Spatial layout recovery from a single omnidirectional image and its matching-free sequential propagation

G. López-Nicolás, J. Omedes and J. J. Guerrero

Instituto de Investigación en Ingeniería de Aragón. Universidad de Zaragoza
C/ María de Luna 1, E-50018 Zaragoza, Spain
gonlopez@unizar.es, jason.omedes@gmail.com, josechu.guerrero@unizar.es

Abstract

The goal of this work is to recover the spatial layout of indoor environments from omnidirectional images assuming a Manhattan world structure. We propose a new method for scene structure recovery from a single image. This method is based on line extraction for omnidirectional images, line classification, and vanishing points estimation combined with a new hierarchical expansion procedure for detecting floor and wall boundaries. Each single omnidirectional image independently provides a useful hypothesis of the 3D scene structure. In order to enhance robustness and accuracy of this single image-based hypothesis, we extend this estimation with a new homography-based procedure applied to the various hypotheses obtained along the sequence of consecutive images. A key point in this contribution is the use of geometrical constraints for computing the homographies from a single line of the floor. The homography parametrization proposed allows the design of a matching-free method for spatial layout propagation along a sequence of images. Experimental results show single image layout recovery performance and the improvement obtained with the propagation of the hypothesis through the image sequence.

Keywords: Omnidirectional vision, Vanishing points, Homography, Manhattan-world, Spatial layout

1. Introduction

In this work, we address the problem of recovering the spatial layout of a scene from a set of lines extracted from indoor omnidirectional images. Although related works have been proposed in the literature, they are exclusively intended for conventional cameras. Omnidirectional cameras provide information in a wide field of view in one single image. Among these cameras we find the catadioptric systems, which are the combination of mirrors and cameras. A survey of the different classes of central catadioptric sensors with one mirror and lens are treated in [1]. For many applications, these camera systems are usually spotted in wheel-based robotic platforms with the mirror and the camera vertically aligned with respect to their symmetry axes. This consideration is useful since vertical lines of the scene become straight radial lines in the image and their vanishing point appears in the center of the image. When this assumption is not satisfied, all the lines of the scene become conic lines in the image and vanishing point detection is harder to execute given the additional geometrical complexity. Another good property in omnidirectional catadioptric systems as a consequence of their wide field of view is the observation of longer lines of the scene than in conventional camera systems. Additionally, vanishing points usually fall inside the omnidirectional image boundaries, being easier and more accurate to detect. Given these characteristics, the omnidirectional images allow a better way to deal with certain problems than conventional cameras, such as the problem of scene layout recovery from single images.

Spatial layout recovery of a scene requires the analysis of the surrounding structures to be able to recognize its global geometry, boundaries between floor and walls, relative orientation of surfaces, corners location, and relative distances between different elements of the scene. In general, a conservative spatial layout is enough for defining a navigability map of the environment, and this can be a powerful tool that provides very useful information for performing tasks such as navigation [2] or obstacle detection [3]. Therefore, rather than a precise and detailed map of the scene [4], [5], [6], we focus in this work on providing a conservative map in which the distribution of the different elements of the scene are classified as floor or walls. An example of the expected result for an input omnidirectional image after recovering its spatial layout is illustrated in Fig. 1.

Works in the literature addressing the problem of spatial layout recovery have been proposed for images acquired by conventional cameras, and most of them work under the Manhattan-world assumption [7]. This assumption states that the scene is essentially composed of three main directions orthogonal to each other, which is usually accomplished for indoor environments. This is used for instance in [8] to create a cubic room model attempting to recognize surfaces in cluttered scenes. Straight lines are usually easy to find and extract in structured environments like indoor scenes of buildings. Many techniques start with line detection as main image feature and impose geometrical constraints in order to find corners or relevant characteristics such as parallelism or orthogonality between elements to generate plausible hypothesis of the scene

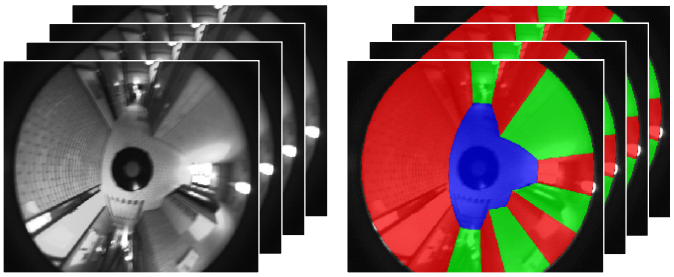

Figure 1: Example illustrating the goal of the presented approach. An input image (left) and the result of the algorithm (right). Walls are in green or red, and floor is in blue. The single view procedure is also enhanced through a sequence of images avoiding image feature matching.

structure [9]. All of these techniques were designed to be used with conventional images. In this context, the use of omnidirectional vision would be advantageous due to the wider field of view and the other characteristics commented above. However, omnidirectional vision presents challenges such as geometrical complexity that makes the previous proposed techniques for conventional images not plausible, and new algorithms are required to take into account this different geometry.

In this paper, we present an efficient method for spatial layout recovery from single omnidirectional images. Additionally, the method proceeds by propagating the individual hypothesis along a sequence composed by consecutive images of the scene. To our knowledge, this is the first spatial layout recovery algorithm that addresses the problem with omnidirectional vision. The first contribution is a new algorithm that provides the spatial layout recovery from a single image. In particular, the algorithm for estimating the structural layout hypothesis for a single image proceeds as follows. First, lines from the omnidirectional image are extracted and classified depending on their orientation by estimating the vanishing points (VPs). Examples of works using VPs are the 3D reconstruction from single standard images presented in [10] or the line clustering to determine the VPs from omnidirectional images [11]. Using the VPs classification, our proposed algorithm selects a set of lines producing different possible wall-floor boundaries. Imposing geometrical constraints an initial four walls-room hypothesis is generated followed by a hierarchical expansion process that accommodates the room hypothesis according to the image data.

This single image based algorithm shows good performance and it is robust to occlusions of the scene lines. However, depending on the complexity of the scene, misclassifications may occur. In order to improve the robustness of the approach, we use the spatial layout recovery algorithm for single images to extract hypothesis for a whole sequence of consecutive images. The idea is to combine the information obtained from the set of hypothesis to get a better approximation of the real structure. Regarding the use of a sequence of images to estimate the spatial layout, a related work is [12], where motion cues are used to compute likelihoods of indoor structure hypotheses by comparing the predicted location of point features on the environment model to their actual tracked locations in the standard image stream. Scene understanding has been also considered by combining geometric and photometric cues learned across multiple standard views [13]. However, that approach requires solving the stereo problem whereas our method relays on monocular information extended through a sequence of images without performing feature matching. In particular, our second contribution is a matching-free homography-based procedure that extends the single view approach to sequences of images improving the accuracy and robustness of the results. In our proposal, we avoid the necessity of classical methods of the prone to error feature matching process, especially harder in omnidirectional images, improving efficiency and robustness. In order to be able to compare and integrate several hypotheses, all of them have to be projected over the same frame. This projection is carried out by using homographies of the floor. The homography computation procedure presented is also novel since it uses geometrical constraints to reduce from four to one the minimal set of line correspondences necessary to compute each homography, which allows to skip computational expensive matching algorithms. Once the different hypotheses for some consecutive frames of the sequence are projected on an image, the algorithm looks for the one with the highest similarity rate with respect the other hypotheses. This one is selected as basic layout, and later on is averaged with the characteristics of the other hypothesis depending on how similar they are.

In our previous work [14], a preliminary version of our single-view spatial layout recovery approach was presented. Here we contribute with an improved version of that method, extend the experimental results with different datasets, and propose the matching-free method for sequential propagation of the layout. Experimental results showing the improvement of that propagation on image sequences are also included.

The paper is organized as follows. In section 2 we present the method for obtaining the spatial layout recovery from a single image. The following section 3 presents the procedure to propagate the hypotheses along a sequence of images avoiding feature matching. In section 4, the proposed approach is tested with real images of three datasets acquired with different cameras. Finally, conclusions are given in Section 5.

2. Single view spatial layout recovery

In this section, we describe the proposed algorithm to come up with the spatial layout of the scene from a single omnidirectional image. We start extracting lines from the image, which are then classified according to their orientation in order to carry out the estimation of vanishing points (VPs). Combining this information with a set of geometrical constraints we generate hypothesis about the floor contour. From the classified lines, a set of points is selected. These points are used to fit conic lines which represent plausible wall-floor boundaries. Then, a conservative four walls-room hypothesis is generated by selecting the four most voted conic lines. Finally, the initial hypothesis is expanded, according to the image data distribution, to obtain a representative hypothesis. This is carried out by replacing the initial floor contours with a set of appropriate conic lines so that successive hypotheses approximate better the actual shape of the scene. In the following, each step is explained in detail.

