

¹ Distributed multi-camera visual mapping using
² topological maps of planar regions

³ E. Montijano^{a,*}, C. Sagües^a

⁴ ^a*DIIS - Instituto de Investigación en Ingeniería de Aragón (I3A), University of Zaragoza,
5 Spain.*

⁶ **Abstract**

This paper presents a solution for cooperative visual mapping using planar regions. Each agent is assumed to be equipped with a conventional camera and has limited communication capabilities. Our approach starts building topological maps from independent image sequences where natural landmarks extracted from conventional images are grouped to create a graph of planar regions. With this approach the features observed in several images belonging to the same plane are stored only once reducing the size of the individual maps. In a distributed scenario this is very important because smaller maps can be transmitted faster, which makes our approach better suited for cooperative mapping. The later fusion of the individual maps is obtained via distributed consensus without any initial information about the relations between the different maps. Experiments with real images in complex scenarios show the good performance of our proposal.

⁷ *Keywords:* Computer Vision, Distributed Systems, Mapping

⁸ **1. Introduction**

⁹ Advances in communication technologies and vision systems have made fea-
¹⁰ sible the idea of sets of cameras performing different perception tasks such as
¹¹ surveillance, tracking or mapping in a cooperative way. In this paper we focus
¹² on the problem of map building using several cameras which communicate their
¹³ perceived information to others. A set of cameras is able to map the environ-

*Corresponding author

Email addresses: emonti@unizar.es (E. Montijano), csagues@unizar.es (C. Sagües)
Preprint submitted to Elsevier April 12, 2010

14 ment faster than a single one would do. However, this adds new challenges that
15 must be solved.

16 The problem of mapping the environment considering a single camera has
17 been deeply studied, specially in the robotics community. The importance of
18 creating and maintaining a map for localization and navigation tasks is obvious
19 and a lot of effort has been made in this research line. A common approach is to
20 simultaneously localize the camera and map the environment (SLAM) [10], [12].
21 View-based maps [19] introduce geometric constraints to the process, making
22 it more robust. Computing the structure of the scene, usually defined with the
23 positions of the features and the cameras, makes the errors and drift grow with
24 the size of the map. Our approach overcomes this limitation because no explicit
25 metric information for the global map is computed. The errors in different
26 planes are uncorrelated because the extraction of each one is independent of the
27 others.

28 Topological maps [20] where no metric information is computed are another
29 widely used approach to organize the visual information. Topological visual
30 maps can be built from conventional [2], [15] or omnidirectional [25], [36] images.
31 The whole image can be stored but usually only the extracted features are saved.
32 Most recent and successful works use visual words [26] to represent the scene
33 in a more compact way. Although topological maps based on images give good
34 results, the space required to manage these maps is considerably big. If we
35 store all the features, many of them will be seen in several images, so that
36 the map will take up a lot of repeated data for the same features. By using
37 planes, features are stored only once independently of the number of images in
38 which they are observed. Moreover, the complexity of the maps is also reduced.
39 Graphs made from raw images are usually dense because of the number of
40 connections among close images whereas with the proposed maps the number
41 of connections between planes is considerably smaller. Semantic meaning [6] in
42 the maps is essential to human-robot interaction. Planes also provide a good
43 semantic information meaningful for humans, making this choice more adequate
44 than just features or raw images.

45 In addition to the previous advantages, it is well known that the estimation
46 processes are improved in terms of accuracy and stability when considering the
47 scene represented by planar regions [32]. Moreover, there are several works in
48 the literature that assume the presence of planes in the environment to solve
49 different tasks such as visual servoing [9], [13], [21], visual navigation with
50 maps [27], [11], structure reconstruction [29] or camera calibration [23]. In
51 all these approaches the 3D structure of the scene is not required and only
52 sets of coplanar features in the image domain are used. Since the proposed
53 solution based on planes follows these guidelines it can also be useful in the
54 above situations.

55 Plane detection in images is a common problem in computer vision [16], [35],
56 [30]. In order to detect the planes in the images we use a triple set, plane-
57 image-image, for feature matching and homography computation; the planes
58 are tracked along the sequence and new features are added to the planes once
59 they are detected [24]. Homology constraints are used to detect new planes and
60 also give a geometric criterion to relate the planes. Our approach is different
61 to existing techniques to detect planes in images because previous works are
62 usually more concerned about detecting planar regions in pairs of images instead
63 of along sequences, as we do.

64 The extension of the problem to a situation in which several cameras coop-
65 eratively map the environment presents new challenges. Problems such as data
66 association of multiple elements [34] [14], different intrinsic parameters of the
67 cameras [1] or the fusion of the different maps [3] appear in this new scenario
68 and new solutions should be given. Surveillance systems deal with the problem
69 of sensor fusion using static cameras [8]. In [17], [31] the global map is man-
70 aged by a centralized entity and the observations of the different cameras are
71 introduced as they are captured. Distributed map merging is considered in [3]
72 but data association is supposed to be given among the different maps. In the
73 previous approaches the initial relative positions and the intrinsic parameters of
74 the cameras are known. Our approach is independent of the relative positions of
75 the cameras or their intrinsic parameters. We consider a set of individual maps

76 created by different cameras with no a priori information about the relation be-
77 tween them. A distributed fusion based on consensus algorithms [7] is proposed.
78 Distributed data association is done in the image domain considering invariant
79 descriptors (e.g., SURF [5]) and homography constraints [4]. Our system is able
80 to transform the initial local maps to similar maps which satisfy the conditions
81 required to execute a decentralized fusion, achieving consensus.

82 As contributions of this paper, we can emphasize our proposal for detection
83 of planar regions in image sequences through a set of homographies using a triple
84 set, plane-image-image, for feature matching. Our approach for building visual
85 topological maps based on planar regions do not require metric information.
86 Besides, a solution for a multi-camera scenario is proposed, giving a technique
87 to fuse several maps based on distributed consensus without knowledge about
88 the cameras, the positions or the data association.

89 The remainder of the paper is arranged as follows. The construction of a
90 topological map using only one camera is explained in section 2. The extension
91 of the problem to several cameras and map fusion using distributed consensus
92 techniques is presented in section 3. In section 4 experimental results with real
93 images illustrate the performance of the proposal. Finally, in section 5 some
94 conclusions are presented.

95 **2. Single robot topological map**

96 Let $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, \dots\}$ be the set of images acquired by a 6DOF camera on-
97 board while moving. Let us assume that several rigid planar regions appear in
98 the scene. There is no knowledge about either the internal parameters of the
99 camera, represented by the calibration matrix \mathbf{K} , nor about the motion between
100 consecutive frames, \mathbf{R} and \mathbf{t} . For an easy understanding of the section, sub-
101 script indices will correspond to images in \mathcal{I} whereas superscript indices will
102 correspond to the planar regions, for example π_k^m will represent the features of
103 the m^{th} planar region seen in the k^{th} image.

104 The topological map is managed using a graph, $\mathcal{G} = (\mathcal{P}, \Pi, \mathbf{A})$, represented

105 by a finite non empty set of planes \mathcal{P} with cardinality $|\mathcal{P}| = P$, a vector $\mathbf{\Pi}$
106 containing the features observed in the planar regions and a matrix of relations
107 between the planes $\mathbf{A} \in \{0, 1\}^{P \times P}$. If $\mathbf{A}(m, n) = 1$ then there exists a relation
108 between the planes m and n , whereas for planes with no relation at all $\mathbf{A}(m, n) =$
109 0.

110 If one plane, m , is visible in two consecutive images of the sequence, \mathcal{I}_k and
111 \mathcal{I}_{k+1} , it is possible to compute a projective mapping (inter-image homography),
112 $\mathbf{H}_{k,k+1}^m$, that relates the features belonging to the plane, $\boldsymbol{\pi}_k^m = \mathbf{H}_{k,k+1}^m \boldsymbol{\pi}_{k+1}^m$.
113 This homography is defined up to a scale factor and has the form $\mathbf{H}_{k,k+1}^m =$
114 $\mathbf{K}(\mathbf{R}_{k,k+1} - (1/d_{k+1}^m)\mathbf{t}_{k,k+1}(\mathbf{n}_{k+1}^m)^T)\mathbf{K}^{-1}$, with d_{k+1}^m and \mathbf{n}_{k+1}^m the distance and
115 normal of the m^{th} plane in the $(k+1)^{th}$ frame, respectively. The homography
116 can be estimated from four correspondences without prior knowledge about the
117 scene or the calibration [18].

