Robotica (2013) volume 31, pp. 687–701. © Cambridge University Press 2013
doi:10.1017/S0263574712000732

A robust, multi-hypothesis approach to matching occupancy grid maps

Jose-Luis Blanco†*, Javier González-Jiménez‡ and Juan-Antonio Fernández-Madrigal‡

†*Department of Engineering, University of Almería, Almería, Spain*
‡*Department of System Engineering and Automation, University of Málaga, Malaga, Spain*

(Accepted December 11, 2012. First published online: January 11, 2013)

SUMMARY
This paper presents a new approach to matching occupancy grid maps by means of finding correspondences between a set of sparse features detected in the maps. The problem is stated here as a special instance of generic image registration. To cope with the uncertainty and ambiguity that arise from matching grid maps, we introduce a modified RANSAC algorithm which searches for a dynamic number of internally consistent subsets of feature pairings from which to compute hypotheses about the translation and rotation between the maps. By providing a (possibly multi-modal) probability distribution of the relative pose of the maps, our method can be seamlessly integrated into large-scale mapping frameworks for mobile robots. This paper provides a benchmarking of different detectors and descriptors, along extensive experimental results that illustrate the robustness of the algorithm with a 97% success ratio in loop-closure detection for ~ 1700 matchings between local maps obtained from four publicly available datasets.

KEYWORDS: Mobile robots; SLAM; Robot localization; Pose estimation and registration; navigation.

1. Introduction

Occupancy grid maps, introduced into the mobile robotics community almost three decades ago,[11] are a very valuable geometrical representation for map building of planar environments.[15,16,36] In this representation, the space is arranged into a metric grid of cells, each one storing the probability of being occupied by obstacles. These maps can be employed in the context of large-scale, hybrid metric-topological map models,[5,8,12] where each node of a topological graph represents a local metric map, e.g. a set of visual landmarks or a grid map.

An important requirement of hierarchical mapping approaches is to detect whether two local maps correspond to the same physical place and, in that case, to compute the relative transformation between those maps (namely, detecting *loop closures*). Solving loop closure in a hierarchical framework, the purpose of the method presented in this work, implies coping with a number of hurdles such as noise in the robotic sensor, ambiguity (different parts of the environment can be indistinguishable) and dynamic scenarios (the map of an area may change over time).

Instead of using grid maps alone, we have adopted a dual representation of local maps where both occupancy grids and point maps are maintained. As described in ref. [29], this approach has a number of advantages since these maps complement each other and their maintenance only requires updating both maps simultaneously with the same sensory data.

Corresponding to this dual representation, our approach for aligning a pair of local maps consists of two differentiated steps: (i) the grid maps are first matched without any *a priori* information; then (ii) the point maps help to refine the matching. Our discussion will preeminently focus on the first step, the *grid-to-grid matching*, since the registration of point maps is a well-understood topic with efficient solutions such as Iterative closest point (ICP).[2] Furthermore, this second step only has to refine an estimation already close to the real solution while the grid-to-grid matching has no such advantage and thus poses a far more challenging problem.

We propose to estimate the transformation between a pair of grid maps by registering the corresponding *map images*, the grayscale images resulting from interpreting grid cells as pixels and occupancy probabilities as gray levels. Since in robotic applications we can select the grid cell size, we can focus on matching maps with identical cell sizes only. Therefore, a pair of maps can be only related by a rigid transformation, fully determined by a two-dimensional (2D) translation plus a rotation, disregarding scale changes.

In general, image registration techniques can be classified into those based on intensity and those based on the extraction of interest points—refer to ref. [38] for an extensive review. Although the former approach has already been applied to grid map matching,[16] there is no previous work based on feature extraction, which is known to be more efficient computationally and therefore more appropriate for being integrated into real-time mapping frameworks. In spite of the existence of previous works devoted to analyzing the performance of different visual feature detectors[14] and descriptors,[27] in this work we present a benchmark which specifically addresses their behavior for grid map images.

* Corresponding author. E-mail: joseluisblancoc@gmail.com

Our approach represents an important contribution due to the reporting of a robust method for finding the transformation between map images in the form of a sum of Gaussians (SOG). This probabilistic representation allows coping with multiple hypotheses and therefore to consistently integrate the method into robotic mapping frameworks, most of them based on probabilistic Bayesian inference.[37] Our probabilistic approach is therefore in contrast to previous works on robust image registration based on vote counting in the space of transformation parameters.[33] Within mobile robotics, Duckett and Nehmzow[10] reported a method very similar to ours, which also obtains an SOG for potential matches between grid maps. However, their work assumes an accurate knowledge of the absolute orientation of the robot (i.e. it should be equipped with a compass); hence, our proposal has a broader applicability to practical situations.

The present work is also related to research in multi-robot mapping, since the map merge problem can be seen there as a special instance of the detection of loop closures in single robot mapping. In that field, a method with a similar purpose to ours has been reported in ref. [3], but it does not consider the possibility of multiple hypotheses in the map merge, and a rough comparison of typical execution times has revealed that our method is about 100 times faster.

The rest of the paper is organized as follows. In the next section, we introduce an overview of our method. A thorough discussion on different detectors and descriptors is provided in Sections 3 and 4, respectively, which are benchmarked in Section 5. The robust matching method, discussed in Section 6, requires a Gaussian model for the optimal rigid transformation for subsets of correspondences, which is discussed in Section 7. Finally, experimental results validate our approach with maps from four publicly available datasets. We must remark that a C++ implementation of the proposed algorithm has been released under the open source GNU General Public License.[1]

2. Overview

Our overall method is summarized in Fig. 1. First, map images are preprocessed to soften out the irregularities commonly found in grid maps, which can be seen as high-frequency noise. Interest points (*features*) are then detected in these filtered images and descriptors computed to model their surroundings. Obviously, the choice of a particular interest point detector and descriptor will determine the performance of our whole method. After comprehensive experiments (refer to discussion in Section 5) we have determined that either the Harris[17] or the Kanade–Lucas–Tomasi (KLT)[24,34] detectors, in combination with a descriptor consisting of a circular patch centered at the feature, provide the best performance in terms of both maximizing the distinctiveness and reducing the computational cost.

Once features have been extracted from map images, a set of all the candidate correspondences $\mathcal{C}$ between features in both images is determined by means of a measure of similarity between their descriptors (as explained in Section 4.2). Due to ambiguity in maps, it is common for a given feature to have several candidate correspondences. From all those candidates, a modified RANSAC algorithm obtains subsets of internally consistent hypotheses $\mathcal{C}_i \subset \mathcal{C}$ by imposing *uniqueness* (each feature must correspond up to just one in the other map) and the *rigid transformation* constraint (the relative position of features must be the same in both maps). The uncertainty of all the variables is accounted for during the whole process; thus, all the decisions are taken upon stochastic tests. Unlike the standard RANSAC algorithm,[13] we propose to keep not only the solution with the largest number of supporting inliers but a

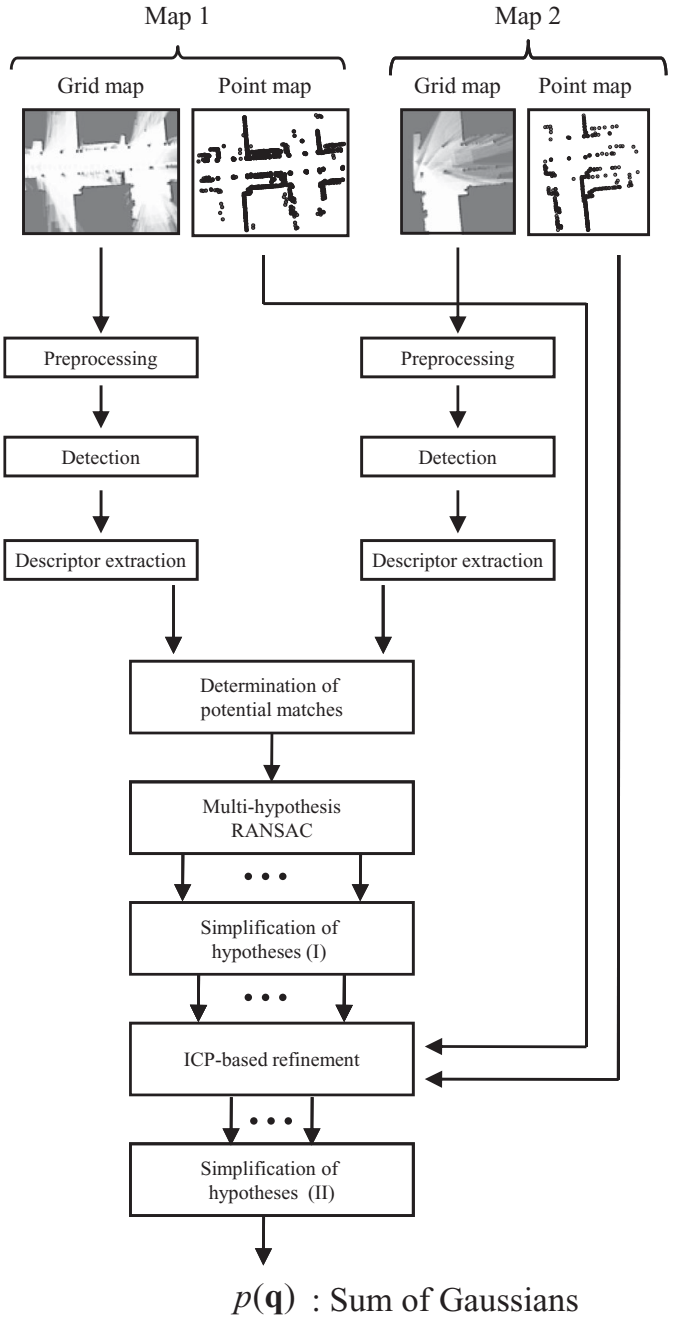


Fig. 1. An overview of the proposed method for map matching, which aligns a pair of maps each comprising a grid map and a point map. It first registers the grid maps to obtain a set of potential transformations $\mathbf{q}$, which are then refined employing the point maps and ICP-based alignment. The result is a probability density distribution for the actual $\mathbf{q}$ in the form of a mixture of Gaussians.

[1] See http://www.mrpt.org/Application:grid-matching

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

dynamic number of them. Each of these detected hypotheses leads to a particular rigid transformation, which is modeled as a Gaussian distribution over the space of translations and rotations.

The general form for the probability distribution of the rigid transformation $\mathbf{q}$ between two maps, as an SOG, can be written as

$$p(\mathbf{q}) = \sum_i \mathcal{N}(\mathbf{q}; \mathbf{q}_i^\star, \mathbf{Q}_i)\omega_i, \tag{1}$$

where each ω_i weights a Gaussian kernel centered at $\mathbf{q}_i^\star$ with covariance matrix $\mathbf{Q}_i$ and such that $\sum_i \omega_i = 1$. The distribution $p(\mathbf{q})$ can also be expanded using the law of total probability over all the potential sets of correspondences $\mathcal{C}_i$ as follows:

$$p(\mathbf{q}) = \sum_{\forall \mathcal{C}_i} p(\mathbf{q}|\mathcal{C}_i)P(\mathcal{C}_i). \tag{2}$$

Comparing Eq. (1) with Eq. (2) it is clear that we can choose $P(\mathcal{C}_i)$ as the SOG weights ω_i and model the density of $\mathbf{q}$ (given a set of correspondences $\mathcal{C}_i$) as a Gaussian distribution, that is,

$$p(\mathbf{q}|\mathcal{C}_i) = \mathcal{N}(\mathbf{q}; \mathbf{q}_i^\star, \mathbf{Q}_i). \tag{3}$$

The parameters of this distribution (its mean and covariance) will be derived in Section 7.