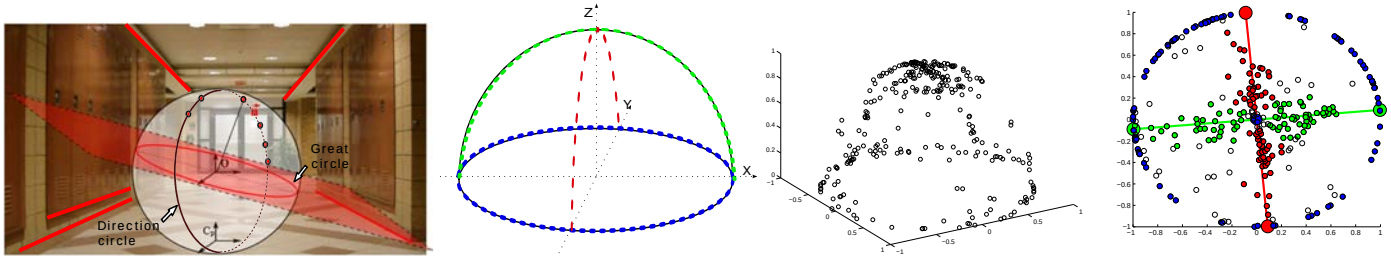

Figure 2: From left to right: Example of the projection of lines from the scene onto the unitary sphere as great circles. The assembling of the normal vectors, which define great circles from lines of the same direction, composes the representative global circle of each main direction. Ideal distribution of normal vectors according to three main directions. Unitary sphere with real normal vectors distribution of an image. Classification of the data using our proposed method.

2.1. Vanishing point estimation through line detection

Given a single omnidirectional image, the first step is the line extraction. Two methods to achieve the extraction of lines for catadioptric systems are [15] and [16]. We use here the approach of Bazin et al. and, although it was intended for paracatadioptric systems ($\xi = 1$), the generalization to hypercatadioptric systems ($0 < \xi < 1$) is trivial. This algorithm starts by detecting edges in the image and builds chains of connected pixels. At this point, omnidirectional camera calibration is needed to proceed [17]. A unifying theory for all central catadioptric systems was proposed in [18] and extended in [19]. In these works, the image formation model is developed by defining the well-known unified sphere model. Basically, a world point $\mathcal{P}$ is projected in a unit sphere as $\mathbf{p^s}$ in such a way that the omnidirectional images can be described by a perspective projection, involving the mirror parameters, related with the sphere by a translation of ξ along the symmetry axis. Given a point $\mathbf{p} = (u, v, 1)^T$ in the omnidirectional image, the point is transformed to a virtual plane using

$$\mathbf{m} = \mathbf{K}^{-1}\mathbf{p}\ , \tag{1}$$

being $\mathbf{K} \in \mathbb{R}^{3\times3}$ a matrix containing the intrinsic camera parameters coupled with mirror parameters:

$$\mathbf{K} = \begin{bmatrix} \gamma_u & 0 & u_0 \\ 0 & \gamma_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}\ . \tag{2}$$

The projection of the point $\mathbf{m} = (x, y, 1)^T$ onto the unit sphere is finally computed with

$$\mathbf{p}^s = (\lambda x, \lambda y, \lambda - \xi)^T\ , \tag{3}$$

where

$$\lambda = \frac{\xi + \sqrt{1 + (1 - \xi^2)(x^2 + y^2)}}{x^2 + y^2 + 1}\ . \tag{4}$$

Then, chains of connected pixels are projected onto the sphere, where each chain of pixels defines a great circle on the sphere (i.e. a section of a sphere that contains its diameter) that can be represented by its normal vector $\mathbf{n} = (n_x, n_y, n_z)$. After a splitting and merging process, these chains are considered as lines if they satisfy the great circle constraint [15].

Next step is line classification according to their relative orientation in the image. As mentioned before, chains of pixels compose a great circle onto the unitary sphere. Each of these great circles can be uniquely defined by its normal vector $\mathbf{n}$, and groups of normal vectors from lines that belong to each dominant orientation (i.e. vertical or horizontal) define also a representative circle onto the sphere. This is illustrated with an example in Fig. 2(left). This step aims at finding circles composed by sets of normal vectors in order to identify and classify the extracted lines according to the main directions that define the scene. We also impose additional constraints that help to speed up the analysis: Since most of the indoor scenes are composed by three orthogonal main directions, we use the Manhattan world assumption [7], which states that the scene is built in a cartesian grid. This hypothesis reduces the search to three orthogonal circles. Catadioptric systems are typically used with the camera system symmetry axis in vertical configuration. Under this assumption, we can identify the three main direction circles as follows:

1. Normal vectors from vertical lines define the first circle at the equator of the unitary sphere. Thus, lines whose normal vectors have the n_z component below a threshold are automatically classified as verticals, and removed for next steps.
2. Project the remaining normal vectors normalized by n_z, so every $\mathbf{n}$ falls in a 2D plane, conforming two perpendicular lines. Using RANSAC [20] we seek these two orthogonal directions which minimize the sum of squares perpendicular distances of the inliers (with a number of inliers greater than a minimum). These lines define the two horizontal (X and Y) main directions.
3. Each extracted line from the image is labelled according to the distance between its normal vector and one of the main directions. Notice that normal vectors with $n_z \simeq 1$ are conflictive since they can not be properly classified, so it is better to remove them in order to avoid misclassifications.
4. Finally, VPs are estimated as the points where the lines defining main directions intersect ($Z = 0$).

The result after performing the previous steps for line classification and vanishing point estimation is illustrated on the unit sphere with an example in Fig. 2. Another example is provided in Fig. 3, where the left image depicts the extracted lines while the image on the right shows the result.

Although we assume a general Manhattan world structure, lines in other main directions may be present. Additionally,

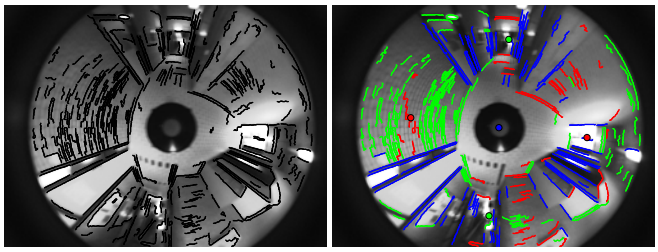

Figure 3: Extracted lines before (left) and after (right) classification according to the three main directions. Verticals are in blue, and horizontals are in red or green. The estimated VPs are depicted with big dots.

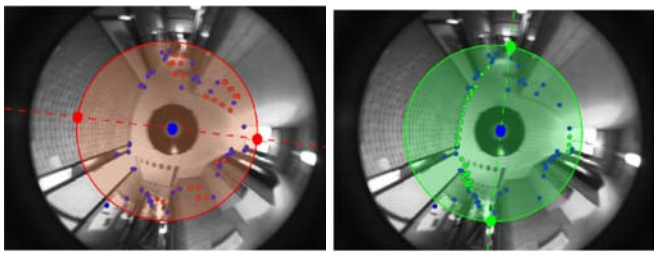

Figure 4: Selected putative points to look for floor-wall boundaries from Z-vertical (blue), X-horizontal (red), and Y-horizontal (green) grouped according to the four possible sectors within the horizon line, defined by the four VPs.

since lines present in the scene become conics in the omnidirectional images, one may think that these conics may represent projections of circles or other conic shapes. With the approach presented in [21] we are able to detect if a conic is the projection of a 3D straight line or not. Moreover, this kind of conics will not intersect in the vanishing points. Therefore, lines of curved walls will be rejected by our algorithm and not considered in the hypotheses.

2.2. *Generation of floor contours*

Lines that define the boundaries between floor and walls in a scene are represented as conic lines on the omnidirectional image. In this section, we describe the use of the VPs and lines previously classified to estimate the scene floor contour in the image. Despite boundaries are easily defined by horizontal lines, these are often noisy, susceptible to misclassification and sometimes do not appear in the image or they do in contours of objects such as furniture, windows or wall/floor decoration that can lead to wrong interpretations. However, it is usual that most vertical lines have their origin around ground level which makes them more reliable, in addition to the fact that vertical lines are easier to detect and classify.

