list of the past images in which it has been previously seen. This list is used to estimate the likelihood $\mathcal{L}(S_t|z_t)$ in a simple voting scheme: a score (originally set to 0) is assigned to every past image and updated when we find a word that has been seen in this image (see figure 1). The update step simply consists in the addition of some statistics about the word to the score. The chosen statistics are obtained from the *term frequency–inverted document frequency (tf–idf)* weighting [20]: it is the product of the term frequency (i.e. the frequency of a word in an image) by the inverted document frequency (i.e. the inverse frequency of the images containing this word). To summarise, when a word is found in the current image, the images where this word has been previously seen have their scores updated with the tf–idf coefficient associated with the pair $\{$*word–image*$\}$. The score associated with each image I_i corresponds to the likelihood $\mathcal{L}(S_t=i|z_t)$ that the current image closes the loop with image I_i given the words z_t.

Special attention must be payed when considering $\mathcal{L}(S_t=-1|z_t)$, the likelihood of the event "no loop-closure occurred at time t". It is here computed as the likelihood of the event "a loop-closure is found with I_{-1}". I_{-1} is a virtual image built at each likelihood computation step with the n most frequently seen words of Z_t (n being the average number of words found per image): it is the "most likely" image. The idea is that the likelihood associated with I_{-1} will be high when I_t contains common words (i.e. in perceptual aliasing situations) or when no loop-closure occured (as I_t will be statistically more similar to I_{-1} than to any other I_i). On the contrary, in a real unambiguous loop-closure situation, the score of I_{-1} will be low compared to the score of the loop-closing image. The likelihood of this virtual image can hence be considered as the likelihood of the "no loop-closure" event. The construction of a virtual image with existing words is similar to the addition of new locations from words sampling used in [18]. The existence of the virtual image can be simulated simply by adding a I_{-1} entry to the inverted index for each of the most frequently seen words. Therefore, if one of them is found in I_t, it will vote for I_{-1} as shown in figure 1 and $\mathcal{L}(S_t=-1|z_t)$ will be computed as for the "true" images.

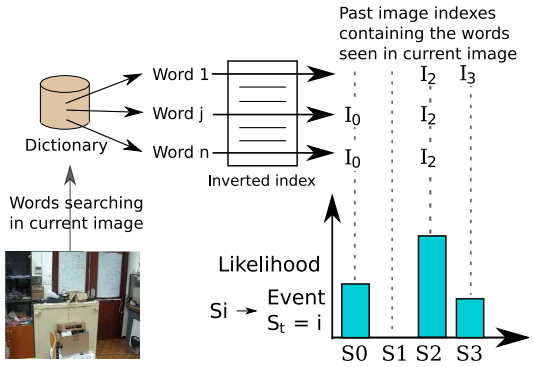

Fig. 1. The voting scheme: the list of the past images in which current words have been seen is obtained from the inverted index and is used to estimate the likelihood.

Once all the likelihoods are computed, we select the subset $H_t\subseteq I^{t-p}$ of images whose score is higher than the mean of the scores plus the standard deviation as the most likely hypotheses. Then, if I_i appears in H_t, the probability $p(S_t=i|z^{t-1})$ is multiplied by the difference between the score of I_i and the mean of the scores at time t, normalized by the mean of the scores (see figure 2). The selection done

on the hypotheses at this stage makes it possible to simplify the update of the posterior, considering that non-selected hypotheses multiply the posterior by 1. When all the images of H_t have been processed, the full posterior is normalized.

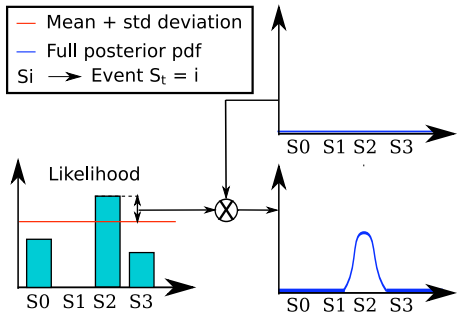

Fig. 2. The full posterior pdf updated with the likelihood: when the likelihood of a hypothesis is above the mean + standard deviation threshold, the corresponding probability is updated.

C. A Posteriori Hypothesis Management

When the full posterior has been updated and normalized, we select as possible candidate for loop-closure detection the hypothesis whose probability is above some threshold (0.8 in our experiments). However, the posterior does not necessarily exhibit a strong single peak for a unique image I_i, but it may rather be distributed over a set of neighboring images (except for I_{-1}). This comes from the similarities among neighboring images. Thus, instead of searching for single peaks among the full posterior, we look for images whose sum of the probabilities over neighboring images is above the threshold (the size of the neighborhood is the same as in section IV-A). After a past image has been selected as a plausible hypothesis for loop-closure with the current image, a multiple view geometry algorithm [9] is used to discard outliers by verifying that the two images satisfy the epipolar geometry constraint and thus come from the same 3D scene.