Finally, we should remark our proposal to *simplify* the SOG distribution generated by the RANSAC stage. This means that, whenever possible, two or more Gaussians are replaced by just one with the appropriate mean and covariance such that it closely covers the same volume than the original pair. We will follow here the method proposed by Runnalls:[30] only those simplifications whose Kullback–Leibler divergence (KLD) between the original and tentative simplified densities is below a threshold will be admitted. One of the reasons to simplify the SOG is reducing as much as possible the cost of the following *refinement* step, in which ICP[2] is applied to the point maps in order to improve the estimate of the mean map transformation $q^\star$. The resulting SOG is then tested again for further potential simplifications, obtaining the final, possibly multi-modal, density distribution for the map transformation.

3. Extraction of Features

In this section, we review some well-known image feature detectors and motivate the need for pre-processing the map images in order to improve the detection process.

3.1. Interest-point detectors

In a typical indoor occupancy grid map, we can easily identify natural features produced by scene elements, like corners, columns or, in general, any sharp edge. They also appear in some outdoor maps originated by vertical poles, building corners, vehicle edges, etc. These natural landmarks are suitable for matching maps of the same areas since they naturally occur in the environment and they are typically static.

All those interest points can be detected by interpreting the grid map as a grayscale image, the map image, and applying existing key-point detectors. The most desirable property of any detector is its *repeatability*, that is, its ability to detect a given feature when it appears in different images.

We are interested in the performance of the following four methods:

- The Harris detector,[17] which searches for points where the structure tensor has two large eigenvalues, revealing the existence of corners.
- The KLT method[24,34] also relies on the structure tensor. It detects salient points where one of the eigenvalues exceeds a given threshold.
- The detection phase of the SIFT algorithm,[23] which identifies scale-space extrema in pyramids of difference of Gaussians. This method aims at detecting *blobs* instead of corners.[26]
- The detector of SURF, based on an approximation to the Hessian matrix.[1]

There exists an issue in map images which affects the process of feature detection and needs to be handled appropriately. Grid mapping from laser range scans typically generates some artifacts in the maps which can be interpreted as high-frequency noise in the image (e.g. those arising from a single ray of the scans). To prevent the detection of spurious interest points in the middle of free space, we propose to pre-process the images by applying first a Gaussian filter and then a median filter to attenuate most of the irregularities. Next, we explain how we have tuned each filter for optimal detection performance.

3.2. Characterization

The set of maps employed in this characterization (available online[2]) consists of 10 pairs of grid maps from real robot data. We must remark that the maps represent real loop-closure situations with partial overlap and small differences in the grids caused by noise and different viewpoints of the robot. Since hundreds of key points are detected in each of these grids, our overall characterization can be considered significant from a statistical point of view.

In order to evaluate the repeatability of each interest point detector, we have applied it to both maps in each pair and then counted the number of common detected features, i.e. the same feature must be detected in *both* grid maps. The correct pairings were obtained then from ground truth transformations between the pairs of maps, computed manually. To avoid a bias in our results due to the number of detected points, we have limited the number of interest points to a fixed value proportional to the extension of each grid map (a typical value of 0.015 features per square meter is appropriate for all the maps employed in our comparison).

The results are summarized in Fig. 2 for each detector and for different values of W_g and W_m, the sizes of the Gaussian and the median filter, respectively. The values $W_g = 0$ and $W_m = 1$ correspond to a null filter in each case; thus, the

[2] Refer to the Web site http://www.mrpt.org/Paper:Occupancy_Grid_Matching.

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

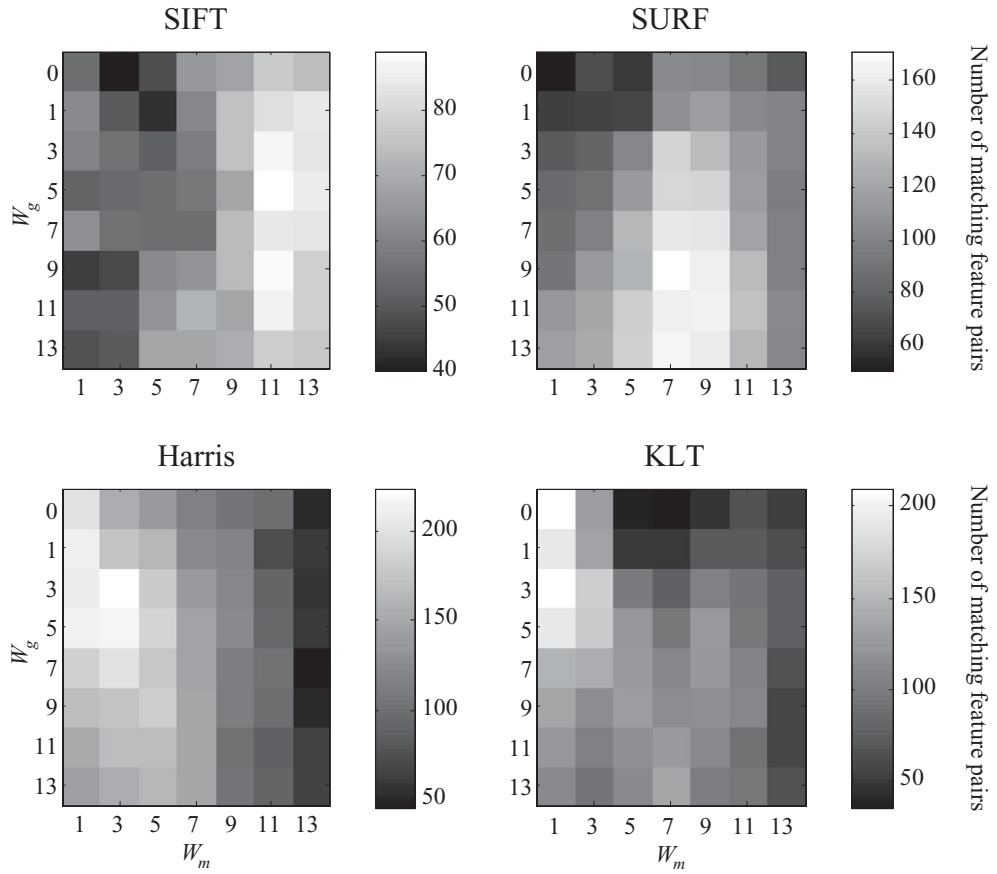


Fig. 2. A measure of the repeatability for each detector and for different sizes of the Gaussian (W_g) and median (W_m) filters used to smooth the map images. Brighter colors indicate a higher number of common features detected in both maps.

cases of applying just one of the filters (or none of them) have also been accounted for.

Observe how blob detectors (SIFT and SURF) perform well for large filter sizes (that lead to more "softened" images), whereas corner detectors (Harris and KLT) have good repeatability for slightly filtered images or even for maps not filtered at all (refer to KLT results in Fig. 2). Figure 3 shows an example of the different filters required by each detector to perform optimally. The best filter configuration for each detector has been employed in the benchmark presented in Section 5.1, and the corresponding overall number of matches can be seen in Fig. 6(f).

4. Descriptors

4.1. Review

Once the key points are detected they are assigned distinctive descriptors in order to establish correspondences. We have studied the performance of the following five image descriptors:[3]

- *SIFT*: This method is based on histograms of image gradients,[23] obtaining a 128-length descriptor vector.
- *SURF*: Based on the responses of Haar wavelets as described in ref. [1].
- *Intensity-domain spin images (Spin)*: A 2D histogram of intensities and distances,[22] with the maximum radius from the interest point determined by the parameter $R_{\max}$. The usage of distances (disregarding angles) makes this descriptor rotation invariant.
- *Linear or logarithmic circular patches*: These two descriptors have many similarities; hence, we discuss them here together. Both map a circular region of radius $R_{\max}$ centered at the interest point into a 2D matrix (the descriptor) of polar coordinates. Let this matrix be denoted by $\mathbf{f}(u, v)$, where the indices u and v denote different values of the distance and the angle from the feature, respectively. The idea is to extract a circular patch of the neighborhood of the feature in a representation which is not invariant to rotations, but where these rotations become just shifts in the angle dimension (v), as illustrated with the examples in Figs. 4(b–c). The only difference between the linear polar descriptor (*lin-polar* for short) and its logarithmic version (*log-polar*) is the usage of a linear or logarithmic scale in the distances.

Next, we address the problem of measuring the similarity between descriptors, a requisite to evaluate their distinctiveness.

[3] OpenCV implementations have been used for all the feature detectors and descriptors mentioned in this paper, except for (i) the SIFT method for which we rely on Hess' implementation[19] and (ii) the *lin-polar* descriptor, coded by the authors and released within OpenCV 2.0.

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

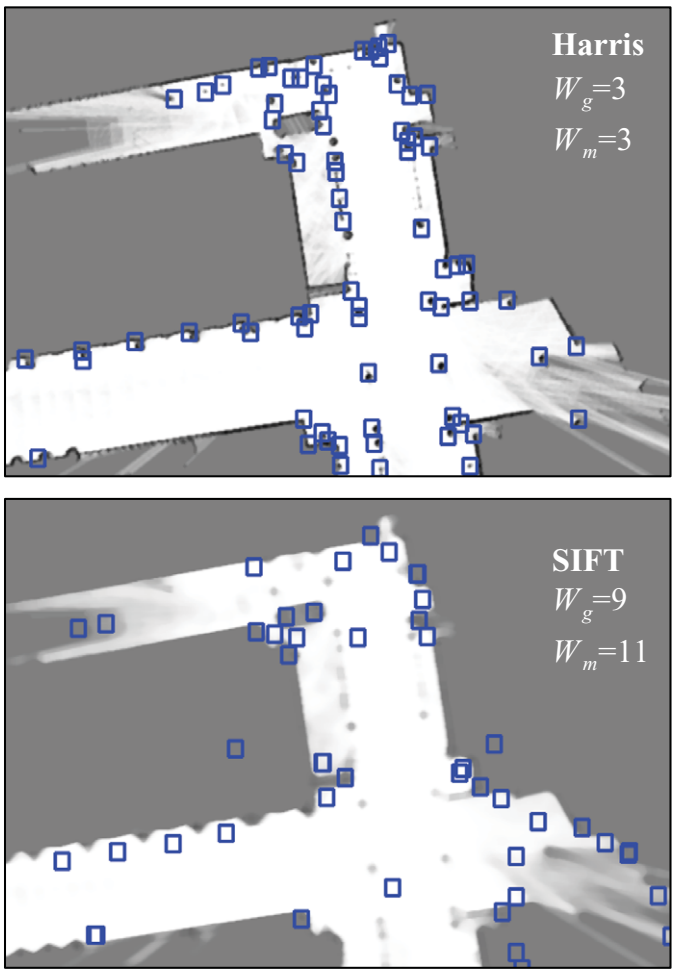


Fig. 3. (Colour online) One of the maps from the dataset, filtered with a Gaussian and median filter of sizes W_g and W_m, respectively. Detected interest points are marked with small squares for the Harris and SIFT detectors. Notice how each method detects a different kind of features (corners or blobs), hence the different filtering requirements.

