Robotics and Autonomous Systems 59 (2011) 840–857

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Appearance-based mapping and localization for mobile robots using a feature stability histogram

B. Bacca [a,b,*], J. Salvi [b], X. Cufí [b]

[a] Universidad del Valle, Calle 13 No 100-00, A.A. 25360, Cali, Colombia
[b] Universitat de Girona, Campus Montilivi, Building PIV, 17071 Girona, Spain

ARTICLE INFO

Article history:
Received 9 December 2010
Received in revised form
26 May 2011
Accepted 9 June 2011
Available online 5 July 2011

Keywords:
Appearance-based
Localization and mapping
Topological maps
Omnidirectional vision

ABSTRACT

The strength of appearance-based mapping models for mobile robots lies in their ability to represent the environment through high-level image features and to provide human-readable information. However, developing a mapping and a localization method using these kinds of models is very challenging, especially if robots must deal with long-term mapping, localization, navigation, occlusions, and dynamic environments. In other words, the mobile robot has to deal with environmental appearance change, which modifies its representation of the environment. This paper proposes an indoor appearance-based mapping and a localization method for mobile robots based on the human memory model, which was used to build a Feature Stability Histogram (FSH) at each node in the robot topological map. This FSH registers local feature stability over time through a voting scheme, and the most stable features were considered for mapping, for Bayesian localization and for incrementally updating the current appearance reference view in the topological map. The experimental results are presented using an omnidirectional images dataset acquired over the long-term and considering: illumination changes (time of day, different seasons), occlusions, random removal of features, and perceptual aliasing. The results include a comparison with the approach proposed by Dayoub and Duckett (2008) [19] and the popular Bag-of-Words (Bazeille and Filliat, 2010) [35] approach. The obtained results confirm the viability of our method and indicate that it can adapt the internal map representation over time to localize the robot both globally and locally.

1. Introduction

These days mobile robots are needed to interact within non-structured environments. They must deal with people, moving obstacles, perceptual aliasing, weather changes, occlusions and robot–human interaction in order to have high levels of autonomy from a decision-making point of view, and to resolve mapping, localization and navigation issues as well as possible. These requirements are useful for service robots designed to conduct surveillance, inspect, deliver, clean and explore. In addition to localization, mapping and navigation problems, they have to guarantee a high level of autonomy through long-term navigation using stable features, which can be extracted from the environment structure or detected using artificial landmarks. Mobile robot mapping and localization methods can be: geometrical, aiming to estimate the absolute position of the robot and landmarks; topological, using graphs to estimate the environment topology; or hybrid, working on two levels, a high level with a topological map and a low level with metric sub-maps corresponding to each node in the graph [1].

In this paper, we consider topological localization and mapping. Topological maps are compact, consume less computer memory, can be stored in efficient data structures, and speed up the navigation process. They use graphs for environmental modeling, and vision sensors to provide the appearance of the environment [1,2]. The goal of a topological map is to obtain a configuration of the nodes in the graph that matches the robot environment topology. Our work focuses on the appearance-based methods to find this topology. The appearance observations we considered were laser scans and omnidirectional vision. The latter has received special attention recently due to its long-term landmark tracking, wide field of view, robustness to occlusions, ability to be fused with range data, and reduced noise sensitivity [3].

The goal of appearance-based methods is to use rich information of color, texture, or environment structure in order to find an association between two datasets, there by reducing the false positives in robot localization. Our review of the literature related to the appearance-based mapping and navigation show that these approaches have been introduced in recent years [4–25]. Most of the related papers use omnidirectional vision as the main sensor, complementing the appearance-based model with range measures, which are primarily used with 3D laser scans that limit their

* Correspondence to: Universitat de Girona, Campus Montilivi, Building PIV, Office 009, 17071 Girona, Spain. Tel.: +34 972 418976; fax: +34 972 418976.
E-mail addresses: bladimir@eia.udg.edu, evbacca@univalle.edu.co (B. Bacca), qsalvi@eia.udg.edu (J. Salvi), xcuf@eia.udg.edu (X. Cufí).

0921-8890/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2011.06.008

applicability in Simultaneous Localization and Mapping (SLAM). Real world environments are dynamic, but most of the cited works assume them to be static, affecting the feature description, possibly failing in the data association process and notoriously decreasing the robot mapping and localization.

Common solutions to these problems include: increasing the feature descriptor strength, so that their invariance to illumination change, rotation and occlusion increases [3,6,17]; building a new kind of image descriptors which depend on the type of captured images [9,11,16]; or fusing different sensors in order to have features with complementary information [11–15,25]. Depending on the technique used, soon or later landmarks are invisible due to illumination changes, occluded, or otherwise disappear definitively. Therefore, it is important to define a process to reinforce stable features, actively forget unstable features, and not forget occluded and stable features, which can recover their stability.

The motivation of our work is to improve appearance-based mapping and localization in long-term operation and in dynamic environments. We propose an indoor appearance-based mapping and a localization method whose main contribution is the Feature Stability Histogram (FSH). This is based on the human memory model [26] to deal with changing environments and long-term mapping and localization. It uses topological maps, such that each discrete location has its own updated appearance model. Unlike [19], we build a histogram using a voting scheme instead of a hard-wired finite-state machine, and we use the FSH to localize the robot by itself. This histogram stores the stability values of local features, while stable features are only used for localization and mapping. This innovative feature management approach for topological mapping and localization is able to cope with changing environments, long-term mapping and localization in the appearance space, and also contributes to the semantic environment representation.

The remainder of this paper is organized as follows. Section 2 discusses the related work which is focused on appearance-based techniques. Then, we explain our memory based model for mapping and localization in Section 3. Subsequently, we describe the system overview and assumptions in Section 4. Section 5 explains the mapping algorithm based on the feature stability histogram, which is included in a Bayesian robot localization framework explained in Section 6. Section 7 presents the experimental results using our implementation of the approach proposed by [19] and the popular Bag-Of-Words approach. Our final remarks are then given in Section 8.

2. Related work

Appearance-based methods for mapping and localization have gained increasing attention in recent years. These mapping techniques can be classified according to the environmental representation method used. Fig. 1 reviews some remarkable studies of appearance-based mapping and localization, where the type of sensor used was also kept in mind. The more obvious and basic approach is based on the global matching of images [23], where the mapping and localization is performed in two separate stages, known as an *offline* process. However, implementing a mapping and localization process in a concurrent way is a more desirable property.

An alternative approach for appearance-based mapping is to model the environment extracting feature descriptors, which are well known in the computer vision community as SIFT and SURF features [4,8,19,20,22,24,25], Discrete Cosine Transforms (DCT) [5], multidimensional histograms (color, edge, texture, gradient and rank) [6], homographies [7] and Fourier transforms [18]. All these methods use $L1$ (Manhattan distance) or $L2$ (Euclidean distance)

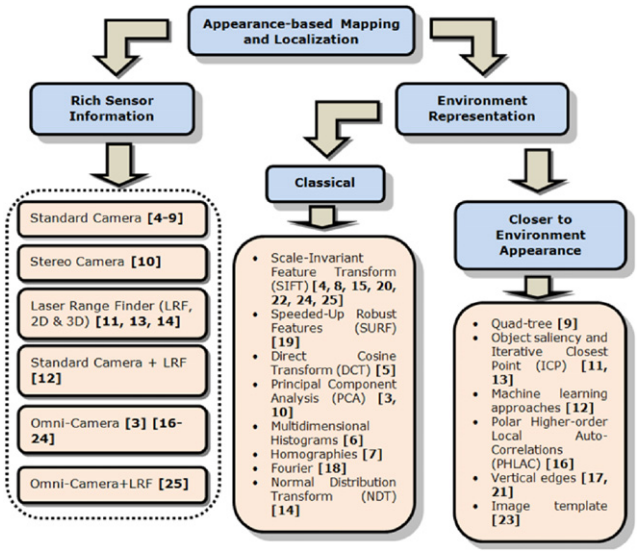

Fig. 1. Review of appearance-based mapping and localization approaches.

metrics to match feature descriptors, and in general terms the mapping and localization process is done assuming a static environment, where its local appearance is not updated over time. Approaches such as [5,8,18,25] involve motion estimation methods based on three-view geometry or essential matrix estimation. It is well known that the more stable a feature is, the more accurate is the motion estimation. Therefore, figuring out the more stable features will improve data association, which will result in better motion estimations. If a pure topological map is used, i.e. without geometric information involved, figuring out the more stable features will improve the topology estimation and further robot localization.

Other approaches take advantage of the type of sensors used (omnidirectional or standard camera, LRF, etc.) and environmental structures surrounding the mobile robot in order to represent the environmental appearance. Intuitively, the appearance-based model of the environment describes the environment as it is, taking advantage of its natural features. That is the case in [9] which uses image information content implementing quad-tree decomposition to find interesting places, of [16] which proposes a new descriptor focused on omnidirectional images called Polar High-Order Local Auto-correlation (PHLAC) and tested using a particle filter framework, but in a small environment and in short-term navigation. An interesting case is reported in [17,21] which uses omnidirectional vision and vertical lines as landmarks projected on the image plane without deformation. These landmarks have promising results because in omnidirectional vision they can be tracked for long baselines, decreasing the probability of being occluded and increasing the probability of being visible.

In contrast to the above approaches, which use one sensor, [25] uses sensor fusion between an omnidirectional camera and a 3D LRF. This approach takes advantage of the metric information provided by the LRF and mixes it with the omnidirectional vision [27]. Then it extracts the vertical lines in the environment and, using a scan matching technique, solves the SLAM problem. However, the authors do not consider occlusions, and illumination changes.

An approach close to our work is presented in [19]. They consider the Atkinson and Shiffrin memory model [26] in order to update the reference view of a particular place. However, they assume that the robot is able to self-localize using other means, since their main goal is to maintain the reference views of the topological map up to date. In contrast to their technique, our approach uses an adapted memory model (still inspired by

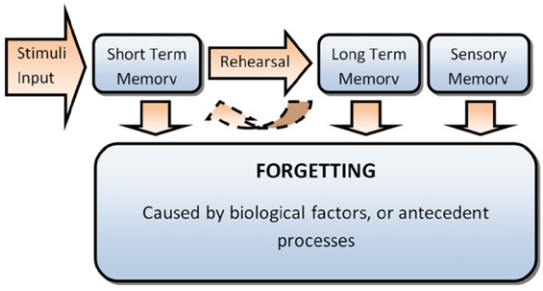

Fig. 2. Atkinson and Shiffrin model of human memory [26].

Atkinson and Shiffrin) in which we modified the way features are considered for robot mapping and localization. We changed the method to figure out if a feature belongs to the short-term memory or the long-term memory, and we used our model to localize the robot in its environment.

One motivation of our approach is to build a feature management system for appearance-based mapping and localization that is able to cope with changing environments and long-term operation. We designed our approach in a way that it can be applied to other mapping and localization techniques. Our system uses SURF [28] descriptors to describe the appearance of indoor environments and a topological map where each node describes a discrete location, and it stores a local appearance model using an FSH. The appearance model is used for mapping and localization extending the results achieved in [29].

3. On memory model based mapping

Early appearance-based mapping and localization approaches such as [23] use full image matching using image templates. Most of the recent appearance-based approaches do not consider updating the map appearance when there are changes in the environment. For years the scientific community has been finding inspiration in nature, even though probabilistic localization models have their origins in how the "place cells" in the hippocampus works. In our case, the Atkinson and Shiffrin memory model [26] can be used to distinguish stable features from unstable ones, and then use the stable features for robot mapping and localization. Fig. 2 shows the Atkinson and Shiffrin memory model, where four main components are shown: the Short-Term Memory (STM) which retains information long enough to use it; the long-term memory, which retains information for longer periods of time or lifetime; the Sensory Model, which was added afterward and was experimentally demonstrated to have the capability of the sensing organs to discriminate information for subsequent processing; the forgetting module, which affects all other components since it was experimentally demonstrated that memories can be forgotten through trace decay. This model proposes entering stimuli inputs in the STM. If these inputs are continuously rehearsed, they become part of the LTM. Information retained in LTM is recalled continuously in lifetime, but it does not reside permanently; if it is not rehearsed it can be forgotten. This memory model has been applied in robot mapping [19], and in robot control architectures [30].

The memory model proposed in [26] has drawn criticism from psychologists and neuroscientists due to its extremely linear representation of the memory process [31,32]. They argue that the Atkinson and Shiffrin model does not take into account the ability of many people to recall information despite the fact that this information has not been rehearsed. This phenomenon is more accentuated in autistic savants. In other words, apparently stimuli inputs can bypass STM to achieve LTM. In addition, this memory model does not consider different levels of memory [31,32]. From the robotics point of view, it would be useful to take into account levels of memory represented in the strength of the feature information.

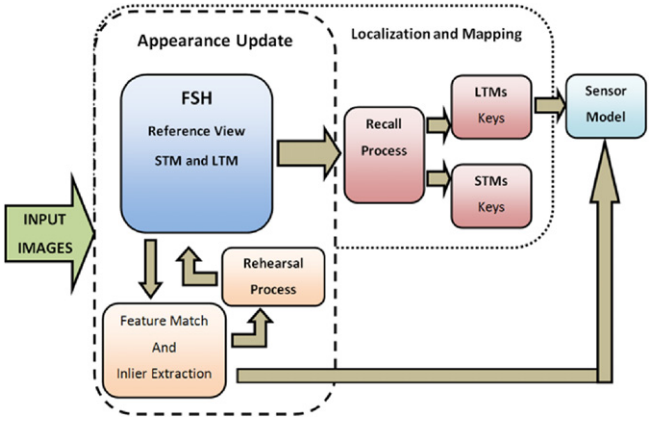

Fig. 3. Our appearance-based update approach.

