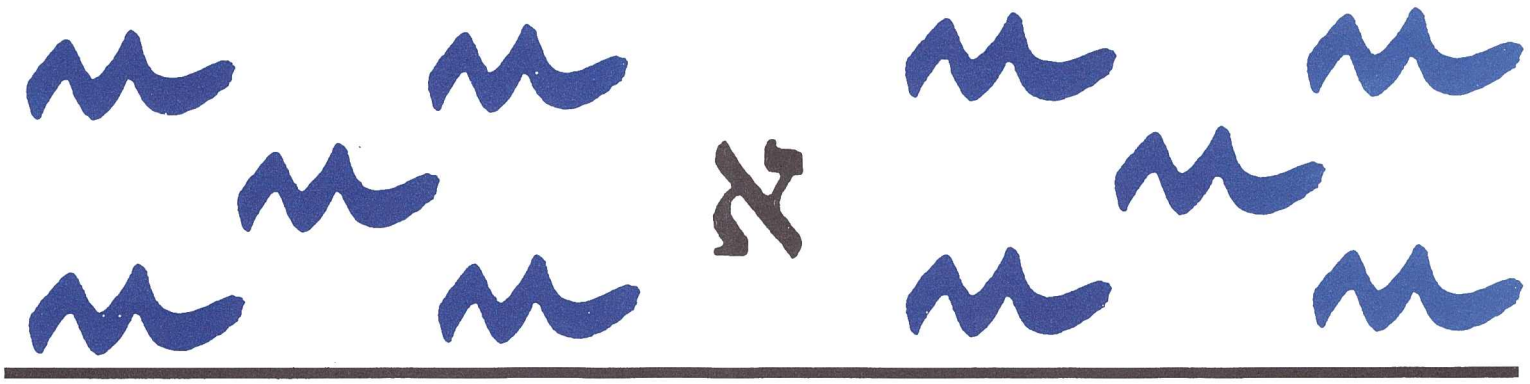
Universitat de les Illes Balears

Departament de Ciències
Matemàtiques i Informàtica

A Solution for Bayesian Loop Closure Detection Based on Local Invariant Features

E. GARCIA-FIDALGO and A. ORTIZ



March / 2013

A-01-2013

TECHNICAL REPORT

A Solution for Bayesian Visual Loop Closure Detection Based on Local Invariant Features

Emilio Garcia-Fidalgo and Alberto Ortiz*

Abstract

Visual loop closure detection in robotics is defined as the ability of recognizing previously seen places given the current image captured by the robot. The *Bag-of-Words* image representation has been widely used for these kinds of tasks. However, in this paper, an appearance-based approach for loop closure detection using local invariant features is proposed. Images are described using SIFT features and, for avoiding image-to-image comparisons, a set of randomized KD-trees are employed for feature matching. Further, a discrete Bayes filter is used for predicting loop closure candidates, whose likelihood is based on these KD-trees. The approach has been validated using monocular image sequences from several environments.

1 Introduction

Localization and mapping are essential problems in mobile robotics. In order to solve them, several approaches have been proposed to perform both tasks at the same time, creating an incremental map of an unknown environment while localizing the robot within this map. These techniques are called SLAM [1] (Simultaneous Localization and Mapping). In SLAM, loop closure detection is a key challenge to overcome. It implies the correct detection of previously seen places from sensor data. This allows generating consistent maps and reduce their uncertainty.

Ultrasounds and laser sensors have been used for years for SLAM and loop closure detection. Nevertheless, in the last decades there has been a significant increase in the number of visual solutions because of the low cost

*E. Garcia-Fidalgo and A. Ortiz are with the Department of Mathematics and Computer Science, University of Balearic Islands, 07122 Palma de Mallorca, Spain. {emilio.garcia, alberto.ortiz} at uib.es

of cameras, the richness of the sensor data provided and the availability of cheap powerful computers. This naturally guides us to an appearance-based SLAM, where the environment is represented in a topological way by a graph. Each node of this graph represents a distinctive visual location visited by the robot while the edges indicate connectivities between locations. Using this representation, the loop closure problem can be solved comparing images directly, avoiding to maintain and estimate the position of feature landmarks.

In the *Bag-of-Words* (BoW) [2] approach, local invariant features obtained from an image are quantized into a vector according to a visual vocabulary. This representation is one of the most used technique for loop closure detection in appearance-based SLAM. However, this method presents some drawbacks. On the one hand, the perceptual aliasing effect [3], where two different places can be perceived as the same, is increased because of the quantization process. On the other hand, an offline training phase is required to build the visual vocabulary on most occasions.

Loop closure detection can be achieved matching raw features directly [3–6]. Despite we have developed a visual-based topological SLAM method in a previous work [7], in this paper we want to focus in the visual loop closure problem. In more detail, we present a Bayesian framework for visual loop closure detection using local invariant features. Given a new image acquired with a monocular image configuration, the probability of loop closure with all previously seen images is computed. When this probability is above a threshold, a further condition derived from epipolar geometry is checked next to confirm that the current image really closes the loop. Experimental results in different environments are provided in order to validate the proposed solution. Matchings between images are obtained efficiently building a set of randomized KD-trees [6].

The rest of the paper is organized as follows: Section 2 enumerates fundamental works related to loop closure detection and visual localization and mapping, Section 3 shows how images are described and matched in our approach, Section 4 exposes a Bayesian loop closure algorithm using visual features, Section 5 shows experimental results obtained from different datasets and Section 6 concludes the paper.

2 Related Work

A high number of appearance-based localization and mapping solutions have been proposed along the last decade. Although many works assume the availability of omnidirectional images [8–11], many others make use of monocular configurations [6, 12–14]. Our approach belongs to this latter class.

Referring to the image description, the BoW approach has become quite popular. Cummins and Newman developed FAB-MAP [12], where a Chow-Liu tree is used for modelling the dependencies between visual words. Angeli *et al.* [13, 14] extended the BoW paradigm to incremental conditions and relied on Bayesian filtering to estimate the probability of loop closure. Despite its well-known general performance, the BoW paradigm is more affected by perceptual aliasing [3]. For this reason, our work follows an approach similar to [13, 14], but using local invariant features for image description and matching.