118 The planes are extracted from the sequence following the approach in [24].
119 The two initial frames of the sequence, \mathcal{I}_1 and \mathcal{I}_2 , are picked up and all the
120 planes seen in both images (Fig. 1-a) are extracted using DLT+RANSAC [18].
121 In order to perform the matching between features we have chosen SURF [5].
122 The process is summarized as follows:

- 123 1. For any plane, m , visible in the first two images:
 - 124 (a) all the features from the plane, $\boldsymbol{\pi}^m$, are stored expressed in the coor-
125 dinates of the first image where they were detected. This first image
126 is marked as the reference image of the plane, \mathcal{I}_{r^m} . The identifiers
127 of the features in every image are also stored to make automatic the
128 future search of these features.
 - 129 (b) The next image in the list is picked up, \mathcal{I}_3 , and the correspondences
130 with \mathcal{I}_2 are found. From the whole set, only those matches that
131 already belong to $\boldsymbol{\pi}^m$ are chosen, searching a new homography among
132 this subset. By looking for the homography only among this subset
133 fewer hypotheses are required in RANSAC because the probability
134 of a sample to be an inlier is bigger.

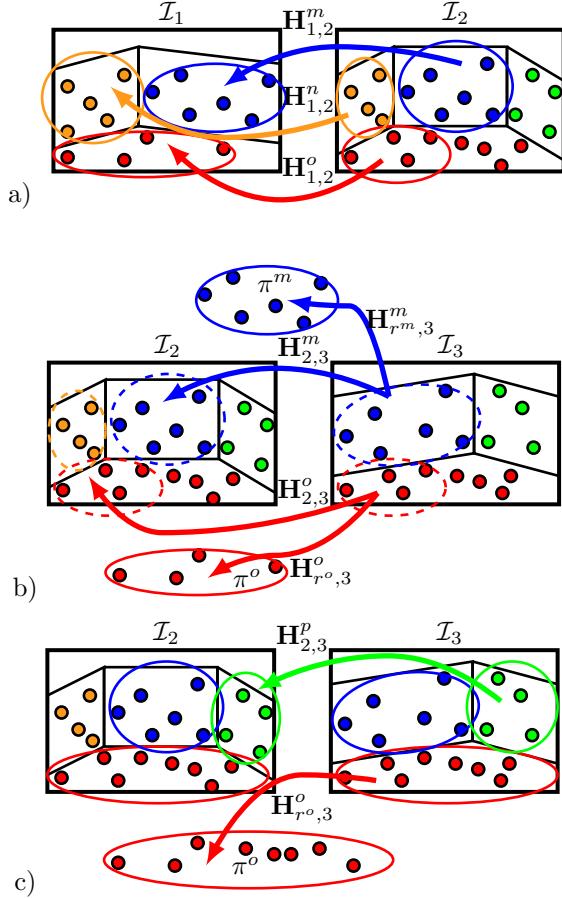


Figure 1: Scheme of the plane segmentation. a) Extraction of the initial planes, m, n, o . b) Triple match Plane-Image-Image for homography computation with the previous and the reference image. c) Addition of new points to the existing planes and detection of new planar regions within the remaining matches.

135 (c) The homography with respect to the points in the reference image,
 136 $\mathbf{H}_{r^m,3}^m$, is also computed so that the voting procedure is more robust,
 137 enforcing every feature to support both homographies instead of just
 138 one (Planes m and o in Fig. 1-b). With the two homographies new
 139 features are added to the existing plane (Plane o in Fig 1-c).

140 2. Once all the matches belonging to existing planes have been processed,

141 new planes between the remaining matches are extracted (Plane p in Fig
142 1-c).

143 3. The next image is selected and the method is recursively repeated until
144 all the images are processed.

145 Every time a new plane is detected it is added to the topological map. One
146 plane, m , is formally added to the graph by

$$\left\{ \begin{array}{l} \mathbf{A} = [\mathbf{I}_P \mid \mathbf{0}_P]^T \quad \mathbf{A} \quad [\mathbf{I}_P \mid \mathbf{0}_P], \\ \boldsymbol{\Pi} = (\boldsymbol{\Pi}^T, \boldsymbol{\pi}^m)^T, \\ \mathcal{P} = \mathcal{P} \cup \{m\}, \end{array} \right. \quad (1)$$

147 with \mathbf{I}_P the identity matrix of $P \times P$ dimensions and $\mathbf{0}_P$ a null vector of dimension
148 P .

149 The final information used to represent one plane is the set of features that
150 belong to the plane with their SURF descriptors. The coordinates of each
151 feature are expressed in the reference image of the plane.

152 2.1. Links between planes

153 Two planes m and n are defined as co-visible if there are at least two consecutive
154 images in which both planes are detected. This idea of co-visibility has a great
155 interest for navigation and localization tasks in future uses of the map. If
156 a camera localizes one plane of the set it will know which other planes it might
157 see when it moves, and so the space searched during the task execution will be
158 reduced.

159 When two planar regions are visible together in two consecutive images
160 it is possible to extract multi-plane constraints between the planes. An homology
161 matrix, also called “relative homography” captures the relative motion
162 between the images through two planes visible in the two images. Let
163 us suppose that m and n are both visible in \mathcal{I}_k and \mathcal{I}_{k+1} . The homology is obtained
164 by multiplying one of the homographies by the inverse of the other
165 one, $\mathcal{H}_{k,k+1}^{mn} = (\mathbf{H}_{k,k+1}^m)^{-1} \mathbf{H}_{k,k+1}^n$. The homology is the chosen criterion to create
166 a link between planes in the topological map. When two planes have been

¹⁶⁷ detected together in two consecutive images the algorithm sets the connecting
¹⁶⁸ edges to 1:

$$\mathbf{A}(m, n) = \mathbf{A}(n, m) = 1 \Leftrightarrow \exists \mathcal{I}_k, \mathcal{I}_{k+1} \in \mathcal{I} \mid \exists \mathcal{H}_{k, k+1}^{mn}. \quad (2)$$

¹⁶⁹ Let us note that although the link is created considering a geometric criterion
¹⁷⁰ between the planes, the map does not include any metric information.

¹⁷¹ The homology has also some properties that can be useful for robust detec-
¹⁷² tion of new planes in the sequence. Using the Sherman-Morrison formula [33],
¹⁷³ as in [35], the homology matrix can be decomposed in $\mathcal{H}_{k, k+1}^{mn} = \mathbf{I} + \mathbf{v}\mathbf{p}^T$, where

$$\mathbf{v} = (v_1, v_2, v_3)^T = \mathbf{K} \frac{\mathbf{R}_{k, k+1}^{-1} \mathbf{t}_{k, k+1}}{1 + \frac{(\mathbf{n}_{k+1}^m)^T}{d_{k+1}^m} \mathbf{R}_{k, k+1}^{-1} \mathbf{t}_{k, k+1}} \quad (3)$$

¹⁷⁴ is a view dependent vector and

$$\mathbf{p} = (p_1, p_2, p_3)^T = \left(\frac{(\mathbf{n}_{k+1}^m)^T}{d_{k+1}^m} - \frac{(\mathbf{n}_{k+1}^n)^T}{d_{k+1}^n} \right) \mathbf{K}^{-1} \quad (4)$$

¹⁷⁵ is a plane dependent vector. The homology is used to separate real planes from
¹⁷⁶ false and repeated ones. This is done using its eigenvalues, $\{\lambda_1, \lambda_2, \lambda_3\}$, which
¹⁷⁷ for a correct homology must have the form $(1, 1, 1 + v_1 p_1 + v_2 p_2 + v_3 p_3)$. Before
¹⁷⁸ adding a new plane to the map the eigenvalues of the homology must hold

$$|\lambda_1 - 1| \leq \epsilon, |\lambda_2 - 1| \leq \epsilon, |\lambda_3 - 1| \geq \epsilon, \quad (5)$$

¹⁷⁹ for a sufficient small ϵ . If the three eigenvalues are close to the unity it means
¹⁸⁰ that the two planes are actually the same one (the homology is an identity
¹⁸¹ matrix), so instead of creating a new plane, the new features are added to the
¹⁸² existing one. On the other hand, if two of the three eigenvalues are not close
¹⁸³ enough to the unity there is an homography that is not describing a real plane.
¹⁸⁴ In this second case the new plane is ignored. Let us notice that the test is pure
¹⁸⁵ image-based and the method still does not need any information about neither
¹⁸⁶ the camera calibration nor the motion between the images.

¹⁸⁷ 2.2. Loop closing

¹⁸⁸ The last problem considered to build the individual maps is to detect when
¹⁸⁹ a plane appears in the sequence because the camera is revisiting the same place

190 (loop closing). To consider this situation every time a new plane is detected, the
 191 algorithm matches the features of the new plane with the rest of existing planes
 192 and tries to compute a robust homography between them. If for some plane the
 193 corresponding homography exists and it is supported by most of the matches it
 194 means that both planes are the same and must be merged. The merging process
 195 is performed by adding to the existing plane the new features and by updating
 196 \mathbf{A} through eq. (2).