In this work, an appearance-based update approach for robot mapping and localization inspired by the Atkinson and Shiffrin memory model is proposed (see Fig. 3). The reference view is composed of both memories, the STM and LTM, not only the LTM as in [19]. It has two main advantages: first, an input feature can bypass the STM and become an LTM, keeping in mind the feature strength, e.g., the feature uncertainty, the Hessian value in the SURF descriptor, or the matching distance; second, using the FSH as the reference view, the feature classification (STM or LTM) is not linear since the rehearsal process can take into account the feature strength. The rehearsal process implemented in our approach is based on the number of times a feature has been observed, but weighted by a function of the matching distance computed using a robust RANSAC outlier rejection and epipolar geometry constraint (Section 5). In this way, the appearance of the environment represented for the FSH is updated according to the presence or absence of pre-observed features, or the inclusion of new features whose vote, in our case, values are weighted by the normalized Hessian value. The rehearsal process proposed in [19] is based on a state machine, such that after four stages a feature is considered as LTM. When the robot starts the mapping and localization a method to distinguish STM and LTM features is needed. Our approach considers this situation when the voting scheme is weighted by a feature importance value.

Once the appearance of the environment is updated, the FSH can be used for mapping and localization. To do so, the recall process distinguishes between STM and LTM features, i.e. differentiating the most stable features (LTM) from the STM features. A feature descriptor is defined as an LTM if it has a high value in the FSH; otherwise it is considered an STM feature. This classification has two main advantages: first, it is a straightforward method to deal with temporal occlusions because, when using the voting scheme of the rehearsal process the FSH value of the corresponding feature suffers a relative decrease, or an increase if the feature is re-observed; second, it is a suitable method to deal with changing environments where illumination changes and pedestrians cause feature appearance or disappearance. In the end, the more stable features will belong to the LTM and will only be used for mapping and localization. The recall process implemented in our approach is threshold-based, i.e., FSH values greater than a threshold are considered LTM features, and those less than the threshold are STM features. The way this threshold was defined is further explained in Section 5.

Once the LTM features are found, they are used to build the sensor model in a Bayesian framework for mapping and localization. The sensor model proposed (Section 6) uses a similarity measure

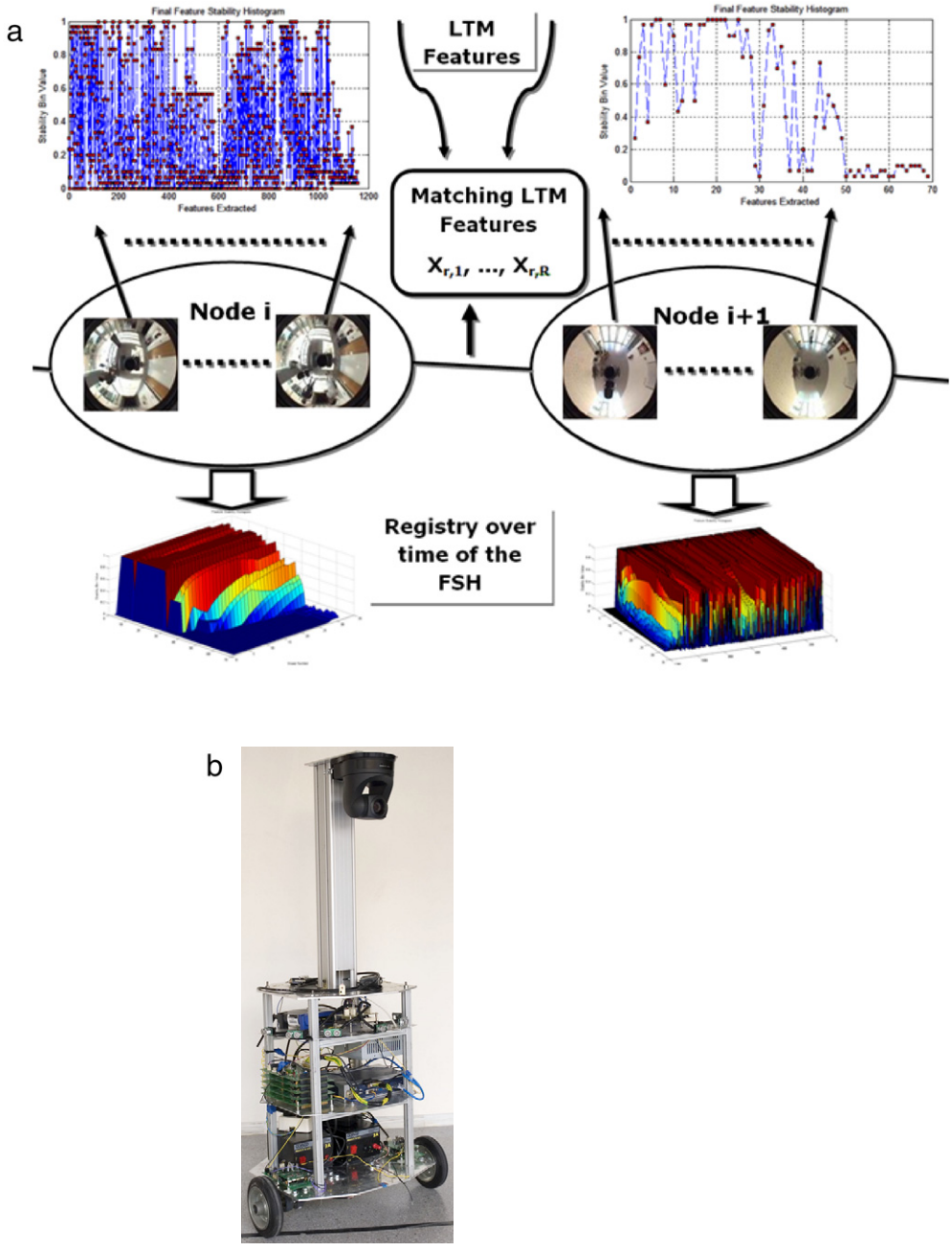

Fig. 4. (a) Internal structure of the topological map considering the feature stability histogram. LTM stands for long-term memory. (b) PRIM mobile robot.

based on the ratio of inliers found and the total number of LTM features of the reference view. The joint work done by the rehearsal and recall processes allows the appearance of the environment to be updated and a sensor model for mapping and localization to be obtained.

4. System overview

The proposed framework for appearance update, mapping and localization depicted in Fig. 3 considers a set of basic assumptions and other constraints related with the mapping and localization phases, which were used to conceive the framework.

4.1. Assumptions

We implemented the proposed approach in a custom-made differential mobile robot (see Fig. 4b). The robot was equipped with an omnidirectional vision setup composed of a *Remote Reality* parabolic mirror with a diameter of 74 mm and a Sony FCB-IX47AP color camera with a resolution of 640×480 pixels. Additionally, the robot was controlled by an embedded computer at 900 MHz. The omnidirectional vision system was the only sensor used and it was previously calibrated. Using these calibration parameters, a binary mask was computed to remove the central texture information where the robot is placed, and the outer texture information from the mirror. In addition, a look-up table (LUT) was computed to lift the omnidirectional image points to the equivalent spherical model. Finally, the mobile robot was assumed to have planar motion and its navigation used collision-free trajectories.

4.2. Topological mapping

Nowadays there are many mapping techniques such as dense 3D maps, sparse 3D maps, and topological maps. Topological

maps are compact, consume less computer memory, can be stored in efficient data structures, and are able to hold high-level information that can be used for semantic environment modeling. Our topological map is composed of several nodes, each of which stores one or more omnidirectional views. The number of views depends on whether a new image descriptor set is similar enough to the descriptors already stored. The structure of our topological map representation, illustrated in Fig. 4a, takes into account the following notation.

- A node defined by ni, $i \in \{1, \ldots, N\}$ where N is the number of nodes in the map; for global localization purposes this number was used to compute an equal prior probability for all nodes.
- In general, a node is composed of a set of SURF descriptors extracted from similar views, which are denoted by $Dn(i, j)$ where i is the node index, and $j \in \{1, \ldots, K\}$. Here, K is the number of feature descriptors stored within a node.
- A node in the topological map stores its own FSH built from the above descriptors. It is denoted as $fsh(i, t), i \in \{1, \ldots, N\}$ at time t, where t denotes the number of time stamps the FSH has been updated; also, the FSH evolution over time is stored and denoted by $rfsh(i, p), i \in \{1, \ldots, N\}, p \in \{1, \ldots, t\}$.
- Edges between nodes define neighboring relationships and store a set of corresponding features extracted from a two-view geometry process between nodes and denoted by $Ed_r, r \in \{1, \ldots, R\}$ where R is the number of edges between nodes. This relationship can be defined as: $Ed_r = match(Dn_i, Dn_{i-1})$, where Dn_{i-1} and Dn_i are the previous $(i-1)$ and current (i) set of SURF descriptors at each $(i-1)$th and ith node, and $match()$ denotes the matching process based on two main steps: a nearest neighbor search and then a robust epipolar geometry estimation using RANSAC for outlier rejection.
- Planar motion estimation is also included and denoted as $m_i = [x_i, y_i, \theta_i]^T$ and recovered up to scale using [33] and the essential matrix estimation done when the image features are robustly matched.

4.3. Mapping phase

The mapping and localization approach proposed is able to automatically construct a topological map from a set of training omnidirectional images, which are taken at regularly spaced intervals since the robot linear velocity is constant. In this work, each image was a node in the topological map, and they were taken with approximately 1 m separation between them. However, it is worth noting that this robot motion is estimated up to scale using only the omnidirectional vision sensor. Later, the topological map was updated eight times with other omnidirectional images obtained under different illumination conditions, times of the day, seasons of the year, and with walking people causing temporary occlusions.

In general terms, the map building algorithm is shown in Fig. 5. After image acquisition and feature extraction, a high similarity check is done to prevent robot stand-by images. Then, the robust matching process based on RANSAC and epipolar constraint begins and a similarity threshold is used to determine whether the robot is stopped or the current image belongs to a new node. If the robot is stopped, the FSH and its register are updated with the current image features. A new node is created taking into account the matching features between the last node and the new one for motion estimation. The current reference view is also updated keeping in mind our appearance-based approach, rehearsing the features in the FSH, and classifying them as STM or LTM.

5. Appearance-based mapping using feature stability histograms

This section presents the pipeline process of our approach. In first place, Algorithm 1 presents how the FSH is built, and then the details of the rehearsal and recall process are described. Also, the experimental support for the threshold selection is presented, as is the pruning technique to avoid the excessive increase of useless STM features.

Algorithm 1. Feature stability histogram algorithm.

```
Definitions:
  fshFeat: Reference features stored in the current node.
  tFshFeat: FSH registry over time.
  fshMatch: Matched referenced features.
  ltmFeat: LTM features of the current node.
  stmFeat: STM features of the current node.
  PRUNVIEW: Threshold to prune STM features.
  thresholdLTM: Threshold to distinguish between LTM/STM
  features
Inputs:
  currNode: Current node.
  currFeat: Current image features.
doFSH()
  // Data association.
  currFeat = getImageFeatures();
  fshFeat = getReferenceFeatures(currNode);
  fshMatch = getMatchedFeaturesThroughEpipolarConstraint
  (currFeat, fshFeat);
  // Feature stability histogram update.
  for (every fshFeat in fshMatch)
    if (fshFeat in ltmFeat)
      increaseFSHvalue(fshFeat); //Rehearsal process (Eq. (1))
    end
  end
  // Feature stability histogram pruning
  for (every fshFeat)
    if(fshFeat value < PRUNVIEW)
      doRemovalFeature(fshFeat);
    end
  end
  tFshFeat = getFSHnormalization(fshFeat);
  // LTM and STM update
  for (every tFshFeat)
    if (fFshFeat >= thresholdLTM)
      ltmFeat = updateLTMfeatures(tFshFeat); // Recall
      process
    else
    stmFeat = updateSTMfeatures(tFshFeat); // Recall process
    end
  end
end
```

Algorithm 1 shows an outline of the FSH creation. Two main inputs are needed, the current node where the FSH is stored and the current image features. In the first place, the data association is done using a robust descriptor matching. First, an $(i-1)$th image and a current image (i) are used to extract tentative correspondent features using the nearest neighbor method described in [28]. Second, the epipolar geometry is computed, given these tentative correspondences; u and p are image and mirror points in the first view, and v and q are image and mirror points in the second view. The essential matrix E was estimated using RANSAC such that for all correspondences $q^T E p = 0$ is satisfied. At this point, a look-up table was created to speed up the process of lifting the image points on the mirror's sphere-equivalent model. A correspondence is regarded as an inlier if in the second image the point v lies within a predefined distance from the conic specified by $v^T C v = 0$ where C is defined in Eq. (1) according to [34].