4.2. A similarity function between descriptors

Given a pair of descriptors $\mathbf{f}_i^a$ and $\mathbf{f}_j^b$ for two key points i and j from maps a and b, respectively, we are interested in measuring their similarity. For the SIFT, SURF and Spin descriptors, the most natural measure is the Euclidean distance between the descriptor vectors. However, the cases of *lin-polar* and *log-polar* deserve more discussion since they are not directly invariant to orientation.

As illustrated in Fig. 4, the descriptors of two matching features only differ by a shift in the angular dimension. Therefore, we propose to measure the distance between two descriptors $\mathbf{f}_i$ and $\mathbf{f}_j$ by their Euclidean distance, given a rotation $\Delta\phi$, that is,

$$d\left(\mathbf{f}_i, \mathbf{f}_j, \Delta\phi\right) = \left(\sum_u \sum_v \left|\mathbf{f}_i(u, v) - \mathbf{f}_j(u, v + \Delta\phi)\right|^2\right)^{\frac{1}{2}}, \tag{4}$$

where the angular polar coordinate v is taken modulo the corresponding size of the matrix.

By computing the distance in Eq. (4) to a pair of descriptors $\mathbf{f}_i^a$ and $\mathbf{f}_j^b$, we obtain a distance vector for each possible shift in orientation $\Delta\phi$. As shown in Fig. 4, these distance vectors have pronounced minima for the true orientation when two features do really match; thus, we propose to measure the inter-feature distance in the cases of *lin-polar* and *log-polar* as

$$d\left(\mathbf{f}_i, \mathbf{f}_j\right) = \min_{\Delta\phi} d\left(\mathbf{f}_i, \mathbf{f}_j, \Delta\phi\right). \tag{5}$$

For all the descriptors in our comparison, we have normalized distances to the range $[0, 1]$ in order to keep homogeneity in the results presented in the next section.

5. Evaluation of Detectors and Descriptors

5.1. Benchmark

After defining a similarity measure for pairs of descriptors in Section 4.2, we are interested in obtaining a set of *candidate correspondences* between the features of two maps a and b, given their descriptors $\mathbf{f}_i^a$ and $\mathbf{f}_j^b$. The goodness of all the potential correspondences must be evaluated such as only the most promising pairings (those passing a given test) are considered as candidates. It is acceptable for each feature to have multiple potential correspondences in the other map, since a subsequent robust matching step (such as RANSAC[13]) can easily manage that ambiguity.

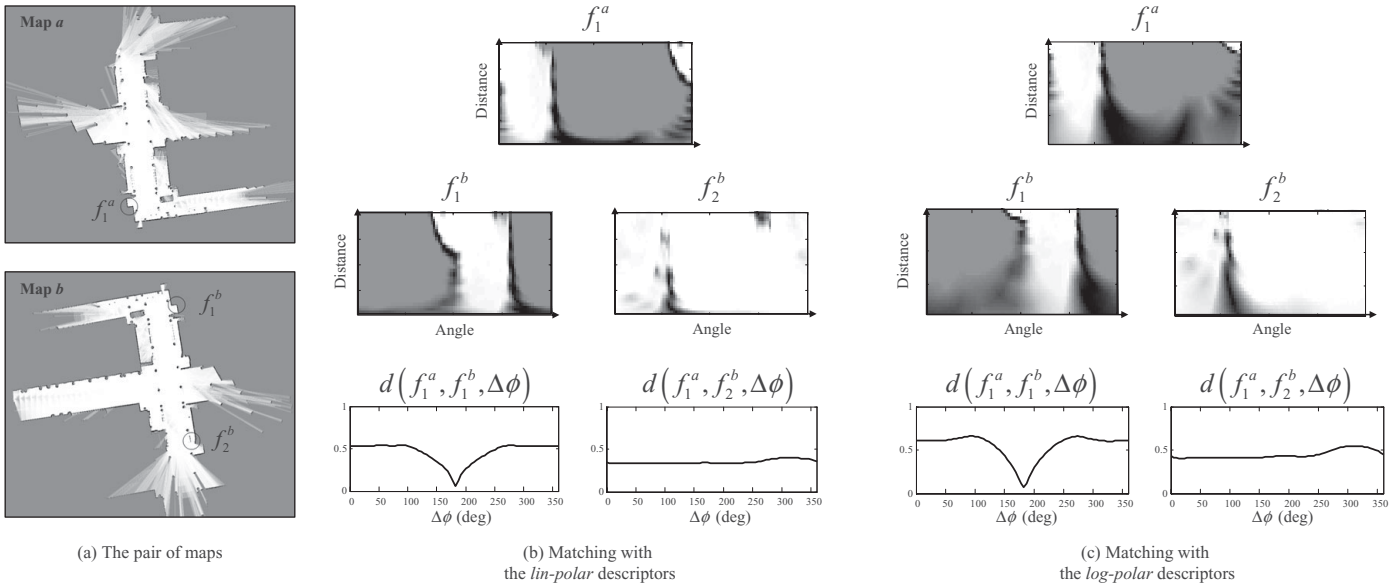
The arguably simplest test for selecting matchings is thresholding, which in our case means to accept a potential match between $\mathbf{f}_i^a$ and $\mathbf{f}_j^b$ only if the distance d_{ij} between their descriptors is below a fixed value T_d. However, this simple scheme has some drawbacks in the context of grid matching, because distance values between actually corresponding pairs may vary in a relatively large range. Thus, any permissive threshold T_d which covers most of the good correspondences would suffer from a high rate of false positives.

Following an idea similar to Lowe's proposal in ref. [23], we introduce a second condition for establishing candidate pairings: the associated distance d_{ij} must be not only below the threshold T_d but also sufficiently close to the best matching of $\mathbf{f}_i^a$ in map b, that is, the minimum of d_{ij} for all values of j (see Fig. 5). This restriction is characterized by a second threshold T_δ which states the maximum acceptable distance δ between a potential pairing and the best one, that is, $\delta_{ij} = d_{ij} - \min_j d_{ij}$. Note that for the extreme case $T_\delta = 0$, each feature will be associated with only one in the other map: the one with the closest descriptor. Both measures d_{ij} and δ_{ij} are illustrated with an example in Fig. 5 for clarity.

A benchmark has been carried out to obtain the optimal values for the thresholds T_d and T_δ from a training set of 10 pairs of submaps with known ground truth and for several combinations of detectors and descriptors. Optimal thresholds have been determined by minimizing the probability P_{err} of misclassifying a correspondence as a valid or an invalid candidate, given by

$$\begin{aligned} P_{\text{err}}(T_d, T_\delta) &= P(w)P_{\text{err}}(T_d, T_\delta|w) + P(v)P_{\text{err}}(T_d, T_\delta|v) \\ &= P(w)P(d_{ij} < T_d, \delta_{ij} < T_\delta|w) \\ &\quad + P(v)[1 - P(d_{ij} < T_d, \delta_{ij} < T_\delta|v)], \end{aligned} \tag{6}$$

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press



(a) The pair of maps (b) Matching with the *lin-polar* descriptors (c) Matching with the *log-polar* descriptors

Fig. 4. Example of matching features with different orientation. (a) An arbitrary reference feature f_1^a is highlighted in map a, and two potential pairings f_1^b (the real correspondence) and f_2^b are marked in map b. (b–c) The similarity between the feature descriptors is displayed as the distance function $d(f_i, f_j, \Delta\phi)$ for the cases of using the *lin-polar* and *log-polar* descriptors, respectively. Notice the pronounced minimum of the distance for the case of the real correspondence $f_1^a \leftrightarrow f_1^b$ close to the 180° relative rotation. We must remark that hundreds of different discrete orientations have been evaluated in this figure with the purpose of generating a clear illustration, while in practice as few as eight discrete orientations are enough for achieving excellent discrimination.

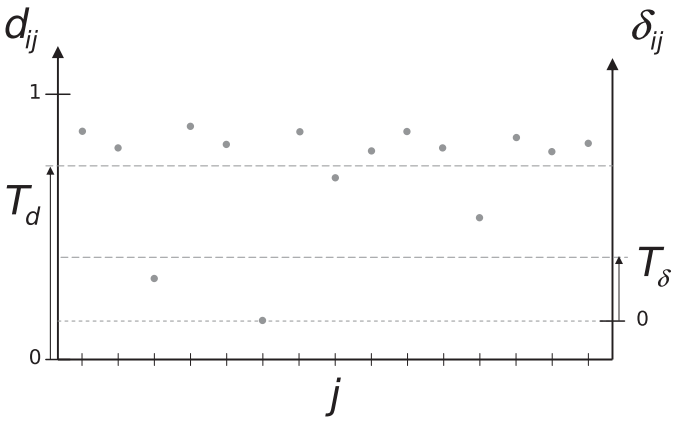


Fig. 5. A schematic illustration of the distance between descriptors d_{ij} and the index δ_{ij}, which measures those distances relative to the closest one for each given feature i. Note that, by definition, the best pairing is always assigned a value $\delta_{ij} = 0$. A pairing will be accepted only if it is below both thresholds T_d (absolute) and T_δ (relative to the minimum distance).

which can be evaluated given knowledge of the joint densities $p(d, \delta|v)$ and $p(d, \delta|w)$, where v and w denote valid and wrong pairings, respectively. The expression above can easily be derived by noticing that a misclassification will occur when (i) a distance d_{ij} passes both thresholds and it was a wrong association (first term in the sum), or (ii) a valid pairing does not pass the thresholds (second term). For our analysis, we assume no *a priori* information about the probability of being in a valid or invalid pairing; thus, we have $P(v) = P(w) = 1/2$. The joint conditional densities $p(d_{ij}, \delta_{ij}|v)$ and $p(d_{ij}, \delta_{ij}|w)$ have been estimated from histograms generated by evaluating all the potential pairings in the 10 pairs of submaps, which amounts to 220 valid and 240,000 invalid correspondences.

The results of the benchmark are summarized in Fig. 6(e), which shows the minimum classification error P_{err} attainable by each combination of feature detector and descriptor, along the associated average computation time (for one whole submap). These times include detection, descriptor extraction and distance computations, but they do not include the preprocessing filters discussed in Section 3.2. This preprocessing would add an average of 10–200 ms, with larger computational burdens associated with SIFT and SURF since they require larger filter kernels than the Harris or KLT methods.

Note that for those descriptors parameterized by a maximum radius R_{max} (see Section 4.1), we present the results only for the value that minimizes the classification error. However, this is a non-critical parameter since any value in the range of 1–3 m gives very similar results. The angular and radial resolutions of the *lin-polar* and *log-polar* descriptors were set to eight and six bins, respectively. Increasing these parameters would in theory make them more distinctive but in practice the impact was little; thus, we employed the minimum values that do not degrade performance significantly.

