Robust Place Recognition With Stereo Sequences

César Cadena, *Member, IEEE*, Dorian Gálvez-López, Juan D. Tardós, *Member, IEEE*, and José Neira, *Senior Member, IEEE*

***Abstract*—We propose a place recognition algorithm for simultaneous localization and mapping (SLAM) systems using stereo cameras that considers both appearance and geometric information of points of interest in the images. Both near and far scene points provide information for the recognition process. Hypotheses about loop closings are generated using a fast appearance-only technique based on the bag-of-words (BoW) method. We propose several important improvements to BoWs that profit from the fact that, in this problem, images are provided in sequence. Loop closing candidates are evaluated using a novel normalized similarity score that measures similarity in the context of recent images in the sequence. In cases where similarity is not sufficiently clear, loop closing verification is carried out using a method based on conditional random fields (CRFs). We build on CRF matching with two main novelties: We use both image and 3-D geometric information, and we carry out inference on a minimum spanning tree (MST), instead of a densely connected graph. Our results show that MSTs provide an adequate representation of the problem, with the additional advantages that exact inference is possible and that the computational cost of the inference process is limited. We compare our system with the state of the art using visual indoor and outdoor data from three different locations and show that our system can attain at least full precision (no false positives) for a higher recall (fewer false negatives).**

***Index Terms*—Bag of words (BoW), computer vision, conditional random fields (CRFs), recognition, simultaneous localization and mapping (SLAM).**

I. Introduction

In this paper, we consider the problem of recognizing locations based on scene geometry and appearance. This problem is, particularly, relevant in the context of large-scale global localization and loop-closure detection in mobile robotics. Algorithms that are based on visual appearance are becoming popular because cameras are easily available and provide rich scene detail. In recent years, it has been shown that taking geometric information also into account further improves system robustness. Most systems rely on a geometrical checking (GC) to verify spatial consistency [1]–[7].

We propose to solve the place recognition problem by using two complementary techniques (see Fig. 1; preliminary versions of this study were described in [8] and [9]). The *first* one is based on the bag-of-words method (BoW) [10], which reduces images to sparse numerical vectors by quantizing their local features. This enables quick comparisons among a set of images to find those which are similar. We use a hierarchical BoW [11] with several novelties that consider the sequential nature of data acquisition. First, we define a novel normalized similarity score

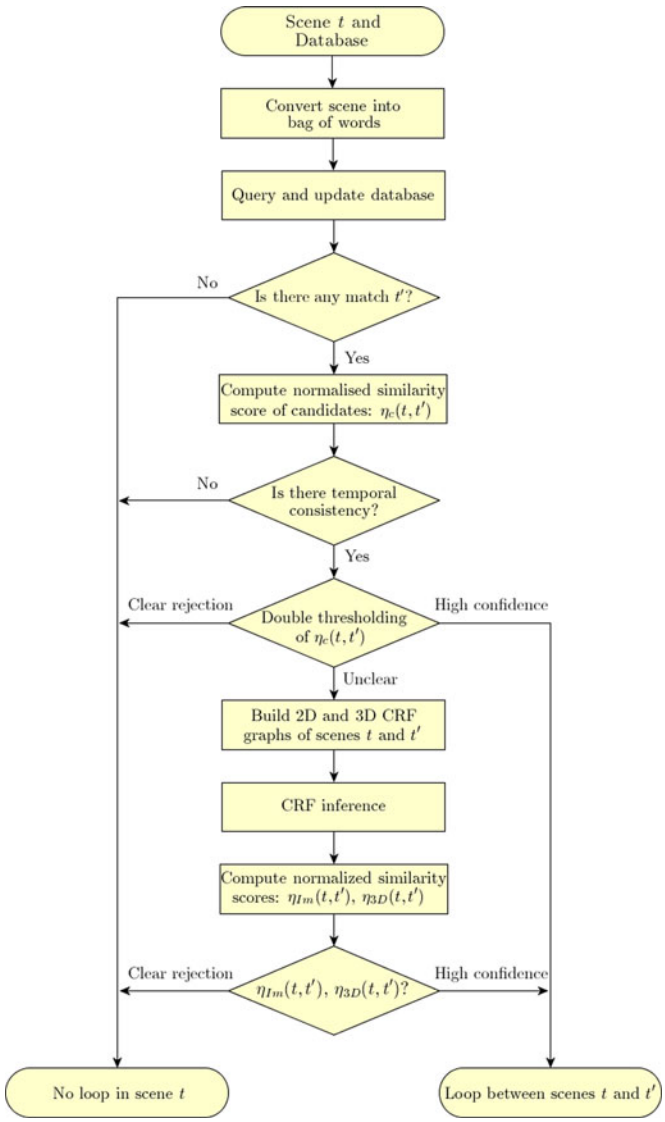

Fig. 1. Scheme of our proposal applied to a scene at time t in the sequence.

Manuscript received March 31, 2011; revised September 23, 2011 and February 9, 2012; accepted February 23, 2012. Date of publication April 3, 2012; date of current version August 2, 2012. This paper was recommended for publication by Associate Editor C. Stachniss and Editor D. Fox upon evaluation of the reviewers' comments. This work was supported in part by the European Union under Project RoboEarth FP7-ICT-248942, the Dirección General de Investigación of Spain under Project DPI2009-13710 and Project DPI2009-07130, and the Ministerio de Educación under scholarship FPU-AP2008-02272. This paper was presented in part at the IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, October 2010, and the 18th World Congress of the International Federation of Automatic Control, Milan, Italy, August–September 2011.

The authors are with the Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, Zaragoza 50018, Spain (e-mail: ccadena@unizar.es; dorian@unizar.es; tardos@unizar.es; jneira@unizar.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2012.2189497

η to evaluate similarity with respect to recent images in the sequence. We also enforce a temporal consistency of the candidate matches to improve robustness, and, finally, we classify the candidates into three categories according to the normalized similarity score: High confidence, unclear, and clear rejection.

Unclear loop-closure candidates are verified by matching the scenes with conditional random fields (CRFs), which is the *second* technique considered. CRF matching is an algorithm that is based on CRFs [12], recently proposed to match 2-D laser scans [13] and image features [14]. CRF matching uses a probabilistic model which is able to, jointly, reason about the association of features. Here, we extend CRF matching to reason about the association of data provided by a stereo camera system in both image space and in 3-D space. This allows our system to consider all information provided by a stereo pair, both near and far. As graph structure for the CRFs in our problem, we propose the use of the minimum spanning tree (MST), where vertices are the features detected in the images, and edge weights are Euclidean distances between them. Because we code near information in the 3-D metric space and far information in image coordinate space, each type of visual information is represented in a separate graph. We also propose accepting the loop-closure candidates based on a normalized similarity score in terms of the likelihoods of the matched scenes, as well as with respect to recent images.

Our basic idea is to exploit the efficiency of BoW to detect revisited places in real time using only appearance (see Section III-A) and the robustness of CRF matching to verify, only in unclear cases, that revisiting matches are correct (see Section III-B).

In Section IV, we analyze the performance of our system using image sequences from the RAWSEEDS project, obtained with a frontal stereo camera in both indoor and outdoor environments. These sequences contain challenging scenes, including many cases of perceptual aliasing, changes in illumination due to time of day, and dynamic environments. We compare our system with the state of the art in visual place recognition, fast appearance-based mapping (FAB-MAP) 2.0 [5], [15]. Our system exhibits better precision–recall performance.

II. Related Work

Place recognition using visual information has been a problem of great interest in robotics for some time. Most successful methods consider appearance or geometric information, or a combination of both. Williams *et al*. [16] compared three loop-closure methods representative of each idea: A map-to-map method that considers mainly geometry, an image-to-image method that considers only appearance, and an image-to-map method that considers both. The best results were obtained for the image-to-image and image-to-map methods, although the image-to-map method does not scale well in large environments.

The image-to-image method considered in the work of Williams *et al*. was FAB-MAP, the first successful appearance-only method, which was proposed by Cummins and Newman [15]. FAB-MAP uses the BoW representation [10], supported by a probabilistic framework. This system proved very successful in large-scale environments. It can run with full precision (no false positives), although at the expense of low recall (the rate of true positives declines). Avoiding false positives is crucial because they result in failure to obtain correct maps, but avoiding false negatives is also important because they limit the quality of the resulting maps, particularly, if large loops are not detected. Geometric information has shown to be important in avoiding false positives while sacrificing less true positives. Angeli *et al*. [1] proposed an incremental version of BoW, using a discrete Bayes filter to estimate the probability of loop closure. Since the Bayes filter can still exhibit false positives in cases where the same features are detected, but in a different geometric configuration, the epipolar constraint was used to verify candidate matchings. Valgren and Lilienthal [2], [3] also verify topological localization candidates using the epipolar constraint, but matching of an image is carried out against the complete image database, which can become inefficient for large environments. Regarding false negatives, Mei *et al*. [17] apply a query-expansion method that is based on the feature covisibility matrix to enrich the information of the locations and facilitate loop detection. In the field of object retrieval, Chum *et al*. [18] also use a query-expansion method to perform image queries against large databases, using appearance information only. All initial candidates are re-ranked using an affine homography, and then, query expansion is carried out. Currently, the performance of such query-expansion methods is heavily dependent on parameter tuning [17]. Furthermore, they do not improve the recall in cases where the image depicts a new place, or there is perceptual aliasing [5].

Other methods have shown the importance of incorporating geometric constraints to avoid false positives. Konolige *et al*. [7] uses a stereo pair to check for a valid *spatial* transformation between two pairs of matching scenes by trying to compute a valid relative pose transformation. The acceptance criteria are based on the number of inliers. A very similar strategy for GC is used by Newman *et al*. [6] with stereo, over the candidates provided from FAB-MAP using an omnidirectional camera. Olson *et al*. [4] test the spatial consistency of a set of candidate matchings by additionally considering their associated pose estimates. This requires odometry or some other source for the priors on the poses. Cummins and Newman [5] incorporated a simplified constraint check for an omnidirectional camera installed on a car in FAB-MAP 2.0. This system was tested using two extremely large (70 km and 1000 km) datasets. They obtain recalls of 48.4% and 3.1%, respectively, at 100% precision. Recently proposed by Paul and Newman [19], FAB-MAP 3-D, additionally, includes 3-D distances between features provided by a laser scanner. This results in higher recall for the same precision in the first urban experiment of FAB-MAP. However, the system can only make inferences about visual features in the laser range.

The place recognition problem has also been addressed using only 3-D range data. Steder *et al*. [20] extract features from range images obtained by a 3-D laser scanner and query a database in order to detect loop closures. This system has high computational requirements compared with systems based on BoWs, but higher recall is attained for the same precision. An important

limitation is that this system cannot distinguish between locations with similar shape but different appearance, for example, corridors, or with different background beyond the sensor's range.