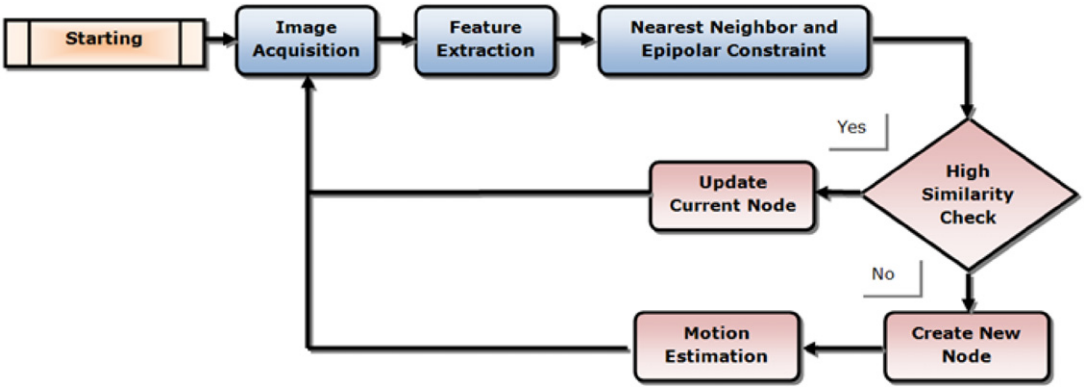

Fig. 5. Outline of the map building algorithm.

$$C = \begin{bmatrix} n_x^2\left(1-\xi^2\right) - n_z^2\xi^2 & n_x n_y(1-\xi^2) & n_x n_z \\ n_x n_y(1-\xi^2) & n_y^2\left(1-\xi^2\right) - n_z^2\xi^2 & n_y n_z \\ n_x n_z & n_y n_z & n_z^2 \end{bmatrix}. \quad (1)$$

Then, the rehearsal process updates the FSH values using the matching distance; this is done according to Eq. (2).

$$f(v, m) = v e^{-\frac{m^2}{\sigma_m^2}} \quad (2)$$

where, v is the nominal vote value (by default 1), m is the matching distance between each corresponding feature and σ_m^2 is the variance of the matching distances. Eq. (2) is valid for re-observed features, but if new features are detected, the feature strength is measured with respect to its Hessian value (SURF key-points) according to Eq. (3).

$$f(v, H_i) = v\left[\frac{H}{\|H\|}\right]_i \quad (3)$$

where, v is the nominal vote value (by default 1), and H_i is the normalized ith Hessian value of the new image feature.

The FSH in each node of the map includes STM and LTM features. According to our experiments, the number of STM features tends to increase. This happens because our approach implements a weighted voting scheme: if the feature is re-observed, it will be promoted; otherwise, it progressively decreases its FSH value. The intuition behind this situation is that computational resources are wasted trying to match these low importance features, increasing the complexity of the data association problem. These features whose FSH values remain at low levels are very often in real environments, because occlusions, walking people, bright spots on the floor and illumination changes are trigged as features. For scalability reasons, we want to maintain a low number of these features, so we have implemented a pruning method that depends on the amount of votes a stored feature has and the minimum number of votes a new feature should have. This is denoted in Eq. (4).

$$Fprun_i = \left\{ vFSH \,|\, vFSH < \frac{1}{\max(FSH)} \,\&\, actView > PRUNVIEW \right\} \quad (4)$$

where, *actView* is the current number of re-observations, *vFSH* is an individual value of the FSH that is related with each feature descriptor in a node, *PRUNVIEW* is a constant which specifies from which map update the pruning process starts (see Algorithm 1), and max(*FSH*) defines the actual maximum value of the FSH. The condition expressed in Eq. (4) is quickly checked for all FSH values, and those that satisfy this equation are removed from the FSH. However, *PRUNVIEW* has to be carefully selected to avoid storing lots of useless features, or deleting new ones which could become future LTM features.

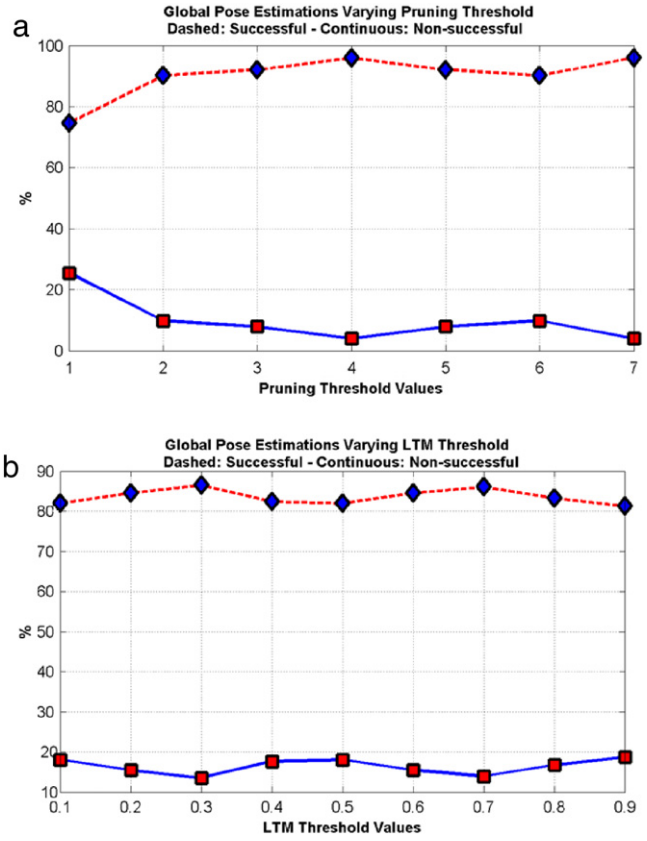

Fig. 6. (a) Successful and non-successful position estimations versus pruning threshold. (b) Successful and non-successful position estimations versus LTM threshold.

We conducted another experiment to figure out the most suitable *PRUNVIEW* value. It consisted of varying the pruning threshold value and the map update as well, and then generating 100 random image sequences to test the successful and non-successful global position estimations. The results are shown in Fig. 6a, where the dashed curve represents the successful localization results and the continuous curve, the non-successful localization results. This figure shows a maximum value of 96.08% of successful results when the *PRUNVIEW* value is 4. A pruning threshold of 1 causes our localization algorithm to have less successful results, because the new features added to the topological map are deleted before they become LTM features. On the other hand, a pruning threshold of 7 causes a scalability

problem: too many features are stored in the FSH, which is very problematic for long-term mapping and navigation and data association. Therefore, a pruning threshold of four is a good compromise between these two extremes.

In [19], this is done in a simpler way. Once the difference between the current view and the current node is computed, for every feature in this set which remains in the first state, it is definitively removed. Our approach is more conservative. Basically, any feature has 3 more chances to increase its FSH value before it is deleted.

The next step in Algorithm 1 is the LTM and STM classification. According to how the FSH is built, a good way is to select a threshold such that FSH values greater than the threshold are considered LTM features, and those less than the threshold are STM features. But, the FSH values change continuously, so an arbitrary threshold (e.g. 0.5) could cause some problems: very few LTM features could be obtained which is not good for appearance-based mapping causing a lack of representativeness for robot localization; a high number of LTM features could increase the data association uncertainty and decrease scalability. We conducted an experiment where we used 100 random image sequences to test the successful and non-successful global position estimations for each change of 0.1 in the threshold value. The results are shown in Fig. 6b, where the dashed curve with diamonds shows the successful localization results, and the solid curve with squares shows the non-successful localization results. As can be seen, two possible values can be extracted: 0.3 or 0.7. If we selected a threshold value of 0.3, it would select many LTM features, which is inconvenient from representativeness and scalability point of view. On the other hand, a threshold of 0.7 (successful localization results of 86.1%) is a good commitment between the inconveniences described above. Then, this value is used in the algorithm as *thresholdLTM*.

Until now we have defined two important thresholds: one to distinguish LTM from STM features and another one to perform a pruning of STM features. We performed two statistical experiments to define them, and we did not assume any constraint with respect to the feature extraction.

6. Robot localization framework

Once the topological map of the environment is built, it can be used for robot localization. Given an omnidirectional image of the surroundings of the robot, the aim of our localization algorithm is to find the node where the robot is likely to be. Note that such a node is related to a world position in the environment.

In general, robot localization involves global and local localization problems. Global localization is considered when no a priori information about the location is available. This situation occurs when the robot is initialized or gets lost. In contrast, if the robot knows the current pose, local localization deals with tracking the robot motion along subsequent poses. This paper proposes an appearance-based mapping and probabilistic localization approach to deal with both global and local localization using the appearance update approach described in Fig. 3.

SURF features are used to describe the appearance of the environment. They are normally used as local features, rather than global ones. Perceptual aliasing in the environment can confuse robots, because the visual appearance of the environment is similar at two or more different locations. However, the proposed mapping and localization approach based on the human memory model allows us to find the most stable features, and reduce the location ambiguity. This can be achieved up to a level of uncertainty that directly affects the position estimation. Therefore, to constrain such uncertainty, a Bayesian filtering-based approach assigns a probability value at each topological location.

Bayesian filters are used for probabilistic estimations of the system dynamic state when noisy observations gather at time t and actions are done at the same time. In particular, our state is defined as a node $x \in \{n_1, \ldots, n_N\}$ in the topological map, e.g. $x_t = n_4$ means the robot at time t is in the node 4; the observation $z_v = v_{desc}$ done at time t is composed of SURF descriptors from the current image. Then, our localization problem formulation is: given a collection of LTM features $Z = \{Dn_1, \ldots, Dn_N\}$ such that $z(x_i) = Dn_i$ relates a location with an observation in the map, the goal of the localization algorithm is to find the node location x_t that matches the current image.

The Bayesian filter recursively calculates the posterior state distribution $p(x_t/z_{1:t})$ as the probability of being at node x_t at time t. We use Bayes' rule to define this posterior distribution as depicted in Eq. (5).

$$p\,(x_t|z_{1:t}) = \frac{p(z_t|x_t, z_{t-1}, \ldots, z_0)p(x_t|z_{t-1}, \ldots, z_0)}{p(z_t|z_{t-1}, \ldots, z_0)} \tag{5}$$

where the denominator can be replaced by a normalization factor, since it does not depend on x_t, and keeping in mind that Bayesian filters assume that the dynamics of the system are Markovian, which means that future locations do not depend on past locations. Eq. (5) can be expressed as Eq. (6) allowing us to calculate the location estimation recursively since $p(x_{t-1}/z_{1:t-1})$ is the last location estimation.

$$p\,(x_t|z_{1:t}) = \alpha p\,(z_t|x_t) \sum_{x_{t-1}\in\{n_1,\ldots,n_N\}} p\,(x_t|x_{t-1})\,p\,(x_{t-1}|z_{1:t-1})\,. \tag{6}$$

Eq. (5) has two unknown distributions: $p(x_t, x_{t-1})$ and $p(z_t/x_t)$. The first, called the motion model, expresses the probability transition between two node locations in the topological map. To define it, we enforce the temporary coherence of the node position estimation and assume that transitions between closer places are more likely than transitions between more distant node locations. We model this as a Gaussian distribution centered at x_t and expressed in Eq. (7),

$$p\,(x_t|x_{t-1}) = \gamma e^{-\frac{\|x_t - x_{t-1}\|}{\sigma_x^2}} \tag{7}$$

where γ is a normalization constant, $\|x_t - x_{t-1}\|$ is the distance between the two nodes in the topological map, and σ_x^2 is the variance of the distances on the map. The second unknown probability distribution in Eq. (7) is $p(z_t/x_t)$, or the sensor model. In our case the sensor model is computed keeping in mind the appearance update approach described in Section 3, where a set of LTM features from the topological map are compared with the current view z_t at x_t such that a similarity measure is computed as depicted in Eq. (8).

$$s_{v,i} = \frac{match\,(v_{desc}, Dn_i)}{\sqrt{match\,(v_{desc}, Dn_i) * Dn_R}} \tag{8}$$

where v_{desc} is the new image descriptor set, Dn_i is the ith LTM node descriptor set, $match(v_{desc}, Dn_i)$ is the number of corresponding features between the new image and the LTM features of the current node, and Dn_R is the number of total LTM features in the current node. We have defined a geometrical average in the denominator of Eq. (8) to constraint the influence of high values of Dn_R, and avoid giving more weight to the stored features than the new image descriptor set. Then, given a set of LTM features stored in the topological map $Z_{LTM} = \{Dn_1, \ldots, Dn_N\}$, the probability of the current observation z_t at x_t belongs to a particular node in the map is depicted in Eq. (9).

$$p\,(z_t|x_t) = \delta e^{-\frac{sim(z_t, Z_{LTM})}{\sigma_z^2}} \tag{9}$$

where δ is a normalization constant, $sim(z_t, Z_{LTM})$ is the similarity measure defined by Eq. (8), and σ_z^2 is the variance of this measure.

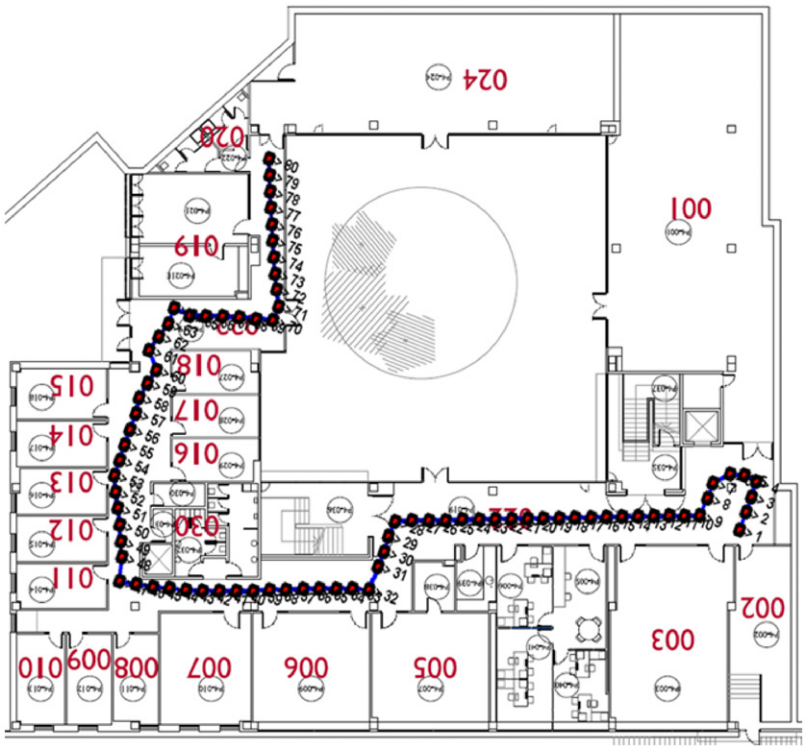

Fig. 7. Camera motion estimation and topological map representation. The motion estimation was placed on the building plans.