Related to conic line generation, it has been demonstrated that knowing the calibration of the catadioptric system, a conic line in the image is defined from only two points [21]. Additionally, every line of the image must pass through a vanishing point. Thus, from any relevant point in the image that lies on the boundary between wall and floor, it is possible to compute the conic line that define such contour.

The identification of the points that fall on the borders of the floor is not an easy task. To achieve good accuracy, extracted lines are discretized into points as follow:

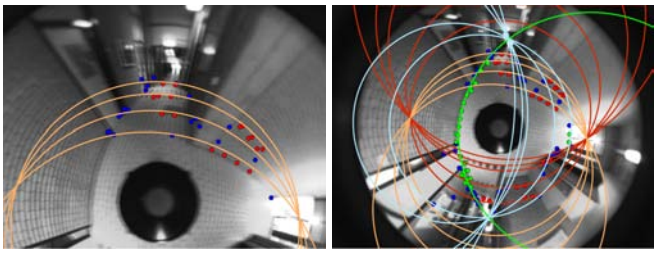

Figure 5: Conic lines for floor-wall boundaries hypothesis: Conic generation process for a set of points (left), and generated conics for every set (right).

- Points from vertical lines (denoted as $\mathbf{p}_z$ with $\mathbf{p}_z = \{\mathbf{p}_z^i,\, i = 1..m_z\}$): From each vertical line, we keep the closest point to the center of the image, since these have the highest probability to belong to the floor contour.
- Points from horizontal lines (denoted as $\mathbf{p}_x$ or $\mathbf{p}_y$ with $\mathbf{p}_x = \{\mathbf{p}_x^i,\, i = 1..m_x\}$ or $\mathbf{p}_y = \{\mathbf{p}_y^i,\, i = 1..m_y\}$, respectively): Horizontal lines with high probability to belong to the floor contour are sought. Giving especial importance to verticals, we select those horizontal lines near the previously selected points $\mathbf{p}_z$. Then, horizontal candidates are homogeneously discretized into a fixed number of points.

Once this distribution of relevant points has been extracted, we take in account two geometrical constraints. First, we define the horizon line as the conic going through the horizontal vanishing points on the catadioptric image. Image points situated within this conic belong to points of the scene with a lower height than the camera and thus may correspond to points of the floor, while those pixels outside the horizon conic correspond to points far from the ground, and therefore are not considered for floor contour generation. Secondly, it is remarkable that each couple of VPs divide the image in two hemispheres (see Fig. 4). Thus, at least one border of the contour of the floor has to belong to each of the four resultant sectors. To accomplish this condition we group the selected relevant points $\{\mathbf{p}_x, \mathbf{p}_y, \mathbf{p}_z\}$ in four sets. Each of these sets is composed by the extracted points from vertical and horizontal lines that fall within the correspondent hemisphere and have the orientation assigned by the VPs that generate the hemisphere (i.e. sectors formed by VPs of horizontal lines in X direction, contain the two sets composed by $\mathbf{p}_x$ and $\mathbf{p}_z$, while sectors generated by VPs of horizontal in Y direction, contain the two sets composed by $\mathbf{p}_y$ and $\mathbf{p}_z$) as shown in Fig. 4.

Points of these four groups are used for conic line generation, which will represent possible floor contours. The cross product between a vanishing point $\mathbf{v}$ and each of these points $\mathbf{p}_i$ generates a normal vector $\mathbf{n_i} = (n_{ix},\, n_{iy},\, n_{iz})^T$,

$$\mathbf{n_i} = \mathbf{v^s} \times \mathbf{p_i^s}\ . \tag{5}$$

Each normal vector $\mathbf{n_i}$ defines a conic $\overline{\mathbf{\Omega}}_\mathbf{i}$ which projects in the image as the conic $\widehat{\mathbf{\Omega}}_\mathbf{i}$ [22]:

$$\overline{\mathbf{\Omega}}_\mathbf{i} = \begin{bmatrix} n_{ix}^2(1-\xi^2) - n_{iz}^2\xi^2 & n_{ix}n_{iy}(1-\xi^2) & n_{ix}n_{iz} \\ n_{ix}n_{iy}(1-\xi^2) & n_{iy}^2(1-\xi^2) - n_{iz}^2\xi^2 & n_{iy}n_{iz} \\ n_{ix}n_{iz} & n_{iy}n_{iz} & n_{iz}^2 \end{bmatrix}$$

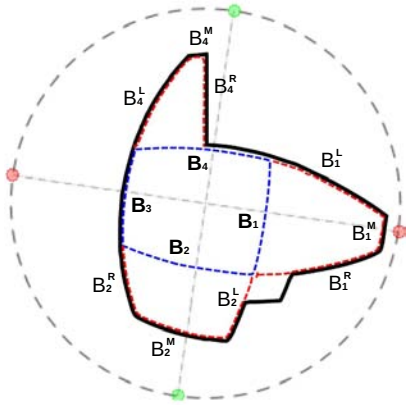

Figure 6: Example depicting the possible cases for boundary expansion (B_1 and B_2 are expandable regions, B_3 will not be expanded, and B_4 corresponds to an occluded corner). Thick line represents the actual room boundaries, first hypothesis is in dashed blue, and final one in dashed red.

$$\widehat{\mathbf{\Omega}}_\mathbf{i} = \mathbf{K}^{-T}\overline{\mathbf{\Omega}}_i\mathbf{K}^{-\mathbf{1}} \ . \tag{6}$$

Computing every combination of points and VPs for each of the four sets would come up with a large number of possible boundaries. Instead, we start generating a single conic from a randomly selected point $\mathbf{p}_i$ of one of the four sets and its respective VP. Then, distance between this conic and every point $\mathbf{p}_j$ $(i \neq j)$ of the current set of points is computed, using for this purpose the efficient approximation of point-conic distance defined in [21]. Those points whose distance to the conic is below a threshold are considered inliers, and they vote the conic. The conic is recomputed to minimize the distance to its inliers and the process is iterated until no more inliers are added. From the remaining points of the current set not voting for any conic, we randomly pick one and rerun the previous process (this is not a greedy algorithm, therefore every point of the current set is considered as a possible inlier even if it is already voting for another conic). This whole process is repeated until every point of the set is voting to at least one conic. Finally, the most voted conics are considered as eligible boundaries in the next steps of the method. Examples of the generated conics for each of the four sets are shown in Fig. 5.

2.3. Hierarchical hypothesis expansion

The estimation process of floor-wall boundaries presented in [23] is based on finding room corners. Our proposal is more robust since it does not rely on finding corners which often are not easy to detect. Additionally, we do not need to specify the number of walls for generating the hypothesis. Our method is also more efficient, given that trying every possible combination of normal vectors to classify the extracted lines or combination of corners to find the room hypothesis was time-consuming. With our new approach we avoid all these long iterations making viable its use in a sequence of images in real-time.

It can be seen in Fig. 5 that the right combination from the computed conics in Section 2.2 leads to an accurate definition of the real contour of the floor. An important issue is to estimate the number of walls that conform the 3D scene and their location. To solve this, we propose a hierarchical process that

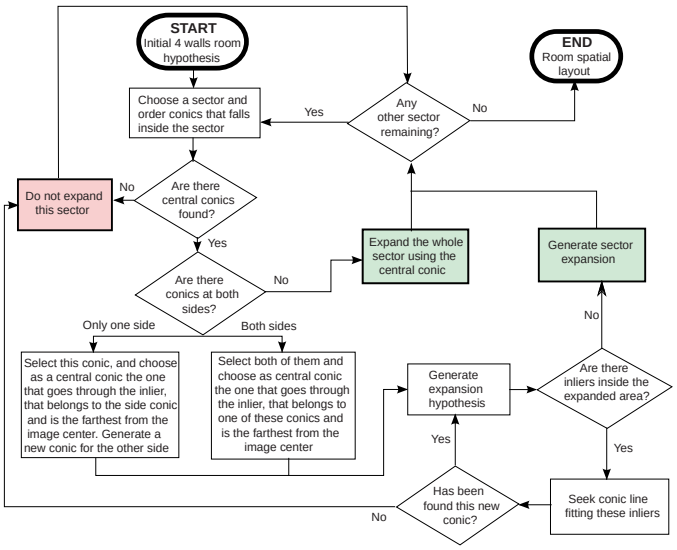

Figure 7: Flow chart explanation of the hierarchical expansion process.

conservatively starts with the hypothesis that the room is composed by four walls, and later on attempts to expand, or extend, the floor area of this first hypothesis.