5.2. Discussion

The first important conclusion we can extract from our comparison is that no descriptor can tell valid pairings from wrong ones with a classification error below $\sim 20\%$, which is clearly a consequence of the ambiguity of features in map images where many look quite similar locally. Still, discarding $\sim 80\%$ of the wrong pairings provides an invaluable improvement to the subsequent robust matching

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

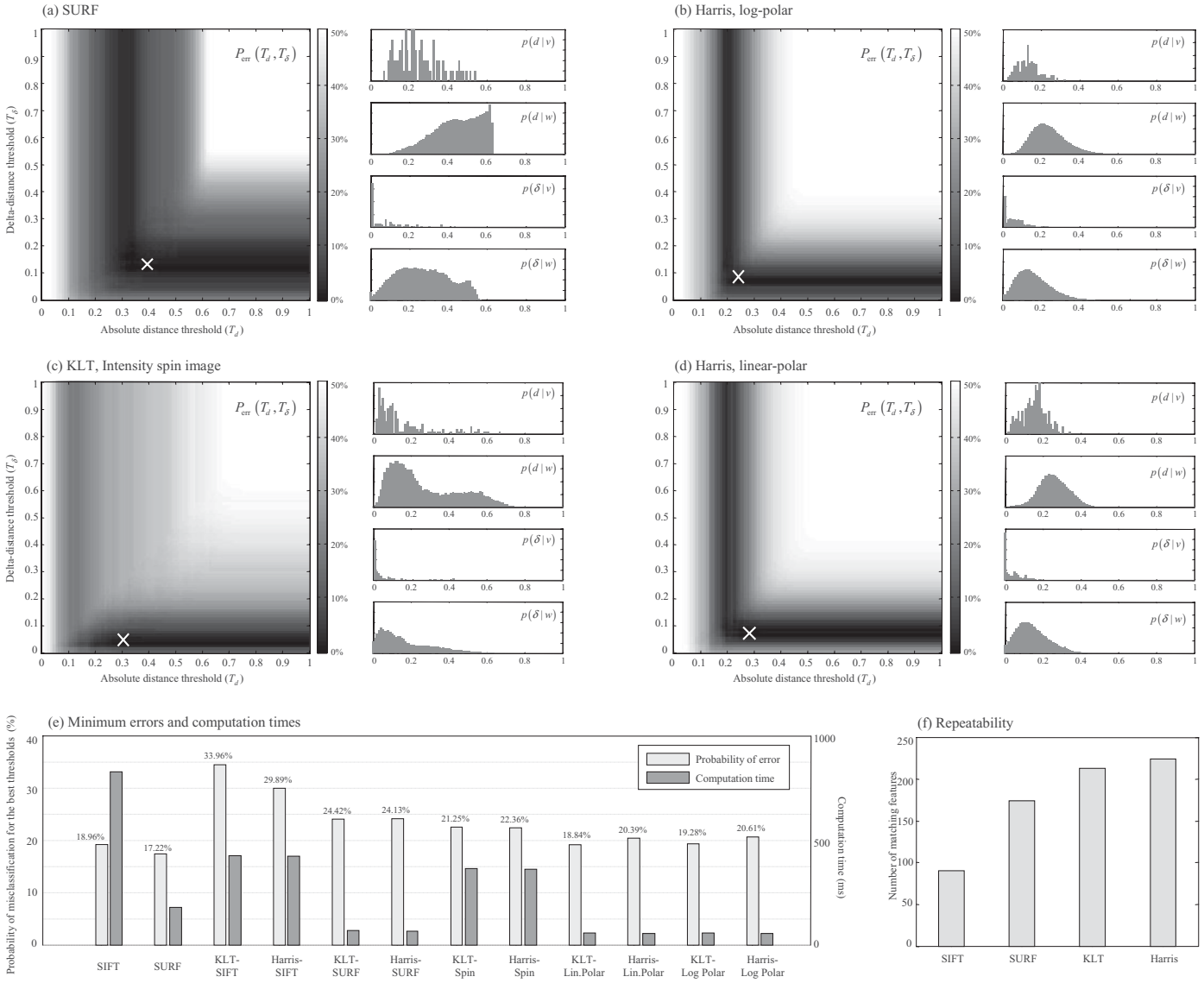


Fig. 6. The benchmark of feature detectors for grid map matching. (a–d) Four examples of the expected P_{err} for different values of thresholds T_d and T_δ. The point with the minimum P_{err} is marked with a cross in each figure. We have also shown the marginal conditional distributions for the distance d and the distance difference δ for valid (v) and wrong (w) associations are shown on the right hand of each subfigure. (e) For each combination of detector and descriptor, the resulting overall probability of classification error P_{err} for its best thresholds, i.e. that marked with a cross in (a–d), along with its average computation time for one map. (f) A measure of the repeatability for each detector.

algorithm (see Section 6), since it will have to deal with a reduced fraction of outliers.

It is interesting to note that the SIFT and SURF descriptors have a much poorer performance when computed for interest points localized by the Harris or the KLT detectors (third to sixth values in the bar graph) than when computed as proposed in their original methods (the first two values in the graph). As commented in Section 3.1 and illustrated in Fig. 6(f), this has important consequences for the practical applicability of those descriptors to grid matching, since the original SIFT and SURF detectors have poorer repeatability than the Harris and KLT methods. Subsequently, we discard the usage of these two descriptors as the optimal solution since they lead to quite similar error ratios (P_{err}) than the other descriptors while severely reducing the number of matched points and implying a higher computational burden, as can be seen in Fig. 6(e).

In Figs. 6(a–d), it represents the computed $P_{\text{err}}(T_d, T_\delta)$ for some selected methods along the marginal distributions obtained in our benchmark. Observe how the marginal $p(\delta_{ij}|v)$ presents a clear peak at the origin ($\delta_{ij} = 0$) for all the methods, which indicates that the closest feature is often the actual correspondence.[4] However, this is not always the case; hence, the optimal T_δ values are not exactly zero.

Note that the worst obtained value for P_{err} (0.5, shown in white in the graphs) is obtained for a wide range of threshold values, while more reduced error ratios only appear for a certain band of the parameters (shown by darker areas). The thickness of these bands is related to the distinctiveness of the descriptors, as can be observed in the densities of descriptor distances for valid and wrong pairings (the histograms at the

[4] Recall that, by definition, $\delta_{ij} = 0$ means that feature $\mathbf{f}_j$ has the minimum distance to feature $\mathbf{f}_i$.

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

Algorithm 1 robust_transform $(\mathcal{C}, \{p_i^A\}, \{p_j^B\}) \rightarrow SOG$

1: $SOG \Leftarrow \emptyset$
2: $iter \Leftarrow 1$
3: **repeat** // *RANSAC iterations*
4: $\quad \hat{\mathcal{C}} \Leftarrow \{c_{k_1}, c_{k_2}\} \subset \mathcal{C} | uniqueness(c_{k_1}, c_{k_2})$
5: $\quad$ **if** $D_M^2(c_{k_1}, c_{k_2}) < \chi_{c,1}^2$ **then** // *Consistency test*
6: $\quad\quad$ **if** $\exists k | \hat{\mathcal{C}} \subset SOG_k.\hat{\mathcal{C}}$ **then** // *Already in?*
7: $\quad\quad\quad$ // *Increment the weight*
8: $\quad\quad\quad SOG_k.\omega \Leftarrow SOG_k.\omega + 1$
9: $\quad\quad$ **else**
10: $\quad\quad\quad$ // *It is a new SOG mode*
11: $\quad\quad\quad \hat{\mathcal{C}}_o \Leftarrow \hat{\mathcal{C}}$ // *Save original minimal set*
12: $\quad\quad\quad$ **repeat** // *Incorporate inliers*
13: $\quad\quad\quad\quad \hat{\mathcal{C}}_t \Leftarrow \hat{\mathcal{C}}_o \cup (i^\star, j^\star)$ // *Tentative set of pairings*
14: $\quad\quad\quad\quad (q_i^\star, \mathbf{Q}_i^\star) \Leftarrow opt\_transf(\hat{\mathcal{C}}_t)$ // *See Eqs.(10),(15)*
15: $\quad\quad\quad\quad (i^\star, j^\star) \Leftarrow \arg\max_{(i,j)} \int p_i(\xi)\tilde{p}_j(\xi)d\xi$
16: $\quad\quad\quad\quad$ **if** $D_M^2(i^\star, j^\star) < \chi_{c,2}^2$ **then**
17: $\quad\quad\quad\quad\quad \hat{\mathcal{C}} \Leftarrow \hat{\mathcal{C}} \cup (i^\star, j^\star)$ // *Accept pairing*
18: $\quad\quad\quad\quad$ **end if**
19: $\quad\quad\quad$ **until** $D_M^2(i^\star, j^\star) \geq \chi_{c,2}^2$
20: $\quad\quad\quad$ **if** $|\hat{\mathcal{C}}| \geq M$ **then** // *Minimum inlier support*
21: $\quad\quad\quad\quad$ // *New Gaussian mode with* $\omega = 1$
22: $\quad\quad\quad\quad (q_i^\star, \mathbf{Q}_i^\star) \Leftarrow opt\_transf(\hat{\mathcal{C}})$ // *Use Eqs.(10), (15) with final set*
23: $\quad\quad\quad\quad SOG \Leftarrow SOG \cup (\hat{\mathcal{C}}, 1, (q_i^\star, \mathbf{Q}_i))$
24: $\quad\quad\quad$ **end if**
25: $\quad\quad$ **end if**
26: $\quad$ **end if**
27: $\quad iter \Leftarrow iter + 1$
28: **until** $iter > maxIters$ // *With maxIters computed as in ref. [18]*

right hand of each P_{err} graph). For instance, compare the histograms $p(d|v)$ and $p(d|w)$ for the SURF and the Spin descriptors in Figs. 6(a–c), where it is clear that in SURF the histograms concentrate in relatively different areas (easing the decision of where to place the threshold) whereas this is definitively not the case for the Spin descriptor.

As a final conclusion from our benchmark, the *lin-polar* and *log-polar* descriptors, both with virtually identical performance, emerge as the best choices for grid matching in combination with either Harris or KLT detector, due to their reduced misclassification probability and faster computation time.

6. Construction of the SOG: The Modified RANSAC Algorithm

Subsets of self-consistent correspondences $\mathcal{C}_i \in \mathcal{C}$ can be extracted with RANSAC, a consensus-based method to distinguish inliers from outliers.[13] However, in our problem, it is not enough to keep the hypothesis with most supporting inliers since ambiguity in grid matching can lead to multiple mutually incompatible but internally consistent subsets $\mathcal{C}_i$. We propose instead to maintain each of those hypotheses as a Gaussian mode in the SOG (refer to Eq. (2)); hence, the need to modify the RANSAC algorithm to allow the existence of multiple hypotheses.