Our system follows a slightly different approach: In order to attain full precision and high recall, an improved BoW technique is used in a first step for the generation of loop closing matches. In unclear cases, loop closing verification is carried out using CRF matching based on both image and 3-D geometry provided by the stereo camera. CRF matching was introduced in [13] to match 2-D laser scans. If visual information is available, texture around laser points can be used for matching, although the remaining visual information is ignored. The authors proposed the possibility of detecting loop closures with CRF by taking the maximum log-likelihood of the match between the current and all previous scans. Comparing the current location against all the previous ones is impractical in real applications. Furthermore, that metric does not provide a way to distinguish between true and false loop closures. The same framework is proposed in [14] to associate features in images considering texture and relative image coordinates. A 2-D Delaunay triangulation is used as graph structure.

In our system, the use of a stereo camera allows us to combine appearance information with 3-D metric information when available. We use the MST as the graph structure instead of the dense Delaunay triangulation. This idea was, previously, used by Quattoni *et al*. [21] in the context of object classification. In that work, equivalent classification performance was shown for MSTs in comparison with more densely connected graphs. In addition, trees allow exact inference algorithms, as compared, for instance, with loopy belief propagation (BP) for cyclic graphs, which is approximate and more expensive. As in [21], our results show that MSTs properly encode connections between the hidden variables and ensures global consistency.

Anguelov *et al*. [22] proposed to use associative Markov networks (another discriminative graphical model) for 3-D dense laser data in the context of segmentation of multiple objects. The graph used by them is a mesh over all the 3-D points. In the same kind of application, Lim and Suter [23] use CRFs and subsample the 3-D laser data with an adaptive data reduction based on spatial properties in order to reduce both learning and inference times. We take advantage of texture in visual information to subsample the 3-D dense information and consider only salient visual features and their coverage areas.

III. Our Proposal

In this section, we describe the two components of our system that constitute the core of our approach: Loop closing detection and loop closing verification. Our place recognition system can be summarized in algorithm 1.

Algorithm 1 Place recognition system

Input: Scene at time t, Database $\langle 1, \ldots, t-1 \rangle$
Output: Time t' of the revisited place, or null

```
Output = Null
Search the bag-of-words database for the best matching scene at t' with score η_c(t, t')
if [t − 1, t'_1], ..., [t − τ_l, t'_{τ_l}] matched and |t'_i − t'_j| ≤ τ_d then
    {Loop candidate detected}
    if η_c(t, t') ≥ α^+ then
        Output = t'        {Accepted}
    else
        if η_c(t, t') ≥ α^− then
            {Loop candidate verification}
            Build G_3D and G_Im
            Infer with CRFs and compute the scores η_G
            if η_3D(t, t') ≤ β_3D ∧ η_Im(t, t') ≤ β_Im then
                Output = t'        {Accepted}
            end if
        end if
    end if
end if
Add current scene to the Database
```

A. Loop Closing Detection

In the spirit of visual BoW [10], we first represent an image of the scene as a numerical vector by quantizing its salient local speed up robust feature (SURF) features [24] [see Fig. 2(a)]. This technique entails an offline stage that consists in clustering the image descriptor space (the 64-D SURF space, in our case) into a fixed N number of clusters. This is done with a rich enough set of training images, which can be independent of the target images. The centers of the resulting clusters are named *visual words*; after the clustering, a *visual vocabulary* is obtained. Now, a set of image features can be represented in the visual vocabulary by means of a vector v of length N. For that, each feature is associated with its approximately closest visual word. Each component v_i is, then, set to a value in accordance with the relevance of the ith word in the vocabulary and the given set, or 0 if that word is not associated with any of the image descriptors. There are several approaches to measure the relevance of a word in a corpus [25]; in general, the more a word appears in the data used to create the visual vocabulary, the lower its relevance is. We use the term frequency-inverse document frequency (tf-idf) as proposed in [10].

This method is suitable to manage a large amount of images; moreover, the authors in [11] present a hierarchical version which improves efficiency. In this version, the descriptor space clustering is done hierarchically, obtaining a visual vocabulary arranged in a tree structure, with a branch factor k and L depth levels. This way, the comparisons for converting an image descriptor into a visual word only need to be done in a branch and not in the whole discretized space, reducing the search complexity to logarithmic. Our implementation[1] of this data structure is used in this paper with $k = 9, L = 6$, and the *k-means++* algorithm [26] as clustering function. This configuration yielded the best performance in our tests and experiments.

1) Normalized Similarity Score: Representing images as numerical vectors is very convenient since it allows us to perform really quick comparisons between images. There are several metrics to calculate the similarity between two image vectors. We use a modified version of the one proposed in [11]. Given two vectors v and w, their similarity is measured as the score

[1] Available at http://webdiis.unizar.es/∼dorian

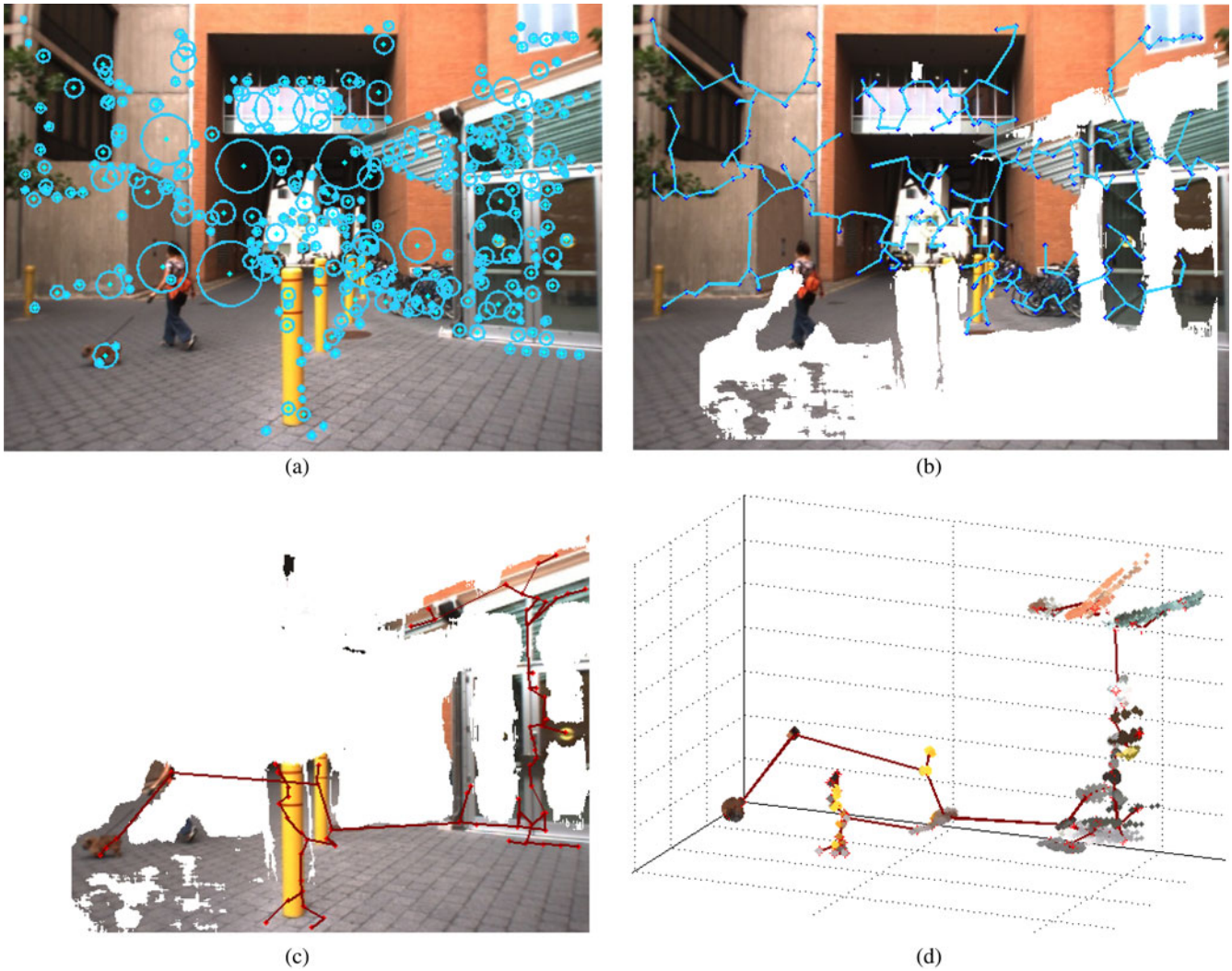

Fig. 2. Scene from an outdoor environment. In each scene, we get the (a) SURF features over one image of the stereo pair and compute the two MSTs: one for features with 3-D information (near features) and the other for the remaining ones (far features). (b) Graph for far features ($\mathcal{G}_{Im}$) in blue. (c) Graph for near features ($\mathcal{G}_{3D}$) in dark red. We apply the CRF matching over both graphs. The MST of $\mathcal{G}_{3\mathrm{D}}$ is computed according to the metric coordinates; here, it is projected over the images only for visualization. (d) $\mathcal{G}_{3\mathrm{D}}$ in metric coordinates with the 3-D point cloud (textured) of each vertex in the tree. The MST gives us an idea of the dependences between features in a scene and enforces the consistency of the feature association between scenes.

$s(v, w)$

$$s(v,w) = 1 - \frac{1}{2}\left\| \frac{v}{\|v\|} - \frac{w}{\|w\|} \right\| \tag{1}$$

where $\|.\|$ stands for the L_1-norm. Note that this score is 0 when there is no similarity at all and 1 when both vectors are the same.

In a general problem with BoWs, when comparing a vector v with several others, choosing the vector w, whose score $s(v, w)$ is maximum, is usually enough to establish a match. However, this is not enough in our context. Ours is a special case where the acquired data are sequential. This means that vectors v and w are associated with instants of time t and t' and that we can take advantage of this fact. Furthermore, it is expected to have lots of very similar images in our problem, since they are collected close in time. In many cases, the matched vector with the highest score s may not be the one we are looking for. We want to distinguish those cases, but the range the score s varies is very dependent on the query image and the words this contains so that it is difficult to set a threshold that works well in every situation. For these reasons, we define a novel metric of similarity, the *normalized similarity score* η_c, as

$$\eta_c(t, t') = \frac{s(v_t, w_{t'})}{s(v_t, v_{t-\gamma})}. \tag{2}$$

We normalize the score obtained from a match between v_t and $w_{t'}$ with the expected score for the query vector v_t. The expected value for v_t is the score obtained when comparing it with a very similar vector. In our case, this is the vector obtained $\gamma = 1$ s ago. If the score $s(v_t, v_{t-\gamma})$ is very small (e.g., if the robot rotated very fast and those two images were not similar at all), the normalized similarity score is not reliable. For this reason, we discard those query vectors v_t such that $s(v_t, v_{t-\gamma}) < 0.1$. For the rest of the cases, the higher the η_c, the more similar the two images. The effect of this normalization is to increase the matching scores of those query images that obtain small scores s because of their number of features, bringing them closer to

those attaining higher s. Note that the normalized similarity score can be defined for any similarity score.