As a result of perceptual aliasing, the sensor model can have more than one maximum value. To overcome this inconvenience and avoid discarding other location hypotheses in the case of initialization or getting lost, Eq. (9) was modified as a sum of Gaussians. The number of Gaussians corresponds to the number of peak values between the maximum of the similarity measure $sim(z_t, Z_{LTM})$ minus σ_z, and the weight of each Gaussian is given by the corresponding peak of the similarity measure, as denoted in Eq. (10); then the assumed sensor model is shown in Eq. (11).

$$l = \max_{Z} \{sim\,(z_t, Z_{LTM}) - \sigma_Z\} \tag{10}$$

$$p\,(z_t|x_t) = \delta_{sum} \sum_{l} w_l e^{-\frac{sim\left(z_{t,l}, Z_{LTM,l}\right)}{\sigma_l^2}} \tag{11}$$

where δ_{sum} is a normalization factor and w_l is the mixture weight, which equals the peak value corresponding to the hypotheses after applying Eq. (10). In this way each new hypothesis is proportional to its similarity measure. It is worth noting that for localization purposes the similarity measure considers the LTM features only. This means the more updates are performed in a node, the more certain is the robot position in the topological map.

Algorithm 2 describes how the Bayesian topological localization is done using the framework explained above, and how the LTM features are used to build the sensor model in order to obtain an estimated node position based on the surrounding appearance. In the first place, the motion model is built using the previous node estimation as depicted in Eq. (7). Then, using the current image features, the more likely nodes where the robot is expected to be are predicted by Eq. (7), and the LTM features of those nodes are used to build the sensor model. Once the robot position in the topological map is estimated, we use this information to update the map with the current features and the FSH algorithm described in Algorithm 1.

Algorithm 2. Bayesian localization using the FSH.

Definitions and inputs:
mapT: Topological map.
currFeat: Current image features.
currNode: Current node estimation.
rMotion: Probabilistic motion model.
rSensor: Probabilistic sensor model.
appSimil: Appearance similarity.
ltmFeat: LTM features of a specific node.
*Xt*_1: Previous node position estimated.
*Pt*_*t*: Previous node position uncertainty.
Xt: Estimated node position.
Pt: Estimated node position uncertainty.
doBayesLocalizationFSH()
currFeat = getImageFeatures();
// Prediction
rMotion = getMotionModel(*mapT*, *Xt*_1, *Pt*_1); // Eq. (7).
// Build the sensor model, Eqs. (8) and (9).
for (every node in rMotion)
ltmFeat = getLTMfeatures(rMotion);
appSimil = getAppearanceSimilarity(currFeat, ltmFeat);
rSensor = doBuildSensorModel(appSimil);
end
rSensor = doSumOfGaussians(rSensor); // Eqs. (10) and (11).
// Node position update
[*XtPt*] = getEstimatedNodePosition(rMotion, rSensor);
// Eq. (6)
// Update map
doFSH(*Xt*, currFeat);
end

7. Experiments

The experiments conducted to test our approach were classified into two groups: first, a static experiment in order to observe the image similarity behavior with and without the environment appearance representation update; second, the global and local

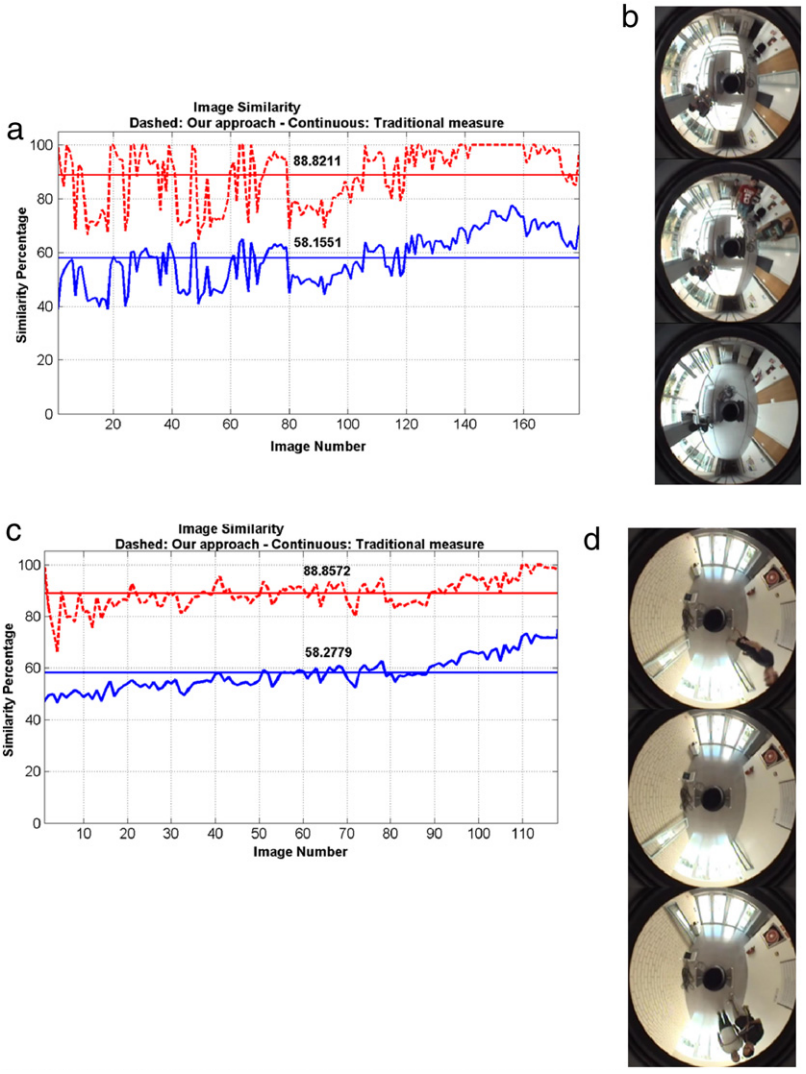

Fig. 8. Static image similarity test. (a) and (c): similarity measure of a place close to big windows, node 20 in the map, and a place close to the cafeteria (node 66). (b) and (d): examples of typical omnidirectional images of both places.

localization of the robot. The purpose of the first experiment is to show how dynamic environments affect similarity measures, and how our approach updates the appearance representation of the environment. In the case of the second set of experiments, our approach is compared with the method proposed by [19] and the Bag-of-Words technique that nowadays is becoming popular for appearance-based mapping and localization [35–37].

Our approach was tested using a value of 0.7 for the threshold value in order to distinguish LTM from STM features in the FSH. Also, the pruning threshold used was 4. Note that these threshold values were experimentally selected according to the experiments described in Section 5. All experimental tests (static and topological localization) use real world images which were captured day and night, in summer, fall and spring, such that a database of 640 images was obtained and they were used to update the appearance of the environment. This dataset was divided into eight equal parts, each one is considered a map update. The images contain a wide variety of occlusions (people walking by and standing), noise, and changes in illumination (day, night and seasons) to ensure a good enough appearance update of the environment.

In addition, one additional image set was taken at completely different positions and orientations compared to the image dataset described above. These images had not been seen previously by the robot. With this set, our approach, the method proposed by [19] and the Bag-of-Words technique were tested using a Bayesian-based simulation framework. For each image in this last dataset, the real topological node in the map was stored, which allowed us to extract the position error (in the topological space) between the real and the estimated location node.

Fig. 7 shows the environment representation and the node locations obtained in the topological map we built. The environment selected was the first floor of the Computer Engineering Department at the University of Girona, where there are normally a lot of people passing by, different passages that are quite similar to each other, and big windows (between nodes 12 to 28, 63 to 66 and 70 to 80) that allow big changes in illumination due to changing weather conditions and seasons. The camera motion estimation was placed on the building plans for visualization purposes only.

The method proposed by [19] was implemented with 4 and 5 stages in STM and LTM finite-state machine, respectively. The feature extraction was done using the SURF algorithm over the original omnidirectional image; these features were not computed over the unwrapped panoramic image as in [19]. The motion model assumed was defined by Eq. (7), and the sensor model by Eq. (9) keeping in mind the reference view specified by [19].

Bag-of-Words methods [36] based their environmental representation on a set of unordered features (the visual words) taken

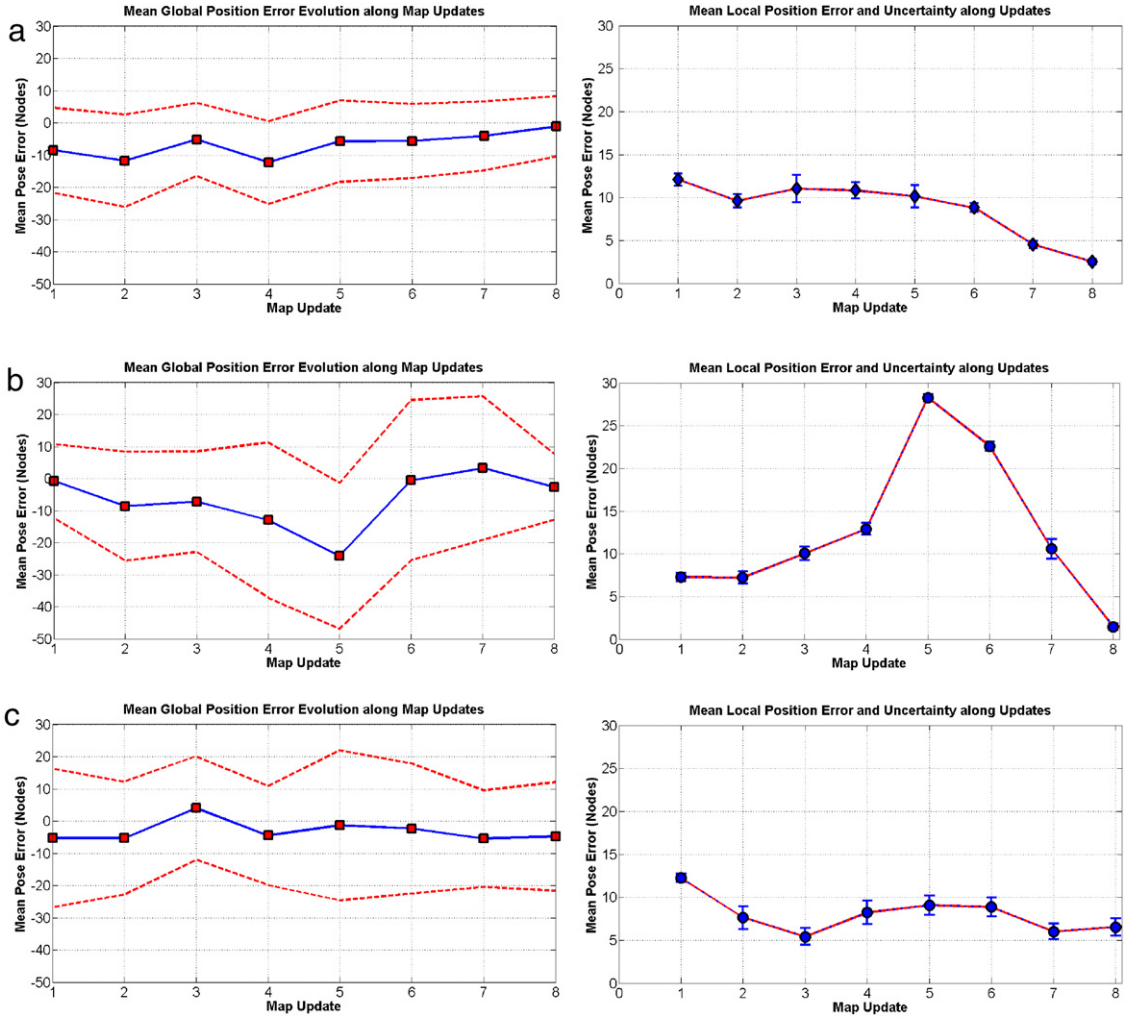

Fig. 9. Mean position error of global and local localization along map updates without noise or artificial occlusion. (a) Our approach. (b) The method proposed in [19]. (c) The Bag-of-Words method.

from a dictionary. The dictionary is built using a clustering technique, commonly *k-means*, and then image classification is based on the occurrence of the visual words in an image to infer its class. The dictionary is built beforehand in an offline process. The matching process is based on a Nearest Neighbor (NN) search among the distance separating the corresponding visual words. The Bag-of-Words toolbox used was the Caltech large scale image [38]. Our implementation uses SURF features computed on the original omnidirectional image. The clustering process involves 80 different classes corresponding to each node in the topological map. Each time the map was updated, the dictionary and clustering process were generated. The motion model assumed was defined by Eq. (7), and the sensor likelihood model was the *term-frequency–inverse document frequency (tf–idf)* weighting depicted in Eq. (12).

$$tf--idf = \frac{n_{wi}}{n_i} \log \frac{N}{n_w} \tag{12}$$

where, n_{wi} is the number of occurrences of word w in an image I_i, n_i is the total number of words in I_i, n_w is the number of images containing word w, and N is the total number of images seen so far.