Other approaches make use of global descriptors, such as Gist [15]. Singh and Kosecka [16] computed Gist descriptors in omnidirectional images of urban environments for detecting loop closures. They presented a novel image matching strategy for panoramas. Bayes filtering is not considered in this work. Liu and Zhang [17] applied Principal Component Analysis (PCA) to Gist descriptors in order to compute the likelihood in a particle filter. This filter is used for detecting loop closures. Siagian and Itti [18] presented a biologically-inspired system to scene classification using Gist as image representation.

Rather than BoW or global descriptors, some authors have used local invariant features for visual localization and mapping as well as for loop closure detection. Zhang [3] presented a method for selecting a subset of Scale-Invariant Feature Transform (SIFT) [19] keypoints extracted from an image. These features are used for matching consecutive images. A location is represented by a set of features that can be matched consecutively in several images. The problem of this approach is that the number of features to manage increases while new images are added, and a linear search for matching becomes intractable. This drawback is overcome in [6] indexing features using a set of randomized KD-trees. Our approach follows these guidelines for image matching.

3 Image Description and Matching

In our approach, each image is described using the SIFT [19] algorithm, where interest points are defined as maxima and minima of a difference of Gaussians function applied in scale space to a series of resampled images. Each feature is then described defining a histogram of gradient orientations around the point at the selected scale, resulting in a 128-dimensional descriptor. These descriptors are compared in this work using Euclidean distance.

A method for an efficient nearest neighbour search is needed in order to match these high-dimensional descriptors. Tree structures have been widely

used to this end, since they reduce the search complexity from linear to logarithmic [6]. To the same purpose, we maintain a set of randomized KD-trees containing all the SIFT descriptors of previously seen images. An inverted index, which maps each feature to the image where it was found, is also created. Given a query descriptor, these structures allow us to obtain, traversing the tree just once, the top K nearest neighbours keypoints among all images.

4 Probabilistic Loop Closure Detection

Given a new image, a discrete Bayes filter is used to detect loop closure candidates. This filter estimates the probability that the current image closes a loop with an already seen image, ensuring temporal coherency between consecutive predictions. Given the current image I_t at time t, we denote z_t as the set of SIFT descriptors extracted from this image. These are the observations in our filter. We also denote L_i^t as the event that image I_t closes a loop with image I_i, where $i < t$. Using these definitions, we want to detect the image I_c whose index satisfies:

$$c = \underset{i=0,\dots,t-p}{\arg\max} \{ P\left(L_i^t | z_{0:t}\right) \}, \tag{1}$$

where $P\left(L_i^t | z_{0:t}\right)$ is the full posterior probability at time t given all previous observations up to time t. As in [14], the most recent p images are not included as hypotheses in the computation of the posterior since I_t is expected to be very similar to its neighbours and then false loop closure detections will be found. This parameter p delays the publication of hypotheses and needs to be set according to the frame rate or the velocity of the camera.

Separating the current observation from the previous ones, the posterior can be rewritten as:

$$P\left(L_i^t | z_{0:t}\right) = P\left(L_i^t | z_t, z_{0:t-1}\right), \tag{2}$$

and then, using conditional probability properties, the next equality holds:

$$P\left(L_i^t | z_t, z_{0:t-1}\right) P\left(z_t | z_{0:t-1}\right) = P\left(z_t | L_i^t, z_{0:t-1}\right) P\left(L_i^t | z_{0:t-1}\right), \tag{3}$$

from where we can isolate our final goal to obtain:

$$P\left(L_i^t | z_t, z_{0:t-1}\right) = \frac{P\left(z_t | L_i^t, z_{0:t-1}\right) P\left(L_i^t | z_{0:t-1}\right)}{P\left(z_t | z_{0:t-1}\right)}. \tag{4}$$

$P\left(z_t|z_{0:t-1}\right)$ is independent of L_i^t, so it can be seen as a normalizing factor. Under this premise and the Markov assumption, the posterior is defined as:

$$P\left(L_i^t|z_{0:t}\right) = \eta P\left(z_t|L_i^t\right) P\left(L_i^t|z_{0:t-1}\right) \ , \tag{5}$$

where η represents the normalizing factor, $P\left(z_t|L_i^t\right)$ is the observation likelihood and $P\left(L_i^t|z_{0:t-1}\right)$ is the probability distribution after a prediction step. Decomposing the right side of (5) using the Law of Total Probability, the full posterior can be written as:

$$P\left(L_i^t|z_{0:t}\right) = \eta P\left(z_t|L_i^t\right) \sum_{j=0}^{t-p} P\left(L_i^t|L_j^{t-1}\right) P\left(L_j^{t-1}|z_{0:t-1}\right) \ , \tag{6}$$

where $P\left(L_j^{t-1}|z_{0:t-1}\right)$ is the posterior distribution computed in the previous time instant and $P\left(L_i^t|L_j^{t-1}\right)$ is the transition model.

Unlike [14], we do not model explicitly the probability of no loop closure in the posterior. If the loop closure probability of I_t with I_c ($P\left(L_c^t|z_{0:t}\right)$) is not high enough, the existence of L_c^t is discarded.

4.1 Transition Model

Before updating the filter using the current observation, the loop closure probability at time t is predicted from $P\left(L_j^{t-1}|z_{0:t-1}\right)$ according to an evolution model. The probability of loop closure with an image I_j at time $t-1$ is diffused over its neighbours following a discretized Gaussian-like function centered on j. In more detail, 90% of the total probability is distributed among j and exactly four of its neighbours ($j-2$, $j-1$, j, $j+1$, $j+2$) using coefficients (0.1, 0.2, 0.4, 0.2, 0.1), i.e. $0.9 \times$ (0.1, 0.2, 0.4, 0.2, 0.1). The remaining 10% is shared uniformly across the rest of loop closure hypotheses according to $\frac{0.1}{\max\{0,t-p-5\}+1}$. This implies that there is always a small probability of jumping between hypotheses far away in time, improving the sensitivity of the filter when the robot revisits old places.

4.2 Observation Model

Once the prediction step has been performed, the current observation needs to be included in the Bayes filter. We want to compute the most likely images given the current image I_t and its keypoint descriptors z_t, but we want to avoid comparing I_t with each previous image, since this is not tractable. To this end, the structures described in section 3 are used. Note that if the robot has revisited the same place several times and the current image I_t closes this

loop again, each descriptor in z_t can be close to descriptors from different previous images in the Euclidean space. This fact is taken into account in the computation of our likelihood.