In the end, all the method can be summarized in the Algorithm 1

Algorithm 1 Single camera topological map

```

1: Extract planes from  $\mathcal{I}_1$  and  $\mathcal{I}_2$ 
2: Create  $\mathcal{G}$  with the initial planes
3:  $\mathbf{A}(m, n) = \mathbf{A}(n, m) = 1 \forall m \neq n$ 
4: for all  $\mathcal{I}_k \in \mathcal{I}$  do
5:   Match features in  $\mathcal{I}_k$  and  $\mathcal{I}_{k+1}$ 
6:   for all visible  $m \in \mathcal{P}$  do
7:     Select the matches that belong to  $\pi^m$ 
8:     Compute  $\mathbf{H}_{k, k+1}^m$  and  $\mathbf{H}_{r^m, k+1}^m$  with DLT+Ransac
9:     Add new features to  $\pi^m$  using  $\mathbf{H}_{r^m, k+1}^m$ 
10:  end for
11:  Search for new planes in the remaining matches
12:  if new plane was already in the map then
13:    Update  $\mathcal{G}$ 
14:  else
15:    Add the new plane to  $\mathcal{G}$  (eq. (1))
16:  end if
17:  Modify  $\mathbf{A}$  with the new homologies (eq. (2)-(5))
18: end for

```

197

198 **3. Multi-Camera Distributed Topological map**

199 Let us consider now a set, \mathcal{C} , of C mobile agents with cameras and limited
200 communication capabilities. Each agent manages an individual topological map
201 $\mathcal{G}_i = \{\mathcal{P}_i, \boldsymbol{\Pi}_i, \mathbf{A}_i\}, i = 1, \dots, C$. The communications among the agents are
202 defined by a graph $\mathcal{G}_{comm} = \{\mathcal{C}, \mathcal{E}\}$, where $(i, j) \in \mathcal{E}$ if agent i can communicate
203 with agent j . The set of neighbors of the agent i is the set of agents that can
204 directly communicate with it, $\mathcal{N}_i = \{j \in \mathcal{C} | (i, j) \in \mathcal{E}\}$. The diameter of the
205 graph $Diam(\mathcal{G}_{comm})$ is the maximum length of a path between any two nodes
206 of the graph.

207 **Assumption 3.1.** *The communication graph, \mathcal{G}_{comm} , is connected, , i.e. there
208 exists a path that connects any two nodes of the network.*

209 Not every agent can directly communicate with each other and the goal is to
210 make all the agents compute an identical global map, $\mathcal{G}^* = (\mathcal{P}^*, \boldsymbol{\Pi}^*, \mathbf{A}^*)$, con-
211 taining the information acquired by the whole set of agents. The computation
212 of the global map can be divided in two parts. On one hand the information
213 about the graph of planes and their relations (\mathcal{P}^* and $\boldsymbol{\Pi}^*$), and on the other
214 hand the reference image of each planar region, how many features it contains
215 and their coordinates (\mathbf{A}^*).

216 A distributed consensus approach is followed to compute the global map.
217 Distributed consensus techniques [7] study how to reach an average of the total
218 information considering only local interactions. A leader election approach is
219 followed to obtain the consensus on the global graph and the SURF descriptors,
220 whereas for the feature coordinates a distributed averaging rule is used. In order
221 to use these techniques several properties must be ensured:

- 222 1. Information of the agents: it is required that all the agents have an initial
223 value of the information.
- 224 2. Data association: in order to perform the fusion it is necessary to know
225 which planes in the local maps are associated. For each plane which
226 features are linked has also to be known.

227 3. Common reference: given two planes from two different maps which cor-
228 respond to the same plane in the world, the corresponding sets of features
229 must be expressed in the same reference.

230 However, local maps composed by planar regions and image features do not
231 satisfy the properties above mentioned. Solutions to overcome this problems
232 are proposed, obtaining a common global map equal for all the agents.

233 *3.1. Information of the agents*

234 It is supposed that none of the agents has the information of all the planes
235 seen by the whole team. Therefore, in the first step the local maps, \mathcal{G}_i , are
236 augmented so that the size and the order of each of them is the same, $P_i^* = P_j^*$
237 and $|\Pi_i^*| = |\Pi_j^*| \forall i, j \in \mathcal{C}$.

238 The ordering of the planes is done by univocal identification of both the
239 agents and the planes. The agents can be identified, for example, using the ip
240 addresses (ID) whereas the planes are ordered as they were detected in the local
241 sequences. These two elements define a global order of the whole set of maps.
242 The function $O : \mathcal{P} \rightarrow \mathbb{N}$ is defined in such a way that it returns the order of
243 a given plane in the map. For example, the first plane observed by the third
244 agent will have a smaller position in the map than the fourth plane observed by
245 the same agent but a larger value in the global order than any plane observed
246 by the second or the first agent.

247 The different size of the initial maps is solved creating fictitious planes in
248 the local maps so that all the agents have a final map of the same size. A
249 fictitious plane, \tilde{m} , is a plane with no relations in the graph and for which all
250 the coordinates of all its features, $\boldsymbol{\pi}^{\tilde{m}}$, are initialized to zero. Every agent creates
251 as many fictitious planes as the total number of planes observed by the other
252 agents. This is done exchanging local messages so that, every agent i eventually
253 has a feature vector

$$\Pi_i^* = (\mathbf{0}_1^T \dots \Pi_i^T \dots \mathbf{0}_C^T)^T \quad (6)$$

²⁵⁴ and an adjacency matrix \mathbf{A}_i^* with the form

$$\mathbf{A}_i^* = \begin{bmatrix} \mathbf{0}_{11} & \dots & \mathbf{0}_{1i} & \dots & \mathbf{0}_{1C} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{0}_{i1} & \dots & \mathbf{A}_i & \dots & \mathbf{0}_{iC} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{0}_{C1} & \dots & \mathbf{0}_{Ci} & \dots & \mathbf{0}_{CC} \end{bmatrix}, \quad (7)$$

²⁵⁵ where $\mathbf{0}_j$ is a vector of zeros with dimension $|\Pi_j|$ and $\mathbf{0}_{ij}$ is a matrix of zeros
²⁵⁶ with dimension $P_i \times P_j, j = 1, \dots, C, j \neq i$. Let us note that in order to create a
²⁵⁷ fictitious plane it is only necessary to know which agent has seen it, the order in
²⁵⁸ the local map and the number of features it contains, which reduces considerably
²⁵⁹ the size of the exchanged messages.

²⁶⁰ When new messages containing information from other agents are received,
²⁶¹ the fictitious planes are added to the local maps,

$$\mathcal{P}_i^* = \mathcal{P}_i^* \cup \tilde{m}. \quad (8)$$

²⁶² Regarding the adjacency matrices, \mathbf{A}_i^* , for any new fictitious plane a new row
²⁶³ and column with zeros is created

$$\mathbf{A}_i^* = \mathbf{P}_{O(\tilde{m})P_i} [\mathbf{I}_{P_i} \mid \mathbf{0}]^T \mathbf{A}_i^* [\mathbf{I}_{P_i} \mid \mathbf{0}] \mathbf{P}_{O(\tilde{m})P_i}, \quad (9)$$

²⁶⁴ where the middle matrices augment the adjacency matrix as in eq. (1) and
²⁶⁵ $\mathbf{P}_{O(\tilde{m})P_i}$ is a permutation matrix that moves the last row, P_i , to the row $O(\tilde{m})$
²⁶⁶ and displaces all the rows in between one position down. Finally, Π_i^* is also
²⁶⁷ updated by adding zeros in the corresponding position. After $\text{Diam}(\mathcal{G}_{comm})$
²⁶⁸ rounds of exchanging information, all the agents will know the size of the global
²⁶⁹ map and all the matrices \mathbf{A}_i^* and vectors Π_i^* will have the same dimension
²⁷⁰ and the form of eqs. (7) and (6) respectively. This process is summarized in
²⁷¹ Algorithm 2.

²⁷² 3.2. Data association

²⁷³ Two issues must be addressed: first which planes (respectively features) are
²⁷⁴ associated with others must be known. Once the data association is known, the

Algorithm 2 Augment local maps - Agent i

```
1: Send information about the local map to all  $j \in \mathcal{N}_i$ 
2: for  $it = 1 \dots \text{Diam}(\mathcal{G}_{comm})$  do
3:   Receive information from all  $j \in \mathcal{N}_i$ 
4:   Create fictitious planes (eq. (8))
5:   Augment  $\mathbf{A}_i^*$  (eq. (9))
6:   Send the new information to all  $j \in \mathcal{N}_i$ 
7: end for
```

²⁷⁵ second issue is to reduce the adjacency matrices (respectively feature vectors)
²⁷⁶ so that they have the correct number of elements and in the right order.