In order to generate the initial hypothesis, we select the conic line of each of the four considered sectors that satisfies the conditions of being one of the most voted and at the same time one of the closest to the center of the image. We impose these conditions in order to be conservative and do not consider as floor the areas of the image that should be labelled as walls.

Let us denote as B_1, B_2, B_3 and B_4 the four boundaries of the initial hypothesis. These first contours may already define an accurate representation of the final shape of the room. However, in most of the cases it will only constitute the central area of the scene and so it can be expanded to find a better approximation to the actual shape of the room. We refer with expansion as replacing the conic line that defines the boundary B_i with others of the extracted set of conics in Section 2.2 which enlarge the area of the initial hypothesis room layout. Let them be B_i^L, B_i^M and B_i^R in clockwise order as shown in Fig. 6.

When looking for expansion of a contour B_i three different cases can be found:

- Case 1: Enough data is available to define the three new expansion boundaries B_i^M, B_i^L and B_i^R. So there will be expansion in the current sector.
- Case 2: There is no middle conic (B_i^M), or the existent ones are close to B_i, this means the most voted wall is still the same and there will not be expansion.
- Case 3: Data only allow us to define one or two boundaries to replace the contour to be expanded. This may be due to noise, luminosity reflects or to the existence of an occluded corner (i.e one of the walls is within a camera death angle and is occluded by other wall).

In order to generalize the procedure including every possible case, we define the flow chart shown in Fig. 7, explained as

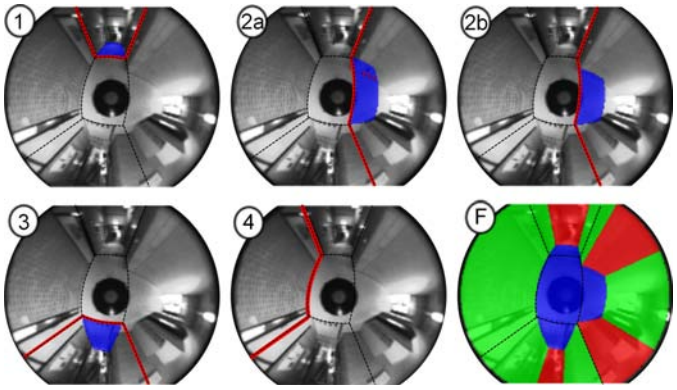

Figure 8: Expansion example: 1) Expansion of first boundary. 2a) First expansion hypothesis for sector 2, due to the presence of inner inliers we select a smaller enlargement. 2b) Next and definitive expansion hypothesis for sector 2. 3) Expansion of sector 3. 4) Due to this contour was already accurate, expansions are not found. F) Final floor hypothesis after expansion of the contours.

follows:

(a) Select a contour B_i to be expanded.

(b) Sort plausible conic lines within the studied sector to expand the current boundary in three groups: Two lateral (B_i^L, B_i^R) and one in the center (B_i^M). Each group can have more than one conic, being the most voted the most plausible. If there are no B_i^M, we are in the case 2, no expansion is done and we move to the next contour B_{i+1}.

(c) If at least one B_i^M is found but no B_i^L or B_i^R are detected, and the most voted B_i^M has more inliers than the current B_i, replacement is done and B_i^L, B_i^R will be selected to be the prolongation from B_{i-1} and B_{i+1}.

(d) If there exist at least one conic of every group B_i^M, B_i^L and B_i^R. Select the most voted conic from each group and generate a possible expansion hypothesis. In case one of the lateral groups has no conic, use B_{i-1} and B_{i+1} in replacement of B_i^L and B_i^R, respectively.

(e) Check if there are numerous inliers voting for other conics within the enlargement area, if not, the expansion is accepted and move to the next contour B_{i+1}.

(f) If the number of inliers is high, the expansion hypothesis is reduced to best fit those inliers by replacing one of the selected conics for the expansion by other with more inliers.

(g) Steps (e) and (f) are repeated until no inliers are found within the expansion area. After that there will be no more expansions.

The steps previously described to generate expansions are illustrated with an example in Fig. 8. Thus, the output of the proposed algorithm is finally the spatial layout of the scene obtained from a single omnidirectional image.

3. Matching-free sequential hypothesis propagation

In section 2, we have described our proposed method to recover the spatial layout of a scene from a single omnidirectional image. However, due to errors and misclassifications, the single view method may not obtain the desired result. Thus, in order to reduce these misclassifications, we recover the spatial layout for each frame of a whole sequence of images, where changes between successive photograms can be assumed to be small. Therefore, the spatial layout obtained for each image should be consistent along the same environment. Subsequently, we propose a propagation model so that each image of the sequence can share information with its adjacent frames improving the accuracy and robustness of the results. In the following, the procedure to propagate the hypothesis along the image sequence is presented. This approach relies on the homography computation of the floor across the views, as explained next.

3.1. Matching-free homography computation

A homography [24] is the transformation matrix $\mathbf{H} \in \mathbb{R}^{3\times3}$ that relates data across two views through a plane of the scene with $\mathbf{p^{s'}} \propto \mathbf{H}\mathbf{p^{s}}$ or $\mathbf{l^{s'}} \propto \mathbf{H^{-T}}\mathbf{l^{s}}$ in the case of point or line correspondences, respectively. The calibrated homography can be related to camera motion ($\mathbf{R}$, $\mathbf{t}$) and plane location as follows

$$\mathbf{H} = \mathbf{R}\left(\mathbf{I} + \mathbf{t}\,\mathbf{n_F}^T/d_F\right), \tag{7}$$

where $\mathbf{n_F}$ is the unit normal of the plane (the floor in our case) with respect to the reference frame and d_F is the distance between the floor and the camera position.

Regarding the homography computation [24], each line correspondence gives two independent equations. Given that $\mathbf{H}$ is defined by 8 degrees of freedom, a set of 4 line correspondences allows to determine the homography up to a scale factor by solving a linear system [25]. In our particular case, we use the conic lines that conform the floor contours obtained in Section 2 to generate line correspondences and compute the homography of the floor.

Feature matching is required in classical methods, but it is time consuming, difficult and usually suffers from mismatching. The usual presence of mismatches may be overcome with a robust method like RANSAC [20]. The RANSAC method proceeds by repeatedly generating hypothesis from a minimal set of correspondences s. In particular, the number N_s of samples required to ensure that at least one of the samples of size s has no outliers with a probability of p is

$$N_s = \log(1-p)/\log(1-(1-\epsilon)^s), \tag{8}$$

where $p = 0.99$ is usually chosen and ϵ is the probability that any selected correspondence is an outlier. However, there is a clear advantage of reducing the minimal set s in terms of computation. This is especially relevant since in our proposal there is no reliable initial matching of lines, and we have to consider all combinations of floor contour lines across the images as putative matches. In particular, we have the probability of outliers correspondences is

$$\epsilon = 1 - \frac{Number\,of\,inliers}{Number\,of\,inliers\,plus\,outliers}, \tag{9}$$

In the following, our goal is to define a new method without using additional costly and prone to error techniques of image feature matching. This process is possible thanks to additional

constraints that reduce the degrees of freedom of the homography. In particular, we consider the planar motion constraint. This implies the camera moves in a horizontal plane, parallel to the floor, so the rotation matrix $\mathbf{R}$ is restricted to roll (θ) around the vertical axis,

$$\mathbf{R} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} . \quad (10)$$

At the same time, since the camera system is vertically aligned, the normal of the floor only has vertical component

$$\mathbf{n_F} = (n_{Fx},\, n_{Fy},\, n_{Fz})^T = (0,\, 0,\, 1)^T , \quad (11)$$

and translation is performed in x and y coordinates,

$$\mathbf{t} = (t_x,\, t_y,\, t_z)^T = (t_x,\, t_y,\, 0)^T . \quad (12)$$

Replacing these values in the homography matrix yields:

$$\mathbf{H} = \mathbf{R}\,(\mathbf{I} + \mathbf{t}\,\mathbf{n_F}^T/d_F) = \begin{bmatrix} \cos\theta & \sin\theta & t_x/d_F \\ -\sin\theta & \cos\theta & t_y/d_F \\ 0 & 0 & 1 \end{bmatrix} \quad (13)$$

Additionally, the computed vanishing points can be tracked to give information about the rotation between two views (rotation angle θ), reducing the unknown homography degrees of freedom. Notice that, although we assume vertical alignment of the camera system for simplicity, a general orientation can be easily handled since the VPs provides the corresponding rotation matrix. Thus, the unknown remaining parameters in $\mathbf{H}$ are: t_x/d_F and t_y/d_F.