For each hypothesis i in the filter, a score $s(z_t, z_i)$ is computed. This score represents the likelihood that the current image I_t closes the loop with image I_i given their descriptors, z_t and z_i respectively. Initially, these scores are set to 0 for all frames from 0 to $t-p$. For each descriptor in z_t, the K closest descriptors among the previous images are retrieved without taking into account the p immediately previous frames, and each of them, denoted by n, adds a weight w_n to the score of the image where it belongs to. This value is normalized using the total distance of the K candidates retrieved:

$$w_n = 1 - \frac{d_n}{\sum_{k \in K} d_k}, \forall n \in K, \tag{7}$$

where d is the Euclidean distance between the considered query descriptor in z_t and the nearest neighbour descriptor found in the tree structure. This value is accumulated according to:

$$s\left(z_t, z_{j(n)}\right) = s\left(z_t, z_{j(n)}\right) + w_n, \forall n \in K, \tag{8}$$

being $j(n)$ the index of the image from where the candidate descriptor n was extracted. The computation of the scores is finished when all descriptors in z_t have been processed. Then, the likelihood function is finally defined according to the following rule [14]:

$$P\left(z_t | L_i^t\right) = \begin{cases} \frac{s(z_t, z_i) - s_\sigma}{s_\mu} & \text{if } s(z_t, z_i) \geq s_\mu + s_\sigma \\ 1 & \text{otherwise} \end{cases}, \tag{9}$$

being s_μ and s_σ respectively the mean and the standard deviation of the set of scores. Only the most likely images given the current observation z_t increases their prior. After incorporating the observation to our filter, the full posterior is normalized in order to obtain a probability function.

4.3 Selection of a Loop Closure Candidate

In order to select a final candidate, we do not search for high peaks in the posterior distribution, because loop closure probabilities are usually diffused between neighbouring images. This is due to visual similarities between consecutive frames in the sequence. Instead, for each image, we add the probabilities in a defined neighbourhood. This neighbourhood is the same as defined in section 4.1: frames $(j-2, j-1, j, j+1, j+2)$ for image j.

Algorithm 1 Visual Loop Closure Detection

/* Variables */
$I = \{I_0, \ldots, I_{N-1}\}$: Sequence of N input images.
K : Set of randomized KD-trees for feature indexing.
B : Discrete Bayes filter.
F_t : Set of SIFT features obtained from image I_t.
c : Candidate image index for closing a loop.
P_c : Probability of candidate image index for closing a loop.
M_i : Set of matchings between image i and images between 0 and $i - p$.
n_{hyp} : Number of hypotheses in the Bayes filter.
E_j^i : Set of matchings surviving the epipolarity constraint-based filter.
L : Output boolean variable for indicating the existence of a loop.
L_{im} : Output integer with the index of the image loop closure.

/* Thresholds */
p : Number of recent images that are not included as hypothesis in the filter.
T_{loop} : Minimum probability to consider a loop candidate.
T_{ep} : Minimum number of surviving matchings after epipolar geometry validation.
T_{hyp} : Minimum number of hypotheses for considering loop candidates.

$n_{hyp} = 0$
for $t = 0$ **to** $N - 1$ **do** /* While there are images */
    F_t = describe(I_t)
    $n = t - p$
    if $n > -1$ **then**
        updateTree(K, F_n) // Adding F_n descriptors to the tree
        addHypotesis(B, n) // Adding a new state in the Bayes filter
        n_{hyp} = n_{hyp} + 1
        M_i = matchTree(K, F_t)
        predict(B) // Applying transition model to the prior distribution
        update(B, M_i) // Incorporating the observation to the prediction
        c, P_c = getCandidate(B) // Getting the best loop candidate
        if $P_c > T_{loop}$ **and** $n_{hyp} > T_{hyp}$ **then**
            E_c^t = epipolarGeometry(F_t, F_c)
            if numberOfElements(E_c^t) $> T_{ep}$ **then**
                $L =$ `True`; $L_{im} = c$
            else
                $L =$ `False`; $L_{im} = -1$
            end if
        else
            $L =$ `False`; $L_{im} = -1$
        end if
    end if
end for

The image I_j with the highest sum of probabilities in its neighbourhood is selected as a loop closure candidate. If this sum is below a threshold T_{loop}, loop closure hypothesis is not accepted. Otherwise, an epipolarity analysis between I_t and I_j is performed in order to validate if they can come from the same scene after a camera rotation and/or translation. Using a RANSAC procedure, the matchings that do not fulfill the epipolar constraint are discarded. If the number of surviving matchings is above a threshold T_{ep}, the loop closure hypothesis is accepted; otherwise, it is definitely rejected.

Another threshold, T_{hyp}, is defined to ensure a minimum number of hypotheses in the filter, so that loop closure candidates are meaningful: first images inserted in the filter tend to attain a high probability of loop closure after the normalization step, what leads to incorrect detections.

The full approach is outlined in Algorithm 1. In detail: *describe* extracts and describes SIFT keypoints from an image, *updateTree* adds a set of SIFT descriptors to the index and trains it, *addHypothesis* adds a new state to the Bayes filter, *matchTree* performs a nearest neighbour search for a set of query SIFT descriptors, *predict* applies the transition function to the previous posterior, *update* updates the filter using the likelihood for the current observation, *getCandidate* returns the image with the maximum probability neighbourhood and *epipolarGeometry* performs the epipolar geometry validation step between two images.

5 Experimental Results

Several experiments have been carried out in order to validate the suitability of our framework for loop closure detection tasks. Datasets from indoor and outdoor environments have been processed, providing results under different environmental conditions. Each dataset is provided with a ground truth, which indicates, for each image in the sequence, which images can be considered as a loop closure with it. The assessment has been performed against this ground truth counting the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) for each whole sequence, where positive is meant for detection of loop closure. Then, several metrics are computed:

- *Precision*: ratio between real loop closures and total amount of loop closures detected ($\frac{TP}{TP+FP}$).
- *Recall*: ratio between real loop closures and total amount of loop closures existing in the sequence ($\frac{TP}{TP+FN}$).