7.1. Static image similarity test

The motivation of our work is to improve appearance-based mapping and localization in long-term operation and in dynamic environments to detect the most stable features in the environment and then use these features for mapping and localization. Our approach was evaluated in one node of the topological map in order to see how it updates the environment information, detecting the most stable features, and computing the image similarity. We conducted a static experiment in which important changes in the environment were present. Three main changes were considered: changes in illumination due to weather conditions, passers-by causing temporary occlusions and moving the furniture present.

Fig. 8a compares the similarity percentage and the image number at node 20, which is close to a big window ensuring real world conditions due to illumination changes. The 180 images were acquired over five days. Fig. 8a shows both similarity measures: the dashed curve was made using our approach, whereas the continuous curve was created without updating the environment appearance. Fig. 8b shows three examples of the typical omnidirectional images obtained at this node, where one can observe the changes in illumination and occlusions due to passers-by. The image similarity means were 88.82% and 58.15% for our approach and without the appearance update, respectively. Dynamic environments as shown in Fig. 8b cause low similarity measures when the representation of the environment is not updated accordingly, but in the case of an LTM-based similarity

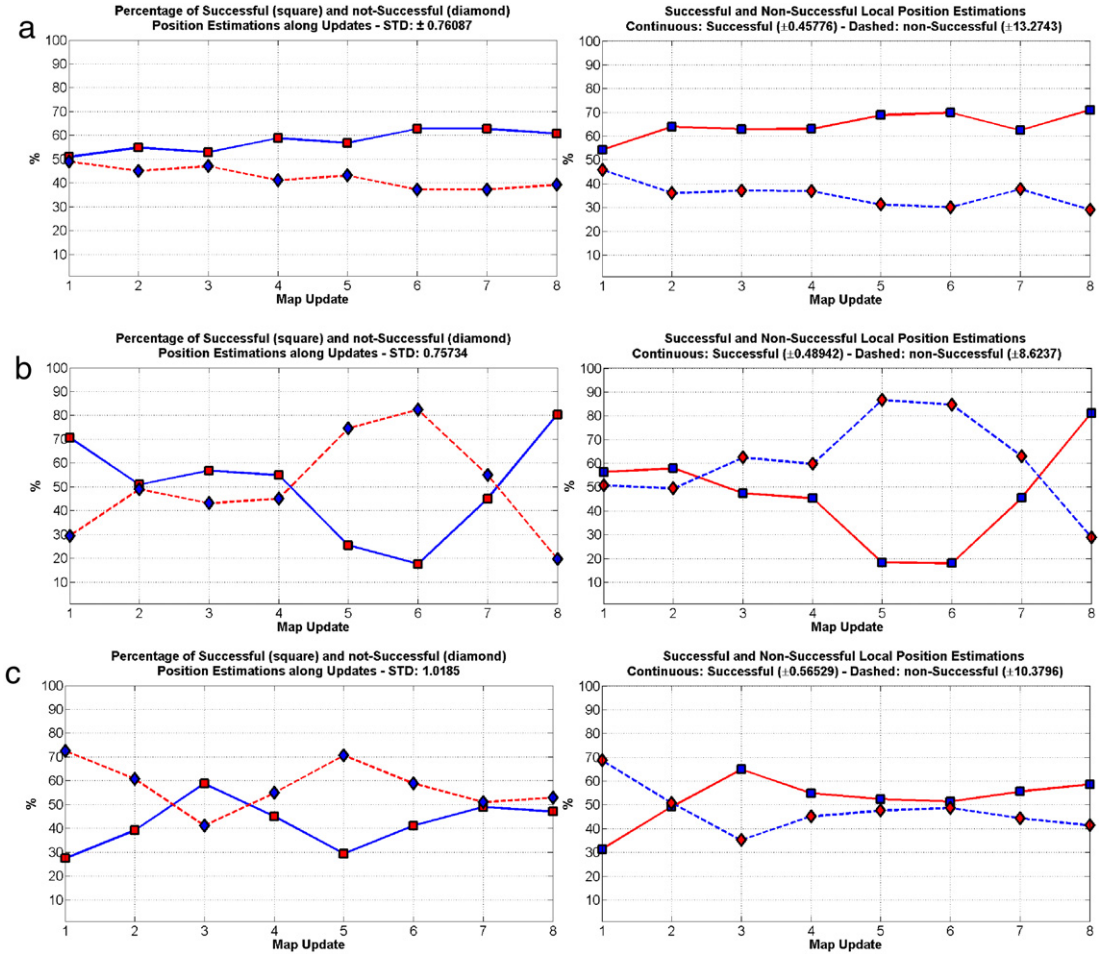

Fig. 10. Successful and non-successful global and local position estimation along map updates without noise or artificial occlusion. (a) Our approach. (b) The method proposed in [19]. (c) The Bag-of-Words method.

measure, this effect is reduced because most LTM features remain and a good representation of the environment is maintained.

A second place was also selected. It is close to a cafeteria area at node 66. In this place a second static experiment, whose results are shown in Fig. 8c and d took place. Again, our approach performed better than the classical approach without appearance update.

7.2. Global and local topological localization

As described at the beginning of this section, comparative global and local topological localization was performed, including the approach proposed by [19], the Bag-of-Words method and our approach. Four types of tests were done: the first one used the original set of test images, without noise or artificial occlusion; in the second a Gaussian noise with $\mu = 0$ and $\sigma = 0.15$ was added to the current image, but without artificial occlusion; in the third and fourth tests a Gaussian noise with $\mu = 0$ and $\sigma = 0.15$ and artificial occlusion was added by randomly removing 25% and 50% of the current image features, respectively.

To evaluate the localization performance, 100 random image sequences were generated from the test dataset for each experiment. In both global and local localization, the estimated location was selected using the winner-takes-all approach. Since we have the real node that each image belongs to, the mean position error in the topological space can be obtained using the 100 random image sequences. Successful position estimation means that the maximum value of the posterior Gaussian belief is within ± 1.25 nodes around the real node location in the map.

At each random image sequence, global localization was evaluated using the first image of the sequence. In this way, we ensured that no previous knowledge about the location was available, since this is the first localization attempt for each image sequence. The remaining images in the sequence were used to evaluate local localization, since in this case the localization algorithm deals with tracking the robot motion along subsequent poses.

7.2.1. Global and local localization without noise or artificial occlusion

Fig. 9 shows the mean position error of global and local localization along the map updates and its uncertainty bounds (3σ). Fig. 9a–c correspond to the results obtained using our approach, the method proposed in [19] and the Bag-of-Words approach implemented using the Caltech toolbox [38], respectively. The left and right sides of each figure show the mean position error for global and local localization, respectively. Because the global localization error was measured using the first image in the random sequence, wide uncertainty bounds are expected to be present. It is also expected that the mean pose error will approach zero as the map updates increase. The experimental results for global localization of Fig. 9 show that our approach, which uses only visual information to figure out where the robot is placed in the topological map, presents a lower position error uncertainty than the others. In our approach, the mean error position for global localization tends to approach zero as the map updates increase. In the method proposed by [19] the effect of the number of states in the finite-state machine for the STM features (4) can be seen, because when the

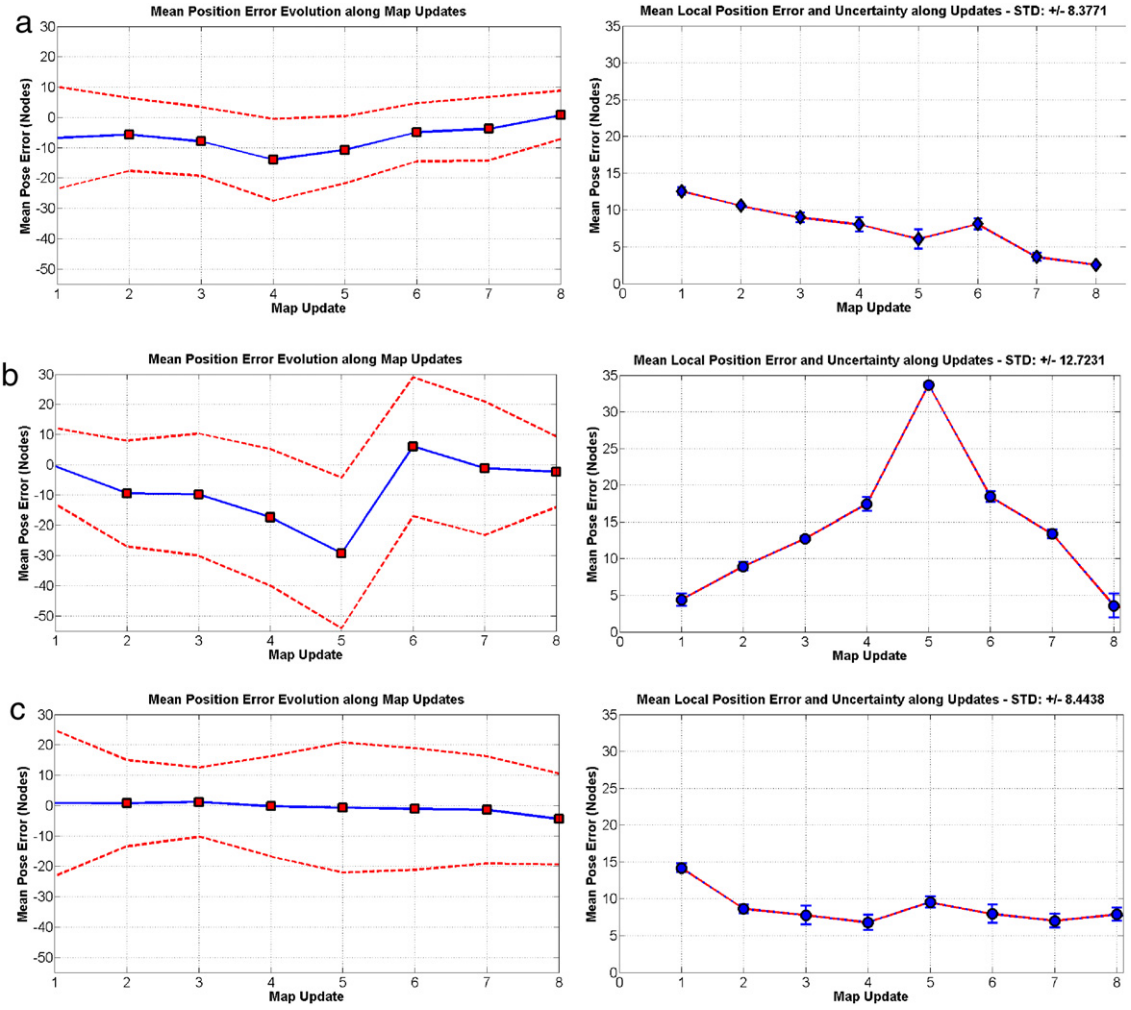

Fig. 11. Mean position error of global and local localization along map updates with Gaussian noise, and without artificial occlusion. (a) Our approach. (b) The method proposed in [19]. (c) The Bag-of-Words method.

map update 5 was presented to the system, it started to decrease the mean position error and was approaching zero, which is not the case for our approach. The Bag-of-Words method holds a more or less constant global position error, but its uncertainty bounds are greater than our approach.

The mean error for local localization behaves in a similar way. As the map updates increase, the mean error tends to decrease. However, in our approach the effect of the environmental appearance update is more evident than in the other approaches. The method proposed by [19] is highly affected by the number of states in the STM finite-state machine, and in the Bag-of-Words approach there is an increase of the mean position error between map updates 4 and 6, which belong to spring and summer according to our dataset. This caused big changes in illumination and bright spots which were considered as features but without meaningful environmental appearance information. It is worth noting that we are dealing with real world images, where natural changes in illumination, walking people and occlusions influence the feature extraction and cause the increase and decrease of the mean position error for the three methods tested. However, Fig. 9 shows that our approach deals with these situations better.

Fig. 10 shows the percentage of successful and non-successful global and local position estimations. Note that our appearance-based approach performs the mapping and localization in the topological space. Successful position estimation means that the maximum value of the posterior Gaussian belief is within ± 1.25 nodes of the real node location in the map; this position estimation was also considered for the other methods tested. Fig. 10 has the same visualization format as the one for Fig. 9 described above. The global localization results of Fig. 10 show that our approach outperforms the method proposed by Dayoub and Duckett [19] and the Bag-of-Words method, because the LTM features obtained from our reference representation (the FSH) are able to maintain the representativeness of the environmental appearance from the beginning. This is done thanks to the weighted voting scheme proposed in Eqs. (2) and (3). The method proposed by [19] tends to increase its percentage of successful position estimations, but after the features state overcomes the STM finite-state machine. Indoor datasets are full of perceptual aliasing. This becomes a great challenge for the Bag-of-Words method because in global localization it finds many position hypothesis within the internal Bag-of-Words environment representation, which is given by the voting schema over the visual words.