2) Temporal Consistency: Our system takes an image at time t from the stereo pair at one frame/s. The image is converted into a BoW vector v_t, which is stored in a set W. At the same time, an *inverted file* is maintained to keep track of the images in which each visual word is present [10]. The set W and the inverted file compose our database of places already visited. The current image vector v_t is compared with all the ones previously stored in set W that have at least one word in common with v_t. The complexity of this operation in the worst case is linear in the number of stored vectors, but checking the common words by looking up the inverted file makes it very quick. The result is a list of matches $< v_t,\ w_{t'} >$, associated with their scores $\eta_c(t, t')$, where $w_{t'}$ are the vectors matched from W. Of these matches, those whose instants t and t' are too close are discarded to avoid matching images taken very close in time. We disallow matches against images taken less than 20 s ago. This value may depend on the length of the loops and the velocity of the robot, but we noticed that this value suffices with the usual environment and platforms we use in our experiments.

To detect loops, we impose a temporal constraint. A loop candidate between images at time t and t_0 is detected if there exist matches $< v_t,\ w_{t_0} >, < v_{t-1},\ w_{t_1} >, < v_{t-2},\ w_{t_2} >, \ldots$, for a short time interval of $\tau_l = 4$ s, that are pairwise consistent. There is consistency if the difference between consecutive timestamps, $t_0, t_1, \ldots$, is small (i.e., within $\tau_d = 2$ s). These temporal values were selected according to the movement speed of the robot in our image sequences, and the expected reliability of the method.

Finally, the match $< v_t,\ w_{t_0} >$, with normalized score $\eta_c(t, t_0)$, is checked by a *double threshold* (α^-, α^+) in order to be accepted as a loop closure. If this score is high enough $(\eta_c(t, t_0) \geq \alpha^+)$, the match is very likely to be correct, so the candidate is accepted as a loop closing. On the contrary, if this score is small $(\eta_c(t, t_0) < \alpha^-)$, the candidate is rejected. In the cases where this normalized score alone is not sufficient to ensure loop closure $(\alpha^- \leq \eta_c(t, t_0) < \alpha^+)$, verification is necessary.

B. Loop Closing Verification

In this section, we describe the process to decide when an unclear loop-closure candidate from BoW is accepted or rejected. This process is done by inferring on the data association, or matching, between SURF points of the candidates' scenes. Here, we refer to SURF point as the pixel in the image where the SURF feature was detected. The matching is carried out using CRFs, a probabilistic undirected graphical model first developed to label sequence data [12]. We model the scene as two graphs: The first graph ($\mathcal{G}_{3D}$) models the near objects, i.e., those pixels with dense information from the stereo, and, hence, with 3-D information [see Fig. 2(c) and (d)]. The second graph ($\mathcal{G}_{Im}$) models the far objects from pixels without disparity information [see Fig. 2(b)]. The nodes of the graphs are the SURF points extracted before, and the edges of the graphs result from computing the MST, according to the Euclidean distances between the pixel coordinates in the case of $\mathcal{G}_{Im}$, and between the 3-D metric coordinates in the case of $\mathcal{G}_{3D}$.

CRFs are a case of Markov random fields (and, thus, satisfy the Markov properties) where there is no need to model the distribution over the observations [27], [28]. If the neighborhood of a node A (i.e., all nodes with edges to A) in the graph is known, the assignment to A is independent of the assignment to another node B outside the neighborhood of A. By definition, the MST connects points that are close in the measurement space, highlighting intrinsic localities in the scene. This implies first that the associations are jointly compatible within neighborhoods, and second that the compatibility is enforced and propagated from neighborhood to neighborhood by the edge between them.

1) Model Definition: Instead of relying on Bayes' rule to estimate the distribution over hidden states $\mathbf{x}$ from observations $\mathbf{z}$, CRFs directly model $p(\mathbf{x}|\mathbf{z})$, the *conditional* distribution over the hidden variables given observations. Due to this structure, CRFs can handle arbitrary dependences between the observations. This makes them substantially flexible when using complex and overlapped attributes or observations.

The nodes in a CRF represent hidden states, which are denoted $\mathbf{x} = \langle \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n \rangle$; observations are denoted $\mathbf{z}$. In our framework, the hidden states correspond to all the possible associations between the n features in scene *A* and the m features in scene *B*, i.e., $\mathbf{x}_i \in \{0, 1, 2, \ldots, m\}$, where the additional state 0 is the outlier state. Observations are provided by the sensors (e.g., 3-D point clouds, appearance descriptors, or any combination of them). The nodes $\mathbf{x}_i$, along with the connectivity structure represented by the undirected graph, define the conditional distribution $p(\mathbf{x}|\mathbf{z})$ over the hidden states $\mathbf{x}$. Let $\mathcal{C}$ be the set of cliques (fully connected subsets) in the graph of a CRF. Then, a CRF factorizes the conditional distribution into a product of *clique potentials* $\phi_c(\mathbf{z}, \mathbf{x}_c)$, where every $c \in \mathcal{C}$ is a clique in the graph, and $\mathbf{z}$ and $\mathbf{x}_c$ are the observed data and the hidden nodes in such clique. Clique potentials are functions that map variable configurations to nonnegative numbers. Intuitively, a potential captures the "compatibility" among the variables in the clique: The larger a potential value, the more likely the configuration. Using the clique potential, the conditional distribution over hidden states is written as

$$p(\mathbf{x}|\mathbf{z}) = \frac{1}{Z(\mathbf{z})} \prod_{c \in \mathcal{C}} \phi_c(\mathbf{z}, \mathbf{x}_c) \tag{3}$$

where $Z(\mathbf{z}) = \sum_{\mathbf{x}} \prod_{c \in \mathcal{C}} \phi_c(\mathbf{z}, \mathbf{x}_c)$ is the normalizing partition function. The computation of this function can be exponential in the size of $\mathbf{x}$. Hence, exact inference is possible for a limited class of CRF models only, e.g., in tree-structured graphs.

Potentials $\phi_c(\mathbf{z}, \mathbf{x}_c)$ are described by log-linear combinations of *feature functions* $\mathbf{f}_c$, i.e., the conditional distribution (3) can be rewritten as

$$p(\mathbf{x}|\mathbf{z}) = \frac{1}{Z(\mathbf{z})} \exp \left\{ \sum_{c \in \mathcal{C}} \mathbf{w}_c^T \cdot \mathbf{f}_c(\mathbf{z}, \mathbf{x}_c) \right\} \tag{4}$$

where $\mathbf{w}_c$ is a weight vector, which represents the importance of different features to correctly identify the hidden states. Weights can be learned from labeled training data.

2) Inference: Inference in a CRF estimates the marginal distribution of each hidden variable $\mathbf{x}_i$, and can, thus, determine the most likely configuration of the hidden variables $\mathbf{x}$ [i.e., the maximum *a posteriori* (MAP) estimation]. Both tasks can be solved using BP [29], which works by transmitting messages containing beliefs through the graph structure of the model. Each node sends messages to its neighbors based on the messages it receives and the clique potentials. BP generates exact results in graphs with no loops, such as trees or polytrees.

3) Parameter Learning: The goal of parameter learning is to determine the weights of the feature functions used in the conditional likelihood (4). CRFs learn these weights discriminatively by maximizing the conditional likelihood of labeled training data. We resort to maximizing the *pseudo-likelihood* of the training data, which is given by the product of all local likelihoods $p(\mathbf{x}_i|\mathrm{MB}(\mathbf{x}_i))$, where $\mathrm{MB}(\mathbf{x}_i)$ is the Markov blanket of variable $\mathbf{x}_i$, which contains the immediate neighbors of $\mathbf{x}_i$ in the CRF graph. Optimization of this pseudo-likelihood is performed by minimizing the negative of its log, resulting in the following objective function:

$$L(\mathbf{w}) = -\sum_{i=1}^{n} \log p(\mathbf{x}_i|\mathrm{MB}(\mathbf{x}_i), \mathbf{w}) + \frac{\mathbf{w}^T\mathbf{w}}{2\sigma_{\mathbf{w}}^2}. \quad (5)$$

The rightmost term in (5) serves as a zero-mean Gaussian prior, with variance $\sigma_{\mathbf{w}}^2$, on each component of the weight vector.

The training data are labeled using RANSAC [30] over the best rigid-body transformation in 6 DoF [31] for $\mathcal{G}_{3\mathrm{D}}$ and over the fundamental matrix for $\mathcal{G}_{Im}$, after SURF matching of two consecutive scenes.

4) Feature Description: The CRF matcher can employ arbitrary local features to describe shape, image properties, or any particular aspect of the data. Our features describe *differences* between shape (only for $\mathcal{G}_{3\mathrm{D}}$) and appearance (for both graphs) of the features. The local features that we use are the following.

Shape difference: These features capture how much the local shape of dense stereo data differs for each possible association. We use the geodesic, principal component analysis (PCA), and curvature distance.

The *geodesic distance*, which is defined as the sum of Euclidean distances between points in the MST, provides information about the density of the neighborhood of each node of the graph. It can be calculated for different neighborhoods representing local or long-term shape information. Given points $z_{A,i}$, $z_{B,j}$, and a neighborhood k, the geodesic distance feature is computed as

$$\mathbf{f}_{geo}(i,j,k,z_A,z_B) = \left| \sum_{l=i}^{i+k-1} \|z_{A,l+1} - z_{A,l}\| - \sum_{l=j}^{j+k-1} \|z_{B,l+1} - z_{B,l}\| \right| \quad (6)$$

where i and j correspond to the hidden state $\mathbf{x}_i$ that associate the feature i of the scene *A* with the feature j of the scene *B*. The neighborhood k of $\mathbf{x}_i$ in the graph corresponds to all the nodes separated k nodes from $\mathbf{x}_i$. In our implementation, this feature is computed for $k \in \{1,2,3\}$. A similar feature is used to match 3-D laser scans in [32].

We also use PCA over the dense 3-D point cloud that is contained within some spheres centered in the graph nodes [textured points in Fig. 2(d)]. The radius of these spheres is given by the keypoint scale provided by the SURF extractor. The *PCA distance* is computed as the absolute difference between the variances of the principal components of a dense point cloud $z_{A,i}^{\mathrm{pca}}$ in scene A and $z_{B,j}^{\mathrm{pca}}$ in scene B

$$\mathbf{f}_{\mathrm{PCA}}(i,j,z_A^{\mathrm{pca}}, z_B^{\mathrm{pca}}) = \left| z_{A,i}^{\mathrm{pca}} - z_{B,j}^{\mathrm{pca}} \right|. \quad (7)$$

Another way to consider local shape is by computing the difference between the curvatures of the dense point clouds. This feature is computed as

$$\mathbf{f}_{\mathrm{curv}}(i,j,z_A^c, z_B^c) = \left\| z_{A,i}^c - z_{B,j}^c \right\| \quad (8)$$

where $z^c = \frac{3s_3}{s_1+s_2+s_3}$, and $s_1 \geq s_2 \geq s_3$ are the *singular values* of the point cloud of each node.