²⁷⁷ The data association is performed based only on local information. Each
²⁷⁸ agent exchanges its initial local map with its set of neighbors. For all the maps,
²⁷⁹ the data association is carried out computing homographies between the local
²⁸⁰ planes and the planes of the other maps. Two planar regions are associated
²⁸¹ if a robust homography exists between them. The homography is computed
²⁸² matching the invariant features and using DLT+RANSAC algorithm.

²⁸³ **Definition 3.1.** Given two maps \mathcal{G}_i^* and \mathcal{G}_j^* the function $M : \mathcal{G}_i^* \times \mathcal{G}_j^* \rightarrow$
²⁸⁴ $\{0, 1\}^{P_i \times P_j}$ returns an association matrix, \mathbf{W}_{ij} , where

$$\mathbf{W}_{ij}(m, n) = \begin{cases} 1 & \text{if } \boldsymbol{\pi}_i^m \text{ and } \boldsymbol{\pi}_j^n \text{ are related by a homography } \mathbf{H}_{ij}^{nn}, \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

²⁸⁵ **Assumption 3.2.** The matching function has the following properties

- ²⁸⁶ • Self association, $M(\mathcal{G}_i^*, \mathcal{G}_i^*) = \mathbf{I}_{P_i}$
- ²⁸⁷ • Planes are associated in a one to one way

$$\sum_{m=1}^{P_i} \mathbf{W}_{ij}(m, n) \leq 1 \text{ and } \sum_{n=1}^{P_j} \mathbf{W}_{ij}(m, n) \leq 1,$$

- ²⁸⁸ • Given two maps \mathcal{G}_i^* and \mathcal{G}_j^* it holds that $M(\mathcal{G}_i^*, \mathcal{G}_j^*) = \mathbf{W}_{ij} = \mathbf{W}_{ji}^T =$
²⁸⁹ $(M(\mathcal{G}_j^*, \mathcal{G}_i^*))^T$.

290 Let us note that these assumptions are satisfied by the robust homography
291 matching.

292 All the local associations can be put together in a general association matrix,
293 $\mathbf{W} \in \mathbb{N}^{P^* \times P^*}$,

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{11} & \dots & \mathbf{W}_{1C} \\ \vdots & \ddots & \vdots \\ \mathbf{W}_{C1} & \dots & \mathbf{W}_{CC} \end{bmatrix}, \quad (11)$$

294 where

$$\mathbf{W}_{ij} = \begin{cases} M(\mathcal{G}_i, \mathcal{G}_j) & \text{if } j \in \{\mathcal{N}_i \cup i\}, \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (12)$$

295 in which each agent has the information of one of the block-rows.

296 The rows give information to the agents about associations of planes with
297 direct neighbors' maps. The question now is how two agents that are not direct
298 neighbors but have seen the same plane are aware of this situation. The global
299 association matrix can also be seen as an adjacency matrix of the whole set of
300 planes, \mathcal{G}^* . Following this reasoning, two planes are associated if there exists a
301 path in \mathbf{W} that links them. As the following lemma states, given a graph \mathcal{G}^* ,
302 the powers of its adjacency matrix contain the information about the number
303 of paths existing between the nodes of \mathcal{G}^* :

304 **Lemma 3.1 (Lemma 1.32 [7]).** *Let \mathcal{G}^* be a graph of order P^* with un-weighted
305 adjacency matrix $\mathbf{W} \in \{0, 1\}^{P^* \times P^*}$, and possibly with self loops. For all $m, n \in$
306 $\{1, \dots, P^*\}$ and $t \in \mathbb{N}$ the (m, n) entry of the t^{th} power of \mathbf{W} , \mathbf{W}^t , equals the
307 number of paths of length t (including paths with self loops) from node m to node
308 n .*

309 In the problem presented here the nodes refer the different planes.

310 We have proposed a distributed algorithm to compute the powers of such
311 adjacency matrix [4]. The algorithm is proved to finish in finite time and finds
312 all the paths between elements in \mathcal{G}^* . After the execution of the algorithm each

313 agent knows all the associations of its planes with the planes of the rest of the
314 agents, even if they are not direct neighbors.

315 Given a plane, m , let $B(m) = \{n \mid \mathbf{W}^t(m, n) = 1\}$ be the set of planes
316 associated to m and $\bar{n}(m) = \arg \min_{n \in B(m)} O(n)$ be the plane with the lowest
317 value in the global order. For all the associations of the agent i , the adjacency
318 matrix \mathbf{A}_i^* is updated to put together all the associated planes

$$\mathbf{A}_i^* = \mathbf{I}_n(\mathbf{A}_i^* \vee \mathbf{P}_{\bar{n}(m), n} \mathbf{A}_i^* \vee \mathbf{A}_i^* \mathbf{P}_{\bar{n}(m), n}) \mathbf{I}_n^T, \forall m \in \mathcal{P}_i, n \in B(m), \quad (13)$$

319 where $\mathbf{P}_{\bar{n}(m), n}$ is a permutation matrix of the rows $\bar{n}(m)$ and n , and \mathbf{I}_n is an
320 identity matrix where the n^{th} row has been deleted. The symbol \vee represents
321 the *or* operation between the matrices, which can be done taking into account
322 that all the elements of the matrices are in the set $\{0, 1\}$. Let us note that row
323 $\bar{n}(m)$ will be the same for all the agents with planes in the set $B(m)$. This
324 means that all the agents will move the information of each association to the
325 same row and will delete the rest of the rows, maintaining the size and the order
326 of their maps.

327 The last problem is to combine the associations of planes in which there is no
328 plane belonging to \mathcal{P}_i . Since the agent i has access only to the i^{th} block of rows
329 within \mathbf{W}^t , all the associations in which there is no plane belonging to \mathcal{P}_i will be
330 unknown. To solve this problem a similar exchange of messages like the one to
331 augment the local maps is carried out. In this case the exchanged information
332 are the sets of associated planes $B(m)$. After $\text{Diam}(\mathcal{G}_{comm})$ iterations all the
333 associations are received by all the agents and using eq. (13) all the updates are
334 done. Algorithm 3 schematizes the data association step for planar regions.

335 Regarding the association of the features a similar process is performed for
336 all the features belonging to the same planar region. After such process all the
337 vectors $\mathbf{\Pi}^*$ are updated, having the same size and order.

338 *3.3. Common reference*

339 At this point the local maps of planes have been associated and all of them
340 have the same size and are equally sorted. Although the adjacency matrices are

Algorithm 3 Data association for planes - Agent i

- 1: Exchange local maps with neighbors
- 2: Compute $\mathbf{W}_{ij}, j \in \mathcal{N}_i$
- 3: **repeat** Update the \mathbf{W}_{ij} as in [4]
- 4: **until** \mathbf{W}_{ij} does not change $\forall j \in \mathcal{C}$
- 5: Update \mathbf{A}_i^* eq. (13)
- 6: **for** $it = 1 \dots \text{Diam}(\mathcal{G}_{comm})$ **do**
- 7: Send block associations to $j \in \mathcal{N}_i$
- 8: Receive block associations from $j \in \mathcal{N}_i$
- 9: Update \mathbf{A}_i^* eq. (13)
- 10: **end for**

341 ready to execute the consensus algorithm, the features still require a common
342 reference image. Otherwise posterior consensus will give erroneous results. In
343 order to solve this problem a leader election algorithm is used to decide, for
344 each plane, which one of the local reference images is defined as the common
345 reference image. A distributed approach of the leader election algorithm can be
346 found in [22]. Initially every agent starts considering itself as leader. At each
347 step the leader value, $\text{lead}_i(t)$, is updated considering the values of its neighbors.

$$\text{lead}_i(t+1) = \max(\text{lead}_i(t), \text{lead}_j(t) \ j \in \mathcal{N}_i). \quad (14)$$

348 After $\text{Diam}(\mathcal{G}_{comm})$ iterations the rule in (14) converges to the same leader for
349 all the nodes in the network.

350 In the case of planar regions, not only a leader (reference image) needs to
351 be chosen for every plane but also the homography that transforms the features
352 from the local references to the global one. The number of observed features in
353 the plane, $|\pi|$, is the criterion chosen to decide the reference. For every plane,
354 algorithm 4 is executed by every agent. In the algorithm, $\mathbf{H}_{j,i}$ represents the
355 homography that transforms the plane from the coordinates in the local ref-
356 erence to the coordinates of the neighbor's reference. This homography is the
357 one computed in the local data association step. The variable $feats$ represents

358 the number of real features that has the plane for which the image reference
 359 is currently the global reference. After $\text{Diam}(\mathcal{G}_{\text{comm}})$ iterations all the agents
 360 know the common reference (the one which contains more real features) and the
 361 homography to transform their coordinates. Applying this transformation and
 362 normalizing the coordinates, all the agents have their features in the same ref-
 363 erence frame. The fictitious planes are not affected by changes of the reference.
 364 Taking this into account the agents with a fictitious plane do not participate in
 the leader election algorithm of such plane.