Table 1: Confusion matrix for the lip6indoor dataset

		Actual	
		True	False
Predicted	Positive	191	0
	Negative	151	31

Precision: 1, **Recall**: 0.86, **Accuracy**: 0.91

- *Accuracy*: Percentage of correctly classified (true positive or true negative) images ($\frac{TP+TN}{TP+TN+FP+FN}$).

Avoiding false positives in a loop closure detection algorithm is essential, since they can introduce errors in mapping and localization tasks. The parameters of our filter have been configured under this premise. As a consequence the classifier always reaches 100% in precision for all datasets.

5.1 Lip6Indoor Dataset

This is an indoor dataset collected by Angeli *et al.*[1] for their work in loop closure detection. It was recorded inside the corridor of a building under strong perceptual aliasing conditions and comprises a total of 388 images of 240×192 pixels. Images were acquired at 1 Hz using a single monocular camera with a 60° field of view and automatic exposure. It performs several loops around the corridor. The path followed can be seen in their original paper ([14] Fig. 4). The algorithm parameters have been configured as $p = 15$, $T_{loop} = 0.7$, $T_{ep} = 7$ and $T_{hyp} = 10$.

Results for this sequence can be seen in Table 1. *Predicted* represents the response given by our framework (loop or not loop) for an input image, while *Actual* is the value in the ground truth for this image. At 100% of precision, our approach shows 86% of recall and 91% of accuracy. No false positives result, as required. There are 31 false negatives, which are due to two main reasons:

- *Sensitivity of the filter.* When an old place is revisited, the likelihood associated to that hypothesis needs to be higher than the other likelihood values during several consecutive images in order to increment the posterior for this hypothesis. This introduces a delay in the loop

[1] http://cogrob.ensta-paristech.fr/loopclosure.html

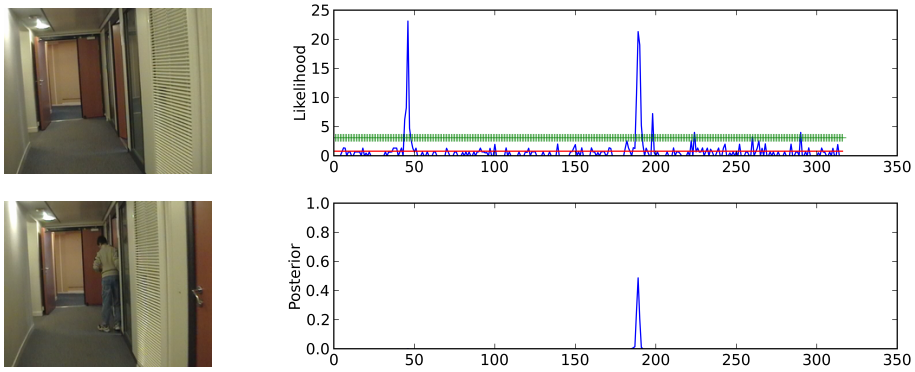


Figure 1: Example of loop closure detection visiting several times the same place. Image 331 (Top, Left) closes a loop with image 189 (Bottom, Left) and image 48 (not shown). As can be seen in (Top, Right), current likelihood presents two strong peaks corresponding to each candidate. After the normalization step, the posterior (Bottom, Right), shows a single peak in the last loop candidate.

closure detection, deriving in false negatives. This sensitivity can be tuned by modifying the transition model of the filter. However, more sensitivity can introduce loop detection errors, i.e. false positives.

- *Camera Rotations.* When the camera is turning around a corner, it is difficult to find and match features in the image, which prevents the hypothesis from satisfying the epipolar constraint and leads to the loop closure hypothesis to be rejected, despite the posterior for this image is higher than T_{loop}.

Fig. 1 shows the suitability of the Bayes framework in a challenging loop closure detection situation. The camera has revisited twice the same place. When it returns to this place again, two high peaks corresponding to the previous visits can be seen in the likelihood, representing possible loop candidates for the current image. After the prediction, update and normalization steps, the posterior presents only one single peak at the second candidate image, i.e. the filter ensures temporal coherency between predictions.

Regarding the ability of the filter to detect loops when the appearance of the environment has changed, Fig. 2 shows an example of situation where a loop is detected despite there is a person in the image who was not in the previous visit. The likelihood function exhibits a clear single peak for the expected loop candidate. After normalizing the posterior, our approach accepts the loop closure since the epipolar constraint between the two images

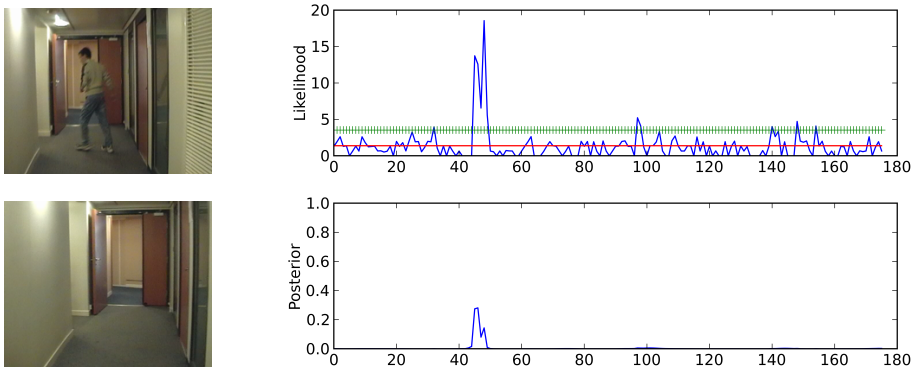


Figure 2: Example of loop closure detection with changes in the environment. Image 190 (Top, Left) closes a loop with image 47 (Bottom, Left). Likelihood (Top, Right) presents a high peak despite there is a person in the current image. (Bottom, Right) is the final posterior.

is satisfied. Our approach is also able to detect loop closures under camera rotations. An example can be found in Fig. 3.

Finally, this dataset has also been used to assess separately the performance of the likelihood function. This is shown in Fig. 4, where the right picture shows the likelihood function values for every pair of frames I_i and I_j and the left picture is the ground truth (only the lower triangles are show). As can be seen, our likelihood presents high values for real loop closures, which are shown as diagonals in the images. There are more noise in the likelihood at the beginning of the sequence because there are less images in the trees, which implies that nearest neighbours for each descriptor are shared between a minor number of images. This effect decreases along the sequence.