Visual appearance: These features capture how much the local appearance from the points in the image differs for each possible association. We use the *SURF distance*. This feature calculates the Euclidean distance between the descriptor vectors for each possible association

$$\mathbf{f}_{\mathrm{SURF}}(i,j,z_A^{\mathrm{descr}}, z_B^{\mathrm{descr}}) = \left\| z_{A,i}^{\mathrm{descr}} - z_{B,j}^{\mathrm{descr}} \right\|. \quad (9)$$

Ramos *et al*. [14] also include as features the distances between the individual dimensions of the descriptor space. In our training and validations data, we do not find a significant improvement in the accuracy of the labeling, and this greatly increases the size of the weight vector.

All previous features described are unary, in that they only depend on a single hidden state i in scene A (indices j and k in the features denote nodes in scene *B* and neighborhood size). In order to generate mutually *consistent* associations, it is necessary to define features, over the cliques, that relate the hidden states in the CRF to each other.

Pairwise distance: This feature measures the consistency between the associations of *two* hidden states $\mathbf{x}_i$ and $\mathbf{x}_j$ and observations $z_{A,i}$, $z_{A,j}$ from scene A and observations $z_{B,k}$ and $z_{B,l}$ in scene B

$$\mathbf{f}_{\mathrm{pair}}(i,j,k,l,z_A,z_B) = \left| \|z_{A,i} - z_{A,j}\| - \|z_{B,k} - z_{B,l}\| \right|. \quad (10)$$

The z_A and z_B are in metric coordinates for $\mathcal{G}_{3\mathrm{D}}$, and in pixels for $\mathcal{G}_{Im}$.

5) Loop-Closure Acceptance: We use the CRF matcher stage over the loop closing candidates provided by the BoW stage. Then, we compute the negative log-likelihood Λ from the MAP associations between the scene in time t, against the loop closing candidate in time t', $\Lambda_{t,t'}$, and the scene in $t-\gamma$, $\Lambda_{t,t-\gamma}$, $\gamma = 1$ s.

The negative log-likelihood $\Lambda^{3\mathrm{D}}$ of the MAP association for $\mathcal{G}_{3\mathrm{D}}$ provides a measure of how similar two scenes are in terms of close range, and Λ^{Im} for $\mathcal{G}_{Im}$ in terms of far range. Thus, in order to compare how similar the current scene is with the scene in t', $\Lambda_{t,t'}$, with respect to how similar the current scene is with the scene in $t-\gamma$, $\Lambda_{t,t-\gamma}$, we use again a normalized similarity

TABLE I
DATASETS

Dataset	Length (m)	Revisited length (m)	# Images	Average speed (m/s)	Overlap with training data
Indoor	774	113	1756	0.44	no
Outdoor	1718	208	2279	0.75	yes*
Mixed	1892	268	2150	0.88	yes*
Malaga	1195	162	461	2.6	no

* Different path and one month apart.

score as

$$\eta_{\mathcal{G}} = \frac{\Lambda^{\mathcal{G}}_{t,t'}}{\Lambda^{\mathcal{G}}_{t,t-\gamma}} \tag{11}$$

where $\mathcal{G}$ indicates the graph.

Score $\eta_{\mathcal{G}}$ is compared with $\beta_{\mathcal{G}}$, a control parameter of the level of similarity that we demand for $(t, t-\gamma)$, where a smaller β means a higher demand. By choosing different parameters for near and far information, we can make a balance between the weight of each in our acceptance.

IV. Experiments

We evaluated our system with the public datasets from the RAWSEEDS Project [33]. The data were collected by a robotic platform in different static and dynamic environments. We used the data corresponding to the stereo vision system with 18 cm of baseline. These are black and white images (640 × 480 pixel) taken at 15 frames/s with the Videre Design STH-DCSGVAR system.

We used a static dataset depicting a mix of indoor and outdoor areas to perform the offline stages of our system. These entail the training of the BoW vocabulary and the learning of the CRF feature weights. We, then, tested the whole system in three datasets: Static indoors, static outdoors, and dynamic mixed. These three datasets along with the training one were collected on different dates and in two different campuses. The trajectory of the robot in the outdoor and mixed datasets has some overlap in location with the dataset used for training; this is not the case with the indoor dataset, which was collected in a different campus. Refer to the RAWSEEDS Project [33] for further details.

In addition, in order to evaluate the final configuration of our system, we used the Malaga parking lot 6L dataset [34]. This is a public dataset with a different vehicle, stereo camera (AVT Marlin F-131C model), and configuration (1024 × 768 pixel at 7.5 frames/s and 86 cm of baseline) than those in the RAWSEEDS Project. Table I shows the information related to these datasets.

In this experimental evaluation section, we first show the effect of each system component in the loop detection stage in Section IV-A. Then, in Section IV-B, we compare the use of the Delaunay triangulation and MST during the CRF learning process, as well as the influence of each feature proposed earlier. Finally, we show the performance of our full system in the aforementioned datasets in Sections IV-C–E.

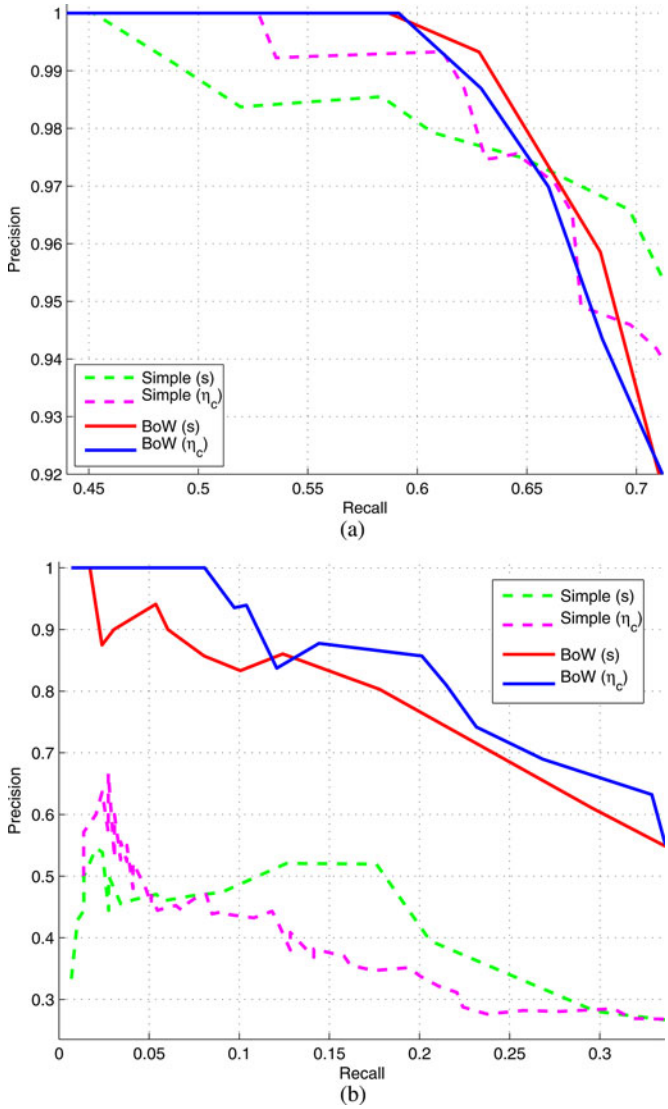

Fig. 3. Precision and recall obtained by different techniques to select loop candidates, with no further geometrical verification. (a) Indoor dataset. (b) Outdoor dataset.

A. Loop Detection Stage

We used 200 images that are uniformly distributed in time from the training dataset to create a BoW vocabulary tree with $k = 9$ branching factor and $L = 6$ depth levels.

We started evaluating how each step of our loop detection stage affects the correctness of the resulting matching candidates. We compared the effectiveness of the score s and η_c when used with both our proposed BoW algorithm (with temporal consistency) and a simple BoW approach. This simple technique consists in selecting the matching image that maximizes the score for a given query. In Fig. 3, we show the precision–recall curves yielded by each configuration as varying the minimum required score. We can see that our BoW with temporal consistency and η_c score obtained the highest recall for 100% precision in the datasets.

Our BoW with temporal consistency outperforms the simple BoW approach. This is specially noticeable in the outdoor

dataset, where the simple is not able to achieve 100% precision so that final results would be hardly reliable. The simple BoW approach failed in this dataset because distant objects, such as buildings, are visible in many images, causing incorrect matches. Requiring temporal consistency reduces these cases because it is unlikely to obtain several consecutive matches with the same wrong place. This makes clear the usefulness of the temporal consistency, as also reported earlier in [8].

Regarding the score, η_c attained higher recall for full precision than the score s for both indoors and outdoors. In the indoor dataset, the behavior of both scores with our BoW approach with temporal consistency was similar, but the advantage of η_c is clear with the simple approach, when just the match with the maximum score is chosen. This shows that η_c is able to provide more discriminative scores than s.

B. Loop Verification Stage

With the same 200 images used to train the BoW vocabulary, we learned the weights for the CRF matcher. For this, we obtained the SURF features from the right image in the stereo rig and selected those with 3-D information from the dense point cloud given by the stereo system. Then, we ran a RANSAC algorithm over the rigid-body transformation between the images at time t and $t - \delta_t$. Since the stereo system has high noise in the dense 3-D information, we selected $\delta_t = 1/15$ s. The same procedure was done over the remaining SURF features with no 3-D information, where we obtained the labels by calculating the fundamental matrix between the images. These two steps resulted in an automatic learning of the CRF labels. Although this automatic labeling can return some outliers, the learning algorithm has demonstrated being robust in their presence. We used the optimization based on the BFGS quasi-Newton method provided by MATLAB to find the weights that minimized the negative log pseudo-likelihood. In both $\mathcal{G}_{3\mathrm{D}}$ and $\mathcal{G}_{Im}$, the weights obtained suggested that the most relevant features in the CRF matcher were $\mathbf{f}_{\mathrm{SURF}}$ and $\mathbf{f}_{\mathrm{pair}}$.

Delaunay versus MST: In order to verify that the accuracy of the data association using the CRFs, as proposed in this paper, is not negatively affected by using MST instead of the Delaunay triangulation, a tenfold cross-validation procedure was carried out. For this, the pairs of images that were used for learning the weights, both in 3-D and image, were randomly permuted and equally divided into ten groups. Nine groups were used for training, and the tenth was used for validation (training and validation data are mutually exclusive). This process was repeated ten times, and the evaluation metrics were computed across folds for all the validation trials.

The tenfold cross validation was performed for $\mathcal{G}_{3\mathrm{D}}$ and $\mathcal{G}_{IM}$, both with the Delaunay triangulation and the MST graph structures. The results of the statistic test in the accuracy of the matching with respect to the labeling given are shown in Table II. The results in the validation data suggest that there is not statistical evidence to favor the Delaunay triangulation as graph structure for our CRF matching processes over the MST. These results agree with the conclusion drawn by Quattoni *et al.* [21].