Algorithm 4 Choice of a common reference - Agent i

```

1: lead =  $i$ ; feats =  $|\boldsymbol{\pi}_i|$ ;  $\mathbf{H}_{r,i} = \mathbf{I}$ 
2: for  $it = 1 \dots \text{Diam}(\mathcal{G}_{\text{comm}})$  do
3:   Send [lead,feats, $\mathbf{H}_{r,i}$ ] to all  $j \in \mathcal{N}_i$ 
4:   Receive [lead $_j$ ,feats $_j$ , $\mathbf{H}_{r,j}$ ] from all  $j \in \mathcal{N}_i$ 
5:   if feats $_j > \text{feats}$  then
6:     lead = lead $_j$ ; feats = feats $_j$ ;  $\mathbf{H}_{r,i} = \mathbf{H}_{r,j}\mathbf{H}_{j,i}$ 
7:   end if
8: end for
9: Transform the features' coordinates of  $\boldsymbol{\pi}$  using  $\mathbf{H}_{r,i}$ 

```

365

366 *3.4. Consensus on the global map*

367 From here on all the nodes have the information needed to perform the
 368 consensus. The graphs \mathcal{G}_i^* satisfy now all the requirements to apply distributed
 369 consensus algorithms.

370 The adjacency matrices of the local maps are updated with the following
 371 rule

$$\mathbf{A}_i^*(m, n) = \mathbf{A}_i^*(m, n) \vee \mathbf{A}_j^*(m, n), \forall j \in \mathcal{N}_i \quad (15)$$

372 **Theorem 3.1 (Convergence of \mathbf{A}_i^*).** *The set of adjacency matrices \mathbf{A}_i^* , un-
 373 der iteration rule (15), converges to a common matrix \mathbf{A}^* that includes all the
 374 links between planes in $\text{Diam}(\mathcal{G}_{\text{comm}})$ iterations.*

³⁷⁵ **Proof.** Let us consider separately each element of the matrices \mathbf{A}_i^* . Considering
³⁷⁶ that the initial value of the elements is $\{0, 1\}$, then it holds that

$$\mathbf{A}_i^*(m, n) \vee \mathbf{A}_j^*(m, n) = \max(\mathbf{A}_i^*(m, n), \mathbf{A}_j^*(m, n)). \quad (16)$$

³⁷⁷ Then eq. (15) can also be seen as an update of a leader election algorithm,
³⁷⁸ which is proved to converge in $\text{Diam}(\mathcal{G}_{comm})$ iterations [22].

³⁷⁹ Since $\mathbf{A}_i^*(m, n) = 1$ implies that there is a link between planes m and n and
³⁸⁰ $\max(0, 1) = 1$ then all the links of the adjacency matrices are preserved. ■

³⁸¹ With respect to the features, since no observations are assumed to be better
³⁸² than others, an average of the matched observations is computed. Let us
³⁸³ consider one feature, \mathbf{f} , belonging to a plane of the global map. Let \mathbf{f}_i denote
³⁸⁴ the value that agent i has of \mathbf{f} after the common reference transformation. The
³⁸⁵ common consensus update rule followed by every agent is

$$\mathbf{f}_i(t+1) = \mathbf{f}_i(t) + \sum_{j \in \mathcal{N}_i} w_{ij} (\mathbf{f}_j(t) - \mathbf{f}_i(t)), \quad (17)$$

³⁸⁶ where w_{ij} are the weights used to fuse the information. If assumption 3.1 holds
³⁸⁷ and the weights w_{ij} have the property of configuring a double stochastic matrix
³⁸⁸ then the iteration rule in (17) converges for all i to the average of the initial
³⁸⁹ values,

$$\forall i \in \mathcal{C} \quad \mathbf{f}_i(t) \rightarrow \mathbf{f}^* = \frac{1}{C} \sum_{i \in \mathcal{C}} \mathbf{f}_i(0) \text{ as } t \rightarrow \infty. \quad (18)$$

³⁹⁰ In order to obtain weights composing a doubly stochastic matrix metropolis for-
³⁹¹ mula is used. Metropolis weights' definition is based on the number of neighbors
³⁹² of each agent:

$$w_{ij} = \begin{cases} \min\left(\frac{1}{|\mathcal{N}_i|}, \frac{1}{|\mathcal{N}_j|}\right) & \text{if } (i, j) \in \mathcal{E} \\ 1 - \sum_{k=1 \dots C} w_{ik} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \quad (19)$$

³⁹³ Let us note that although the fictitious planes do not apport any information
³⁹⁴ to the final consensus, they affect it in the sense that the final value is divided
³⁹⁵ by the total number of agents. The third coordinate of the features plays here

396 a fundamental role. Let us recall that for the fictitious features the third co-
397 ordinate, usually related with the scale of the point, was also set to zero. Let
398 $\mathcal{C}_f \subseteq \mathcal{C}$ the set of agents that have observed f in their local maps.

399 **Theorem 3.2 (Average of features).** *If $\mathcal{C}_f \neq \emptyset$, then the iteration rule (17)
400 converges, in normalized coordinates, for all the agents in the network to*

$$\mathbf{f}_i(t) \rightarrow \frac{1}{|\mathcal{C}_f|} \sum_{j \in \mathcal{C}_f} \mathbf{f}_j(0) \quad \text{as } t \rightarrow \infty \quad \forall i \in \mathcal{C}. \quad (20)$$

401 **Proof.** By the properties of the consensus rule the features converge to

$$\mathbf{f}_i(t) \rightarrow \frac{1}{C} \sum_{j \in \mathcal{C}_f} \mathbf{f}_j(0). \quad (21)$$

402 The third coordinate of the features is either 1 if the agents belongs to C_f or 0
403 if it does not. Therefore, this coordinate converges to

$$[\mathbf{f}_i(t)]_3 \rightarrow \frac{1}{C} \sum_{j \in \mathcal{C}_f} 1 = \frac{|\mathcal{C}_f|}{C}. \quad (22)$$

404 Once the coordinates are normalized, the convergence to the desired value is
405 proven. ■

406 Let us note that the consensus is only carried out for the coordinates of the
407 features and not for the whole descriptor. For the SURF descriptors a leader
408 election algorithm is used so that every agent has the same set of SURF after
409 $\text{Diam}(\mathcal{G}_{comm})$ steps.

410 4. Experimental Results

411 Several experiments have been carried out in order to evaluate the properties
412 and the behavior of our proposal. We have tested it using different real image
413 data sets that correspond to different locations of man-made environments with
414 plenty of planar regions. The first data set has been recorded indoors (House
415 data set). It consists of 3600 frames from different rooms. The second data
416 set is composed by nine different sequences recorded outdoors in a downtown
417 Zaragoza area (Downtown data set). Figure 2 shows a view of the map where
418 these sequences have been acquired and the topology used in the experiments.

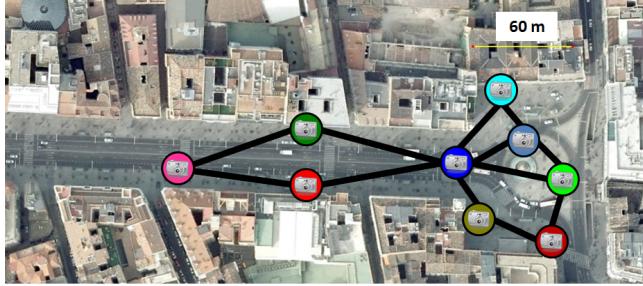


Figure 2: Map of downtown Zaragoza, where the nine sequences have been acquired and the communication graph among the agents. Each camera represents one agent with its local map and the black edges are the communication links in the network.

419 The camera used in all the cases has been a *Panasonic Lumix FX-500*. In
 420 all the cases the camera has moved with 6DOF. For all the images we have used
 421 SURF features [5] for matching. The computation of the homographies has
 422 been done using DLT+RANSAC algorithm. It is well known that under pure
 423 rotations or small motions all the features can be fitted to the same homography.
 424 In video sequences with high frame rates this is a common situation. To avoid
 425 this problem we have followed the idea of [28] to select key frames among the
 426 sequence:

- 427 • There are as many images as possible between the key frames \mathcal{I}_k and \mathcal{I}_{k+1} .
 428 • There are at least M matches between the key frames \mathcal{I}_k and \mathcal{I}_{k+1} .
 429 • There are at least N matches between the key frames \mathcal{I}_k and \mathcal{I}_{k+2} .