Therefore, considering planar motion reduces the minimal set of lines required to compute the homography to $s = 2$. Additionally, taking into account the information provided by the estimated vanishing points, the minimal set is reduced to $s = 1$. Table 1 gives examples of the number of samples N_s required for different cases given putative matching. Assuming that there is at least an inlier set, the number of outliers are the rest of possible correspondence combinations: $(N_L \cdot N'_L) - s$, where N_L and N'_L are the number of extracted conics in each image. Therefore $\epsilon = 1 - s/(N_L \cdot N'_L)$. For simplicity, we consider in this example that $N_L = N'_L$. As it can be seen in examples of Table 1, the computation of the general homography ($s = 4$) is computationally infeasible, and the cases of $s = 2$ or $s = 1$ are also rather costly. Thus, we choose not to use RANSAC here given that the exhaustive search in the number of samples $(N_L)^2$ is feasible in practice, and avoids the probability of failing in the random search.

The use of the minimal set of one line for computing the homography is a key point for the good performance of the method. This is more related with the feasibility and robustness of the method rather than with the accuracy of the computed homography. A main problem for the feasibility of line-based approaches in omnidirectional vision is that there is not yet a general or robust line descriptor for matching. Therefore, the matching-free computation method provides an important advantage in terms of robustness and efficiency. In terms of accuracy, the computed homographies are similar to the case of using standard methods (once skipped the line matching problem)

Table 1: Given N_L lines in each image: Number of samples N_s required for a RANSAC solution with 99% probability of no fail, to robustly compute the homography given a sample size of $s = 1$, $s = 2$, and $s = 4$. We propose performing an exhaustive search using $(N_L)^2$ cases.

N_L	$s = 1$	$s = 2$	$s = 4$	$(N_L)^2$
4	72	293	1,177	16
5	113	718	7,025	25
6	164	1,490	30,213	36
7	224	2,762	103,701	49
8	293	4,714	301,803	64
9	371	7,552	774,363	81
10	459	11,511	1,798,893	100
20	1,840	184,205	460,517,014	400

Figure 9: Left: Hypothesis of image I (red). Center and right represent two homography cases where the white contour represents hypothesis of image I' and red contour is the hypothesis projection from I to I'. Center: Case where homography does not perform well. Right: Case of the best homography.

because the most voted homography is used as well to compute the final homography from all the voting lines. In terms of efficiency, it can be seen in Table 1 that, for example, once 10 initial matches were provided with at least four correct, the standard homography procedure would require 1,798,893 iterations to guarantee obtaining a correct homography, whereas the constrained homography would only require 100 iterations. This is an important advantage even if the time per iteration is small.

Let $\mathbf{n}$ be the normal defining a line in the image I, and $\mathbf{n}'$ the correspondent normal of $\mathbf{n}$ seen in the image I', these lines are related by the homography with:

$$\mathbf{n}' \propto \mathbf{H}^{-T}\mathbf{n} . \quad (14)$$

Taking the inverse and transpose matrix of $\mathbf{H}$, and using the notation: $\mathbf{H}^{-T} = [h_{ij}]$ where $i, j = 1, 2, 3$, it depends on two unknown parameters (h_{31} and h_{32}). The resulting equations system given a line correspondence is:

$$\begin{bmatrix} n_x n_y{}' & n_y n_y{}' & a_{13} \\ -n_x n_x{}' & -n_y n_x{}' & a_{23} \end{bmatrix} \begin{pmatrix} h_{31} \\ h_{32} \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} , \quad (15)$$

where

$$\begin{aligned} a_{13} &= n_z n_y{}' - n_x n_z{}' \sin\theta - n_y n_z{}' \cos\theta \\ a_{23} &= n_x n_z{}' \cos\theta - n_y n_z{}' \sin\theta - n_z n_x{}' \end{aligned} \quad (16)$$

Solving for h_{31} and h_{32}, and through the transpose and inverse of $\mathbf{H}^{-T}$ we recover $\mathbf{H}$. Therefore, given that correspondences between conic lines are not known, the homographies for every $(N_L \cdot N'_L)$ combination is computed (in fact, some of these

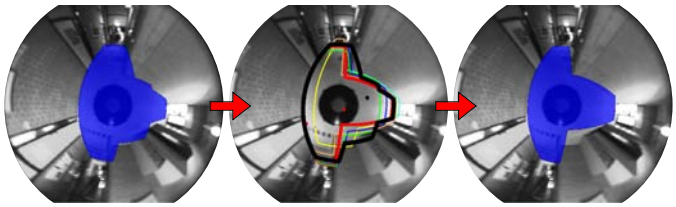

Figure 10: Left. Example of wrong hypothesis (walls area identified as floor). Center: Room hypothesis of actual image (black). Rest of hypothesis are projected on the same frame, and the one with highest similarity rate is chosen as basic layout (red). Right: Final result obtained from the combination of the hypothesis sequence.

combinations will not have physical meaning and could be ignored) and the one that generates the best similarity between the floor contour of both images, I and I', is chosen (see example in Fig. 9).

3.2. Sequential propagation of single-view spatial layout

The last step of the proposed method consists in comparing hypotheses obtained for a sequence of images, in order to obtain an averaged hypothesis which best fits the set of floor contours. In order to compare hypothesis extracted from different frames, it is necessary to project all of them onto the current image. For hypothesis projection, we use the homographies computed with the procedure defined in section 3.1 so that a set of N previous hypothesis $(h_1, ..., h_N)$ is projected over the image I.

At this point, hypothesis from different images can be compared. In case of images with wrong results, and specially in those cases when the camera is going from one room to another, the layout hypothesis obtained for the N frames projected on the image I are likely to be represented by different shapes or number of boundaries. Figure 10(center) shows and example of several hypothesis projected into the same frame. In order to remove bad hypothesis or choose in which of both transition rooms the camera is, we propose to select a basic layout. This basic layout is defined as the hypothesis among the N frames projected on I which satisfies the highest similarity characteristics rate of the set. In our proposal we discretize every hypothesis $(i = 1, ..., N)$ contour in K points $(p_1^i, ..., p_K^i)$ and define the similarity rate of each one as the minimum point to point distance between it and the set of the other N hypothesis:

$$SimRate(i) = \sum_{j=1}^{N} \|(p_1^i, ..., p_K^i) - (p_1^j, ..., p_K^j)\| \ . \qquad (17)$$

The basic layout defines then the final hypothesis number of walls for the image I, and their approximate location:

$$BasicLayout = Hypothesis(\arg\min_i |SimRate(i)|) \ . \quad (18)$$

Nevertheless, for a best fitting we average the basic layout with the other N hypothesis. Let us denote as $F(i)_j$ each line j composing the floor hypothesis of the image i. The final layout result is defined as a weighted average of correspondent lines between the N different hypothesis as shown in Fig. 10(right).

$$F(final)_j = \frac{\sum_{i=1}^{N} w(i)F(i)_j}{\left\| \sum_{i=1}^{N} w(i)F(i)_j \right\|} \ , \qquad (19)$$

where the weights correspond to the similarity rate between each hypothesis i and the basic layout:

$$w(i) = \frac{SimRate(i, BasicLayout)}{\max(SimRate)} \ . \qquad (20)$$

The set of lines $F(final)_j$ is the final hypothesis (h^F) and represents the definitive contour of the scene floor under study, I. However, for next iterations of the sequential algorithm we do not replace the contour of the initial hypothesis for the image I with the new result obtained. In case of replacement every voting hypothesis would turn out to have a similar contour with only small variations between them, and the algorithm would not be flexible to allow changes for the successive contour shapes. To avoid this, but also take in account the final results obtained, we store up to M final hypothesis that we propagate next to the initial hypothesis so both, initial and final, weight in the floor contour decision.

Therefore, it is possible to store a number $M < N$ of final hypothesis $({h_1}^F, ..., {h_M}^F)$ so that the larger M is, the more robust to errors the results are, but the less flexible the algorithm is to introduce changes in shape of the rooms.

In our experiments we have chosen $N = 7$ and $M = 2$, where N is the number of voting frames preceding the current evaluated image and M the number of final hypothesis, with the objective of increasing the results robustness, but at the same time be flexible to possible changes in the scene.

Figure 11: Devices used for omnidirectional image acquisition: a helmet (left), a backpack (center), and a mobile robot (right).