5.2 Lip6Outdoor Dataset

In this second experiment we process an outdoor dataset also recorded by Angeli *et al.* It performs a big loop around the city under good weather conditions. The sequence comprises 1063 images of 240×192 pixels. The path followed by the camera is shown again in their original paper ([14] Fig. 10). Only one parameter of the algorithm, $p = 20$, has been modified with respect to the previous experiment, since the velocity of the camera differs.

In general terms, this is a more complex environment than the indoor experiment, since it has much more images and usually there are more changes in appearance inside a city, due to the traffic, for instance. Table 2 shows the results for this sequence. Again, a high number of correct detections (TP

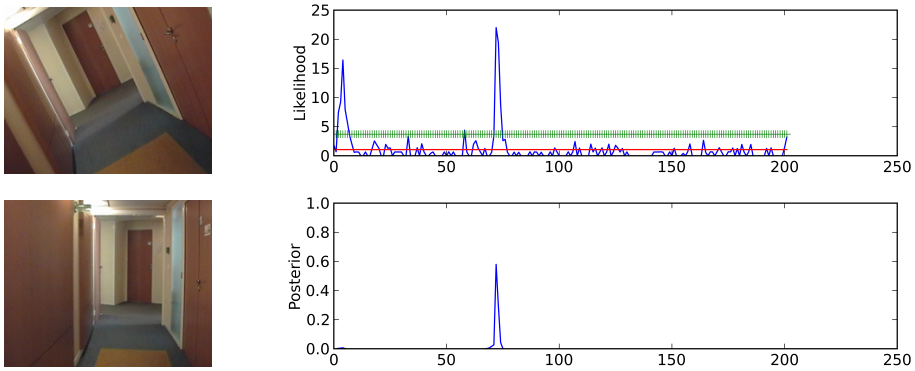


Figure 3: Example of loop closure detection under camera rotations. Despite there is a camera rotation, image 216 (Top, Left) closes a loop with image 72 (Bottom, Left). The likelihood (Top, Right) presents two high peaks since it is the third time the camera visits this place. (Bottom, Right) shows the final posterior, proving that the filter ensures the temporal coherency between loop detections.

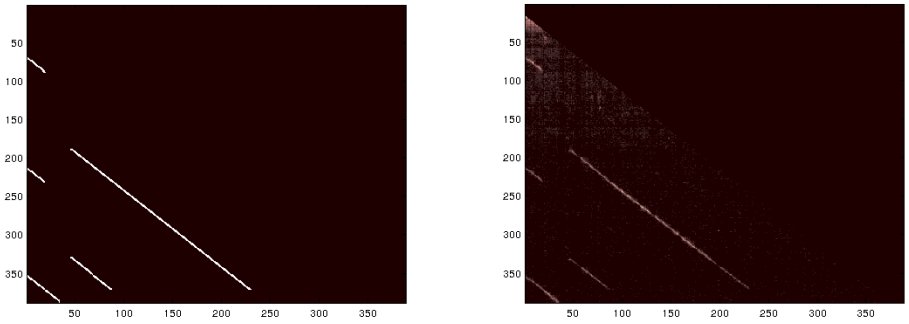


Figure 4: (Left) Ground truth loop closure matrix for the *Lip6Indoor* dataset. (Right) Likelihood matrix computed using our approach.

Table 2: Confusion matrix for the lip6outdoor dataset

		Actual	
		True	False
Predicted	Positive	551	0
	Negative	435	52

Precision: 1, **Recall**: 0.91, **Accuracy**: 0.95

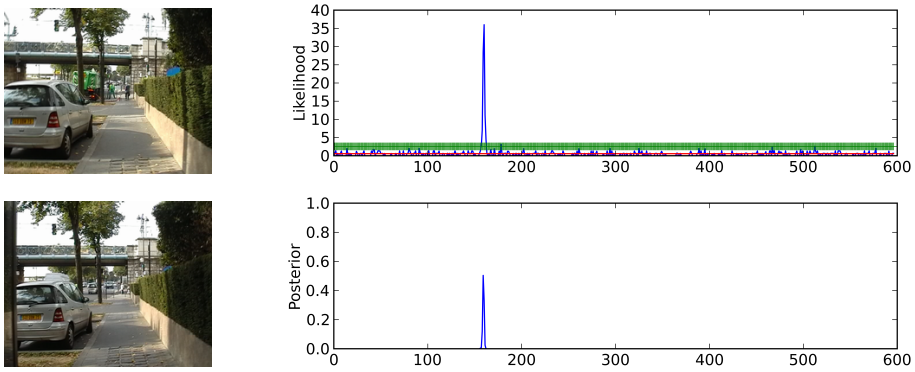


Figure 5: Example of loop closure detection in the outdoor environment. Image 621 (Top, Left) closes a loop with image 159 (Bottom, Left). (Top, Right) Likelihood given the current image. (Bottom, Right) Full posterior after the normalization step.

Table 3: Confusion matrix for the UIB small loop dataset

		Actual	
		True	False
Predicted	Positive	194	0
	Negative	172	2

Precision: 1, **Recall**: 0.99, **Accuracy**: 0.99

and TN) are made, while no false positives arise. At 100% of precision, our approach obtains a recall of 91% and 95% of accuracy. False negatives are due to the same reasons outlined for the previous experiment. However, a minor number of false negatives have resulted regarding to the length of the sequence. This is due to the fact that the perceptual aliasing effect is less present in this environment, and less images are needed to adapt the filter to previous hypotheses. An example of loop closure in this environment is shown in Fig. 5. Fig 6 shows the robustness of the framework under changes in the environment. In this case, our approach is able to find a loop closure between images despite a truck has disappeared in the scene and the camera is rotated.