430 The results are divided in three sections. In the first experiment (sec. 4.1)
 431 we analyze how the triple matching step works and the properties of the ex-
 432 tracted planes. In the second experiment (sec. 4.2) we analyze the properties
 433 of the individual maps created using the sequences of images from each camera
 434 separately. We have compared the resulting graphs obtained for both data sets
 435 with graphs made by images [36]. For the latter approach we have stored the
 436 SURF descriptors of each image and we have imposed the homography con-
 437 straint between frames to observe the pros and cons of using images or planes.

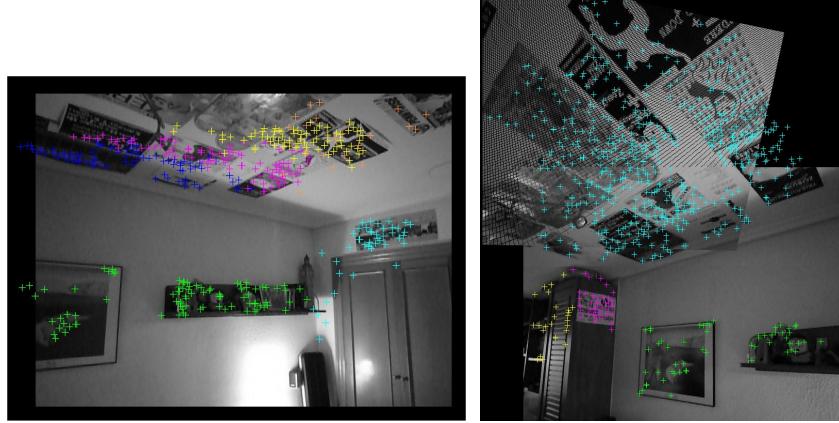


Figure 3: Planar regions extracted from the House sequence. Each region is represented with a different color. In the right figure we have added points corresponding to the ceiling transformed with the computed homography.

438 In the last experiment (sec. 4.3) we have tested the multi-camera distributed
439 approach for the Downtown data set.

440 *4.1. Extraction of the planar regions*

441 The House data set has helped us to test the detection of planes since it has
442 a lot of different planar regions. In Fig. 3 some of the segmented regions are
443 depicted. We observe that although the method detects several planes which
444 are the same in the ceiling, there are no wrong planes segmented.

445 The nine sequences in the Downtown data set are more challenging because
446 in this case the scene contains dynamic elements such as people and cars. There
447 are also a lot of trees that occlude the building facades and, in some cases, there
448 are also illumination changes when the camera points towards the sun. Figure 4
449 and 5 depict two examples of two different planes extracted using our method.
450 Even when the extracted planes contain some outlier features the results are still
451 quite good and, what is more important, we observe that the method follows
452 the planar regions correctly adding new features as they appear.

453 We have observed that sometimes in a real scenario many dynamic objects
454 may generate planar regions. These planar regions are undesirable in a practical



Figure 4: Planar region extracted from one sequence in downtown Zaragoza. The algorithm is able to track and grow the plane over 63 different frames with dynamic elements and some illumination changes. The top figure represents the plane with the 482 detected features. The rest of the images are the frames where the plane was observed.

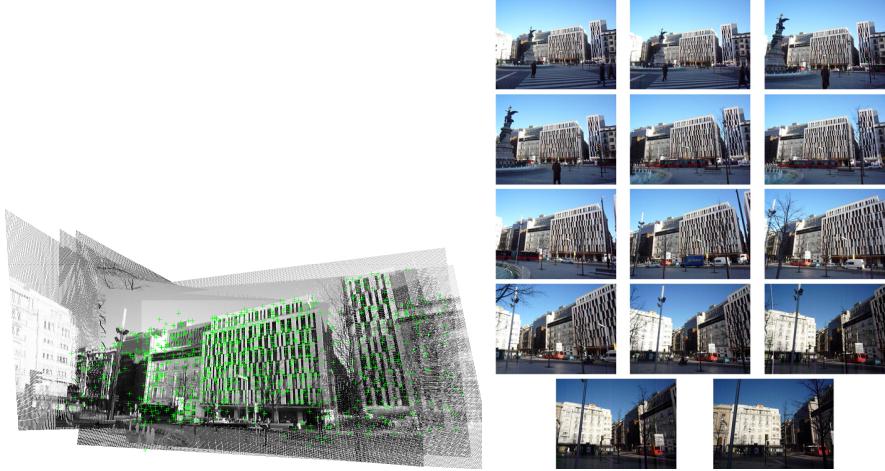


Figure 5: Another planar region extracted from one sequence in downtown Zaragoza.

455 situation. A lower bound on the number of features of a plane after the map
 456 generation clears almost all the undesired planes. Other times the algorithm
 457 considers as a planar region a set of features belonging to different planes but
 458 coplanar between them. Usually in the next steps the algorithm grows this plane
 459 considering only the biggest number of real coplanar features. The last problem
 460 observed comes from the homology test. The homology depends both on the
 461 motion between the images and the parameters of the planes. If the motion is
 462 small or the planes very similar, the three eigenvalues will be close to one and
 463 the test will fail. The selection of the images in such a way that they are as far
 464 as possible from each other reduces this problem.

465 4.2. Single camera topological map of planar regions

466 Using the planar regions extracted with our algorithm from the sequences of
 467 images, we have computed the associated graphs of planes. We have compared
 468 the resulting graphs with the image graphs created following the approach in
 469 [36]. Table 1 shows the comparison of the graphs generated using the house
 470 sequence and Table 2 shows the same comparison for the nine sequences of the
 471 city.

Table 1: Results for the house sequence

Map	Nodes	Edges	Feats	Feats/node	Size (MB)
Images	140	1728	78124	558	29.0
Planes	30	78	10832	361	4.0

Table 2: Results for the downtown Zaragoza sequences

Agent	Map	Nodes	Edges	Feats	Feats/node	Size (MB)
1	Images	300	3966	315830	1052	117.4
	Planes	104	340	15694	150	5.89
2	Images	78	796	58219	746	21.6
	Planes	17	62	3958	232	2.28
3	Images	33	230	29777	902	11.1
	Planes	8	28	1084	135	0.4
4	Images	465	7192	431607	928	160.6
	Planes	92	288	18497	201	6.95
5	Images	86	820	74123	861	27.5
	Planes	34	110	4104	120	1.54
6	Images	101	1434	79700	781	29.6
	Planes	11	40	409	361	1.68
7	Images	32	304	29258	914	10.9
	Planes	6	14	1611	268	0.61
8	Images	35	230	30434	869	11.3
	Planes	9	18	2021	224	0.77
9	Images	32	318	29237	913	10.9
	Planes	6	12	1162	193	0.44
Total	Images	1162	15290	1078185	927	400.9
	Planes	287	912	48540	169	20.20

472 In both cases the graph made of planes has less nodes and edges than the
 473 graph composed by images. The amount of space for storing the information
 474 is drastically reduced using our approach (Tables 1 and 2). Notice that in our
 475 approach the size of each plane is not bounded and there can be big differences

476 between nodes. In a visual memory made by images all the nodes will have
477 similar size (the features per node can be assumed to be bounded) whereas the
478 graph made of planes may contain very small planar regions with just a few
479 features, and other nodes can represent large planar regions with hundreds of
480 features and many homographies.

481 If the camera moves too fast or if there is a sequence of images in which there
482 are no planar regions the topological map will be unconnected. To prevent that
483 situations we have also imposed that consecutive planes are connected in the
484 local maps.

485 *4.3. Multi-Camera Distributed Topological map*

486 The distributed building of a topological map has been done using the maps
487 generated with the nine sequences of the city. The limited communications
488 between the agents are shown in fig 2. The diameter of the graph is 4, which
489 means that most part of the algorithms will finish only in four steps.

490 As previously commented, the triple matching algorithm finds a lot of small
491 planes which are a mix of different outliers (trees, buses, people and noisy fea-
492 tures). These planes do not apport real information and it is better to discard
493 them. In order to do so we have set a threshold of 50 features, so that planes
494 containing less features than the threshold are not considered for the distributed
495 global map. As shown in Table 3 only 132 of the 287 planes take part in the
496 distributed process. These planes amount a total of 46981 features.

497 Initially the agents exchange the information about their maps to create
498 fictitious planes. After four steps every agent has a map with 132 planes and
499 46981 features. Then the agents exchange their maps with their neighbors and
500 perform the local data association step. Figure 6 shows one plane seen by three
501 different neighbor agents with the found matches. The local associations delete
502 a total of 11 planes and 340 features (remaining 121 planes and 46351 features,
503 third row in Table 3). We have observed that the small number of associations
504 is mainly due to the different points of view of the trajectories and not because
505 of mismatching. Even so, in the multi-camera mapping it is advisable to use

506 larger RANSAC thresholds.