Observing the experimental results for the successful and non-successful local position estimations, our approach holds its tendency to increase the successful position estimations as the map updates increase. The method proposed by [19] again suffers the consequences of the delayed appearance update representation, but in the end its percentage increases drastically. Despite the fact the map updates 4–6 are challenging, the Bag-of-Words method shows a positive difference between the successful

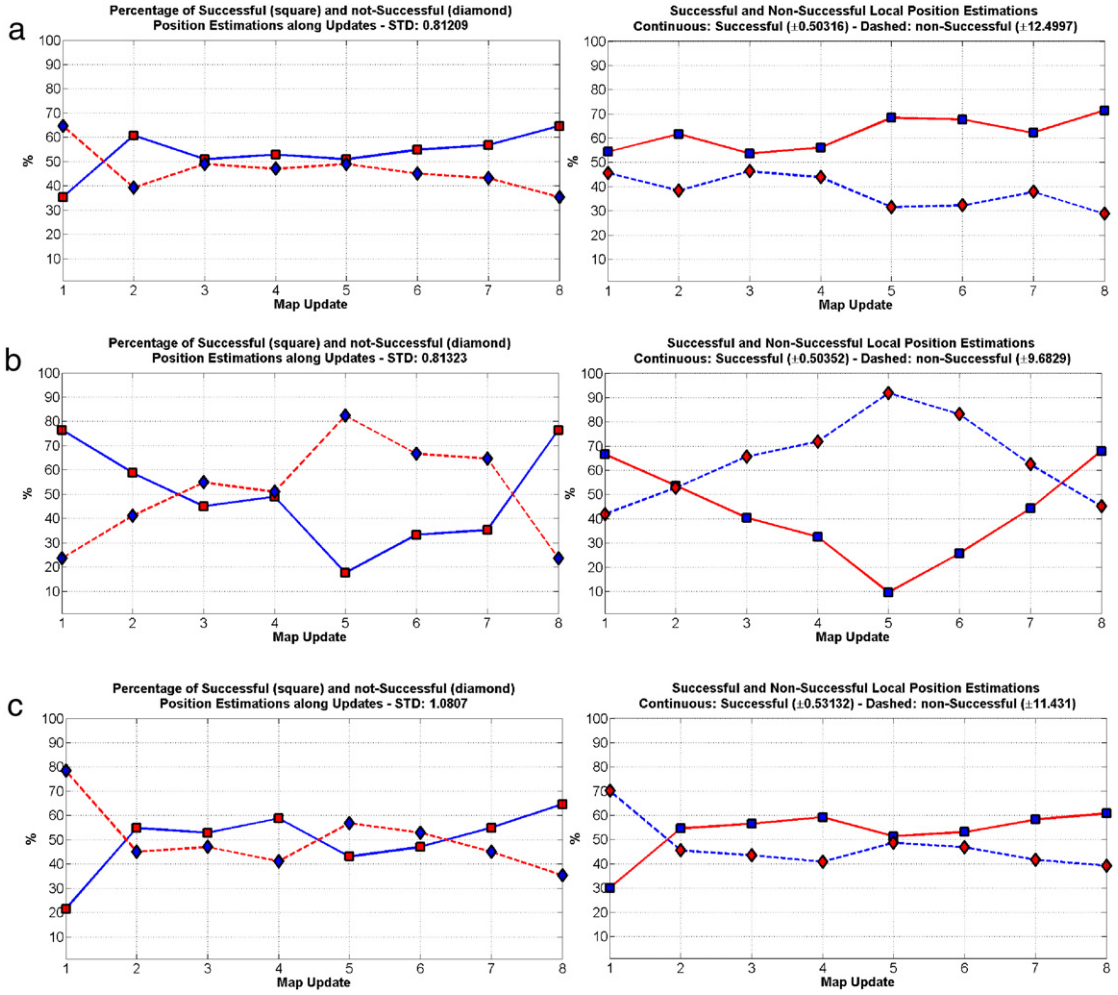

Fig. 12. Successful and non-successful global position estimation along map updates with Gaussian noise, and without artificial occlusion. (a) Our approach. (b) The method proposed in [19]. (c) The Bag-of-Words method.

and non-successful position estimations along the map updates. Fig. 10a and b show clear experimental differences between our approach and the method proposed by [19]; these differences are the consequence of changing the reference view model at each node in the topological map for the FSH, which considers the strength of the environment features and then classifies them as STM or LTM features. In addition, the suitable thresholds are used to distinguish LTM from STM features, and to prune useless features stored in the FSH.

7.2.2. Global and local localization with Gaussian noise, no artificial occlusion

Fig. 11 shows the mean position error for global and local localization using a corrupted input image, but without artificial occlusion. This section and the two following it aim to evaluate our approach in the presence of Gaussian noise and occlusions. These occlusions are artificially generated by randomly removing a percentage of the input features. Note that these occlusions are in addition to the ones naturally present in the original omnidirectional images in our dataset. For the mean position error in global localization, the last three approaches have a mean error close to zero, but the levels of uncertainty are lower in our approach. This also means that our matching and outliers removal method is performing well despite the Gaussian noise added.

From the local localization point of view, Fig. 10 shows that in the end the mean position error is similar to that achieved without adding Gaussian noise in the three methods tested, but again the levels of uncertainty are lower in our approach. The method proposed by Dayoub and Duckett [19] shows the negative effect of having a hard-wired finite-state machine for the rehearsal stage in the STM. The Bag-of-Words method shows an increase of the uncertainty as in both global and local mean position error between map updates 4 and 6, despite a two-view geometry check is being done.

Fig. 12 shows the percentage of successful and non-successful global and local position estimations in the presence of Gaussian noise in the input image. For global localization, it is observed that at the beginning the noise added has a negative impact in our approach, but one map update more is enough to have a positive difference between successful and non-successful position estimation. This difference increases as the map updates increase, which does not happen with the other two approaches. This demonstrates that our approach coherently deals with the original illumination changes and occlusions, and the Gaussian noise added. This can be observed in the continuous curve going up and down on the left side of Fig. 12b and c.

The right part of Fig. 12 shows the local position estimations. Clearly, our approach outperforms the other two approaches, since it always shows a positive difference between the successful and non-successful position estimations. Observing the experimental results between Figs. 10 and 12, the Bag-of-Words method performs better than the method proposed by Dayoub and Duckett [19], whose major weakness is the finite-state machine conception, which adds a highly sequential component to the feature classification process.

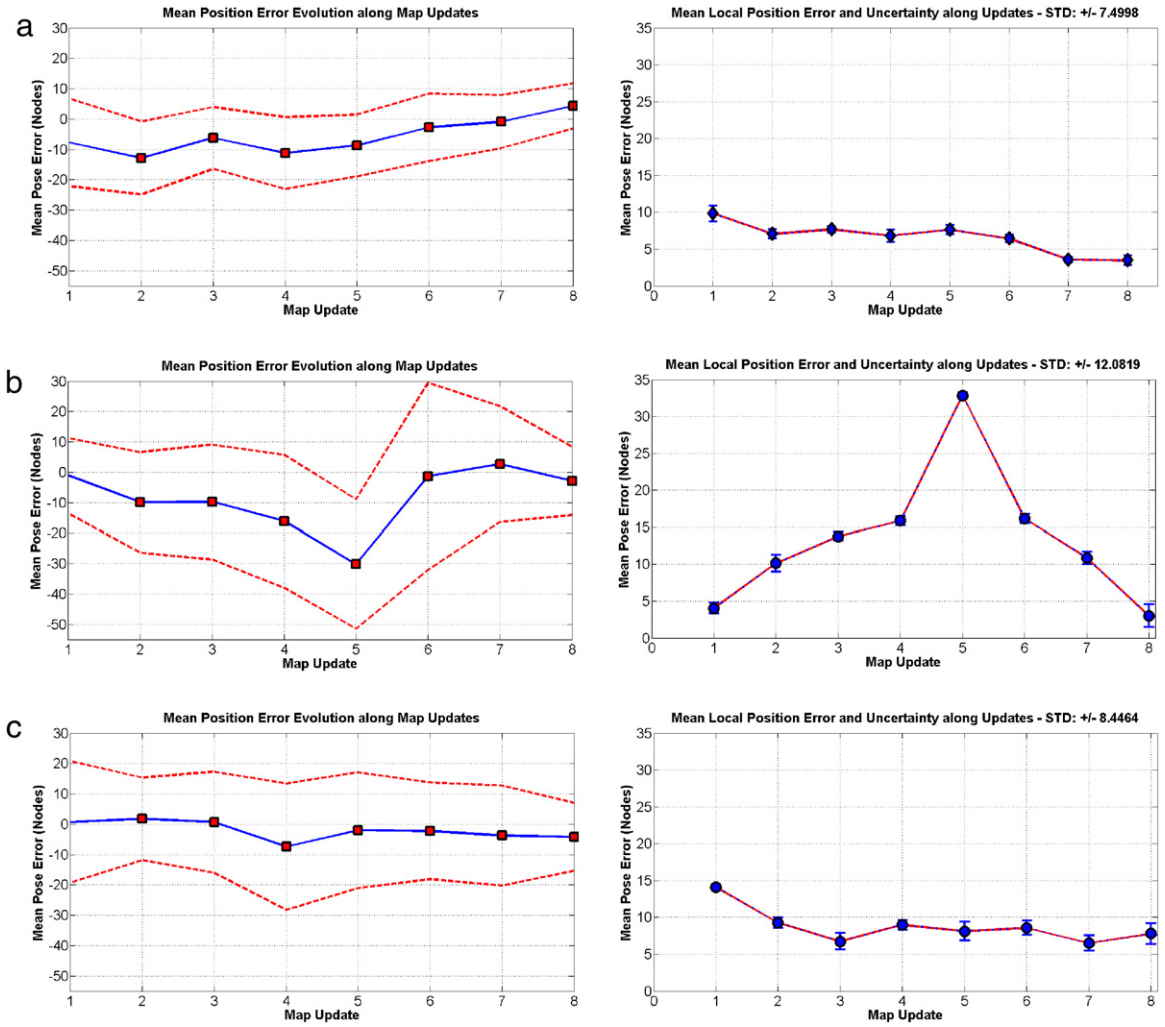

Fig. 13. Mean position error of global localization along map updates with Gaussian noise, and with artificial occlusion of 25%. (a) Our approach. (b) The method proposed in [19]. (c) The Bag-of-Words method.

7.2.3. Global and local localization with Gaussian noise and artificial occlusion of 25%

In this section, we describe how the input images are corrupted with Gaussian noise and artificial occlusion of 25%. The artificial occlusion was implemented by randomly removing features from the input image. As described earlier, this occlusion is in addition to what is naturally present in the original omnidirectional images, and is caused by pedestrians, bright spots or illumination changes due to season or time of the day. The left part of Fig. 13 shows the mean position error for global localization with Gaussian noise and artificial occlusion of 25%. Our approach maintains a decreasing error position to zero along the map updates, and has a lower uncertainty level than the other approaches. In terms of local localization error, the method proposed by [19] still presents an error peak at map update 5, and the Bag-of-Words method behaves similarly to our approach. Comparing the experimental results of Figs. 11 and 13, our approach and the method proposed by [19] behave in an especially similar way. Despite the noise and artificial occlusion added, our approach maintains a good representativeness of the environment's appearance encoded in the LTM features, which allows it to better estimate the robot position than the other approaches. This position estimation is done saving time, computing resources and storage, because the Bag-of-Words method requires the creation of a new dictionary each time the map is updated, and the size of the dictionary greatly depends on the number of images and features. For instance, our approach has an appearance-based map size of 7.73 Mbytes at the eighth update, while in the Bag-of-Words method the map at the eighth update has a size of 64.9 Mbytes.

The percentages of successful and non-successful global and local position estimations are presented in Fig. 14. Here, our approach has some difficulties at the beginning, but after the third map update the successful position estimations are better. It is noteworthy that although the global localization estimation might be erroneous, the local position estimation is correct. This means that the sensor model selection as a sum of Gaussians was correct, because it allowed our approach to recover from wrong global locations, and then progressively obtain good localization estimations as the image sequence continued. The main idea behind this was that the robot would continue its path to collect more evidence and estimate where it is, which was not far from reality because active map building and localization algorithms often use this technique [39].

Given these challenging conditions of noise and artificial occlusion, the method proposed by [19] increases the peak values of non-successful position estimations, since less features in the STM finite-state machine are promoted to the LTM. Despite the fact that the Bag-of-Words method finally achieves a positive difference between the successful and non-successful position estimations, the noise and the artificial occlusion added increases the effect of its main weakness: dealing with perceptual aliasing, which is very common in indoor environments.