Next, we describe the complete process, which has been also specified in Algorithm 1 for clarity. First, two correspondences (the minimum number required to unequivocally determine the distribution of the associated map transformation $p(\mathbf{q}|\mathcal{C}_i)$) are randomly chosen from $\mathcal{C}$ to initialize the subset $\mathcal{C}_i = \{c_{k_1}, c_{k_2}\}$ (line 4 of the algorithm). The *uniqueness* constraint is tested first, that is, in a valid pairing one given feature cannot appear in both correspondences c_{k_1} and c_{k_2} simultaneously. Then, the feasibility of this pair is tested by a χ^2 test (line 5) which detects inconsistencies between the inter-feature spatial distances d_a and d_b measured in the two maps a and b (refer to the example in Fig. 7a). As shown in the Appendix, if

$$\frac{\left(d_a^2 - d_b^2\right)^2}{8\sigma^2\left(d_a^2 + d_b^2\right)} < \chi_{1,c}^2 \tag{7}$$

holds, we can accept that the distances are consistent within a confidence of c, where $\chi_{n,c}^2$ stands for the inverse χ^2 cumulative distribution with n degrees of freedom.

Next, it must be determined the number of inliers supporting the hypothesis $p(q|\mathcal{C}_i)$ defined by each set of initial pairings $\mathcal{C}_i$. This is achieved by establishing associations between all the features in map b and those in a transformed by $\mathbf{q}$. Note that this is a stochastic data-association problem since all feature locations, and the transformation itself, have associated uncertainties.

A robust method for stochastic data association is the Joint Compatibility Branch and Bound (JCBB),[28] but unfortunately its exponential time complexity makes it impractical for our problem, where each map will typically contain about 100 features.

Our alternative, detailed in Algorithm 1, consists of sequentially incorporating (lines 12–19) matches which optimize the integral of the product of the two Gaussians, which can be interpreted as the likelihood of the two points sharing the same position in space—that is, it is the *matching likelihood*[7] of the pairing. The incorporation of inliers stops when the next best pairing candidate (i, j) has a squared Mahalanobis distance $D_M^2(i, j)$ above a given threshold $\chi_{c,2}^2$.

The above process is repeated a number of times and updated dynamically as new inliers are found, as described in ref. [18]. Regarding the weights of the SOG, each Gaussian mode is initially assigned a unit weight, which is incremented each time the same subset of correspondences is found in subsequent iterations (lines 6–8). An optimization of this approach is to test whether the two first correspondences $\mathcal{C}_i$ are already part of another $\mathcal{C}_j$, and in that case, to increment the weight ω_j. This heuristic is justified by the observation that the same set of self-consistent pairings will be obtained if the two first ones were different but belonging to the final subset.

Note as well the existence of a minimum number of required inliers M in order to accept a hypothesis (line 20 of the algorithm). In our experiments, this threshold has been heuristically set to a $\sim$ 15% of the average number of features found in each map. This restriction prevents the detection of

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

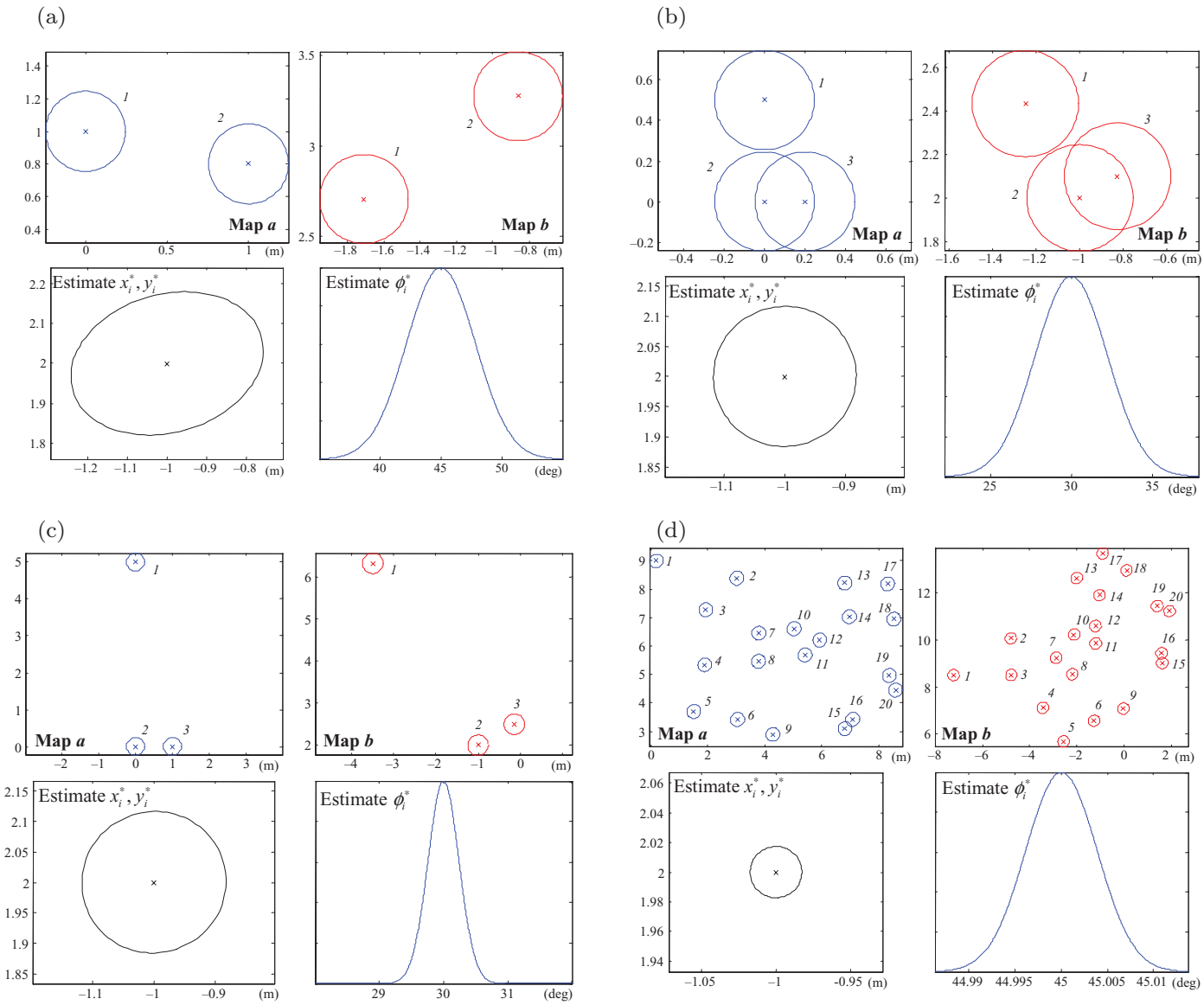


Fig. 7. (Colour online) Four sets of correspondences between two synthetic maps a and b for different spatial distributions and number of detected features. Here, the position uncertainty for all the features has been set to $\sigma_p = 0.10$ and ellipses represent 95% confidence intervals. The inter-feature distances measured in the different maps, d_a and d_b, as employed in Eq. (7), are shown in (a) as an example.

spurious hypotheses with very few supporting inliers caused by pure chance when two maps do not really match.

7. Uncertainty of the Optimal Transformation

Given a set of point correspondences from a pair of maps, it is well known that a closed-form solution exists for finding the rigid transformation between them that is optimal in the sense of least mean-square error (LMSE).[2,20,25] We contribute here with a derivation of the *uncertainty* associated with this optimal solution with the purpose of making our formulation usable within probabilistic localization and mapping frameworks. Taking such uncertainty into account is essential, since the position of any feature is always prone to error, mainly because of the discrete nature of maps and because of the limited precision of the interest point detectors (in the order of one pixel, that is, the size of one grid cell—typically in the range of 5–20 cm for mobile robot grid maps). Additionally, the spatial distribution of features on the map is crucial to the precision in the transformation, as discussed at the end of this section.

Given a certain set of feature correspondences $\mathcal{C}_i$, we model the probability density of a rigid transformation between maps $\mathbf{q} = [x\ y\ \phi]^\top$ as a Gaussian distribution, that is,

$$p(\mathbf{q}|\mathcal{C}_i) = \mathcal{N}(\mathbf{q}; \mathbf{q}_i^\star, \mathbf{Q}_i), \tag{8}$$

where $\mathbf{q}_i^\star$ and $\mathbf{Q}_i$ represent the corresponding mean and covariance matrix, respectively. In the following, we derive expressions for the parameters of this distribution. The basic idea is to take the optimal solution for the map transformation as the mean of the Gaussian, while the covariance matrix is approximated by uncertainty propagation through a first-order Taylor series approximation of the resulting function, as explained below.

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

Let $\mathbf{p}_k^a = [x_k^a \ y_k^a]^\top$ and $\mathbf{p}_k^b = [x_k^b \ y_k^b]^\top$ be the position of the kth feature in maps a and b, respectively. Then, given a set $\mathcal{C}_i$ of correspondences, i.e. pairs of map feature indices in each map (i_a, i_b), we can define the squared error of the feature matching for any rigid transformation $\mathbf{q}$ as

$$E_{\mathcal{C}_i}(\mathbf{q}) = \sum_{\forall (i_a, i_b) \in \mathcal{C}_i} \left| \mathbf{p}_{i_b}^b - (\mathbf{q} \oplus \mathbf{p}_{i_a}^a) \right|^2, \tag{9}$$

where $\oplus$ represents the pose composition operator.[4] In the 2D case, the optimal transformation $\mathbf{q}_i^\star = [x_i^\star \ y_i^\star \ \phi_i^\star]^\top$ that minimizes this error can be obtained by equaling to zero the derivative of Eq. (9) with respect to the transformation $\mathbf{q}$, which leads to the closed-form solution:[25]

$$\frac{\partial E_{\mathcal{C}_i}(\mathbf{q}_i^\star)}{\partial \mathbf{q}} = 0 \qquad \rightarrow$$

$$\mathbf{q}_i^\star = \begin{bmatrix} \bar{x}^a - \bar{x}^b \frac{\Delta_x}{\sqrt{\Delta_x^2+\Delta_y^2}} + \bar{y}^b \frac{\Delta_y}{\sqrt{\Delta_x^2+\Delta_y^2}} \\ \bar{y}^a - \bar{x}^b \frac{\Delta_y}{\sqrt{\Delta_x^2+\Delta_y^2}} - \bar{y}^b \frac{\Delta_x}{\sqrt{\Delta_x^2+\Delta_y^2}} \\ \tan^{-1}\left(\frac{\Delta_y}{\Delta_x}\right) \end{bmatrix}, \tag{10}$$

where $\bar{x}^a$, $\bar{y}^a$, $\bar{x}^b$ and $\bar{y}^b$ are the means (average values) of the vectors $\mathbf{x}^a$, $\mathbf{y}^a$, $\mathbf{x}^b$ and $\mathbf{y}^b$, respectively, which contain the 2D coordinates of features within maps a and b. We have also introduced the auxiliary scalar terms Δ_x and Δ_y, defined as

$$\begin{aligned} \Delta_x &= N\left(\sum_k x_k^a x_k^b + \sum_k y_k^a y_k^b\right) - N^2\left(\bar{x}^a \bar{x}^b + \bar{y}^a \bar{y}^b\right), \\ \Delta_y &= N\left(\sum_k y_k^a x_k^b - \sum_k x_k^a y_k^b\right) + N^2\left(\bar{x}^a \bar{y}^b - \bar{y}^a \bar{x}^b\right), \end{aligned} \tag{11}$$

with $N = |\mathcal{C}_i|$ denoting the number of pairings in $\mathcal{C}_i$.