TABLE II
MEAN AND STANDARD DEVIATION OF THE ACCURACY IN A TENFOLD CROSS-VALIDATION TEST WITH BOTH GRAPH STRUCTURES: DELAUNAY TRIANGULATION AND MST

	$\mathcal{G}_{3D}$		$\mathcal{G}_{IM}$	
	Delaunay	MST	Delaunay	MST
Training set				
Mean	76.65%	88.01%	81.38%	79.53%
Std. dev.	1.70%	0.67%	0.16%	0.17%
Validation set				
Mean	75.05%	87.34%	81.36%	79.53%
Std. dev.	8.30%	5.10%	1.32%	0.99%

TABLE III
ACCURACY IN TRAINING AND VALIDATION SETS FOR THE DATA ASSOCIATION FOR BOTH GRAPHS, REMOVING ONE FEATURE AT A TIME IN THE LEARNING STAGE

	$\mathcal{G}_{3D}$		$\mathcal{G}_{IM}$	
	Training	Validation	Training	Validation
no $\mathbf{f}_{geo}$	86.65%	84.48%		
no $\mathbf{f}_{SURF}$	65.87%	60.92%	19.55%	19.28%
no $\mathbf{f}_{PCA}$	87.63%	84.99%		
no $\mathbf{f}_{curv}$	87.86%	84.68%		
no $\mathbf{f}_{pair}$	85.10%	83.27%	77.84%	77.83%
All features	87.70%	86.45%	78.92%	78.99%

Relative importance of features: The influence of each feature proposed in the CRF matching is studied in the learning stage. The set used for learning was randomly divided into two 60–40% groups: 60% for training and 40% for validation. The learning was then carried out with all the features but one at a time. The accuracy in data association is shown in Table III for both graphs.

The accuracy obtained in each case shows that $\mathbf{f}_{\mathrm{SURF}}$ and $\mathbf{f}_{\mathrm{pair}}$ are the most relevant features in the inference process. However, in the validation set for $\mathcal{G}_{3\mathrm{D}}$, we lose about 2% in the mean accuracy of data association when we remove any other feature. This is a short analysis about the influence of each feature in the inference process that could be extended. For instance, we could analyze many more combinations by adding or removing more than one feature.

Although we have considered a certain set of features in our system, CRFs are amenable to the use of different or additional features that might become available through other sensors or sensing modalities.

C. Full System

In this section, we analyze the performance of our detection and verification stages put together. In addition, we compare our system with the state-of-the-art technique FAB-MAP 2.0 [5]. The FAB-MAP software[2] provides some predefined vocabularies. We used the FAB-MAP indoor vocabulary for our indoor dataset and the FAB-MAP outdoor vocabulary for the mixed and outdoor datasets. This technique has a set of parameters to tune in order to obtain the best performance in each experiment. We give a short description of the parameters in the Appendix (for further details, see [15]). We chose the following two parameter sets in order to obtain different results (see Table IV).

[2]The software and vocabularies were downloaded from http://www.robots.ox.ac.uk/∼mobile/

TABLE IV
FAB-MAP 2.0—PARAMETERS FOR THE EXPERIMENTS

	default	Outdoor	Indoor modified	Mixed
p	0.99	0.96	0.5	0.3
$P(\text{obs}\|\text{exist})$	039	0.39	0.31	0.37
$P(\text{obs}\|\neg\text{exist})$	0.05	0.05	0.05	0.05
$P(\text{newplace})$	0.9	0.9	0.9	0.9
σ	0.99	0.99	1.0	1.0
Motion Model	0.8	0.8	0.8	0.6
Blob Resp. Filter	25	25	25	25
Dis. Local	20s	20s	20s	20s

1) The default parameter set that is provided by the authors. The probability threshold p is taken as 0.99, considering obtaining as few false positives as possible. When we use this configuration, we check the results yielded by FAB-MAP for geometrical consistency.
2) A modified parameter set is tuned to obtain the maximum possible recall at full precision. The idea behind of this tuning is to use as place recognition system only the FAB-MAP 2.0, without geometrical verification. For the outdoor dataset, this parameter set is the same as the default set, only changing the probability threshold.

We filter the results of FAB-MAP 2.0 when using the default configuration with a GC. Since the last available version of the FAB-MAP software does not implement the GC described by Cummins and Newman [5], we implemented a GC based on epipolar geometry. This epipolar constraint consists in computing the fundamental matrix (by using RANSAC and the 8-point algorithm [35]) between two matched images. This test is passed if a well-conditioned fundamental matrix can be obtained.

First, we compared the correctness of our BoW detector with that of FAB-MAP 2.0, both with no geometrical verification. Fig. 4 shows the precision–recall curves resulting in the three RAWSEEDS datasets. We obtained them by varying the minimum confidence value expected for a loop-closure candidate of BoW α^- (with fixed minimum confidence level for a trusted loop closure $\alpha^+ = 0.6$), and the probability of acceptance p of FAB-MAP 2.0. We can observe that the curve of BoW dominated those of FAB-MAP 2.0, even without GC. As was expected, when we choose carefully the parameters of FAB-MAP 2.0, the results that we obtain are much better than when using the configuration by default. This is specially noticeable in the indoor dataset, where there were false positives in all the cases with the default parameters. This is due to the several similar-looking corridors and libraries this dataset presents.

Later, we added the geometrical verification stage to BoW and FAB-MAP 2.0 and compared the results of our system (BoW with CRF matching) and other approaches: FAB-MAP 2.0 with GC, and BoW with GC. We show an example of the results obtained in the indoor dataset in Fig. 5. These were obtained by varying α^- and p, with $\alpha^+ = 0.6$. For our system, we set the β parameters of the CRF matcher in order to obtain 100% precision. We performed this test to compare recall with the state of the art. In view of results shown in Fig. 4, we selected the working value $\alpha^- = 0.15$. Since these datasets are fairly heterogeneous, we think these α values can work well in many situations. It might depend on the vocabulary size, though. All the parameters used are shown in Table V.

The results of FAB-MAP 2.0 over the datasets are shown in Figs. 6(a), 7(a), and 8(a) for the default set of parameters plus the GC with the epipolar constraint, and in Figs. 6(b), 7(b), and 8(b) for the modified set of parameters. Note that FAB-MAP 2.0 with the modified configuration does not need geometrical verification, since we selected the parameters aiming to obtain no false positives. Again, as expected with the modified parameters, FAB-MAP 2.0 obtained greater recall at full precision than with the parameters by default, although some loop closures were not detected. We detail some cases: In the indoor dataset [see Fig. 6(a)], the big area on the beginning of the map (start–end) is especially important in the experiment because if no loop is detected in that area, a SLAM algorithm could hardly build a correct map after having traversed such a long path (around 300 m). In outdoors, as shown in Fig. 7(a) and (b), the biggest loop was missed in the starting and final point of the experiment, in the marked area (O1) in the map. An example of a false negative in this area is shown in Fig. 10(a). This dataset is challenging due to the illumination and blurring present in the images, and this entails an added difficulty for FAB-MAP since the significant overlap of distant objects between consecutive images decreases its discriminative ability [36]. For the experiment in the dynamic mixed environment, important loop closures were missed again, e.g., M1 and M2 areas in Fig. 8(a). Examples of those false negative cases are shown in Fig. 10(b) and 10(c). In the false negative cases that we show in Fig. 10, both configurations of FAB-MAP 2.0 reported a probability of *new place* greater than 0.999.

In order to show the improvements of our loop-closure verification stage, we checked the candidates given by BoW with the same GC technique that we described previously. The results are shown in Figs. 6(c), 7(c), and 8(c). In Fig. 6(c), all the loop-closure areas were detected but with too many false positives due to the perceptual aliasing (see Fig. 9); this is disastrous for any SLAM algorithm. In the outdoor and mixed datasets, the precision was 100%, sacrificing recall and, more importantly, the detection of loop-closure areas. As we can see in Fig. 5, we can tune the parameters of BoW + GC to attain full precision, but at the cost of sacrificing recall. This also makes the performance of this system not good and stable across environments and conditions.

The results of our system over the datasets are shown in Figs. 6(d), 7(d), and 8(d), and the comparative statistics of all experiments is made in Table VI. In the indoor experiment, we can detect all the loop-closure areas at 100% precision. In the outdoor and mixed datasets, we keep full precision, higher recall level, and most of the loop-closure areas detected.

Our system detected successfully the loops in Fig. 10 as true positives. The three cases shown were verified by the CRF stage. Our CRF matcher reports the following η scores: In O1 [see Fig. 10(a)], $\eta_{3\text{D}} = 1.24$ and $\eta_{Im} = 1.37$; in M1 [see Fig. 10(b)], $\eta_{3\text{D}} = 0.4$ and $\eta_{Im} = 1.67$; and in M2 [see Fig. 10(c)], $\eta_{3D} = 1.29$ and $\eta_{Im} = 1.24$. Note that with the β parameters for indoors, such cases would be rejected.

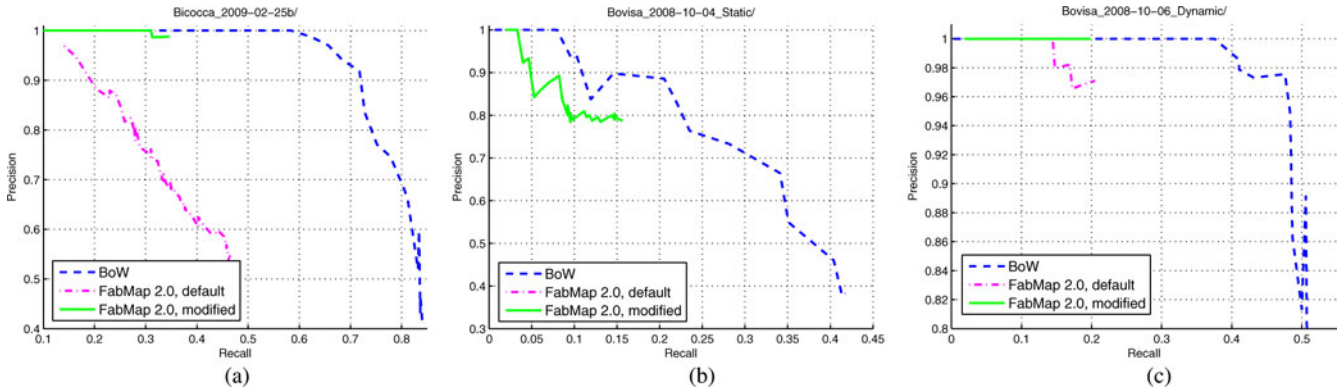

Fig. 4. Precision and Recall without CRF verification or epipolar constraint with BoW and FAB-MAP 2.0 with two parameter sets. (a) Indoor dataset. (b) Outdoor dataset. (c) Mixed dataset.