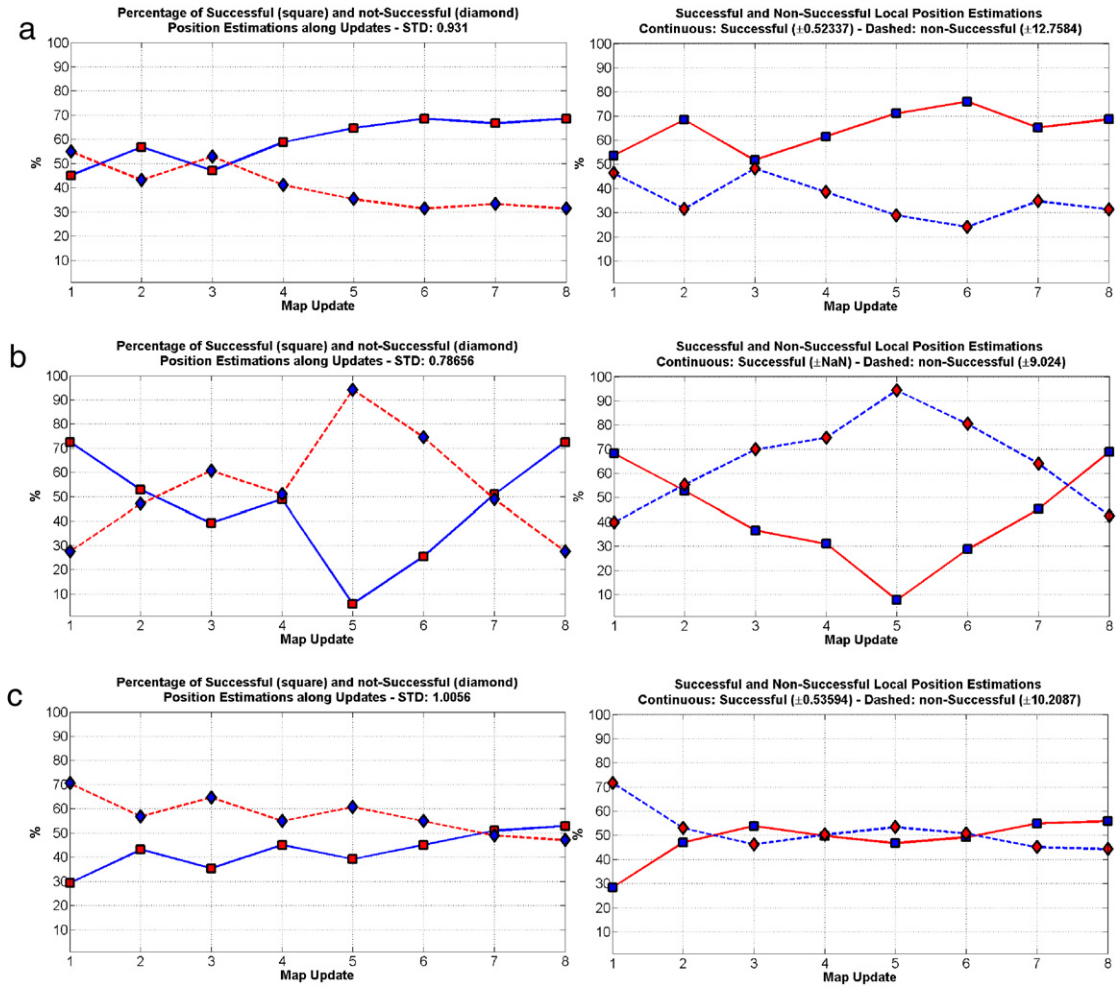

Fig. 14. Successful and non-successful global position estimation along map updates with Gaussian noise, and with artificial occlusion of 25%. (a) Our approach. (b) The method proposed in [19]. (c) The Bag-of-Words method.

7.2.4. Global and local localization with Gaussian noise and artificial occlusion of 50%

Fig. 15 shows the experimental results for the global and local mean position errors when the input images were corrupted with noise and artificial occlusion of 50%. Fig. 15a shows how our approach performed well on this challenging test, the global position estimation approaches zero and the local position estimation decreases as the map updates increase. The similitude of the results presented for our approach in all the four of these tests is evidence of the importance of getting a suitable appearance representation of the environment. For instance, the threshold selection to distinguish STM from LTM features was statistically and experimentally found, as in other computer vision studies did such as the Fast–Hessian/Hessian–Laplace selection in SURF [28]. In our case, if the threshold is too low, lots of LTM features are considered, which causes perceptual aliasing. If the threshold is high, fewer LTM features are considered and the reference view in each node lacks representativeness of the environment appearance.

Fig. 15b shows the global and local position error for the method proposed by [19]. The error peak at map update five is still present, but one might wonder what happens if the rehearsal stages at the STM are reduced? This test was performed, but the mean position error did not decrease at the same rate as our approach, because many LTM features were considered as a reference view and most of them did not deserve to be promoted. In addition, the discrete increments of the state in the finite-state machine framework do not support a real value, which causes a lack of flexibility choosing the more suitable number of states. Fig. 15c shows the results for the Bag-of-Words method which has some similarities with the results of our approach, but it involves an offline process, the uncertainty levels are bigger and as described in Section 7.2.3 the time, computing and storage costs are high.

Fig. 16 shows the percentages of successful and non-successful position estimation in the presence of Gaussian noise and artificial occlusion of 50%. Fig. 16a shows the experimental results for our approach, which after the fourth map update achieve a positive difference between the successful and non-successful global position estimation. This figure also shows evidence of how the assumed sensor model allows progressive improvements in the local position estimation. The increasing tendency of the successful local position estimation curve is also evidence of how the posterior Gaussian belief progressively shrinks.

Fig. 16b and c show the experimental results for the method proposed in [19] and the Bag-of-Words method. The former recovers in the end, but while the reference view at each node does not take into account the LTM features the non-successful position estimations prevail. The latter has no successful global position estimation at all, but the local position estimations are not reliable.

7.2.5. Scalability

An important motivation behind this work is being able to deal with large environments and long-term navigation. Then, the mean size of the LTM features set can provide evidence for the

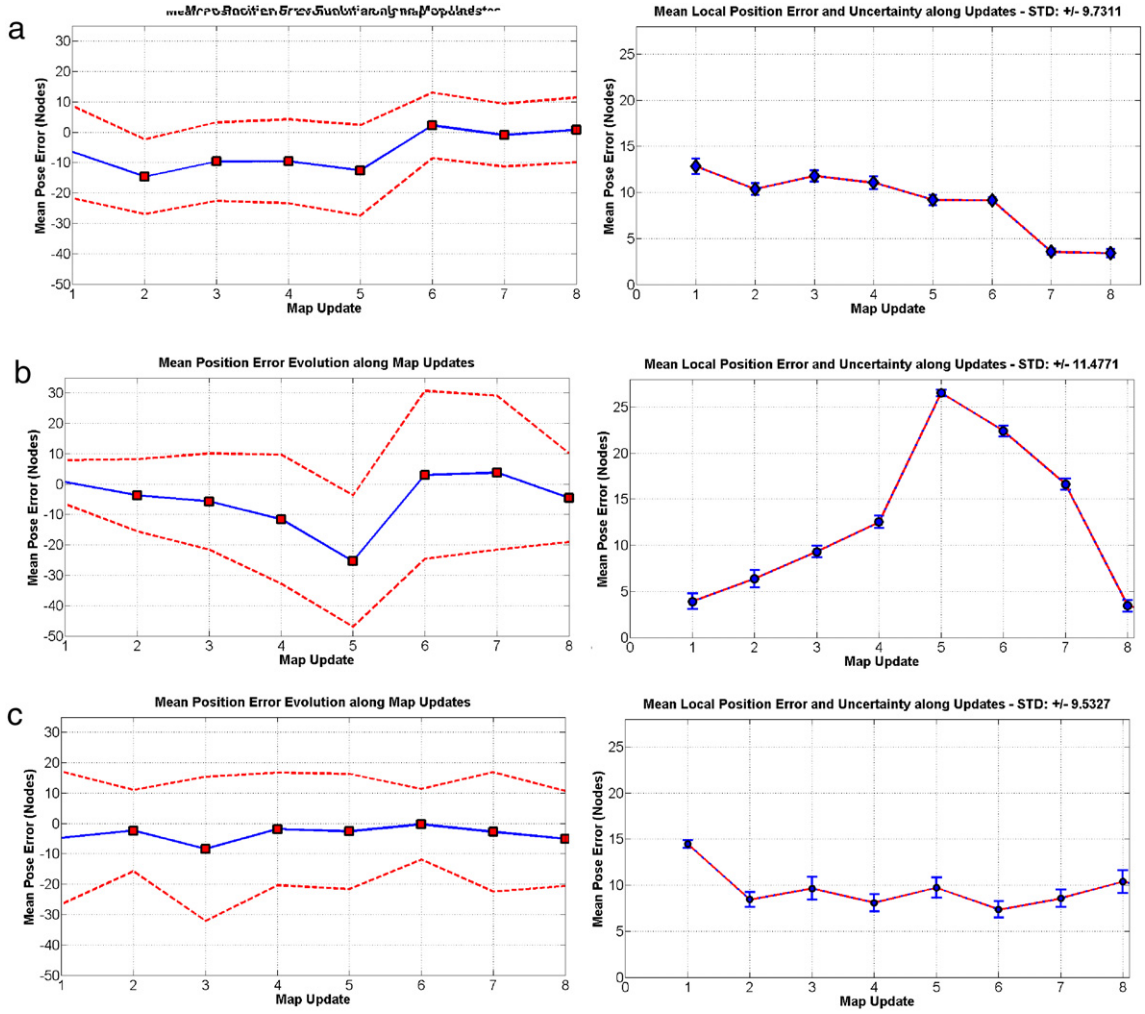

Fig. 15. Mean position error of global and local localization along map updates with Gaussian noise, and with artificial occlusion of 50%. (a) Our approach. (b) The method proposed in [19]. (c) The Bag-of-Words method.

scalability of our approach. Fig. 17 shows the evolution of the number of LTM and STM features as the map updates increase. The continuous curve with diamonds shows the evolution of the number of LTM features, the dotted curve with circles the evolution of the number of STM features, and the dashed curves their corresponding uncertainty. Fig. 17 shows that the number of LTM features remains almost constant, or at least it does not exceed 284. Although the number of STM features has a tendency to grow after initialization, after map update 4 it is reduced thanks to the pruning method discussed earlier, and remains almost constant or at least does not exceed 480 features after the fifth update. This means that our approach deals well with large environments, because LTM features are only used for robot mapping and localization.

8. Conclusions

This article has proposed an innovative feature management approach for topological mapping and localization and appearance-based indoor environment representation. Our approach, based on a modified human memory model, implements concepts such as Long-Term (LTM) and Short-Term Memory (STM) as mechanisms to classify features as either stable or non-stable. Unlike other approaches, our method considers a weighted voting scheme to outperform the Atkinson and Shiffrin memory model linearity, which allows our proposed appearance-based mapping and navigation approach to pass to the STM only those strong features in the environment. LTM and STM concepts were applied to topological mapping and localization using a Feature Stability Histogram (FSH), which stores at each node statistics about what features have been observed repeatedly. STM and LTM features are distinguished using a threshold, and only LTM features were used for robot mapping and localization. Using the weighted voting scheme implemented using the FSH, our method can deal with temporal occlusions caused by dynamic environments and illumination changes caused by time of the day and seasons.

Our method was tested in static and dynamic environments. The former included two sets of images acquired over a long period of time in order to show that our approach and the image similarity measure proposed offer better results than a static description of the environment. The latter used a topological map which was updated as many as eight times and a Bayesian-based localization approach for global and local localization experiments. According to these results, the FSH, the weighted voting scheme and the classification of STM and LTM features seems like a promising way to improve appearance-based mapping and localization. The LTM features obtained from our reference representation (the FSH) are able to maintain the representativeness of the environmental appearance from the beginning of the map creation, which does not happen with the other methods we compared. The sensor model assumed as a sum of Gaussians has been proven correct, because it allowed our approach to recover from wrong

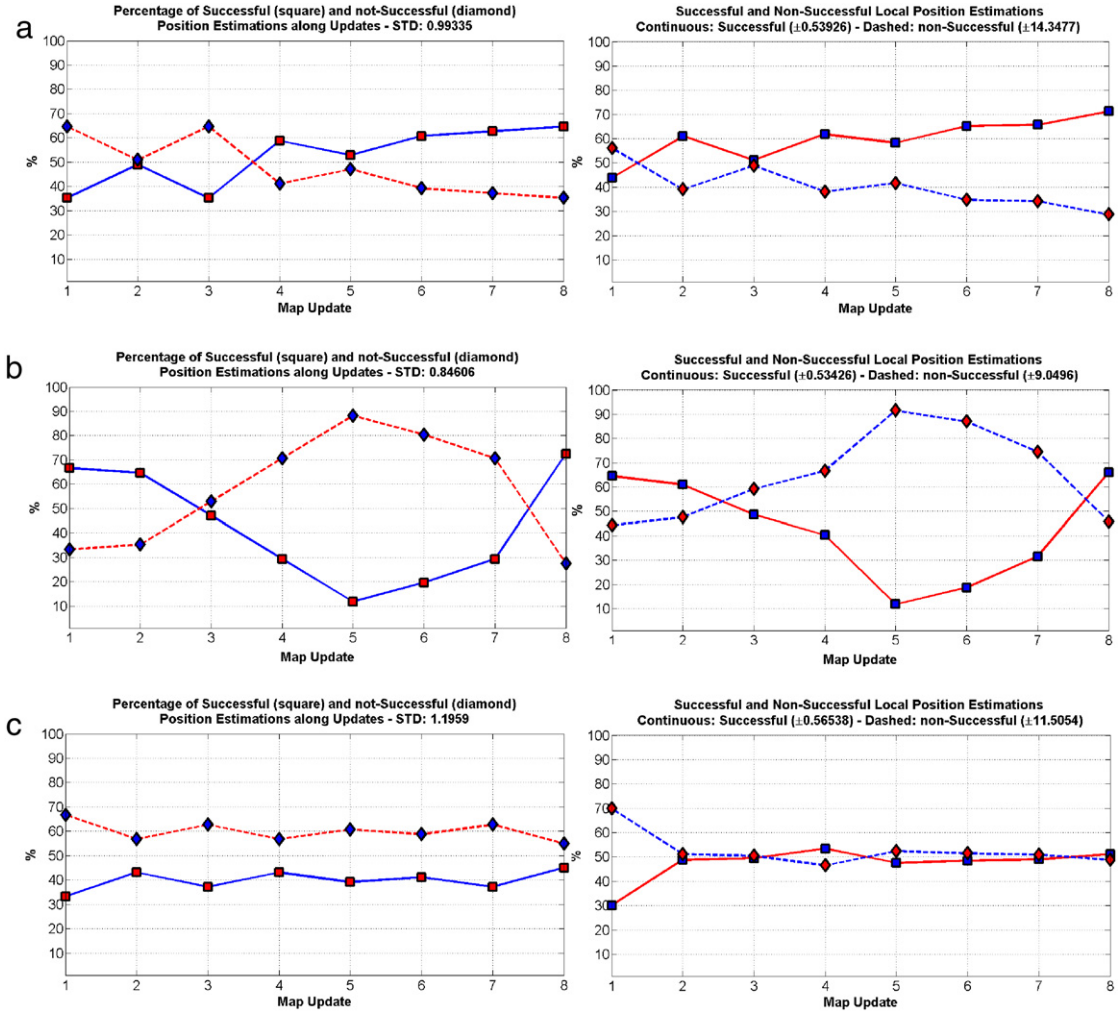

Fig. 16. Successful and non-successful global and local position estimation along map updates with Gaussian noise, and with artificial occlusion of 50%. (a) Our approach. (b) The method proposed in [19]. (c) The Bag-of-Words method.