The optimal transformation in Eq. (10) can then be seen as a function $\mathbf{q}_i^\star = \mathbf{q}(\mathbf{z})$ of six auxiliary variables, which we can stack into the vector $\mathbf{z} = \left[\bar{x}^a \ \bar{y}^a \ \bar{x}^b \ \bar{y}^b \ \Delta_x \ \Delta_y\right]^\top$. In order to estimate the covariance matrix $\mathbf{Q}_i$ that models the uncertainty of the optimal transformation, we use first-order uncertainty propagation, for which it is first needed the multivariate Gaussian distribution of the vector of auxiliary variables $\mathbf{z}$. This vector is a function of the 2D coordinates of all the features $\mathbf{x}^a$, $\mathbf{y}^a$, $\mathbf{x}^b$ and $\mathbf{y}^b$ (which all are known input data). Assuming that these coordinates are corrupted with an additive, zero-mean Gaussian noise with known covariance matrices $\mathbf{X}^a$, $\mathbf{Y}^a$, $\mathbf{X}^b$ and $\mathbf{Y}^b$, we can approximate the covariance of $\mathbf{z}$ by

$$\Sigma_{\mathbf{z}} = \mathbf{J}_z \begin{bmatrix} \mathbf{X}^a & 0 & 0 & 0 \\ 0 & \mathbf{Y}^a & 0 & 0 \\ 0 & 0 & \mathbf{X}^b & 0 \\ 0 & 0 & 0 & \mathbf{Y}^b \end{bmatrix} \mathbf{J}_z^\top. \tag{12}$$

Since $\mathbf{z}$ depends on the whole set of feature coordinates, the Jacobian matrix $\mathbf{J}_z = \frac{\partial \mathbf{z}}{\partial \{\mathbf{x}^a, \mathbf{y}^a, \mathbf{x}^b, \mathbf{y}^b\}}$ has a dimensionality of $6 \times 4N$. In despite of the large size of the matrices involved in Eq. (12), important simplifications are possible because of the following properties of the feature covariances:

- Since in most feature detectors each point is detected independently, Gaussian errors in the coordinates of different features are uncorrelated.
- As a consequence of this independent detection, all features may be assigned the same covariance.
- It is plausible for most interest point detectors to assume an isotropic distribution for the localization errors.

These assumptions are widely accepted in the computer vision literature.[9,31,32,35] To sum up, it seems plausible to accept that $\mathbf{X}^a$, $\mathbf{Y}^a$, $\mathbf{X}^b$ and $\mathbf{Y}^b$ are diagonal matrices with the same variance for all the coordinates, which we will name σ_p^2. By replacing the covariance matrices by their values in Eq. (12), we end up with the following diagonal matrix:

$$\Sigma_{\mathbf{z}} = \sigma_p^2 \begin{bmatrix} \frac{1}{N}\mathbf{I}_4 & \mathbf{0}_{4\times 2} \\ \mathbf{0}_{2\times 4} & \beta \mathbf{I}_2 \end{bmatrix}, \tag{13}$$

with β given by

$$\beta = N^2(N-1)\left(\hat{\sigma}_{x^a}^2 + \hat{\sigma}_{y^a}^2 + \hat{\sigma}_{x^b}^2 + \hat{\sigma}_{y^b}^2\right), \tag{14}$$

where the constants $\hat{\sigma}_{x^a}^2$, $\hat{\sigma}_{y^a}^2$, $\hat{\sigma}_{x^b}^2$ and $\hat{\sigma}_{y^b}^2$ represent the unbiased estimates of the variance for their corresponding vectors.

At this point, we can proceed with the derivation of the covariance of $\mathbf{q}_i^\star$. By computing the Jacobian of Eq. (10) with respect to $\mathbf{z}$, $\mathbf{J}_q = \mathbf{q}_i^\star / \partial \mathbf{z}$, it follows that the covariance $\mathbf{Q}_i$ is proportional to the uncertainty of the individual features σ_p^2, that is

$$\mathbf{Q}_i = \mathbf{J}_q \Sigma_z \mathbf{J}_q^\top = \sigma_p^2 \begin{pmatrix} C_{11} & C_{12} & C_{13} \\ C_{12} & C_{22} & C_{23} \\ C_{13} & C_{23} & C_{33} \end{pmatrix}, \tag{15}$$

where the matrix terms are given by

$$\begin{aligned} C_{11} &= \frac{2}{N} + \beta\left(\frac{\bar{x}^b \Delta_y + \bar{y}^b \Delta_x}{\Delta_x^2 + \Delta_y^2}\right)^2, \\ C_{22} &= \frac{2}{N} + \beta\left(\frac{\bar{x}^b \Delta_x - \bar{y}^b \Delta_y}{\Delta_x^2 + \Delta_y^2}\right)^2, \\ C_{33} &= \frac{\beta}{\Delta_x^2 + \Delta_y^2}, \\ C_{12} &= \beta \frac{\left(\bar{x}^b \Delta_y + \bar{y}^b \Delta_x\right)\left(\bar{y}^b \Delta_y - \bar{x}^b \Delta_x\right)}{\left(\Delta_x^2 + \Delta_y^2\right)^2}, \\ C_{13} &= \beta \frac{\bar{x}^b \Delta_y + \bar{y}^b \Delta_x}{\left(\Delta_x^2 + \Delta_y^2\right)^{\frac{3}{2}}}, \\ C_{23} &= \beta \frac{\bar{y}^b \Delta_y - \bar{x}^b \Delta_x}{\left(\Delta_x^2 + \Delta_y^2\right)^{\frac{3}{2}}}. \end{aligned} \tag{16}$$

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

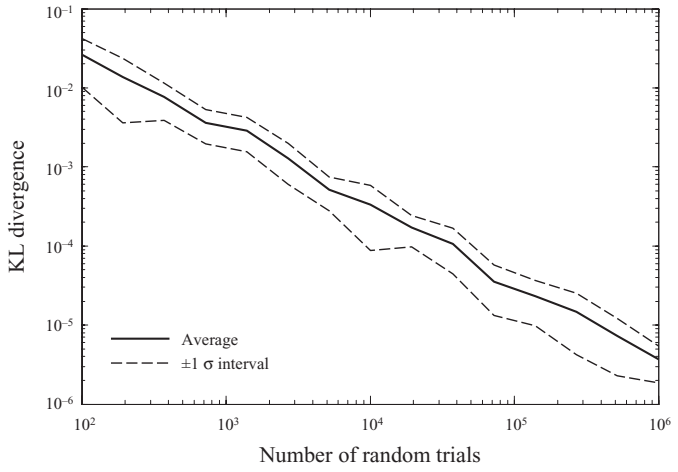


Fig. 8. The Kullback–Leibler divergence (KLD) between our theoretical model for the covariance $\mathbf{Q}_i$ and its value from a Monte Carlo simulation for an increasing number of trials. Confidence intervals are shown for the KLD since the values at each point were computed for 50 different maps generated by randomly positioned features.

To illustrate some results for this covariance estimation, the transformations computed from four sets of feature correspondences are shown in Fig. 7, along with their 2D uncertainty ellipses for $[x_i^\star \; y_i^\star]^\top$ and the densities of $\phi_i^\star$. These examples illustrate some interesting properties of the resulting uncertainty. First, the uncertainty in the orientation $\phi_i^\star$ strongly depends on the spatial distribution of the features, since more precise estimations can be made from features distributed over larger areas. This can be clearly observed by comparing the two cases shown in Figs. 7(b–c). Second, the uncertainty in the 2D coordinates of $\mathbf{q}_i^\star$ decreases with the number of features N only for very low values of N. This can be explained by the term $2/N$ becoming negligible in the expressions for C_{11} and C_{22}, where the second term does not decrease for increasingly larger values of N.

In order to validate our model of the covariance $\mathbf{Q}_i$, we have evaluated the Kullback–Leibler divergence between our model and the covariance obtained from a Monte Carlo simulation comprising six pairs of correspondences between randomly located features corrupted with Gaussian noise. The results, in Fig. 8, reveal that the experimentally obtained covariance approaches the theoretical model as the number of Monte Carlo trials increases.

At this point, we have described a closed-form, optimal solution for the map transformation and derived a Gaussian approximation to its associated uncertainty for any given set of correspondences. The derived expressions are needed during the RANSAC stage discussed in Section 6, specifically in the step denoted as *opt_transf* in Algorithm 1.

8. Results

In this section, we present experiments aimed at testing the robustness of our approach. For all these results, we have employed the Harris corner detector and the linear-polar descriptor to establish correspondences between 10 cm resolution grid maps.

8.1. Performance under errors and noise

Maps built by a mobile robot at different moments in time may present significant differences due to both dynamic objects and errors in the robot localization while mapping. To quantify the accuracy of our method against such differences, we have matched a reference map, built from real data, to a transformed one with known ground-truth translation and rotation—see the left column in Fig. 9.

Two sources of errors have been evaluated. First, the estimated robot path in the environment (which in turn determines the accuracy of the map itself[16]) has been deliberately corrupted by Gaussian noise with a standard deviation of σ_p. As σ_p increases, so does the degradation of the test map, as illustrated in the top row of the figure. It can be seen how the corresponding errors in the map transformation as detected by our method increases with larger σ_p, which is explained by both the more erroneous locations of detected features and their more reduced repeatability, shown in the rightmost column of the figure. Note that repeatability is a desired property of any feature detector since it ensures that the same physical point is detected in two different maps in spite of potential changes in orientation or minor differences in the feature surroundings. For completeness, we also repeated this experiment with a standard RANSAC implementation. Note that since in this test map there exist no chances for multi-hypothesis matching (that is, the map does not present a real ambiguity), the accuracy of standard RANSAC should match that of our method, and indeed this is what we verified.

Second, we also evaluated the effects of noise in the laser scanner ranges, characterized by a standard deviation of σ_l. As reflected in Fig. 9, our method is less sensitive to this kind of error, probably because the preprocessing of map images smoothes out part of the noisy measurements.

We must remark that the error and noise levels probed in this characterization are much higher than those expected in real-world conditions (the ranges of realistic values are marked in the graphs). Therefore, the errors of our method under normal conditions are expected to be below 10 cm, approximately.

8.2. Performance in loop-closure detection

The following benchmark characterizes the performance of our method in its natural application to hierarchical SLAM,[5,12] that is, in detecting loop closures from local, metric submaps. For this aim, we have selected four publicly available datasets. Three of them, the Freiburg campus dataset, the Intel dataset and the MIT dataset, are published in the Radish repository,[21] while the fourth was collected by the authors at the Málaga campus.[5] See Fig. 10 for example submaps from each dataset.