4. Experimental results

The proposed method has been tested with three different image datasets. All the images have been acquired indoors using hypercatadioptric camera systems, composed by a standard camera aiming to an hyperbolic mirror. The first camera system consists of a compact hypercatadioptric camera spotted on the top of a helmet. This device is designed in the framework of personal assistance applications (Fig. 11 left). Notice that, due to head movements, the image data acquired with this system

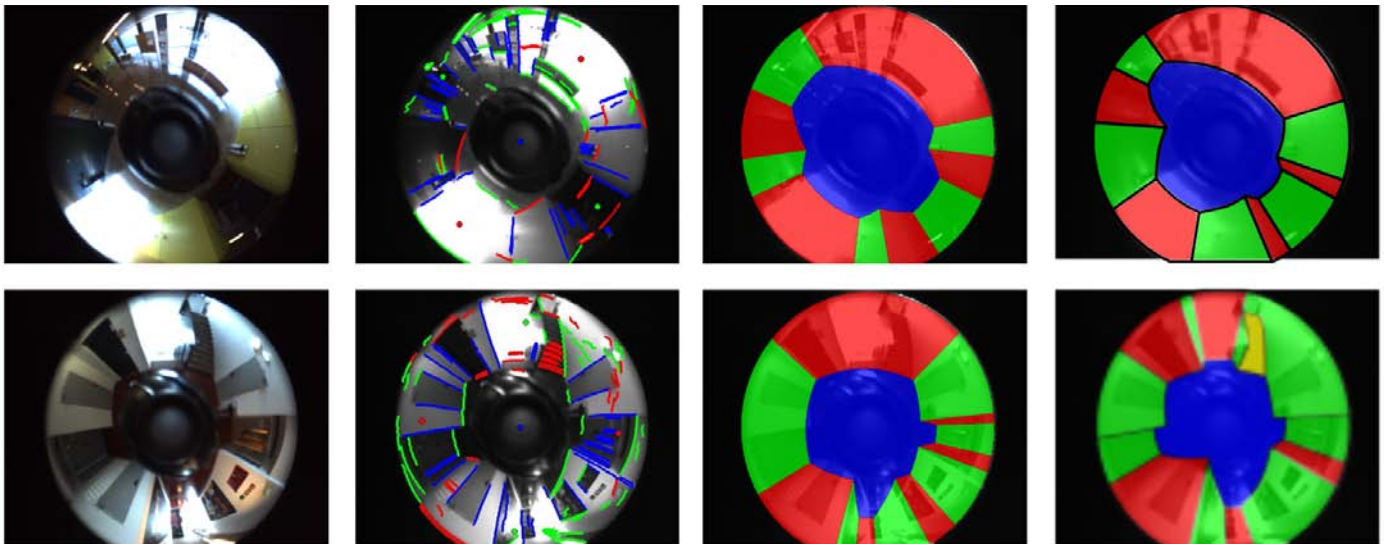

Figure 12: Experimental results for the helmet dataset. From left to right: Input image fed into our algorithm. Extracted lines and estimated VPs (big dots) classified according to the 3 main directions. Third column are the results obtained by our proposed method. Last column, is the ground truth manually labelled.

Table 2: Calibration parameters for the different omnidirectional systems.

	Helmet	Backpack	COGNIRON
Size (pixel)	1024×768	1280×1024	1024×768
u_0 (pixel)	523	639.1	530
v_0 (pixel)	408	524.7	389
γ_u (pixel)	296.8	352.8	262.49
γ_v (pixel)	296.8	353	262.76
ξ (mm)	0.9737	0.9733	0.93

barely holds the vertical camera alignment assumption, but the method shows to be robust to vertical misalignment. The second camera system is composed by a hypercatadioptric camera attached to a backpack carried by the user. In this case, the camera motion is more stable since it rests on the carrier shoulders (Fig. 11 center). The third image dataset counts with a numerous amount of indoor scenes and belongs to the project COGNIRON [26] which is available online. These images were taken with a hypercatadioptric system spotted on a mobile robot, a Pionner PeopleBot such as the one depicted in (Fig. 11 right). The area occluded by this device is minimum, and since the camera is spotted on a robot, verticality is assured. Image size and camera calibration of each of these systems is given in Table 2.

In the following sections, the algorithm for single view spatial recovery and the sequential hypothesis propagation method are tested. The experiments have been executed using Matlab®, running at 3 seconds per frame.

4.1. Results of single view spatial layout recovery

In this section we show some results obtained for the different datasets when applying the layout recovery algorithm for a single omnidirectional image. The ground truth was obtained by manually labelling the walls and floor of the studied scenes to compare with our results.

Images in Fig. 12 are taken by the helmet-camera. It is noticeable that the helmet covers a wide area of the center of the image. This makes more difficult to extract floor boundaries. In this case, first picture was taken in a wide hall with two large crystal walls. Complexity increases in the second case, where the floor is barely visible and the scene presents a big amount of branching and stairs. Even though, our method is able to overcome these difficulties and comes up with a close approximation of the correct room layout. Stairs are not classified because our hypothesis only takes in account three main directions and, since they are in a diagonal direction, they are not considered. Images in Fig. 13 are taken with the backpack-camera system. The better visibility of the floor in this system allows to obtain more reliable lines. First image corresponds to a long hall section with an enlargement in one of its sides. Despite the high level of luminosity coming through the window, our method is able to identify every element of the scene. Next image shows another narrow hall next to two-way stairs. As in the previous case, stairs cannot be identified as a separated element since they do not correspond to the three main directions, nonetheless the rest of the room is properly approximated.

Figure 14 shows some results of the database COGNIRON. First example corresponds to an L shape hall. Walls are not too saturated with objects so the result is accurate. We also observe an occluded corner at the superior part of the image. Second case represents a hall with a desk and a shelf, where our algorithm is able to recognize these obstacles. However, it does not detect the open door situated at the top part of the image, probably due to all the light going through it. Third picture is taken in an intersection of corridors and despite the complexity of the scene our method still achieves a good approximation of the structure layout. The last example is a hall with a huge number of horizontal lines on the wall situated at the bottom of the image. In this case our method fails by detecting one of these lines as the most likely floor boundary.

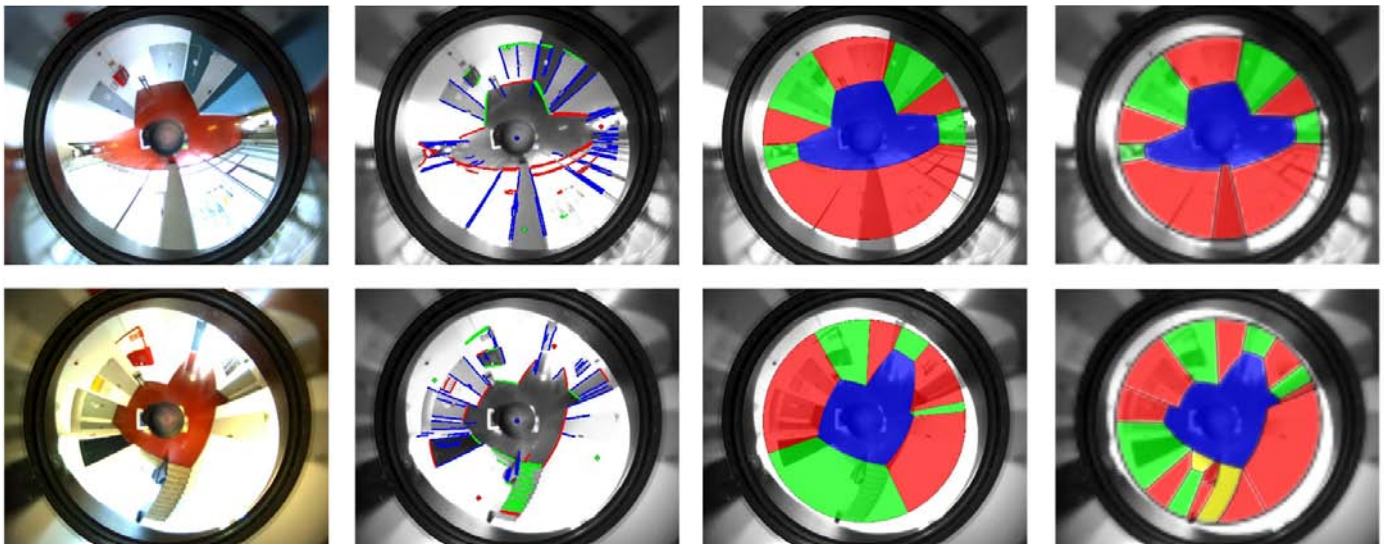

Figure 13: Experimental results for the backpack dataset. From left to right: Input image fed into our algorithm. Extracted lines and estimated VPs (big dots) classified according to the 3 main directions. Third column are the results obtained by our proposed method. Last column, is the ground truth manually labelled.