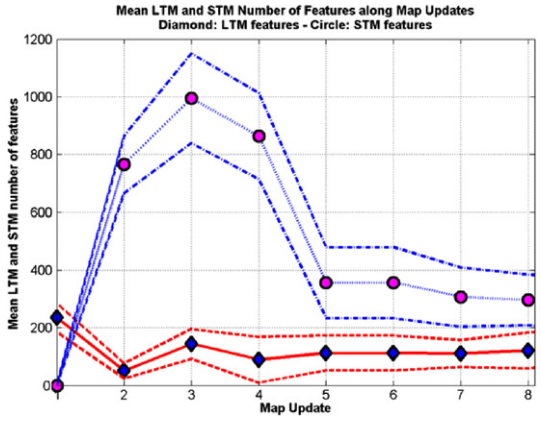

Fig. 17. Mean number of LTM and STM features as map update increases.

global locations, and then progressively obtain good localization estimations as the image sequence continued.

The major weakness of the method [19] is its finite-state machine conception which adds a highly sequential component in the feature classification process. This inconvenience is solved in our approach thanks to the weighted voting scheme, which allows an earlier feature classification (STM or LTM features), based on the feature strength (uncertainty). Perceptual aliasing becomes a great challenge for the Bag-of-Words method, because in global localization it finds many position hypotheses within the internal Bag-of-Words environment representation. The effect of perceptual aliasing is considerably lower in our approach than in the Bag-of-Words method, because our method uses the FSH, which maintains a local representation of the environment appearance. Normally, the big changes in illumination and bright spots have no meaningful environmental appearance information, but the Bag-of-Words method does not have an explicit way to distinguish them from relevant features. In our case, the FSH and the feature classification in STM or LTM seems a promising way to extract relevant or stable features from the environment.

Pure topological maps give coarse localization estimations, but they behave very well in long-term navigation and large environments, and they are a suitable method for appearance-based environment models. For instance, our approach deals well with large environments, because our tests have shown that the number of LTM features was maintained within reasonable limits. Finally, in general, SURF features behaved well in our experiments, but they are not features with enough representative information to be used as global features; so, another improvement to our approach will be focused on the use of features that are closer to an appearance-based model of the robot environment as proposed in [40], using a fusion sensor method as described in [27], rather than interesting key points. Despite this additional issue, our approach is able to perform a global and local localization with the results shown.

As future work, the STM/LTM threshold will be learned depending on how the environment changes, and the pruning threshold according to the FSH value exponential decay. However, in this work these threshold selections gave good localization results, and it is worth noting that the dataset used was collected during different seasons of the year, ensuring a wide variety of illumination changes, walking people and in general a dynamic indoor environment.

Acknowledgments

This work has been supported by the publicly funded Spanish project DPI-2007-66796-C03-02, the LASPAU-COLCIENCIAS grant 136-2008, the University of Valle contract 644-19-04-95, and the consolidated research group's grant SGR2005-01008.

References

[1] T. Bailey, H. Durrant-Whyte, Simultaneous localisation and mapping (SLAM): part II state of the art, Robotics and Automation Magazine 13 (2006) 108–117.
[2] H. Durrant-Whyte, T. Bailey, Simultaneous localisation and mapping (SLAM): part I the essential algorithms, Robotics and Automation Magazine 20 (2006) 99–110.
[3] J. Gaspar, N. Winters, E. Grossmann, J. Santos-Victor, Toward robot perception through omnidirectional vision, Studies in Computational Intelligence 70 (2003) 223–270.
[4] A. Angeli, S. Doncieux, J. Meyer, D. Filliat, Incremental vision-based topological SLAM, in: Intelligent Robots and Systems, IEEE/RSJ International Conference on, 1, 2008, pp. 1031–1036.
[5] J. Porta, B. Krose, Appearance-based concurrent map building and localization using a multi-hypotheses tracker, Intelligent Robots and Systems, in: IEEE/RSJ International Conference on, 4, 2004, pp. 3424–3429.
[6] C. Zhou, Y. Wei, T. Tan, Mobile robot self-localization based on global visual appearance features, in: Robotics and Automation, in: Proceedings ICRA, IEEE International Conference on, 1, 2003, pp. 1271–1276.
[7] A. Remazeilles, F. Chaumette, P. Gros, 3D navigation based on a visual memory, in: Robotics and Automation, ICRA, in: Proceedings IEEE International Conference on, 1, 2006, pp. 2719–2725.
[8] S. Šegvić, A. Remazeilles, A. Diosi, F. Chaumette, A mapping and localization framework for scalable appearance-based navigation, Computer Vision and Image Understanding 113 (2) (2009) 172–187.
[9] V.A. Sujan, M.A. Meggiolaro, F.A. Belo, Information based indoor environment robotic exploration and modeling using 2-D images and graphs, Autonomous Robots 21 (1) (2006) 15–28.
[10] J.M. Porta, J.J. Verbeek, B.J. Kröse, Active appearance-based robot localization using stereo vision, Autonomous Robots 18 (1) (2005) 59–80.
[11] J. Nieto, T. Bailey, E. Nebot, Recursive scan-matching SLAM, Robotics and Autonomous Systems 55 (1) (2007) 39–49.
[12] O. Martínez, A. Rottmann, R. Triebel, P. Jensfelt, W. Burgard, Semantic labeling of places using information extracted from laser and vision sensor data, in: Proceedings of the IEEE/RSJ IROS Workshop: From Sensors to Human Spatial Concepts, Beijing, China, pp. 1742–1747. October, 2006.
[13] A. Nüchter, J. Hertzberg, Towards semantic maps for mobile robots, Robotics and Autonomous Systems 56 (11) (2008) 915–926.
[14] M. Magnusson, H. Andreasson, A. Nuchter, A.J. Lilienthal, Appearance-based loop detection from 3D laser data using the normal distributions transform, in: Robotics and Automation, ICRA IEEE International Conference on, 1, 2009, pp. 23–28.
[15] P. Newman, D. Cole, K. Ho, Outdoor SLAM using visual appearance and laser ranging, in: Robotics and Automation, ICRA, Proceedings IEEE International Conference on, 1, 2006, pp. 1180–1187.
[16] F. Linaker, M. Ishikawa, Real-time appearance-based Monte Carlo localization, Robotics and Autonomous Systems 54 (3) (2006) 205–220.
[17] T. Goedemé, M. Nuttin, T. Tuytelaars, L. Van Gool, Omnidirectional vision based topological navigation, International Journal of Computer Vision 74 (3) (2007) 219–236.
[18] H.M. Gross, A. Koenig, S. Mueller, Omniview-based concurrent map building and localization using adaptive appearance maps, in: Systems, Man and Cybernetics, 2005 IEEE International Conference on, 4, 2005, pp. 3510–3515.
[19] F. Dayoub, T. Duckett, An adaptive appearance-based map for long-term topological localization of mobile robots, in: Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, 1, 2008, pp. 3364–3369.
[20] Z. Zivkovic, B. Bakker, B. Krose, Hierarchical map building and planning based on graph partitioning, in: Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, 1, 2006, pp. 803–809.
[21] A.C. Murillo, C. Sagüés, J.J. Guerrero, T. Goedemé, T. Tuytelaars, L. Van Gool, From omnidirectional images to hierarchical localization, Robotics and Autonomous Systems 55 (5) (2007) 372–382.
[22] Z. Zivkovic, O. Booij, B. Kröse, From images to rooms, Robotics and Autonomous Systems 55 (5) (2007) 411–418.
[23] Y. Matsumoto, K. Ikeda, M. Inaba, H. Inoue, Visual navigation using omnidirectional view sequence, in: Intelligent Robots and Systems, IROS, Proceedings. IEEE/RSJ International Conference on, 1, 1999, pp. 317–322.
[24] H. Andreasson, T. Duckett, A.J. Lilienthal, A minimalistic approach to appearance-based visual SLAM, Robotics, IEEE Transactions on 24 (5) (2008) 991–1001.
[25] G. Gallegos, P. Rives, Indoor SLAM based on composite sensor mixing laser scans and omnidirectional images, in: Robotics and Automation, ICRA, IEEE International Conference on, 3, 7, 2010, pp. 3519–3524.
[26] R. Atkinson, R. Shiffrin, Human memory: a proposed system and its control processes, in: K.W. Spence, J.T. Spence (Eds.), The Psychology of Learning and Motivation, Academic Press, New York, 1968, pp. 89–195.
[27] B. Bacca, E. Mouaddib, X. Cuffi, Embedding range information on omnidirectional images through laser range finder, in: IEEE/RSJ Internal Conference on Intelligent Robots and Systems, IROS, 1, 2010, pp. 2053–2058.
[28] H. Bay, A. Ess, T. Tuytelaars, L.V. Gool, SURF: speeded up robust features, Computer Vision and Image Understanding (CVIU) 110 (3) (2008) 346–359.
[29] B. Bacca, J. Salvi, J. Batlle, X. Cufì, Appearance-based mapping and localisation using feature stability histograms, Electronics Letters 46 (16) (2010) 1120–1121.
[30] R. Barber, Desarrollo de una arquitectura para robots móviles autónomos: aplicación a un sistema de navegación topológica, Ph.D. dissertation, Dept. Elect. Eng., U. Carlos III, Madrid, Spain, 2000.
[31] A. Baddeley, Working memory: looking back and looking forward, Nature Reviews in Neuroscience 4 (10) (2003) 829–839.
[32] R. Llinás, I of the Vortex: From Neurons to Self, MIT Press, Cambridge, MA, 2001.
[33] D. Scaramuzza, F. Fraundorfer, R. Siegwart, Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC, in: Proc. IEEE Int. Conf. on Rob. Aut., 1, 2009, pp. 4293–4299.
[34] J.P. Barreto, H. Araujo, Geometric properties of central catadioptric line images and their application in calibration, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (8) (2005) 1327–1333.
[35] S. Bazeille, D. Filliat, Combining odometry and visual loop-closure detection for consistent topo-metrical mapping, RAIRO Operations Research (2010) 1–6.
[36] M.E. Nilsback, A. Zisserman, A visual vocabulary for flower classification, in: IEEE Conf. on Comp. Vision and Pattern Recog., 2006, pp. 1447–1454.
[37] K.L. Ho, P. Newman, Detecting loop closure with scene sequences, Internal Journal of Computer Vision 74 (3) (2007) 261–286.
[38] M. Aly, M. Munich, P. Perona, Indexing in large scale image collections: scaling properties and benchmark, Application of Computer Vision (2011) 418–425.
[39] S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, MIT Press, Cambridge, MA, 2005.
[40] B. Bacca, J. Salvi, X. Cufí, Appearance-based SLAM for mobile robots, in: Proceedings of Conference on Artificial intelligence. Research and Development, Frontiers in Artificial Intelligence and Applications, 202, 2009, pp. 55–64.

B. Bacca was graduated in Electronic Engineering in 1999, and he received his M.Sc. in 2004 both from the Universidad del Valle, Cali, Colombia. Actually, he is a Ph.D. student at the University of Girona and assistant professor at the Universidad del Valle. His current research interests are in the field of localization and mapping for mobile robots, computer vision, focusing on SLAM and appearance-based environmental models.

J. Salvi was graduated in Computer Science from the Technical University of Catalonia in 1993, received the DEA (M.Sc.) in Computer Science in 1996 and the Ph.D. in Industrial Engineering in 1998 both from the University of Girona, Spain. He is Professor of Computer Vision at the Computer Architecture and Technology Department and at the Computer Vision and Robotics Group, University of Girona. His current interests are in the field of computer vision and mobile robotics, focused on visual SLAM, structured light, stereovision, and camera calibration. He is the leader of the 3D Perception Lab and charter member of the spinoffs AQSense and BonesNotes. Dr. Salvi received the Best Thesis Award in engineering for his Ph.D.

X. Cufí received a BS in Physical Sciences from the Universitat Autònoma de Barcelona (UAB) in 1987, and Ph.D. in Computer Vision from the Universitat Politècnica de Catalunya (UPC) in 1998. He is lecturer and the academic director of the Computer Engineering department at the Universitat de Girona (UdG), and he belongs to the Computer Vision and Robotics (VICOROB) research group of the UdG. He is also involved in the Academic Board of an International Erasmus-Mundus master on Vision and Robotics. His recent research interests are underwater image processing, underwater robotics, and mobile robotics. Dr. X. Cufi is a member of the Spanish Committee on Automatics (CEA - GTRobotics) and Spanish member of the International Federation of Automatic Control (IFAC).