All these datasets have been processed within our Hybrid Metric-Topological (HMT) SLAM framework, presented elsewhere.[5] In this framework, the original sequence of robot observations is grouped into segments of consecutive observations (the submaps) according to a natural metric of similarity.[6] For convenience, we disabled topological

[5] Available online at http://www.mrpt.org/Malaga_2006_campus_dataset

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

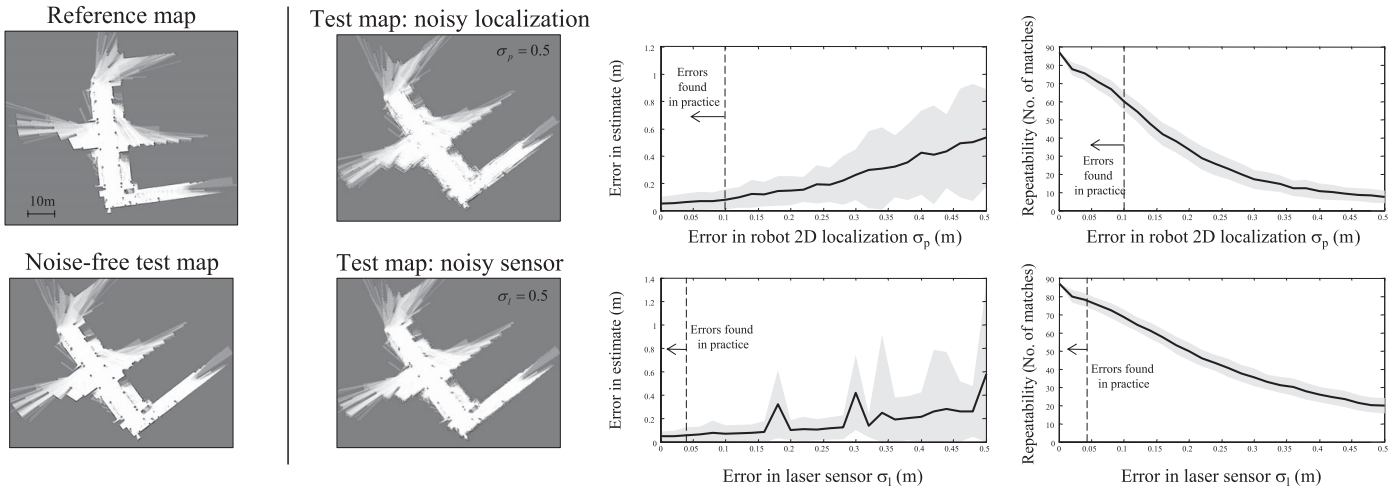


Fig. 9. Characterization of our method under the presence of localization errors (σ_p) and laser sensor errors (σ_l). The average error in the map transformation (from the most likely Gaussian mode in the SOG) and the average number of matches between the pair of maps are shown by the thick plot, while the $\pm 1\sigma$ confidence intervals are represented by the shaded region.

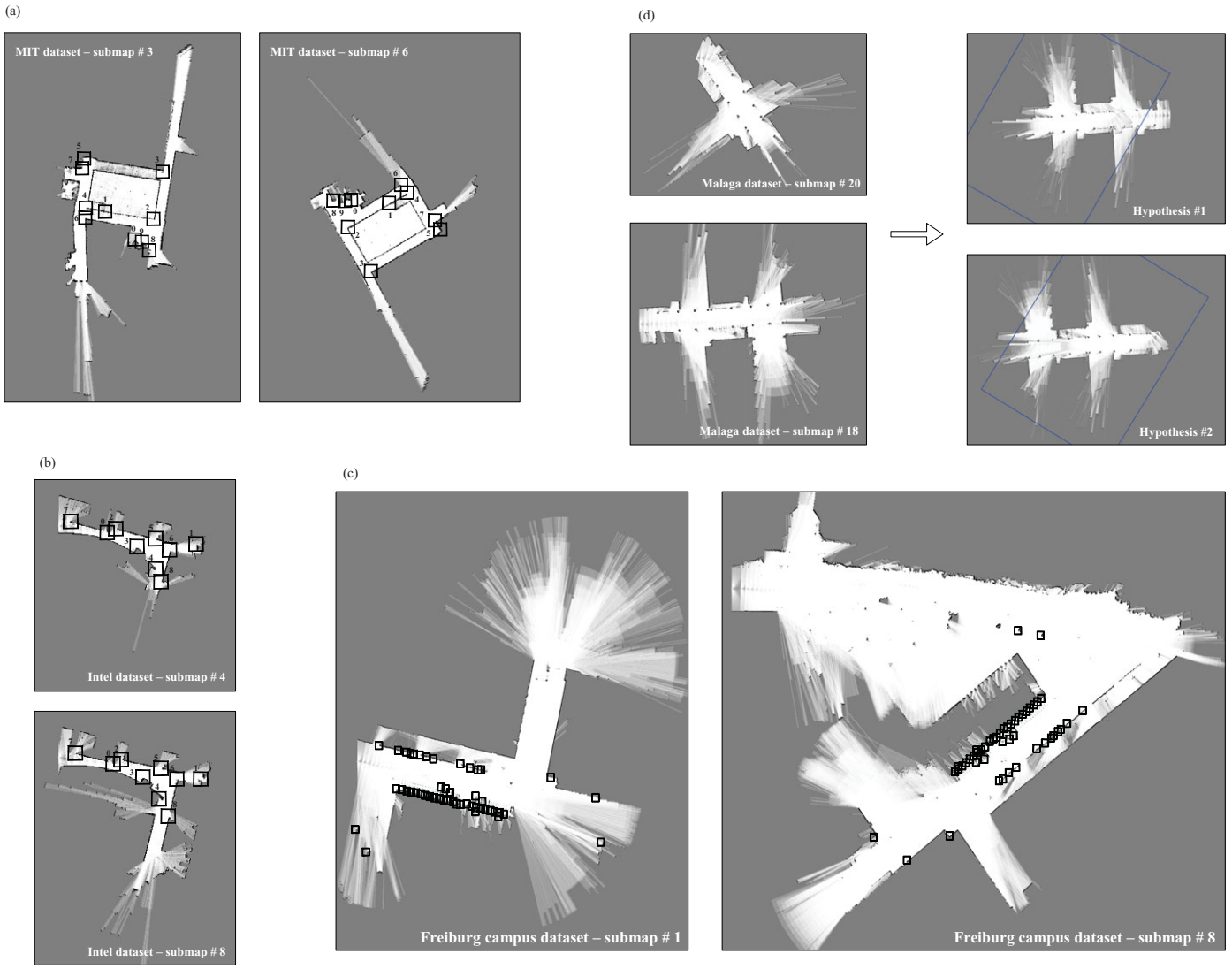


Fig. 10. (Colour online) (a–c) Some examples of map-to-map matchings as detected by the proposed method. (d) A pair of submaps for which a multi-modal transformation is detected. The two different hypotheses are represented by the overlay of the submap #20 over submap #18 in the right-hand images.

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

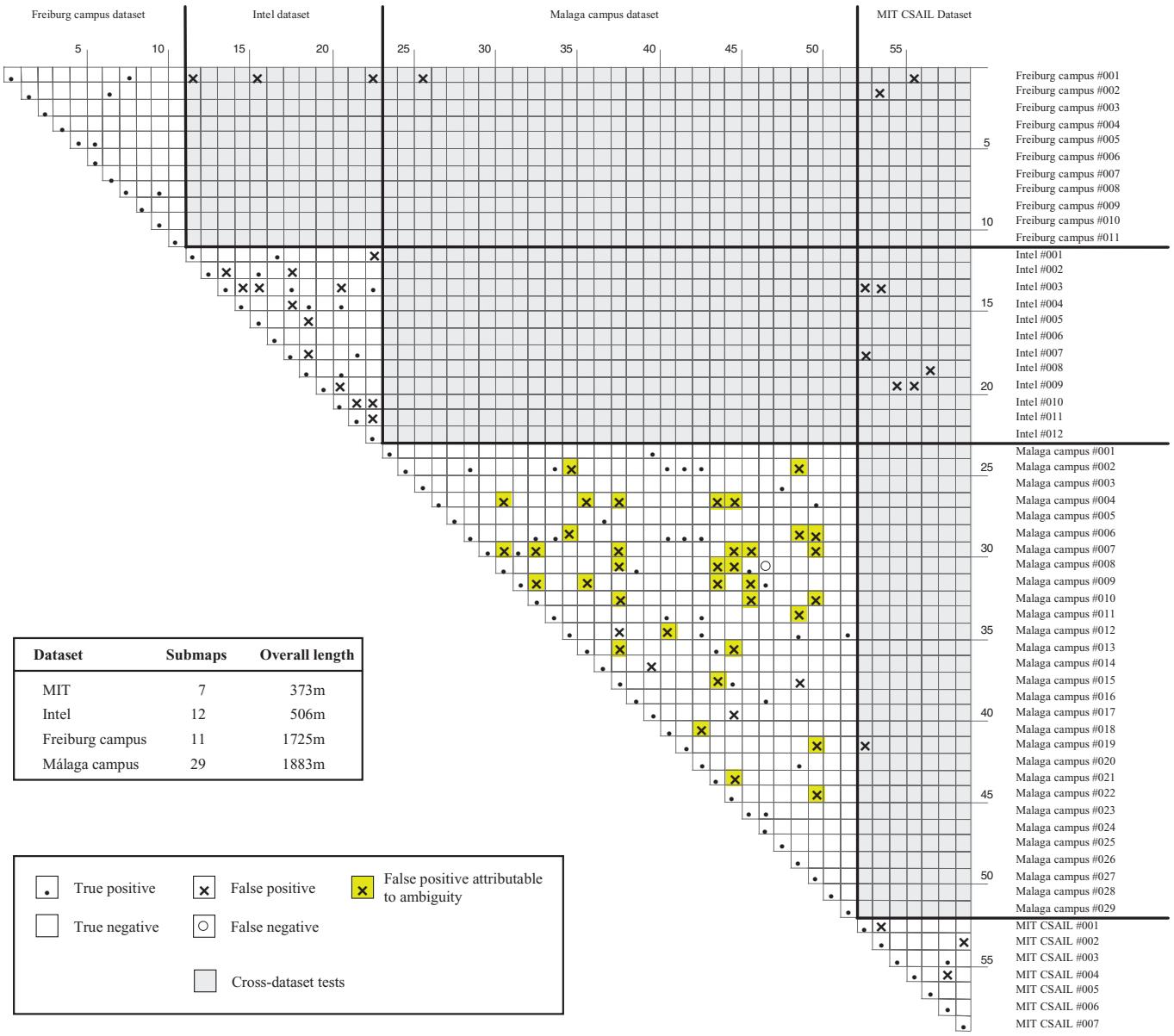


Fig. 11. (Colour online) Results of the loop-closure benchmark. The submaps corresponding to each of the two datasets have been separated by thick lines and inter-dataset blocks have been shaded for clarity.

loop-closure detection in this framework to obtain the raw sequence of submaps in each dataset. Among them, some areas will appear several times corresponding to loop closures.