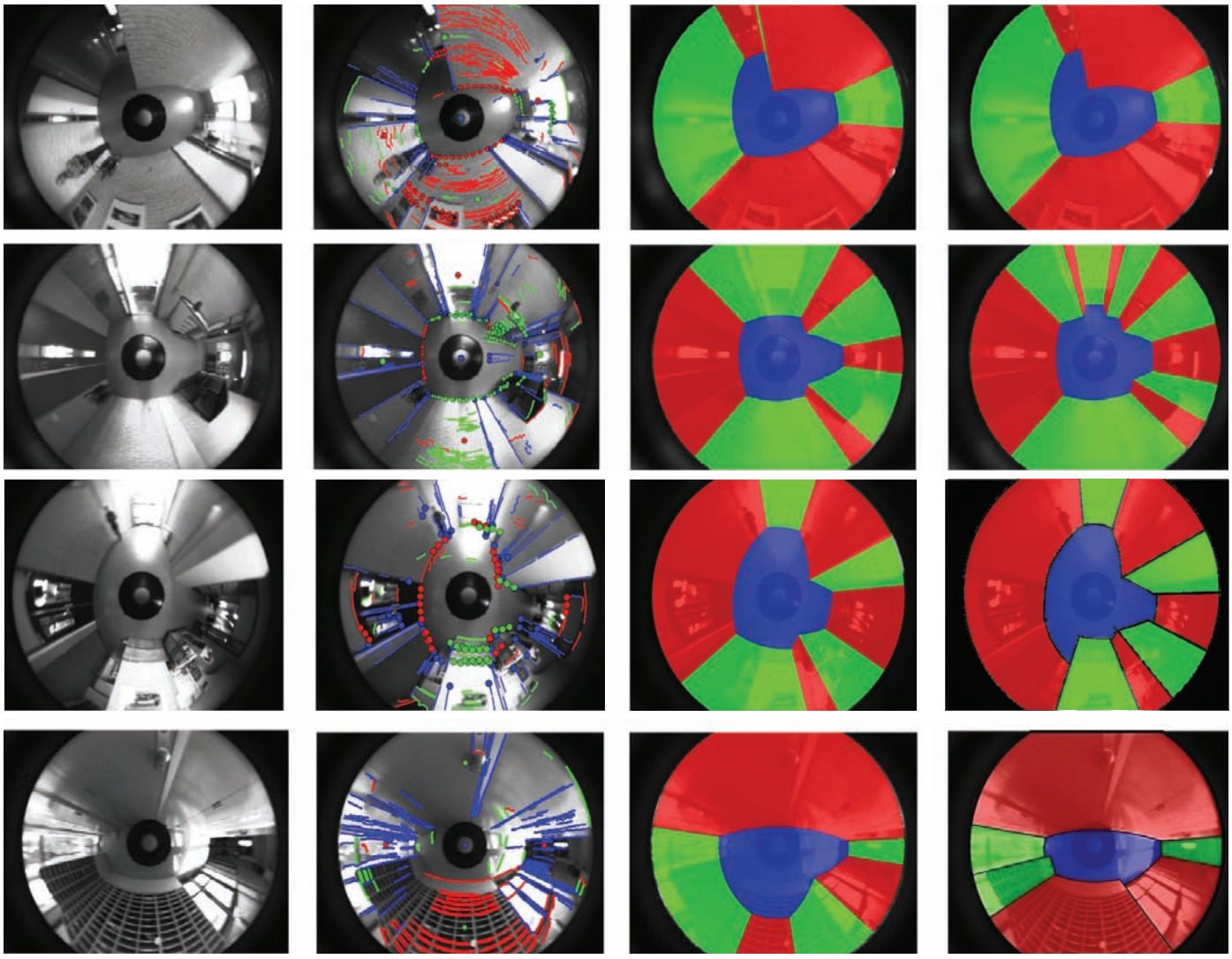

Figure 14: Experimental results for the COGNIRON dataset. From left to right: Input image fed into our algorithm. Extracted lines and estimated VPs (big dots) classified according to the 3 main directions. Third column are the results obtained by our proposed method. Last column is the ground truth manually labelled.

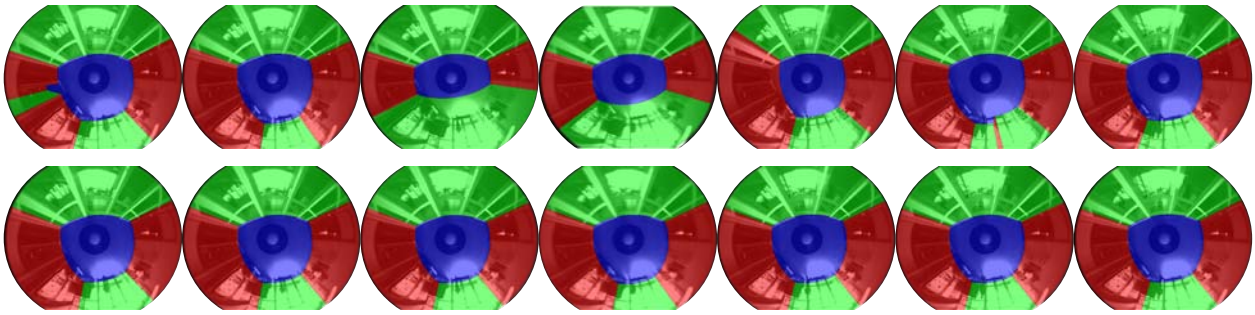

Figure 15: Sequence composed by 7 consecutive frames. First row shows results obtained without homography-based hypothesis propagation. Second row shows how including the use of homographies leads to more homogeneous results and correct possible errors in the original hypotheses.

4.2. Results of sequential propagation

The performance of this method has been tested with different sequences of images of the dataset COGNIRON. First sequence (Fig. 15) is composed by 7 consecutive frames, depicting how the application of the homography-based hypothesis propagation achieves final results more robust to misclassification, preserving similar contours among frames.

The second sequence is composed by 14 consecutive images (Fig. 16). In this case, the selected scene is a transition zone where the camera moves from a hall with a T-shape into one of its ramifications, where the shape of the scene becomes rectangular (I-shape). It is shown how in the initial stage of the transition, the first images of the sequence match the shape of the hall. However, as soon as the camera starts leaving the T-shape area and enters into the next part of the hall, the sequential algorithm attempts to preserve the original shape of the corridor leading to erroneous results (images 10, 11 and 12 of the sequence). Nevertheless, as the camera moves further our method quickly adapts and the resultant hypothesis transforms itself to match with the new surroundings.

Additional results are presented in the attached **video** composed by a sequence of 200 frames. We have labelled a ground truth for some images of this sequence and compared the results obtained for single view spatial layout defined in Section 2 and results obtained once we apply the proposed sequential propagation method in Section 3. We define as true positives (t_p) the number of pixels that our results and the floor ground truth have in common, false positives (f_p) the number of pixels identified as floor in our method but do not correspond to the floor in the ground truth, and false negatives (f_n) the number of pixels identified as not floor when they are labelled as floor in the ground truth. With these values we can compute several performance indexes to compare the improvement of our method. Let us denote *precision* as $\frac{t_p}{t_p+f_p}$, *recall* as $\frac{t_p}{t_p+f_n}$ and F_1 a harmonic mean of precision and recall as $(\frac{2 \cdot precision \cdot recall}{precision+recall})$. Computing the average of the performance values obtained for frames of the sequence, we observe in Table 3 that we achieve an improvement of precision and recall using our proposed method.

As commented previously, we have designed a conservative method of spatial layout recovery in the sense that we prefer this algorithm to be reliable in detecting walls or obstacles rather than detecting the whole floor with a high degree of accuracy, even if sometimes it means to classify complex floor parts as obstacles. In general, this implies lower recall than precision, as can be seen in Table 3. However, the difference between precision and recall is not very high because, in general, the floor area close to the user is bigger and easier to classify than the area in the limits of the floor with the walls. Therefore, a floor patch near the walls classified as obstacle has less impact in the recall than a wrong classification of the floor close to the user, which otherwise could be more critical from a practical point of view.

Table 3: Averaged performance values obtained from an image sequence for the single view spatial recovery method and by using the hypothesis propagation through the image sequence.