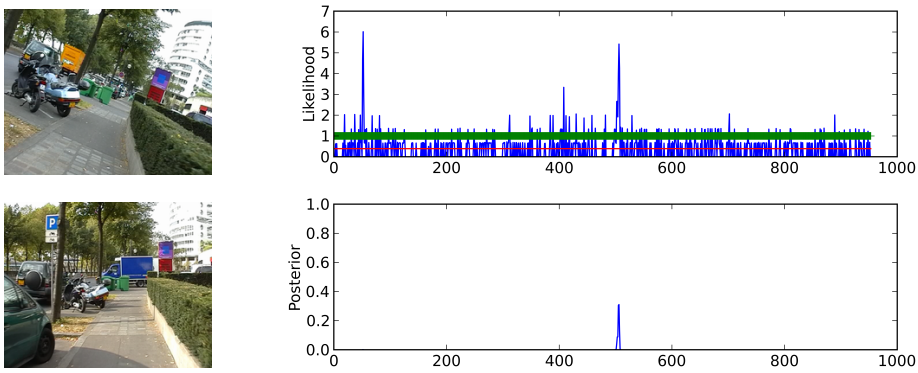


Figure 6: Example of loop closure detection with apperance changes and camera rotations. Despite the traffic changes in the image and the camera rotation, image 978 (Top, Left) closes a loop with image 505 (Bottom, Left). The likelihood (Top, Right) presents two high peaks since it is the third time the camera visits this place. (Bottom, Right) shows the final posterior.

5.3 UIB Small Loop Dataset

Our approach has been further validated using several sequences recorded by ourselves. This experiment involves a loop around the Anselm Turmeda Building of the University of Balearic Islands Campus. The sequence comprises 388 images of 300×240 pixels, and has been recorded using a handheld camera at 1 Hz. In contrast to the previous experiment, it is an outdoor dataset under bad weather conditions. The parameters used for processing this sequence are the same as for the second dataset.

The results for this experiment are shown in Table 3. At 100% of precision, the recall is 99% and the accuracy is 99%. These values suggest that our approach can also be used under bad weather conditions. An example of loop closure detection for this experiment can be found in Fig 7.

The path followed by the camera is shown in Fig 8 using a representation similar to [14]. As can be seen, during the first loop, no detections were reported, meaning that any candidate had a probability above T_{loop}. When the camera came back to the beginning of the sequence, the algorithm starts detecting loop closures. Several images are needed until closing the first loop. Due to the filter inertia, these images correspond to the false negatives found. The algorithm is then able to detect correctly the rest of the loops until the end of the sequence.

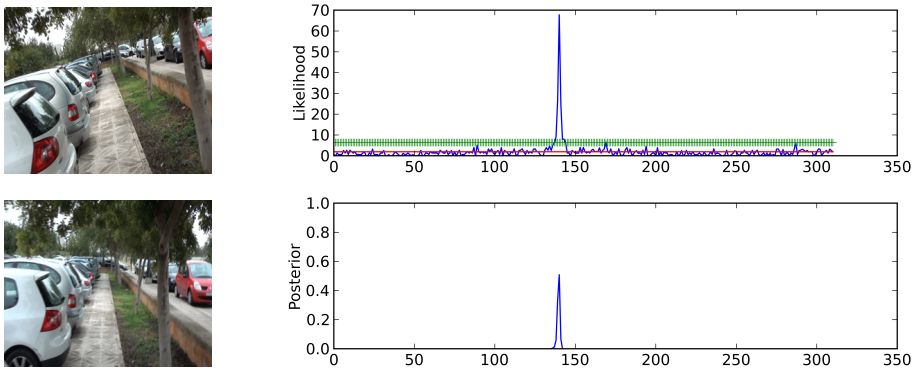


Figure 7: Example of loop closure detection under bad weather conditions and camera rotations for the UIB small loop dataset. Image 330 (Top, Left) closes a loop with image 139 (Bottom, Left). (Top, Right) Likelihood given the current image. (Bottom, Right) Full posterior after the normalization step.

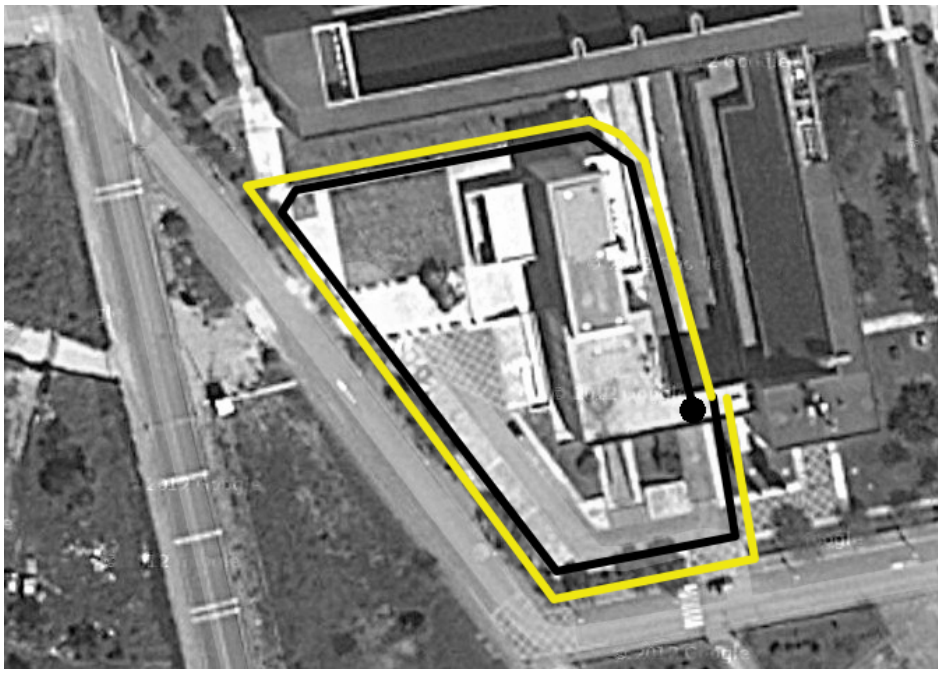


Figure 8: Path followed by the camera during the UIB small loop experiment. The black point indicates the beginning of the sequence, the black lines show no loop closure detections (highest posterior probability is under T_{loop}) and the yellow lines represent loop closure detections (highest probability is above T_{loop} and the epipolar constraint is satisfied).