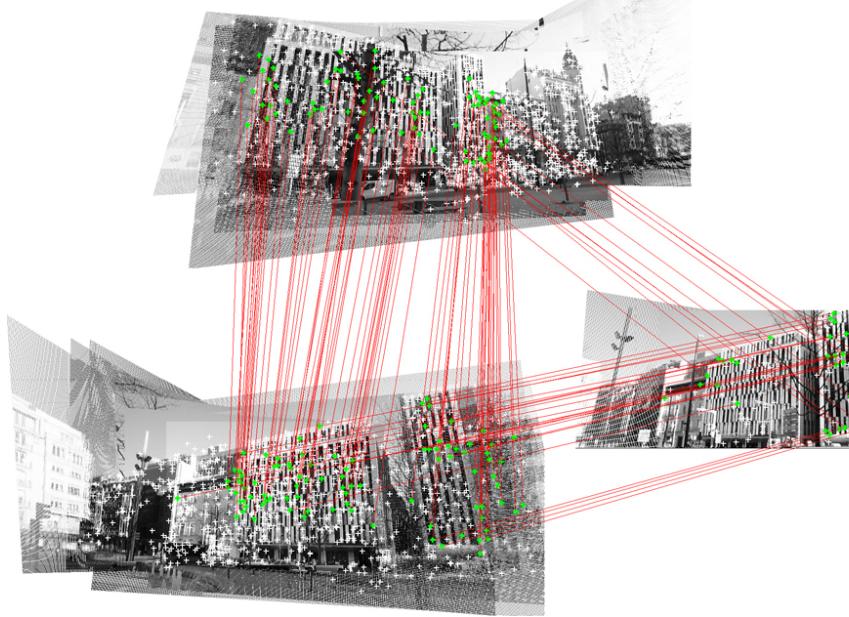


Figure 6: Multi-Camera Distributed Topological Map. Example of a planar region viewed in three different sequences. Red lines are the matches among the planes. In this example the reference plane is the top one.

Table 3: Evolution of the global map's size

Step	Nodes	Feats
Initial global map	287	48540
Erase Small planes	132	46981
Data Association	121	46351

507 After data association the agents execute the leader election algorithm in
508 order to fix the common references for each plane (algorithm 4). In the example
509 of the figure 6 the reference plane is the top one because is the one with the
510 most features.

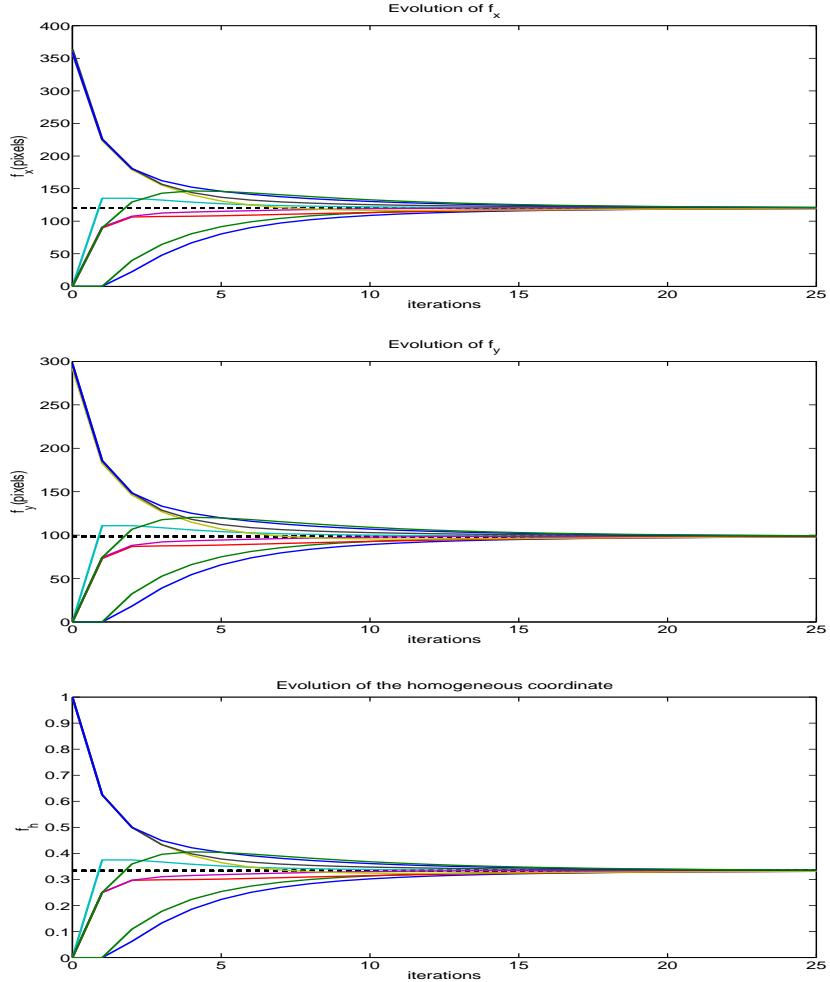


Figure 7: Consensus process for feature coordinates. Initially each agent has a different value of the feature coordinates. The nine agents exchange the information they have with their neighbors. It is observed that after 25 iterations consensus has been achieved and all the agents have the same value of the coordinates.

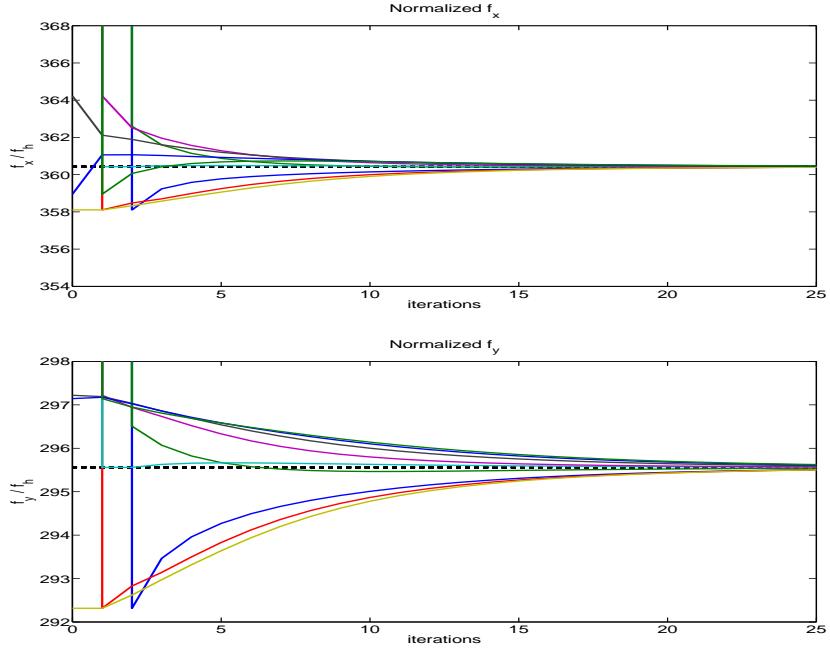


Figure 8: Consensus process for normalized feature coordinates. Evolution in normalized coordinates of the value of the same feature than in figure 7. Only three agents have a real observation of the feature, which are $(358.1072, 292.3148)^T$, $(364.2285, 297.2203)^T$ and $(358.9520, 297.1431)^T$. The rest of the agents have fictitious features with initial value equal to zero. At the beginning the agents with a fictitious coordinate have ∞ values. We can see that eventually the nine agents reach the same value, which corresponds to the average of the measurements of the three agents that observed the feature.

511 Finally, the agents execute the consensus rule to reach an average on the
 512 features and a consensus on the adjacency matrices of the topological maps.
 513 Figure 7 shows the consensus evolution of the three coordinates by the nine
 514 agents; we can see that they do not reach the desired average in f_x and f_y
 515 because only three of the nine agents have information about the feature but
 516 the average considers the value of the nine. However, the normalized coordinates
 517 (fig. 8) converge to the desired value, as stated in theorem 3.2.

518 A final comparison with the maps made by images has been done in order
 519 to analyze the amount of information transmitted through the network. This
 520 analysis has been carried out analytically considering the network topology and

Table 4: Amount of information transmitted (MB)

Total using images	$\simeq 5612$
Augment maps	< 1
Share maps	45.2
Data Association	13.2
Block Association	< 1
Leader Election	< 1
Consensus	1091.5
Total using planes	$\simeq 1150$

521 the information exchanged. An upper bound on the number of messages re-
 522 quired to transmit some information to the considered network (flooding) is 14.
 523 That means that for the map made of images, the 400.9 megabytes (Table 2)
 524 are transmitted 14 times, giving a total of 5612 transmitted megabytes. The
 525 breakdown of the transmitted information using topological maps is in table 4.
 526 The results show that the total information transmitted through the network
 527 using processed information, as planes in our proposal, is considerably smaller
 528 (20%). Also the final global map is better using planar regions since every match
 529 mixes the information reducing the total amount of data, whereas with images
 530 the total size is the sum of the local maps.