If successful, the algorithm returns the 3D transformation between x_t and x_i (i.e. the viewpoints associated with I_t and I_i): a loop-closure is detected. Otherwise, the hypothesis is discarded. However, even if a hypothesis has been discarded by the reconstruction algorithm, its a posteriori probability will not fall to 0 immediately: it will diffuse over neighboring images during the propagation of the full posterior from t to $t+1$. Thus, correct hypotheses erroneously discarded by epipolar geometry will be reinforced by the likelihoods of further time instants until a valid 3D transformation is found.

V. Experimental Results

We obtained results from several indoor image sequences grabbed at 1Hz with a simple monocular handheld camera[1]. In this paper, we only present the results obtained from one experiment where perceptual aliasing is particularly strong. The overall camera trajectory followed during this experiment is shown in figure 3 using three different colors. When the posterior is below the 0.8 threshold, the trajectory is shown in blue. When it is above the threshold and the epipolar constraint is satisfied, a loop-closure is detected and the trajectory is shown in green. But, when the posterior is above the threshold and the epipolar constraint is not satisfied, the loop-closure hypothesis is rejected and the trajectory is shown in red. This especially happens in case of perceptual aliasing: since our bag of words algorithm relies on the occurrence of the words rather than on their position, the current image may look like a past image but the structure of the scene may not be consistent and hence, the epipolar constraint cannot be satisfied.

[1]Videos available at http://animatlab.lip6.fr/∼AngeliVideosEn

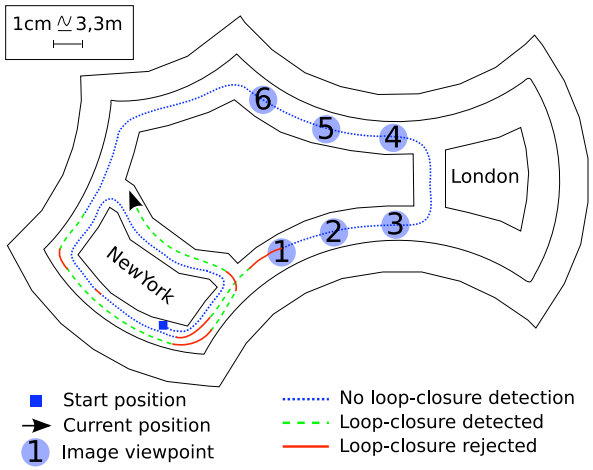

Fig. 3. Overall camera trajectory for the "lip6kennedy" sequence. A first loop is done around the "New York" elevators on the left before going to the "London" elevators on the right. The first loop is travelled again when the camera is back from the "London" elevators following the top-most corridor on the plan. The numbers in the blue circles indicate the positions from which the images composing the mosaic of the figure 4 were taken. See text for details about the trajectory.

As we can see in figure 3, the trajectory is shown in blue every time the camera is discovering unexplored areas, in spite of the strong preceptual aliasing present in the corridors to and from the "London" elevators (see figure 4 for examples of the images composing the sequence). During the run, no *false positive* detections were made (i.e. when a loop-closure is detected whereas none occurred), proving the robustness of our solution to perceptual aliasing.

From figure 3, we can also see that the trajectory is shown in green most of the time spent in previously visited places, indicating that *true positive* detections were made (i.e. when a loop-closure occurs and it is correctly detected). The figure 5 gives an example of a true positive detection.

During the travel of the camera in already explored places, we can note that the color of the trajectory is always switching from green to red when the camera is turning around corners. In these particular cases, loop-closure detection fails only because the epipolar constraint fails to be satisfied, due to the large and fast rotations made by the camera. This corresponds to *false negative* detections (i.e. when a loop-closure occurs but it is not detected).

When considering the trajectory of the camera with more attention, we can observe that the first loop-closure detection

Fig. 4. Top-most corridor (top row) and bottom-most corridor (bottom row) image examples, showing the high level of perceptual aliasing in the environment. The numbers in the blue circles help associating the images with the positions labelled in the figure 3.