Table 4: Confusion matrix for the UIB Large Loop dataset

		Actual	
		True	False
Predicted	Positive	439	0
	Negative	491	47

Precision: 1, **Recall**: 0.90, **Accuracy**: 0.95

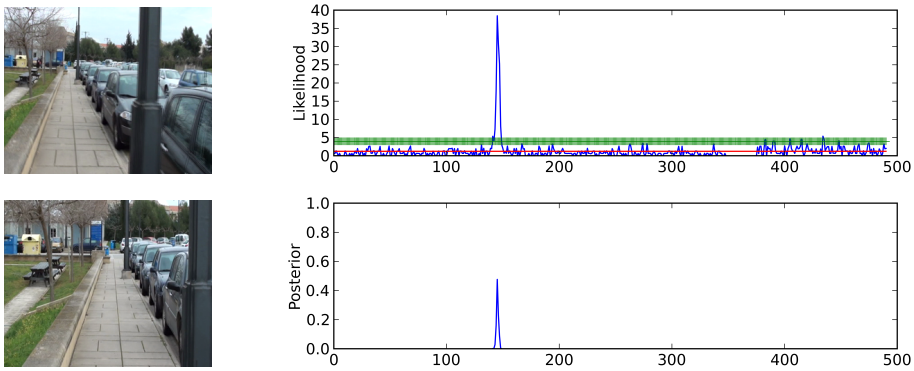


Figure 9: Example of loop closure detection corresponding to the UIB large loop dataset. Image 515 (Top, Left) closes a loop with image 145 (Bottom, Left). (Top, Right) Likelihood given the current image. (Bottom, Right) Full posterior after the normalization step.

5.4 UIB Large Loop Dataset

During this experiment, one small and one large loop were performed around two nearby campus buildings at the University. The sequence comprises around 16 minutes of video, resulting into 997 images of 300×240 pixels grabbed at 1 Hz. There are no modifications in the input parameters of the algorithm regarding previous datasets.

The trajectory followed by the camera and the response of our filter are roughly outlined in Fig 10. Most of the time the camera returned to previously seen places, the filter was able to detect loop closures. Table 4 shows again that a high number of true positives and true negatives resulted, while no false positives arose. At 100%, the recall is 90% and the accuracy is 95%. An example of loop closure detection for this experiment is shown in Fig 9.

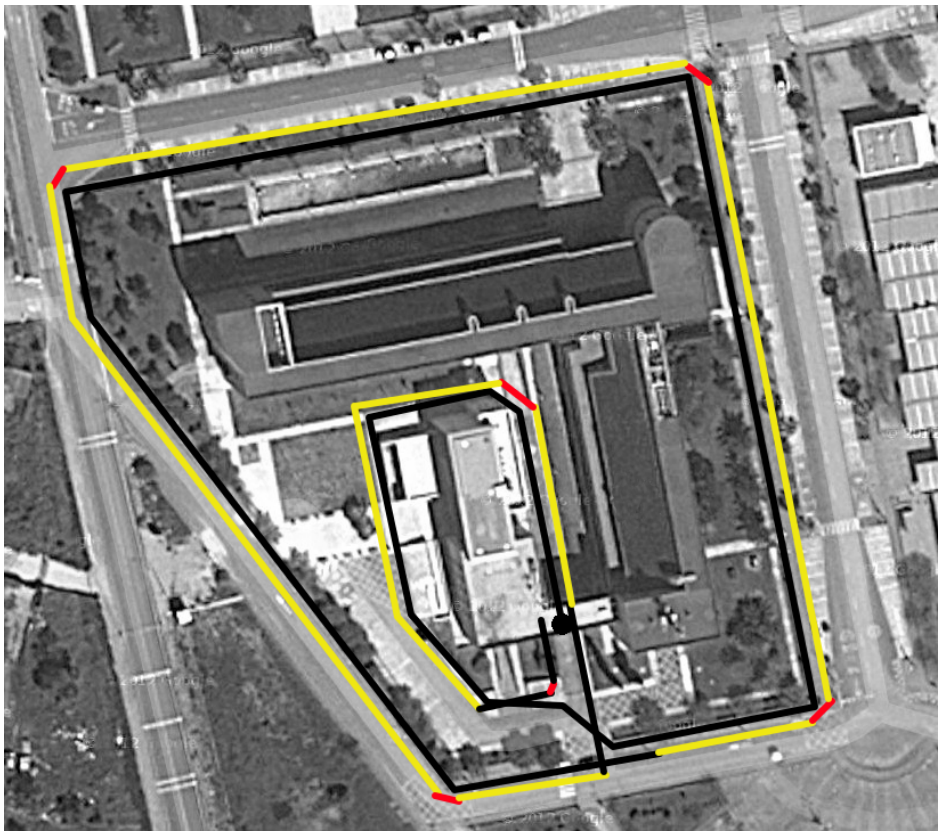


Figure 10: Path followed by the camera during the UIB large loop experiment. The black point indicates the beginning of the sequence, the black lines show no loop closure detections (highest posterior probability is under T_{loop}), the red lines show rejected hypoteses (no epipolar geometry is satisfied) and the yellow lines represent loop closure detections (highest probability is above T_{loop} and the epipolar constraint is satisfied).

Table 5: Confusion matrix for the UIB Indoor dataset

		Actual	
		True	False
Predicted	Positive	157	0
	Negative	177	30

Precision: 1, **Recall**: 0.84, **Accuracy**: 0.92

5.5 UIB Indoor Dataset

This experiment involves an indoor dataset obtained inside the Anselm Turmeda building of the University of Balearic Islands campus. The sequence consists of 384 images of 300×240 pixels, and comprises a loop along different floors of the building. As well as for the last experiment, the algorithm parameters were not changed.

This sequence presents a very challenging environment, since it entails several difficulties to be overcomed by our approach. First of all, the camera velocity is not constant. This is due to the fact that we needed to climb and down the stairs during the recording. This difficulty enables us to validate the ability of the filter to self-adapt under camera speed changes. By the way, when the camera is at the stairs, there are several images looking at white walls, which present very few image features and, consequently, means an interesting challenge. Moreover, the dataset presents some parts where illumination changes, what makes the camera adapt its operating parameters to the environment, generating several overexposed images. Some examples of these problems are shown in Fig. 11.