531 5. Conclusions

532 We have presented an algorithm for topological map building using auto-
 533 matically extracted planar regions. The planar regions are extracted using the
 534 information from previous images performing a Plane-Image-Image matching
 535 that tracks and grows the planar regions as new areas of the plane become vis-
 536 ible. The planar regions are organized in a graph, built simultaneously to the
 537 extraction. The idea of storing planes as information in the top graph rather
 538 than whole images (or each image feature set) presents several advantages. It
 539 includes some intrinsic information about the scene structure and it reduces the

graph size and complexity. This reduction is seized on multi-camera applications. We have proposed how to use distributed consensus techniques to merge the local maps obtained by different agents and obtain a distributed consensus on the global map. The initial problems observed with our approach to use these techniques have been exposed and solved also using distributed techniques, proving final convergence to the real average of the data. Experimental results with real images indoors and outdoors show the good performance of the method.

6. Acknowledgments

This work was supported by projects DPI2009-08126 and grant AP2007-03282 Ministerio de Educacion y Ciencia.

- [1] K. Achour and M. Benkhelif. A new approach to 3D reconstruction without camera calibration. *Pattern Recognition*, 34(12):2467 – 2476, 2001.
- [2] A. Angeli, D. Filliat, S. Doncieux, and J.A. Meyer. Fast and incremental method for loop-closure detection using bag of visual words. *IEEE Transactions on Robotics*, 24(5):1027–1037, 2008.
- [3] R. Aragues, J. Cortes, and C. Sagües. Distributed map merging in a robotic network. In *Workshop: Network Robot Systems: human concepts of space and activity, integration and applications. IROS*, pages 104–110, 2008.
- [4] R. Aragues, E. Montijano, and C. Sagües. Consistent data association in multi-robot systems with limited communications. In *Robotics: Science and Systems*, 2010. To appear.
- [5] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision*, pages 404–417, 2006.
- [6] M. R. Boutella, J. Luob, X. Shena, and C. M. Brownna. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

- 566 [7] F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009. Electronically available at <http://coordinationbook.info>.
- 569 [8] F. Castanedo, M.A. Patricio, J. Garcia, and J.M. Molina. Robust data
570 fusion in a visual sensor multi-agent architecture. *10th International Con-*
571 *ference on Information Fusion*, pages 1–7, 2007.
- 572 [9] J. Chen, W.E. Dixon, M. Dawson, and M. McIntyre. Homography-based
573 visual servo tracking control of a wheeled mobile robot. *IEEE Transactions*
574 *on Robotics*, 22(2):406–415, 2006.
- 575 [10] J. Civera, A. J. Davison, and J. M. M. Montiel. Inverse depth parametriza-
576 tion for monocular SLAM. *IEEE Transactions on Robotics*, 55(5):372–382,
577 2008.
- 578 [11] J. Courbon, Y. Mezouar, and P. Martinet. Indoor navigation of a non-
579 holonomic mobile robot using a visual memory. *Autonomous Robots*,
580 25(3):253–266, 2008.
- 581 [12] E. Eade and T. Drummond. Scalable monocular SLAM. In *IEEE Inter-*
582 *national Conference on Computer Vision and Pattern Recognition*, pages
583 469–476, 2006.
- 584 [13] Y. Fang, D.M. Dawson, W.E. Dixon, and M.S. de Queiroz. Homography-
585 based visual servoing of wheeled mobile robots. In *IEEE International*
586 *Conference on Decision and Control*, pages 2866–2871, 2002.
- 587 [14] V. Ferrari, T. Tuytelaars, and Luc Val Gool. Wide-baseline multiple-view
588 correspondences. In *IEEE International Conference on Computer Vision*
589 *and Pattern Recognition*, pages 718–725, 2003.
- 590 [15] F. Fraundorfer, C. Engels, and D. Nister. Topological mapping, localiza-
591 tion and navigation using image collections. In *IEEE/RSJ International*
592 *Conference on Intelligent Robots and Systems*, pages 3872–3877, 2007.

- 593 [16] F. Fraundorfer, K. Schindler, and H. Bischof. Piecewise planar scene re-
 594 construction from sparse correspondences. *Image and vision computing*,
 595 24(4):395–406, 2006.
- 596 [17] A. Gil, O. Reinoso, M. Ballesta, and M. Juliá. Multi-robot visual slam
 597 using a rao-blackwellized particle filter. *Robotics and Autonomous Systems*,
 598 58(1):6880, 2009.
- 599 [18] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*.
 600 Cambridge University Press, Cambridge, 2000.
- 601 [19] K. Konolige, J. Bowman, JD. Chen, P. Mihelich, M. Calonder, V. Lepetit,
 602 and P. Fua. View-based maps. In *Robotics: Science and Systems*, 2009.
- 603 [20] B. Kuipers. Modeling spacial knowledge. In *Int. Joint Conference on*
 604 *Artificial Intelligence*, pages 292–298, 1978.
- 605 [21] G. Lopez-Nicolas, C. Sagües, and J.J. Guerrero. Homography-based vi-
 606 sual control of nonholonomic vehicles. *IEEE International Conference on*
 607 *Robotics and Automation*, pages 1703–1708, 2007.
- 608 [22] N. Lynch. *Distributed Algorithms*. Morgan Kaufmann publishers, 1997.
- 609 [23] J.F. Menudet, J.M. Becker, T.Fournel, and C.Mennessier. Plane-based
 610 camera self-calibration by metric rectification of images. *Image and Vision*
 611 *Computing*, 26(7):913–934, 2007.
- 612 [24] E. Montijano and C. Sagües. Topological maps based on graphs of planar
 613 regions. In *IEEE/RSJ International Conference on Intelligent Robots and*
 614 *Systems*, pages 1661–1666, 2009.
- 615 [25] A. C. Murillo, C. Sagüés, J. J. Guerrero, T. Goedemé, T. Tuytelaars, and
 616 L. Van Gool. From omnidirectional images to hierarchical localization.
 617 *Robotics and Autonomous Systems*, 55(5):372–382, 2007.
- 618 [26] J. Qin and N.H.C.Yung. Scene categorization via contextual visual words.
 619 *Pattern Recognition*, 43(5):1874 – 1888, 2010.

- 620 [27] A. Remazeilles and F. Chaumette. Image-based robot navigation from an
621 image memory. *Robotics and Autonomous Systems*, 55(4):345–356, 2007.
- 622 [28] E. Royer, M. Lhuillier, M. Dhome, and J.M. Lavest. Monocular vision for
623 mobile robot localization and autonomous navigation. *International Journal*
624 *on Computer Vision*, 74(3):237–260, 2007.
- 625 [29] C. Sagüés, A.C. Murillo, F. Escudero, and J.J. Guerrero. From lines to
626 epipoles through planes in two views. *Pattern Recognition*, 39(3):384–393,
627 2006.
- 628 [30] G. Silveira, E. Malis, and P. Rives. Real-time robust detection of planar
629 regions in a pair of images. In *IEEE/RSJ International Conference on*
630 *Intelligent Robots and Systems*, pages 49–54, 2006.
- 631 [31] J. Sola, A. Monin, M. Devy, and T. Vidal-Calleja. Fusing monocular infor-
632 mation in multicamera slam. *IEEE Transactions on Robotics*, 24(5):958–
633 968, 2008.
- 634 [32] R. Szeliski and P. H. S. Torr. Geometrically constrained structure from
635 motion: Points on planes. In *European Workshop on 3D Structure from*
636 *Multiple Images of Large-Scale Environments*, pages 171–186, 1998.
- 637 [33] S. A. Teulosky, W. T. Vetterling, and B.P. Flannery. *Numerical Recipes*
638 *in C: The art of Scientific Computing*. Cambridge University Press, Cam-
639 bridge, 2002.
- 640 [34] J. Yao and W. Cham. Robust multi-view feature matching from multiple
641 unordered views. *Pattern Recognition*, 40(11):3081–3099, 2007.
- 642 [35] L. Zelnik-Manor and M. Irani. Multiview constraints on homogra-
643 phies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*,
644 24(2):214–222, 2002.
- 645 [36] Z. Zivkovic, O. Booij, and Ben Kröse. From images to rooms. *Robotics and*
646 *Autonomous Systems*, 55(5):411–418, 2007.