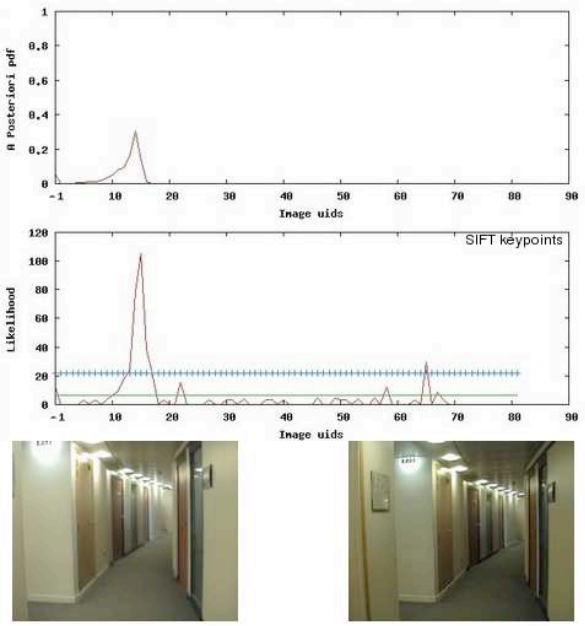

Fig. 5. First loop-closure detection for the "lip6kennedy" sequence. Shown are the full posterior and the likelihood computed from the SIFT feature space, along with the current image I_t (bottom left) and the loop-closing image I_i (bottom right). Likelihoods are defined as the scores (tf–idf) for the different image candidates. Also shown with the likelihood are the score mean (solid green) and the score mean + standard deviation threshold (blue crosses). As it can be seen, the likelihood is very strong around images 12 to 17, causing the posterior to reach the 0.8 threshold. Also, it clearly appears here that I_t and I_i come from very close viewpoints.

that should be done (i.e. when the camera reaches again its starting position for the first time, during its first travel behing the "New York" elevators) is missed and the trajectory remains colored with blue. This is imputable to the slow reactivness of the probabilistic framework: the likelihood associated with a particular hypothesis has to be very high relative to the other likelihoods to provoke a fast loop-closure detection. Usually, the likelihood associated with a hypothesis must have a good support during 2 or 3 consecutive images in order to trigger a loop-closure detection. This tardiness enhances the robustness of the detection to transient detection errors.

During the run, there were only 4 cases where the probability was above the threshold but the selected hypothesis was wrong and has hopefully been rejected by the multiple view geometry algorithm. These events, that can be considered as *false alarms*, can be identified in figure 3 as the two only red portions of the trajectory that do not correspond to camera rotations around corners. The false alarms occurring just after the first serie of true positive detections, at the begining of the bottom corridor from "New York" to "London" elevators, can be explained by the slow decrease of the posterior after a loop-closure detection (see section IV-C): the posterior is still high, but the structure of the scene is no longer consistent, the hypothesis is rejected.

In order to test the robustness of the detection to camera orientation changes, the camera was rotated when passing behind the "New York" elevators for the second time. As show by the green color of the trajectory during this second passing, the loop-closure detection results were not affected (see figure 6).

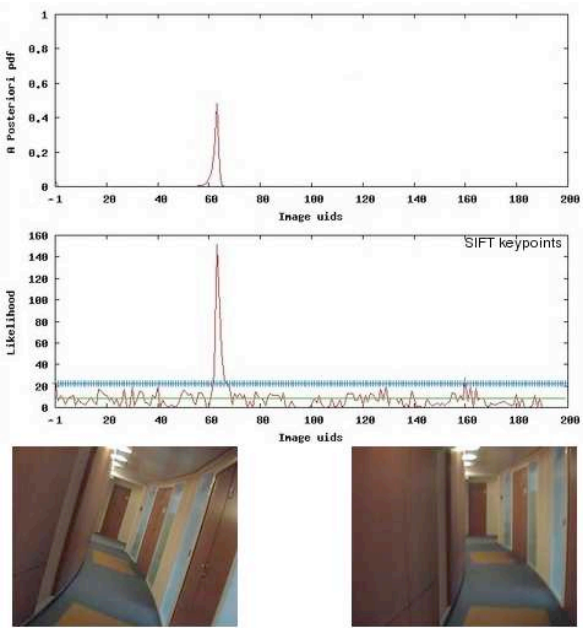

Fig. 6. Another loop-closure detection for the "lip6kennedy" sequence. Although camera orientations are different, the loop-closure is clearly detected as shown by the very peaked a posteriori pdf.

During the experiment, the vocabularies were built online in an incremental fashion from the 234 images of size 240x192 pixels taken at 1Hz, enabling real-time performances with a Pentium Core2 Duo 2.33GHz laptop: CPU time was **??1m24s??56s??** to process the 3m54s of the sequence. The evolution of the computation time per image is given in the figure 7. We can see that the time needed to extract the features in the images is nearly constant. When adding the word searching time, the evolution scales logarithmically with time. Finally, the overall image processing time seems to evolve linearly with time, making it possible to approximate the number of images at which the system will no longer be realtime to 1500.

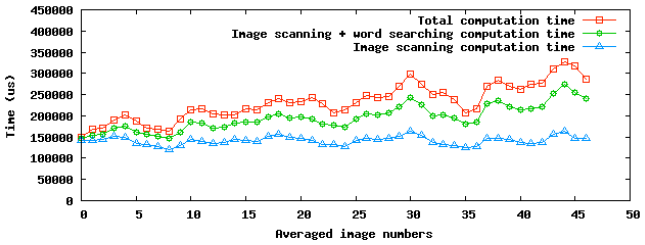

Fig. 7. Evolution of the processing time per image: given is the time needed to extract the features in the images (triangles), to which is added the time required to find the corresponding words in the vocabulary (circles), along with the total computation time per image (squares). To enhace readability, computation times have been averaged every 5 images.