Despite the drawbacks described above, our approach is able to succeed, as it is shown in Table 5. It obtains a recall of 84% and and accuracy of 92%. If an overexposed image or with not enough features arrives at the filter, the full posterior does not present high peaks and a false negative is generated. When the image stream becomes stable, the algorithm reacts and starts detecting loop closures again. This shows that our approach is able to manage these challenging kinds of situations. Fig. 12 shows an example of loop closure detection from this indoor dataset.

Figure 11: Examples of images from the indoor environment. (Top, Left) First image in the sequence. (Top, Right) Image taken from the stairs. (Bottom, Left) Overexposed image. (Bottom, Right) Image after camera stabilization.

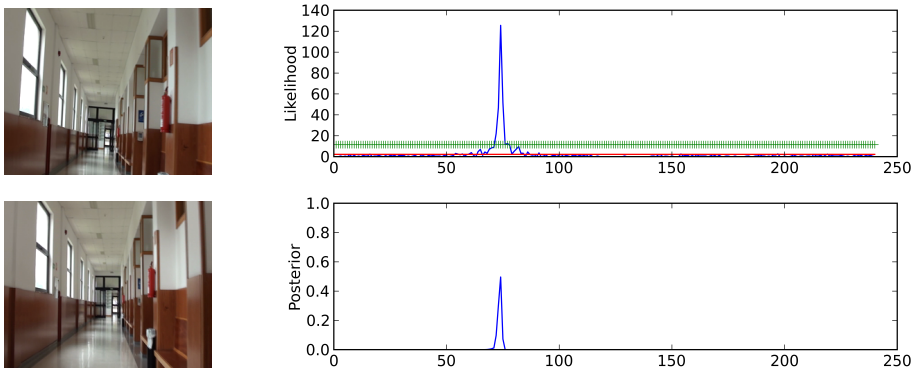


Figure 12: Example of loop closure detection from the UIB indoor dataset. Image 260 (Top, Left) closes a loop with image 73 (Bottom, Left). (Top, Right) Likelihood given the current image. (Bottom, Right) Full posterior after the normalization step.

6 Conclusions and Future Work

An appearance-based loop closure detection approach using a single monocular camera and SIFT features has been presented. When a new image is acquired, a discrete Bayes filter is used to obtain loop closure candidates, using a likelihood based on matching the current image descriptors with the descriptors of the previously-seen images in an efficient way. Then, the image that presents the highest probability in the full posterior is selected as a loop closure candidate. If this probability is higher than a threshold, a further validation step based on the epipolar geometry constraint is performed to confirm if both images can come from the same place. Otherwise, the loop closure hypothesis is rejected.

Despite other works make use of BoW approach for image representation, in this work raw local invariant features have been used. For managing features, an index based on a set of randomized KD-trees is employed. Experiments using datasets from different environments have been reported.

Referring to future work: (a) we want to use this algorithm for visual mapping and localization tasks for robotics; (b) matching images using other kinds of features, such as binary descriptors, is a solution to explore, since it can improve our approach in computational terms; (c) our Bayes filter can be executed in a GPU in order to speed up the loop closure detection; and (d) we would like to investigate the use of our approach in long-term tasks, reducing significatively the number of hypothesis in the filter.

Acknowledgment

This work is supported by the European Social Fund through grant FPI11-43123621R (Conselleria d'Educacio, Cultura i Universitats, Govern de les Illes Balears).

References

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms," *Robotics and Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[2] J. Sivic and A. Zisserman, "Video Google: A Text Retrieval Approach to Object Matching in Videos," in *International Conference on Computer Vision*, pp. 1470–1477, 2003.

[3] H. Zhang, "BoRF: Loop-Closure Detection with Scale Invariant Visual Features," in *International Conference on Robotics and Automation*, pp. 3125–3130, 2011.

[4] A. Kawewong, N. Tongprasit, S. Tungruamsub, and O. Hasegawa, "Online and Incremental Appearance-Based SLAM in Highly Dynamic Environments," *International Journal of Robotics Research*, pp. 33–55, 2011.

[5] H. Zhang, B. Li, and D. Yang, "Keyframe Detection for Appearance-Based Visual SLAM," in *International Conference on Intelligent Robots and Systems*, pp. 2071–2076, 2010.

[6] H. Zhang, "Indexing Visual Features: Real-Time Loop Closure Detection Using a Tree Structure," in *International Conference on Robotics and Automation*, pp. 3613–3618, 2012.

[7] E. Garcia-Fidalgo and A. Ortiz, "Probabilistic Appearance-Based Mapping and Localization Using Visual Features," in *Iberian Conference on Pattern Recognition and Image Analysis*, 2013.

[8] Z. Zivkovic, B. Bakker, and B. Krose, "Hierarchical Map Building Using Visual Landmarks and Geometric Constraints," in *International Conference on Intelligent Robots and Systems*, pp. 2480–2485, 2005.

[9] I. Ulrich and I. Nourbakhsh, "Appearance-Based Place Recognition for Topological Localization," in *International Conference on Robotics and Automation.*, pp. 1023–1029, 2000.

[10] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool, "Markerless Computer Vision Based Localization using Automatically Generated Topological Maps," in *European Navigation Conference*, pp. 235–243, 2004.

[11] D. G. Sabatta, "Vision-based Topological Map Building and Localisation using Persistent Features," in *Robotics and Mechatronics Symposium*, pp. 1–6, 2008.

[12] M. Cummins and P. Newman, "FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance," *International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[13] A. Angeli, S. Doncieux, J.-A. Meyer, and D. Filliat, "Real-Time Visual Loop-Closure Detection," in *International Conference on Robotics and Automation*, pp. 1842–1847, 2008.

[14] A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer, "A Fast and Incremental Method for Loop-Closure Detection Using Bags of Visual Words," 2008.

[15] A. Oliva and A. Torralba, "Modeling the Shape of the Scene : A Holistic Representation of the Spatial Envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[16] G. Singh and J. Kosecka, "Visual Loop Closing using Gist Descriptors in Manhattan World," in *International Conference on Robotics and Automation*, 2010.

[17] Y. Liu and H. Zhang, "Visual Loop Closure Detection with a Compact Image Descriptor," in *International Conference on Intelligent Robots and Systems*, pp. 1051–1056, 2012.

[18] C. Siagian and L. Itti, "Rapid Biologically-Inspired Scene Classification using Features Shared with Visual Attention," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 300–12, 2007.

[19] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.