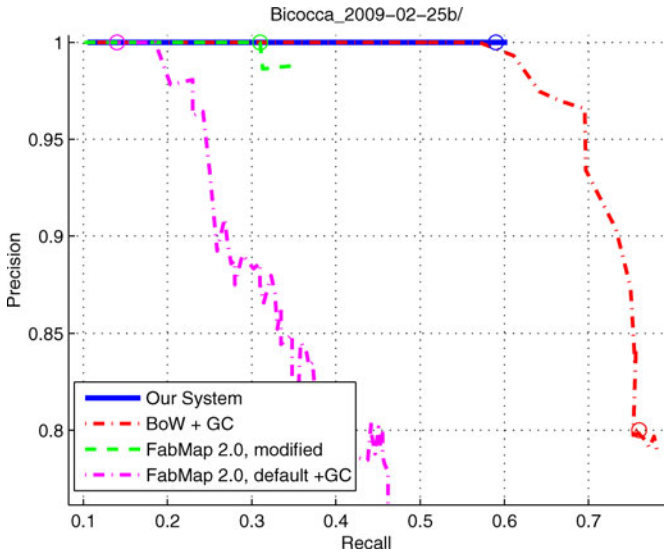

Fig. 5. Precision–recall curves for the indoor dataset. In each curve, the working point is marked, as in Tables IV and V. We also show our BoW stage with GC as verification. The GC checks the epipolar constraint with RANSAC. Note that FAB-MAP 2.0 with the modified configuration needs no GC.

TABLE V
OUR SYSTEM—PARAMETERS FOR THE EXPERIMENTS

	Indoor	Outdoor	Mixed
α^+	0.6	0.6	0.6
α^-	0.15	0.15	0.15
β_{3D}	1	1.5	1.5
β_{Im}	1.3	1.7	1.7
τ_l	4s	4s	4s
τ_d	2s	2s	2s
Min $s_{t,t-\gamma}$	0.1	0.1	0.1
Dis. Local	20s	20s	20s

Furthermore, our CRF matcher is robust against perceptual aliasing. For instance, the false positives obtained with the GC in the indoor sequence was, correctly, discarded (see Fig. 9). In the case of F1, our CRF matcher rejected it by both graphs $\mathcal{G}_{3D}$ and $\mathcal{G}_{Im}$ with $\eta_{3D} = 1.45$ and $\eta_{Im} = 1.79$ [see Fig. 9(a)]. In F2, our CRF matcher rejected it by the far information coded in $\mathcal{G}_{Im}$, with $\eta_{3D} = 0.95$ and $\eta_{Im} = 1.47$ [see Fig. 9(b)].

With the parameters by default of FAB-MAP 2.0, we cannot obtain full precision in the indoor dataset, even with $p = 0.99$. As explained in Fig. 9(c), we have to verify the loop closures detected with the GC to attain full precision, obtaining lowest recall in the three datasets. With the modified configuration, we tuned the parameters of FAB-MAP 2.0, aiming to maximize the precision. With this approach, we obtained 100% precision in the indoor, outdoor, and mixed datasets with 30.6% recall and 2/6 loop areas detected, 3.3% recall and 3/9 areas, and 19.9% recall and 3/8 areas, respectively.

We also tried to tune the parameters of FAB-MAP to maximize recall without paying attention to the precision, which can be improved later by using the geometrical constraint. With that approach, we could attain 100% precision in the outdoor and mixed datasets, but false positives remained indoors, obtaining 75% precision only. As in the case of BoW + GC shown in Table VI, the GC was not able to filter out all the incorrect loop candidates suffering from perceptual aliasing. In the mixed dataset, the recall obtained, 15%, was lower than that observed with the other FAB-MAP 2.0 configurations. The same situation occurred outdoors, except for unrealistically low thresholds, like $p = 0.3$, that yielded a recall up to 5%.

In light of these results, we can see that our verification stage is better suited to discriminate near–far information for decision making.

D. Timing

The online system runs at 1 frame/s. We have a research implementation in C++ using the OpenCV library. In Table VII, we show the average and maximum times for each stage of the system on a 2.3-GHz IntelCore i3 CPU M350 and 4-GB RAM. For the whole system, the average and the maximum times were computed only when all the stages were executed. Note that the maximums for each stage happened in different cases. That is more evident in the inference process for $\mathcal{G}_{3D}$ and $\mathcal{G}_{Im}$: When an image provides more 3-D points, less background information remains. In an image, the number of nodes and hidden states between $\mathcal{G}_{3D}$ and $\mathcal{G}_{Im}$ are complementary. The execution time reported for the CRFs in the graphs includes computing the

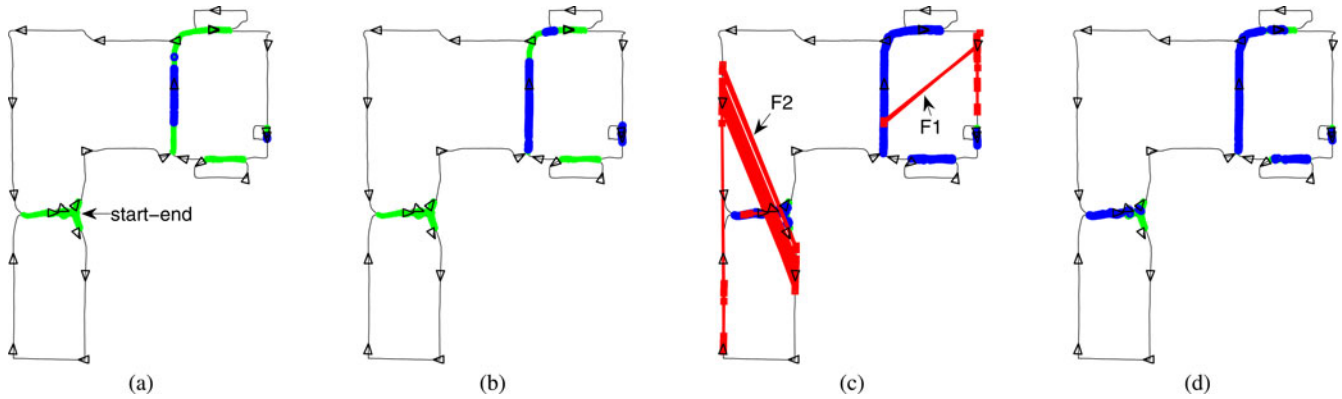

Fig. 6. Loops detected by each of the methods in the indoor dataset. Black lines and triangles denote the trajectory of the robot, light green lines denote actual loops, deep blue lines denote true loops detected, and light red lines denote false loops detected. In Fig. 9, we show the false positive cases F1 and F2. (a) FAB-MAP 2.0 (default) + GC. (b) FAB-MAP 2.0 (modified). (c) BoW + GC. (d) *Our system*.

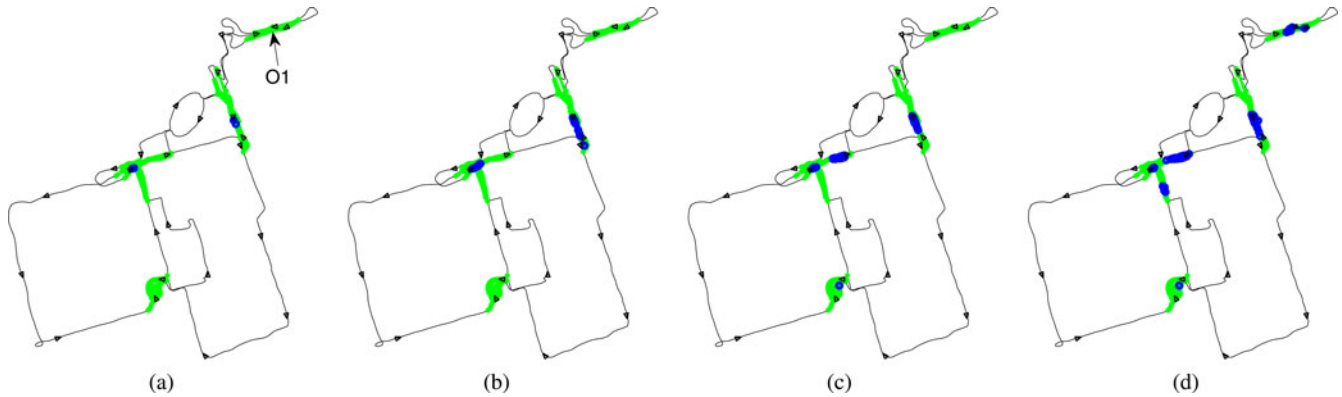

Fig. 7. Loops detected by each of the methods in the outdoor dataset. Black lines and triangles denote the trajectory of the robot, light green lines denote actual loops, and deep blue lines denote true loops detected. In Fig. 10, we show the false negative case O1. (a) FAB-MAP 2.0 (default) + GC. (b) FAB-MAP 2.0 (modified). (c) BoW + GC. (d) *Our system*.

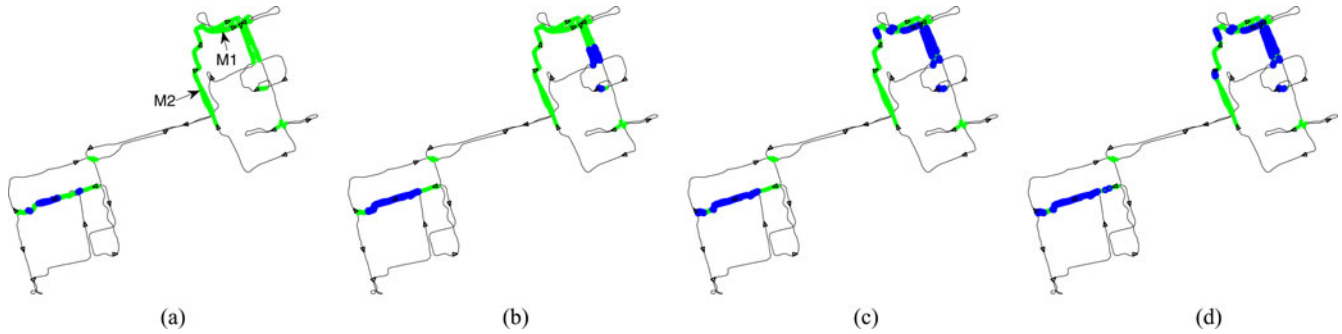

Fig. 8. Loops detected by each of the methods in the mixed dataset. Black lines and triangles denote the trajectory of the robot, light green lines denote actual loops, and deep blue lines denote true loops detected. In Fig. 10, we show the false negative cases M1 and M2. (a) FAB-MAP 2.0 (default) + GC. (b) FAB-MAP 2.0 (modified). (c) BoW + GC. (d) *Our system*.

MSTs, the corresponding features, and the inference for each one. The time for the whole system includes computing the 3-D point cloud from the disparity map and writing and reading the SURF descriptors and point clouds on disk.

E. No Hands Test

After obtaining the best results in the different datasets of the RAWSEEDS Project comparatively, we tested our system over a different dataset, i.e., the Malaga parking lot 6 [34]. As before, we carry out the place recognition task at 1 frame/s. This dataset, as the indoor one described earlier, was collected in a completely different location from the one where our training images were acquired. The main challenge is to test our system with the configuration, already, used in the previous experiments on a different vehicle and stereo camera system.