The so obtained set of 59 submaps is an ideal testbed for the method proposed in this paper, since we can now try to match each submap with the rest, including those in different datasets (from which no valid transformation should result). The detailed results of executing the 1711 map-to-map matchings are shown in Fig. 11, where each entry in the table specifies the outcome from our method and whether the two maps actually do correspond or not, that is, it shows the loop-closure ground truth (obtained by human inspection). Note that there are two possible kinds of errors in this experiment: false positives (our method detecting a loop closure that does not really exist) and false negatives (where a real loop closure is overlooked). In global SLAM, the former is far more important because a single false positive may completely ruin the map. However, note that in HMT–SLAM, candidate false positives may not be that critical as long as they can be discarded if the uncertainty of the metric information is not too high.[5]

As can be seen in the figure, our method correctly detects as non-matchings virtually all the cases where each submap belongs to a different dataset. Two datasets deserve additional attention. First, the Intel dataset leads to several false positives, which is explained by the symmetry of the environment, i.e. all its submaps are very similar. Second, the Málaga 2006 campus dataset also suffers from many false positives, most of them attributable to the environment consisting of an array of three exactly identical buildings. Such potential errors, as mentioned above, can be easily discarded in a posterior stage by checking the consistency of the loop-closure hypothesis and the metric information within a hierarchical map.

The overall performance is also summarized in Table I, where for the sake of a fair validation we do not count the

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

Table I. Results for the loop-closure detection benchmark.

	Result	Disregarding ambiguity
True positives	97.56% (40/41)	–
False positives	3.47% (58/1670)	1.38% (23/1670)
True negatives	96.53% (1612/1670)	98.62% (1647/1670)
False negatives	2.44% (1/41)	–

elements in the main diagonal of Fig. 11 (matching each submap to itself), which were correctly detected by our method. It is remarkable that only one loop closure out of 41 was not recognized (a $\sim 2.4\%$ fail rate). We also show in the table the ratio of false positives modified by disregarding the errors clearly attributable to a real repetitive environment, not to errors in our detection method.

Regarding the computation time of this benchmark, it took 1740 s to compute the 1711 matchings in a Pentium Core Duo @ 2.2 GHz (using a single execution thread), yielding an average 1.02 s per match. Note that this includes the detection and descriptor extraction phases, not only the descriptor matching.

9. Conclusions

In this paper, we have proposed a new approach to grid matching, based on existing computer vision techniques (detectors and descriptors) and providing the modifications required by the ambiguity typically found in our problem by means of a multi-hypothesis RANSAC stage. The resulting method has been demonstrated to assess a 97% success ratio in detecting loop closures while also being reliable against sensor noise and errors in the robot positioning. In contrast to previous works, our proposal does not rely on an accurate knowledge of the robot heading, thus making it suitable to a larger number of real-world SLAM problems. Also, by keeping the probabilistic nature of the problem throughout the whole process, potentially including multimodal distributions, our method has important and direct applications to hierarchical robot map building of large-scale environments.

Appendix : Pairings test of consistency

In the following, we derive the test for the hypothesis that a given pair of features in maps a and b do actually match. Our statistical test relies solely on the rigid-body constraint that dictates that both inter-feature distances d_a^2 and d_b^2, measured in each map, must be equal. Note the usage of squared distances due to convenience during the derivation. A schematic illustration of these distances can be observed with an example in Fig. 7(a).

Following the assumptions presented in Section 7, the uncertainty in the feature points is modeled by a 2D isotropic Gaussian with a standard deviation of σ. Then, each of the squared distances d_i^2 is

$$d_i^2 = |p_{i,1} - p_{i,2}|^2 = (x_{i,1} - x_{i,2})^2 + (y_{i,1} - y_{i,2})^2, \quad (17)$$

and, by means of linear uncertainty propagation, we can model each d_i^2 as a Gaussian with mean $\bar{d}_i^2$ and variance $\sigma_{d_i^2}^2 = \mathbf{J}\boldsymbol{\Sigma}\mathbf{J}^\top$, where $\mathbf{J}$ is the Jacobian of Eq. (17) and $\boldsymbol{\Sigma}$ is the covariance of the feature point coordinates. It is clear that, assuming independence for the coordinates, this covariance amounts to $\sigma^2\mathbf{I}_4$; thus, by replacing the values of the Jacobians, we obtain

$$\sigma_{d_i^2}^2 = \sigma^2\mathbf{J}\mathbf{J}^\top = 8\sigma^2\bar{d}_i^2. \quad (18)$$

Having the distribution of each variable d_i^2, we can define the auxiliary variable z as the difference between the two squared distances, that is, $z = d_a^2 - d_b^2$. Under the hypothesis of the pairing to be valid, both distances d_a and d_b should be equal, thus z should be null. This allows us to test the hypothesis with a confidence c by means of the following χ^2 test:

$$\chi^2 = \frac{\left(\bar{d}_a^2 - \bar{d}_b^2\right)^2}{8\sigma^2\left(\bar{d}_a^2 + \bar{d}_b^2\right)} < \chi_{1,c}^2, \quad (19)$$

where $\chi_{n,c}^2$ is the inverse of the χ^2 cumulative distribution function with n degrees of freedom. In the denominator, we also use the fact that the variance of z is the sum of the variances of the individual squared distances.

References

1. H. Bay, T. Tuytelaars and L. Van Gool, "Surf: Speeded up robust features," *Lecture Notes Comput. Sci.* **3951**, 404 (2006).
2. P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(2), 239–256 (1992).
3. A. Birk and S. Carpin, "Merging occupancy grid maps from multiple robots," *IEEE Proc.* **94**(7), 1384 (2006).
4. J.-L. Blanco, "A tutorial on se(3) transformation parameterizations and on-manifold optimization," Technical report, University of Malaga (Sep. 2010).
5. J.-L. Blanco, J.-A. Fernández-Madrigal and J. Gonzalez, "Towards a unified Bayesian approach to hybrid metric-topological SLAM," *IEEE Trans. Robot.* **24**(2), 259–270 (2008).
6. J.-L. Blanco, J. Gonzalez and J.-A. Fernández-Madrigal, "Subjective local maps for hybrid metric-topological SLAM," *Robot. Auton. Syst.* **57**(1), 64–74 (2009).
7. J.-L. Blanco, J. González-Jiménez and J.-A. Fernández-Madrigal, "An alternative to the Mahalanobis distance for determining optimal correspondences in data association," *IEEE Trans. Robot.* **28**(4) (2012).
8. M. Bosse, P. Newman, J. Leonard, M. Soika, W. Feiten and S. Teller, "An Atlas Framework for Scalable Mapping," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2 (2003) pp. 1899–1906.
9. A. J. Davison, I. Reid, N. Molton and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(6), 1052–1067 (2007).
10. T. Duckett and U. Nehmzow, "Mobile robot self-localisation using occupancy histograms and a mixture of Gaussian location hypotheses," *Robot. Auton. Syst.* **34**(2–3), 119–130 (2001).
11. A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer* **22**(6), 46–57 (1989).
12. C. Estrada, J. Neira and J. D. Tardos, "Hierarchical SLAM: Real-time accurate mapping of large environments," *IEEE Trans. Robot.* **21**(4), 588–596 (2005).
13. 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* **24**(6), 381–395 (1981).

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press

14. A. Gil, O. M. Mozos, M. Ballesta and O. Reinoso, "A comparative evaluation of interest point detectors and local descriptors for visual slam," *Mach. Vision Appl.* **21**(6), 905–920 (2010).
15. G. Grisetti, G. D. Tipaldi, C. Stachniss, W. Burgard and D. Nardi, "Fast and accurate SLAM with Rao-Blackwellized particle filters," *Robot. Auton. Syst.* **55**(1), 30–38 (2007).
16. J. S. Gutmann and K. Konolige, "Incremental mapping of large cyclic environments," **In**: *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation* (1999) pp. 318–325.
17. C. Harris and M. Stephens, "A combined corner and edge detector," **In**: *Proceedings of Alvey Vision Conference*, vol. 15 (1988) pp. 147–151.
18. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision* (Cambridge University Press, Cambridge, 2003).
19. R. Hess, "An open-source SIFTLibrary," **In**: *Proceedings of the international conference on Multimedia*, (2010) pp. 1493–1496.
20. B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. Opt. Soc. Am. A*, **4**(4), 629–642 (1987).
21. A. Howard and N. Roy, The robotics data set repository (radish) (2003). available at: http://radish.sourceforge.net/
22. S. Lazebnik, C. Schmid and J. Ponce, "A sparse texture representation using affine-invariant regions," **In**: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2 (2003) pp. 319–324.
23. D. G. Lowe, "Object recognition from local scale-invariant features," **In**: *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2 (1999) pp. 1150–1157.
24. B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proc. DARPA Image Understanding Workshop*. **121**, 130 (1981).
25. J. L. Martínez, J. González, J. Morales, A. Mandow and A. García-Cerezo, "Mobile robot motion estimation by 2D scan matching with genetic and iterative closest point algorithms," *J. Field Robot.* **23** (Jan. 2006) pp. 21–34.
26. K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," **In**: *Proceedings of European Conference on Computer Vision*, vol. 1 (2002) pp. 128–142.
27. K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(10), 1615–1630 (2005).
28. J. Neira and J. D. Tardós, "Data association in stochastic mapping using the joint compatibility test," *IEEE Trans. Robot. Autom.* **17**(6), 890–897 (2001).
29. J. I. Nieto, J. E. Guivant and E. M. Nebot, "The hybrid metric maps (HYMMS): A novel map representation for DenseSLAM," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation* (2004) pp. 391–396.
30. A. R. Runnalls, "Kullback–Leibler approach to Gaussian mixture reduction," *IEEE Trans. Aerosp. Electron. Syst.* **43**(3), 989–999 (2007).
31. P. Saeedi, P. D. Lawrence and D. G. Lowe, "Vision-based 3-D trajectory tracking for unknown environments," *IEEE Trans. Robot.* **22**(1), 119–136 (2006).
32. S. Se, D. Lowe and J. Little, "Local and global localization for mobile robots using visual landmarks," **In**: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. **1** (2001) pp. 414–420.
33. C. Shekhar, V. Govindu and R. Chellappa, "Multisensor image registration by feature consensus," *Pattern Recogn.* **32**(1), 39–52 (1999).
34. J. Shi and C. Tomasi, "Good features to track," **In**: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (1994) pp. 593–600.
35. H. Tamimi, H. Andreasson, A. Treptow, T. Duckett and A. Zell, "Localization of mobile robots with omnidirectional vision using particle filter and iterative SIFT," *Robot. Auton. Syst.* **54**, 758–765 (2006).
36. S. Thrun, "Learning occupancy grid maps with forward sensor models," *Auton. Robot.* **15**(2), 111–127 (2003).
37. S. Thrun, W. Burgard and D. Fox, *Probabilistic Robotics* (MIT Press, Cambridge, MA (USA), 2005).
38. B. Zitova and J. Flusser, "Image registration methods: A survey," *Image Vis. Comput.* **21**(11), 977–1000 (2003).

https://doi.org/10.1017/S0263574712000732 Published online by Cambridge University Press