VI. Discussion and Future Work

The solution proposed in this paper is, to our knowledge, the first fully incremental and online vision-based method allowing loop-closure detection in real-time. The results presented here show the robustness of our solution to perceptual aliasing. However, the more complex probabilistic framework introduced by the authors of [18] handles it more properly, taking it into account at the word level (i.e. the input information level) while in our case, it is managed at the detection level (i.e. the output level), when hypotheses are checked by the epipolar geometry algorithm. Still, the evaluation of the distinctiveness of every word proposed in [18] cannot be done incrementally, because to evaluate the co-occurrences of the words, representative images of the entire environment have to be processed beforehand. In our method, the distinctiveness of the words is taken into account using the inline calculated tf–idf coefficient when computing the likelihood: the words seen multiple times in the same location will vote with a high score for this location (i.e. high tf), while the words seen in every locations will have a small contribution (i.e. low idf).

In a future work, we will adapt the current approach to a purely vision-based SLAM system like [4] so as to reinitialize the SLAM algorithm when the camera position is lost or when there is a need to self-relocalize in a map acquired beforehand. The metrical information about the camera's pose coming from SLAM could be used to improve the definition of the neighborhood of a location, using spatial transitions between adjacent locations instead of time indexes. Moreover, very close viewpoints could be agglomerated into a single location, scaling better with the number of images.

An evaluation of other feature spaces should be done. As stated in [7], several feature spaces could be used together in order to improve the performances, each feature space giving a specific image representation: color histograms could be useful in textureless images, for example. Also, geometric information from relative spatial positions between the visual words could be used to improve matching.

VII. Conclusion

In this paper, we have presented a fast and incremental bag of visual words method for performing loop-closure detection in real-time, with no false positive detections even under strong perceptual aliasing conditions. Our approach calls upon a Bayesian filtering framework with likelihood computation in a simple voting scheme and should be extended to SLAM reinitialization in a near future.

References

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous localisation and mapping (slam): Part i," *IEEE Robotics and Automation Magazine*, vol. 13, no. 1, pp. 99–110, 2006.

[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] A. Angeli, D. Filliat, S. Doncieux, and J. Meyer, "2d simultaneous localization and mapping for micro aerial vehicles," in *European Micro Aerial Vehicles (EMAV)*, 2006.

[4] 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, pp. 1052–1067, June 2007.

[5] M. E. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," in *IEEE conference on Computer Vision and Pattern Recognition*, 2006.

[6] J. Sivic, B. C. Russel, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering objects and their locations in images," in *International Conference on Computer Vision*, 2005.

[7] D. Filliat, "A bag of words method for interactive visual qualitative localization and mapping," in *IEEE International Conference on Robotics and Automation*, 2007.

[8] G. Csurka, C. Dance, L. Fan, J. Williamowski, and C. Bray, "Visual categorization with bags of keypoints," in *ECCV04 workshop on Statistical Learning in Computer Vision*, pp. 59–74, 2004.

[9] 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.

[10] F. Dellaert, F. Dieter, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *IEEE International Conference on Robotics and Automation*, May 1999.

[11] J. Wolf, W. Burgard, and H. Burkhardt, "Robust vision-based localization by combining an image retrieval system with monte carlo localization," *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 208–216, 2005.

[12] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, and E. Nebot, "Fastslam: An efficient solution to the simultaneous localization and mapping problem with unknown data association," *Journal of Machine Learning Research*, vol. -, pp. –, 2004.

[13] M. Pupilli and A. Calway, "Real-time visual slam with resilience to erratic motion," in *IEEE Computer Vision and Pattern Recognition*, 2006.

[14] 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.

[15] I. Ulrich and I. Nourbakhsh, "Appearance-based place recognition for topological localization," in *IEEE International Conference on Robotics and Automation*, 2000.

[16] J. Wang, H. Zha, and R. Cipolla, "Coarse-to-fine vision-based localization by indexing scale-invariant features," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 36, pp. 413–422, April 2006.

[17] P. Newman, D. Cole, and K. Ho, "Outdoor slam using visual appearance and laser ranging," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2006.

[18] M. Cummins and P. Newman, "Probabilistic appearance based navigation and loop closing," in *Proc. IEEE International Conference on Robotics and Automation (ICRA'07)*, 2007.

[19] D. Lowe, "Distinctive image feature from scale-invariant keypoint," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[20] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *IEEE International Conference on Computer Vision (ICCV)*, 2003.