For that, we kept the same vocabulary and CRFs' weights, as well as the parameters used in the outdoor dataset, as shown

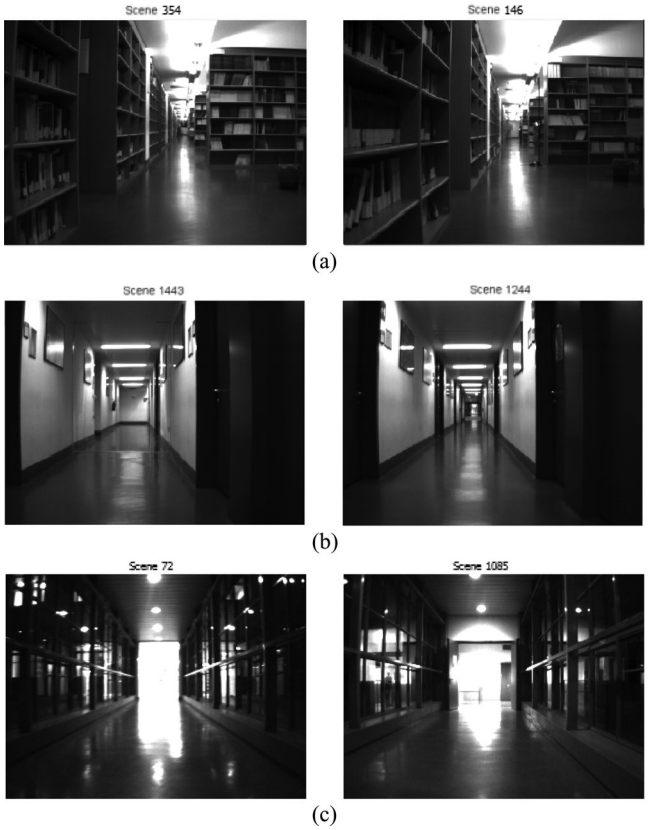

Fig. 9. (a) and (b) False positive cases obtained by BoW plus GC in the indoor dataset. (c) Two different corridors make FAB-MAP 2.0 produce a false positive ($p = 0.9989$) with the default parameters. However, it is correctly rejected by the GC.

Fig. 10. False negatives in the outdoor and mixed datasets that our method can successfully detect but FAB-MAP 2.0 misses. FAB-MAP 2.0 sets query image of case (a) as a new place with a probability of 0.99947, of (b) with 0.99997 and 0.99902 with the default and modified set of parameters, respectively, and (c) with 1.0 and 0.9993. These scenes correspond to the biggest loops in the trajectories.

TABLE VI
RESULTS FOR RAWSEEDS DATASETS

	Precision	Recall	loop zones found/actual
RAWSEEDS Indoor			
FAB-MAP 2.0 def. + GC	100%	14.1%	2 / 6
FAB-MAP 2.0 mod.	100%	30.6%	2 / 6
BoW + GC	79.8%	76.3%	6 / 6
Our System	**100%**	**59.1%**	**6 / 6**
RAWSEEDS Outdoor			
FAB-MAP 2.0 def. + GC	100%	0.7%	2 / 9
FAB-MAP 2.0 mod.	100%	3.3%	2 / 9
BoW + GC	100%	7.0%	3 / 9
Our System	**100%**	**11.15%**	**6 / 9**
RAWSEEDS Mixed			
FAB-MAP 2.0 def. + GC	100%	3.7%	1 / 8
FAB-MAP 2.0 mod.	100%	19.9%	3 / 8
BoW + GC	100%	29.9%	4 / 8
Our System	**100%**	**32.8%**	**5 / 8**

TABLE VII
COMPUTATIONAL TIMES FOR OUR SYSTEM (IN SECONDS)

	SURF extraction	BoW	CRF Matcher $\mathcal{G}_{3D}$	CRF Matcher $\mathcal{G}_{Im}$	Whole System
Average	0.15	0.01	0.15	0.15	0.47
Maximum	0.30	0.04	0.36	0.65	1.04

in Table V. In order to compare the results, we ran FAB-MAP 2.0 with the configuration that obtained a better result for the outdoor experiment, as well as with the default parameter set $p = 0.99$, as well as filter the results with the epipolar constraint GC, as before.

The results over the Malaga parking lot are shown in Fig. 11 and Table VIII. With no changes in our system, we attain full precision despite the increased speed of this vehicle. Using FAB-MAP with the configuration for best performance in the outdoors experiment, we can obtain higher recall here, but precision falls down to 42%, unacceptable for any SLAM system. If we use the configuration that exhibited bad performance in recall in the outdoors experiment, FAB-MAP 2.0 default plus GC attains higher recall compared with our system (68% versus 42%); both methods find four out of five loop-closure zones. This makes our system more stable across different environments and conditions. We show in Fig. 12 two examples of those loops found by one and not by the other: Ma1 and Ma2.

Note in Fig. 11(c) that FAB-MAP alone has bad performance. It returns a large number of detections, more than half false loops. The increase in the number of alarms as compared with the RAWSEEDS experiments is due to the higher speed of the vehicle, 2.6 m/s versus 0.8 m/s (see Table I). This results in less overlap between consecutive processed frames, increasing the maximum values of the probability distribution over the sequence. Still, it is susceptible to perceptual aliasing due to overlapping in the far information. This is corrected with GC

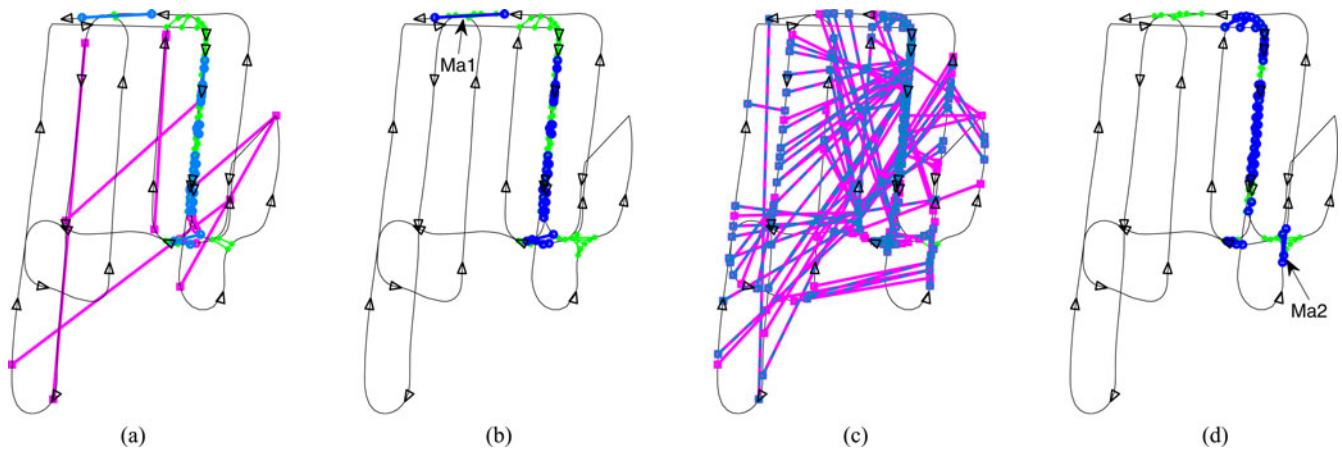

Fig. 11. Loops detected by our system and FAB-MAP 2.0 in the Malaga parking lot dataset. Black lines and triangles denote the trajectory of the robot; light green lines denote the actual loops. (a) BoW, high confidence detections (light blue) are accepted and unclear detections (magenta) are subject to verification. (c) FAB-MAP 2.0, detections with p =0.99, (default) are the dashed light blue; detections with p =0.96 (modified) are in magenta. (d) Detections with p =0.99, which are verified with GC. In (b) **Our system** and (d) FAB-MAP (default) + GC, deep blue lines denote true loops detected.

TABLE VIII
RESULTS FOR MALAGA DATASET

	Precision	Recall	loop zones found/actual
FAB-MAP 2.0 def. + GC	100%	67.9%	4 / 5
FAB-MAP 2.0 mod.	41.5%	81.2%	4 / 5
Our System	100%	41.8%	4 / 5

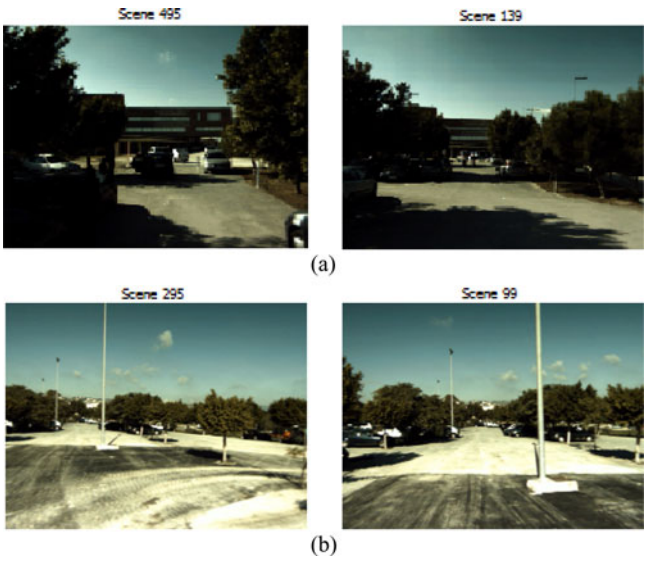

Fig. 12. Cases marked in 11(b) and 11(d) from Malaga dataset. These correspond to loops found by one and not by the other method.

because this dataset does not suffer from strong perceptual aliasing in near information, in contrast with the indoor dataset (see Fig. 9).

Our loop closing detection stage discriminates better, still detecting the most of loop-closure zones [see Fig. 11(a)]. As expected, our verification stage, correctly, decides over the unclear cases [magenta lines in Fig. 11(a)]. The final result of our full system is shown in Fig. 11(b).

V. CONCLUSIONS AND FUTURE WORK

We have presented a system that combines two powerful matching algorithms, i.e., BoW and CRFs, to robustly solve the place recognition problem with stereo cameras. We have evaluated our place recognition system in different environments (indoor, outdoor, and mixed) from public datasets. In all cases, the system can attain 100% precision (no false positives) with higher recall than the state of the art (less false negatives), and detecting the most (especially important) loop-closure zones.

No false positives means that the environment model will not be corrupted, and less false negatives means that it will be more precise. The important lesson that we can learn from this is that we must always apply a verification stage over detected loops based on appearance. As we have seen in situations of perceptual aliasing, our verification stage with the CRF matcher is more robust than the GC using the epipolar constraint.

As mentioned in [36], the effectiveness of FAB-MAP decreases when the camera looks forward, because FAB-MAP models the environment as "a collection of discrete and disjoint locations" [15]. However, in our experiments, the stereo camera system faces forward, and distant objects (e.g., buildings in outdoor scenes) persist for many frames, making scenes overlap and be less discriminative. This causes the matching probability mass of FAB-MAP to be flattened over the scenes. It is easier for our system to overcome those cases because our normalized similarity scores (η_c, η_{3D}, η_{Im}) to match acceptance are computed at each frame and take into account the similarity between consecutive frames.