	Precision	Recall	F_1
Single view	0.8645	0.8140	0.8385
Sequence propagation	0.9511	0.9274	0.9391

5. Conclusion

We have developed a new method for spatial layout recovery from a single omnidirectional image based on conic lines classification and its propagation along a sequence of images without requiring feature matching. Experimental results show that the obtained spatial layouts from single images are quite good and provides useful information for applications such as generation of navigability maps. As expected, the method accuracy decreases if the quality of the scene lines detected is low. Nevertheless, a small number of extracted lines can provide good results. In order to enhance the robustness of the method, we propagate the hypotheses along a sequence of images by means of homographies, each one computed from a single floor line, and still skipping the prone to error and computational expensive matching algorithms. Experiments carried out in real sequences of images, taken indoors on a robot or a person with a wearable omni-camera, show good performance with considerable improvement in the accuracy and robustness of the results.

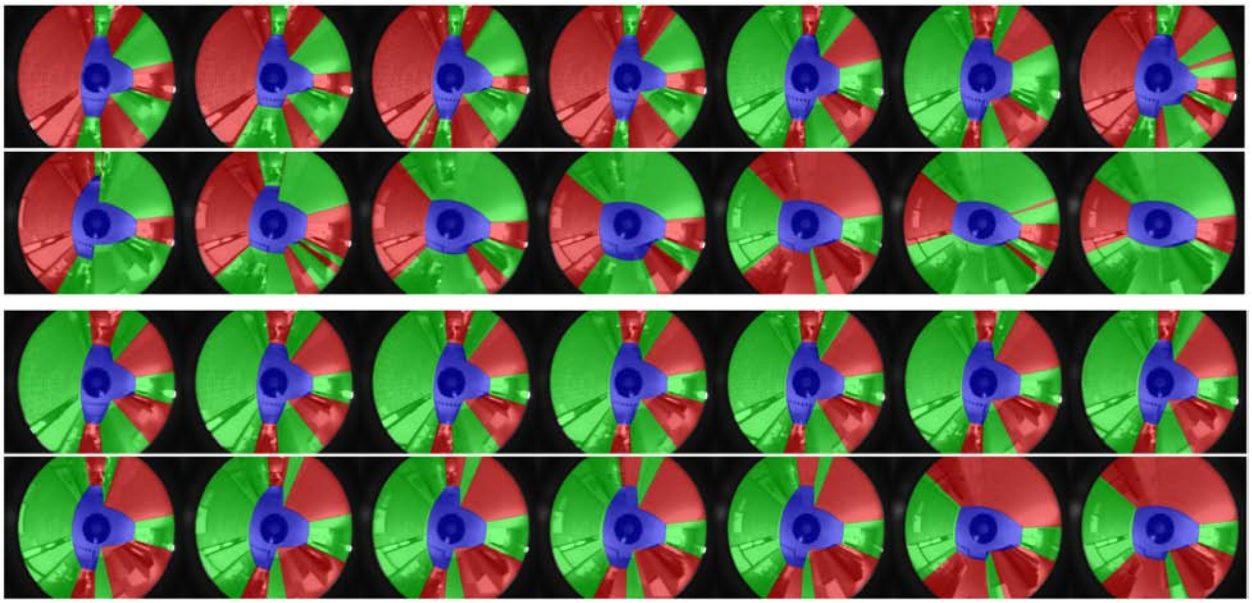

Figure 16: First two rows show a sequence of 14 frames without applying the homography-based hypothesis propagation. Note how red and green walls alternate between images. The bottom two rows show the same sequence applying the propagation process. Here color code keeps the same along the whole sequence thanks to the VPs tracking. Last images show transition between T hallway into an I shape corridor.

6. Acknowledgments

This work was supported by Ministerio de Economía y Competitividad (project DPI2012-31781) and FEDER funds, and by DGA-FSE (group T04).

References

[1] S. Baker, S. K. Nayar, A theory of single-viewpoint catadioptric image formation, Int. Journal of Computer Vision 35 (2) (1999) 175–196.

[2] Y. Matsumoto, K. Ikeda, M. Inaba, H. Inoue, Visual navigation using omnidirectional view sequence, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 1999, pp. 317–322.

[3] K. Yamazawa, Y. Yagi, M. Yachida, Obstacle detection with omnidirectional image sensor hyperomni vision, in: IEEE International Conference on Robotics and Automation, 1995, pp. 1062–1067.

[4] P. Sturm, S. Maybank, A method for interactive 3D reconstruction of piecewise planar objects from single images, in: British Machine Vision Conference, 1999, pp. 265–274.

[5] S. Fleck, F. Busch, P. Biber, W. Strasser, H. Andreasson, Omnidirectional 3D modeling on a mobile robot using graph cuts, in: IEEE International Conference on Robotics and Automation, 2005, pp. 1748–1754.

[6] A. Rituerto, L. Puig, J. J. Guerrero, Visual SLAM with an omnidirectional camera, in: Int. Conf. on Pattern Recognition, 2010, pp. 348–351.

[7] J. M. Coughlan, A. L. Yuille, Manhattan world: Compass direction from a single image by bayesian inference, in: Int. Conf. on Computer Vision, 1999, pp. 941–947.

[8] V. Hedau, D. Hoiem, D. Forsyth, Recovering the spatial layout of cluttered rooms, in: IEEE International Conference on Computer Vision, 2009, pp. 1849–1856.

[9] D. Lee, M. Hebert, T. Kanade, Geometric reasoning for single image structure recovery, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2009.

[10] A. Criminisi, I. D. Reid, A. Zisserman, Single view metrology, in: International Conference on Computer Vision, 1999, pp. 434–441.

[11] J. C. Bazin, Y. Seo, C. Demonceaux, P. Vasseur, K. Ikeuchi, I. Kweon, M. Pollefeys, Globally optimal line clustering and vanishing point estimation in Manhattan world, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2012, pp. 638–645.

[12] G. Tsai, C. Xu, J. Liu, B. Kuipers, Real-time indoor scene understanding using bayesian filtering with motion cues, in: IEEE International Conference on Computer Vision (ICCV), 2011, pp. 121–128.

[13] A. Flint, D. Murray, I. Reid, Manhattan scene understanding using monocular, stereo, and 3D features, in: IEEE International Conference on Computer Vision (ICCV), 2011, pp. 2228 –2235.

[14] J. Omedes, G. López-Nicolás, J. J. Guerrero, Omnidirectional vision for indoor spatial layout recovery, in: Frontiers of Intelligent Autonomous Systems, Springer-Verlag, 2013, pp. 95–104.

[15] J. C. Bazin, I. Kweon, C. Demonceaux, P. Vasseur, A robust top-down approach for rotation estimation and vanishing points extraction by catadioptric vision in urban environment, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008, pp. 346–353.

[16] J. Bermudez, L. Puig, J. J. Guerrero, Line extraction in central hypercatadioptric systems, in: OMNIVIS, 2010.

[17] L. Puig, J. Bermudez, P. F. Sturm, J. J. Guerrero, Calibration of omnidirectional cameras in practice: A comparison of methods, Computer Vision and Image Understanding 116 (1) (2012) 120–137.

[18] C. Geyer, K. Daniilidis, A unifying theory for central panoramic systems and practical implications, in: Proceedings of the 6th European Conference on Computer Vision, Part II, Springer-Verlag, 2000, pp. 445–461.

[19] J. P. Barreto, H. Araujo, Issues on the geometry of central catadioptric image formation, in: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, 2001, pp. 422–427.

[20] M. A. Fischler, R. C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM 24 (6) (1981) 381–395.

[21] J. Bermudez-Cameo, L. Puig, J. J. Guerrero, Hypercatadioptric line images for 3D orientation and image rectification, Robotics and Autonomous Systems 60 (6) (2012) 755–768.

[22] J. Barreto, General central projection systems: Modeling, calibration and visual servoing, Ph.D. thesis (2003).

[23] N. D. Ozisik, G. López-Nicolás, J. J. Guerrero, Scene structure recovery from a single omnidirectional image, in: OMNIVIS, 2011, pp. 359–366.

[24] R. I. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, 2nd Edition, Cambridge University Press, 2004.

[25] C. Mei, S. Benhimane, E. Malis, P. Rives, Homography-based tracking for central catadioptric cameras, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 2999–3004.

[26] Z. Zivkovic, O. Booij, B. Krose, From images to rooms, Robotics and Autonomous Systems 55 (5) (2007) 411–418.