By jointly using the CRF-matching algorithm over visual near 3-D information (here provided by stereo vision but also possible with range scanners, etc.) and far information, we have demonstrated that challenging false loop closures can be rejected. Furthermore, CRF-matching is also able to fuse any other kind of information, such as image color, with ease.

Our place recognition system is able to run in real time, processing scenes at one frame/s. In most cases, after extracting the SURF features (maximum 300 ms), our system only takes

11 ms to detect if there are possible loop closures, and 300 ms to check them when necessary. We are considering the use of cheaper feature extractors that can speed up this process without a negative impact in precision and recall.

In our experiments, the best β thresholds for acceptance of the CRF matching turned out to be clearly different for indoor and outdoors scenarios. These parameters will also depend on the velocity of motion, mainly because we use images from the previous second as reference in the comparisons. Incorporating the computation of these thresholds as part of the learning stage would also make the system more flexible. Nevertheless, our system has demonstrated a stable performance, always at full precision, for different environments, cameras, and conditions. Systems such as simple BoW or FAB-MAP, both aided by GC, can obtain good results if adequately tuned in each case. However, the same configuration can result in very poor performance in others.

An important line of future work is addressing the place recognition problem over time. Our system performs well in multiday sessions using parameters learned in different months, and this is also true of alternative systems such as FAB-MAP. The environment can also change during the operation in the same session (see Fig. 10). Our algorithm is also able to detect places revisited at different times of day, while alternative systems sometimes reject them in order to maintain high precision.

Several extensions are possible for operation in longer periods of time. The vocabulary for the BoW has shown to be useful in different environments, which suggests that a rich vocabulary does not require frequent updates. The learned parameters in the CRF stage can be relearned in sliding window mode depending on the duration of the mission. The system will then be able to adjust to changing conditions. In cases of periodical changes, such as times of day or seasons, we will need to maintain several environment models and select the most appropriate for a given moment of operation.

APPENDIX

FAST APPEARANCE-BASED MAPPING PARAMETERS DESCRIPTION

The parameters that we have modified are the following ones (for further details, see [5] and [15]).

1) p: Probability threshold. The minimum matching probability required to accept that two images were generated at the same place;
2) $P(\text{obs}|\text{exist})$: True positive rate of the sensor. Prior probability to detect a feature given that it exists in the location;
3) $P(\text{obs}|\neg\text{exist})$: False positive rate of the sensor. Prior probability to detect a feature given that it does not exist in the location;
4) $P(\text{newplace})$: Probability for new place. Prior probability to determine whether the last image is a new place;
5) σ: Likelihood smoothing factor. Factor for smoothing the likelihood values through consecutive places;
6) *Motion Model*: Model Motion Prior. This biases the matching probabilities according to the expected motion of the robot. A value of 1.0 means that all the probability mass goes forward, and 0.5 means that probability goes equally forward and backward;
7) *Blob Resp. Filter*: Blob Response Filter. All the SURF points with a blob response below this threshold are discarded;
8) *Dis. Local*: Disallow N local matches. Set the prior to be zero on the last N places. We use the same parameter in our system during the BoW stage for not producing matches against the last N scenes.

REFERENCES

[1] A. Angeli, D. Filliat, S. Doncieux, and J. Meyer, "A fast and incremental method for loop-closure detection using bags of visual words," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1027–1037, Oct. 2008.

[2] C. Valgren and A. J. Lilienthal, "SIFT, SURF and seasons: Long-term outdoor localization using local features," in *Proc. Eur. Conf. Mobile Robots*, Sep.19–21, 2007, pp. 253–258.

[3] C. Valgren and A. J. Lilienthal. (2010). Sift, surf & seasons: Appearance-based long-term localization in outdoor environments. *Robot. Auton. Syst.* [Online]. vol. 58, no. 2, pp. 149–156. Available: http://www.sciencedirect.com/science/article/B6V16-4X908T5-6/2/679a6246b247d1b8329211a2b9df49f4

[4] E. Olson, "Recognizing places using spectrally clustered local matches," *Robot. Auton. Syst.*, vol. 57, no. 12, pp. 1157–1172, Dec. 2009.

[5] M. Cummins and P. Newman. (2010). Appearance-only SLAM at large scale with FAB-MAP 2.0. *Int. J. Robot. Res.* [Online]. vol. 30, no. 9, pp. 1–24. Available: http://ijr.sagepub.com/content/early/2010/11/11/0278364910385483.abstract

[6] P. Newman, G. Sibley, M. Smith, M. Cummins, A. Harrison, C. Mei, I. Posner, R. Shade, D. Schrter, L. Murphy, W. Churchill, D. Cole, and I. Reid, "Navigating, recognising and describing urban spaces with vision and laser," *Int. J. Robot. Res.*, vol. 2, nos. 11–12, pp. 1406–1433, 2009.

[7] K. Konolige, J. Bowman, J. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua, "View-based maps," *Int. J. Robot. Res.*, vol. 29, no. 8, pp. 941–957, 2010.

[8] C. Cadena, D. Gálvez-López, F. Ramos, J. Tardós, and J. Neira, "Robust place recognition with stereo cameras," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, Taipei, Taiwan, Oct. 2010, pp. 5182–5189.

[9] C. Cadena, J. McDonald, J. Leonard, and J. Neira, "Place recognition using near and far visual information," presented at the 18th World Congr. Int. Fed. Automat. Control (IFAC), Milano, Italy, Aug. 2011.

[10] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. Int. Conf. Comput. Vis.*, Oct. 2003, vol. 2, pp. 1470–1477.

[11] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2006, vol. 2, pp. 2161–2168.

[12] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," presented at the 18th Int. Conf. Mach. Learn., San Francisco, CA. [Online]. Available: citeseer.ist.psu.edu/lafferty01conditional.html.

[13] F. Ramos, D. Fox, and H. Durrant-Whyte, "CRF-matching: Conditional random fields for feature-based scan matching," in *Proc. Robot.: Sci. Syst.*, 2007, pp. 201–208.

[14] F. Ramos, M. W. Kadous, and D. Fox, "Learning to associate image features with CRF-matching," in *Proc. Int. Symp. Exp. Robot.*, 2008, pp. 505–514.

[15] M. Cummins and P. Newman, "FAB-MAP: Probabilistic localization and mapping in the space of appearance," *Int. J. Robot. Res.*, vol. 27, no. 6, pp. 647–665, 2008.

[16] B. Williams, M. Cummins, J. Neira, P. Newman, I. Reid, and J. Tardós, "A comparison of loop closing techniques in monocular slam," *Robot. Auton. Syst.*, vol. 57, pp. 1188–1197, 2009.

[17] C. Mei, G. Sibley, and P. Newman, "Closing loops without places," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2010, pp. 3738–3744.

[18] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, "Total recall: Automatic query expansion with a generative feature model for object retrieval," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.

[19] R. Paul and P. Newman, "FAB-MAP 3D: Topological mapping with spatial and visual appearance," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 2649–2656.

[20] B. Steder, G. Grisetti, and W. Burgard, "Robust place recognition for 3D range data based on point features," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 1400–1405.
[21] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell, "Hidden conditional random fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1848–1852, Oct. 2007.
[22] D. Anguelov, B. Taskar, V. Chatalbashev, D. Koller, D. Gupta, G. Heitz, and A. Ng, "Discriminative learning of Markov random fields for segmentation of 3D scan data," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 2, pp. 169–176.
[23] E. H. Lim and D. Suter, "Conditional random field for 3D point clouds with adaptive data reduction," in *Proc. Int. Conf. Cyberworlds*, 2007, pp. 404–408.
[24] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. 9th Eur. Conf. Comput. Vis.*, 2006, vol. 3951, pp. 404–417.
[25] J. Yang, Y. Jiang, A. Hauptmann, and C. Ngo, "Evaluating bag-of-visual-words representations in scene classification," in *Proc. Int. Workshop Multimedia Inf. Retrieval*. New York: ACM, 2007, p. 206.
[26] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2007, pp. 1027–1035.
[27] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA: MIT Press, 2009.
[28] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer-Verlag, 2006.
[29] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA: Morgan Kaufmann, 1988.
[30] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
[31] E. Olson, "Robust and efficient robotic mapping," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Mass. Inst. Technol., Cambridge, MA, Jun. 2008.
[32] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "SCAPE: Shape completion and animation of people," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 408–416, 2005.
[33] "RAWSEEDS FP6 Project," (2009). [Online]. Available: http://www.rawseeds.org
[34] J.-L. Blanco, F.-A. Moreno, and J. González. (2009, Nov.). A collection of outdoor robotic datasets with centimeter-accuracy ground truth. *Auton. Robots* [Online]. vol. 27, no. 4, pp. 327–351. Available: http://www.mrpt.org/Paper:Malaga_Dataset_2009.
[35] R. Hartley, "In defense of the eight-point algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 6, pp. 580–593, Jun. 1997.
[36] P. Piniés, L. M. Paz, D. Gálvez-López, and J. D. Tardós, "Ci-graph simultaneous localization and mapping for three-dimensional reconstruction of large and complex environments using a multicamera system," *J. Field Robot.*, vol. 27, pp. 561–586, 2010.

César Cadena (M'11) received the electronic engineering and mechanical engineering degrees in 2005 and the M.Sc. degree in 2006, all from the Universidad de los Andes, Bogotá, Colombia. He received the Ph.D. degree in computer science from the University of Zaragoza, Zaragoza, Spain, in 2011.

He is currently with the Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza. His current research interests include persistent mapping in dynamic environments, semantic data association with machine learning techniques, and control systems.

Dorian Gálvez-López received the Graduate's and Master's degrees in computer science from the Centro Politécnico Superior, Zaragoza, Spain, in 2007 and 2009, respectively, where he is currently working toward the Ph.D. degree in visual semantic simultaneous localization and mapping.

He was with the Centre for Autonomous Systems, KTH Royal Institute of Technology, Stockholm, Sweden, until 2008, where he was involved in object detection research. His current research interests include fast object detection and visual loop closure for robotic applications.

Juan D. Tardós (M'05) was born in Huesca, Spain, in 1961. He received the M.S. and Ph.D. degrees in electrical engineering from the University of Zaragoza, Zaragoza, Spain, in 1985 and 1991, respectively.

He is currently a Full Professor with the Departamento de Informática e Ingeniería de Sistemas, University of Zaragoza, where he is in charge of courses in robotics, computer vision, and artificial intelligence. His current research interests include simultaneous localization and mapping, perception, and mobile robotics.

José Neira (SM'09) was born in Bogotá, Colombia, in 1963. He received the M.S. degree from the Universidad de los Andes, Bogotá, in 1986 and the Ph.D. degree from the University of Zaragoza, Zaragoza, Spain, in 1993, both in computer science.

He is currently a Full Professor with the Department of Computer Science and Systems Engineering, University of Zaragoza, where he teaches compiler theory, computer vision, machine learning, and mobile robotics. His current research interests include autonomous robots, computer vision, and artificial intelligence.