Citation: J. Omedes, G. López-Nicolás and J. J. Guerrero. Omnidirectional Vision for Indoor Spatial Layout Recovery. The 12th International Conference on Intelligent Autonomous Systems, Jeju Island, Korea, June 26-29, 2012

Omnidirectional Vision for Indoor Spatial Layout Recovery

J. Omedes, G. López-Nicolás and J. J. Guerrero
Instituto de Investigación en Ingeniería de Aragón. Universidad de Zaragoza, Spain
jason.omedes@gmail.com, gonlopez@unizar.es, jguerrer@unizar.es

***Abstract*— In this work, we study the problem of recovering the spatial layout of a scene from a collection of lines extracted from a single indoor image. Equivalent methods for conventional cameras have been proposed in the literature, but not much work has been done about this topic using omnidirectional vision, particulary powerful to obtain the spatial layout due to its wide field of view. As the geometry of omnidirectional and conventional images is different, most of the proposed methods for standard cameras do not work and new algorithms with specific considerations are required. We first propose a new method for vanishing points (VPs) estimation and line classification for omnidirectional images. Our main contribution is a new approach for spatial layout recovery based on these extracted lines and vanishing points, combined with a set of geometrical constraints, which allow us to detect floor-wall boundaries regardless of the number of walls. In our proposal, we first make a 4 walls room hypothesis and subsequently we expand this room in order to find the best fitting. We demonstrate how we can find the floor-wall boundary of the interior of a building, even when this boundary is partially occluded by objects and show several examples of these interpretations.**

I. Introduction

Indoor structure recovery from images is an easy task for humans but not that easy for computers. At the same time, it is a very useful task since knowing floor-wall boundaries can give us valuable information for navigation, motion planning, obstacle detection or 3D reconstruction.

This problem has been studied several times and still attracts the effort of many researchers to implement each time better algorithms. Most of these contributions work under the Manhattan-World assumption [1], which assumes the scene is composed of 3 main directions orthogonal to each other. Indoor environment usually satisfies this condition so is understandable this hypothesis is extensively used. Some examples are [2], that uses extracted lines and geometric reasoning to generate hypothesis and select the best fit, or [3] which represents the room as a 3D box and tries to recognize floor-wall boundary in cluttered rooms. There are also other works as [4] that uses Bayesian filtering over a set of floor-wall boundary hypotheses without the restriction of Manhattan-World assumption, but there still being 3 main directions without the imposition of orthogonality between them.

Lately, research on omnidirectional vision is taking more importance due to the wide range of vision of these images, which helps in the detection of VPs and makes visible lines much longer. However, in central catadioptric images, straight lines from the real world become conics adding the issue of geometrical complexity, implying that many of existing algorithms for conventional images are not applicable. So it is needed to come up with new methods that take in account characteristics of this kind of images.

This work was supported by the Spanish project VISPA DPI2009-14664-C02-01 and FEDER funds.

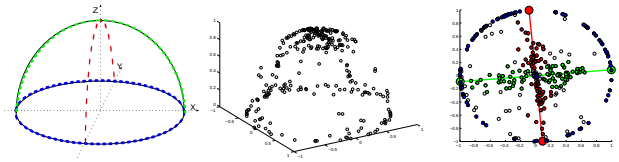


Fig. 1. In the sphere model, every line from the image is represented by its normal on the sphere. The figure represents the sphere where each point corresponds to a normal vector (Colorcode: X=Red, Y=Green, Z=Blue). From left to right: Sphere with perfect data; Sphere of a real image; classification of the previous data using our algorithm in the horizontal plane. Big dots represent VPs.

Here, we present our work for structure recovery from images. Starting from a single omnidirectional indoor image, we extract lines from it, classify them depending on their orientation. From this classification, we select a set of points which will lead us to generate possible wall-floor boundaries, and imposing geometrical constraints we generate a first 4 walls-room hypothesis to later on expand or not this room according on how the data is distributed.

Inspired by [5], we propose a new method. This approach is more robust since it does not rely in finding corners which often are not easy to detect, also we do not need to specify the number of walls we are looking for. In addition, it is much faster, as trying every possible combination of normals vector to classify the extracted lines or combination of corners to find the room hypothesis was high time-consuming. With our new approach we avoid all these long iterations making viable its use in a sequence of images at real-time.

II. Vanishing Point Estimation through Line Detection

The first step of our proposal begins with extraction of lines from the image. Regarding to line extraction for catadioptric systems two methods are [6] [7]. Both start using Canny edge detector and linking edge pixels. The main difference is that [7] works on the catadioptric image, where lines and VPs are extracted by RANSAC. Whereas [6] uses the unitary sphere model proposed in [8] where points from the image $\mathbf{p_I} = (X_o, Y_o, 1)$ are projected as

$\mathbf{p_S} = (X_S, Y_S, Z_S)$. By doing this, each chain of pixels from a line in the image defines a $great\ circle$ on the sphere which can be represented by its normal vector $\mathbf{n} = (n_x, n_y, n_z)$.

We use Bazin's Matlab toolbox[1], but adapting the equations from para-catadioptric ($\xi = 1$) to hyper-catadioptric ($0 < \xi < 1$) system in order to generalize the method for a more general mirror shape. From this point, [6] proposes to test every possible combination between pairs of normal vectors to identify main directions (1 vertical, 2 horizontal) at the same time as the 3 corresponding VPs. However, this method is time-consuming and sometimes comes up with misclassifications.

We propose a new robust and fast method to classify lines parallel to the 3 dominant directions, taking in consideration two hypotheses: **a**) Manhattan-world assumption [1] which states the scene is build on a cartesian grid, **b**) $\mathbf{Z}$ camera's axis is aligned with $\mathbf{Z}$ reference's axis of the world, since catadioptrical systems are mainly used in wheel-based robots, so planar-motion is assumed. It is easily demonstrable that under these assumptions and with perfect data, normal vectors $\mathbf{n} = (n_x, n_y, n_z)$ corresponding to the three different directions $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ does not has its own component (i.e. line from the real world belonging to direction $\mathbf{X}$, has a normal vector $\mathbf{n}$ whose component $n_x = 0$), Fig. 1. However, often data is not perfect so it will suffer deviations from this configuration. The classification process for these normals is done as follows:

1) Under assumption **b**, image lines whose normal vectors has n_z component below a threshold (experimentally we find 0.2 is a good value) are automatically classified as vertical lines, and removed for next steps.

2) Suppress n_z component of remaining normals so every $\mathbf{n} = (n_x, n_y, 0)$ will fall in a 2D plane, and using RANSAC we seek two orthogonal lines which minimize $\frac{error}{inliers}$ (with number of $inliers$ greater than a minimum). These lines will define the two horizontal main directions.

3) Image lines are labeled depending on the distance between its normal vector and one of the two main directions. It is remarkable that normal vectors whose component $n_z \simeq 1$ are conflictive as they are conics which degenerate into circles and can not be properly classified, so it is better remove them to avoid errors.

4) Finally, VPs are estimated as the points where the lines defining main directions cut the sphere at the hemisphere ($Z = 0$), see Fig. 1.

III. Hierarchical Layout Hypothesis Method

Due to noise and imperfections of real images often there is not enough information to clearly define where the floor-wall boundary is, so with the extracted lines and a set of geometric constraints we must seek the best approach to find where these boundaries are. In order to do this we generate conics (possible boundaries) from a set of points belonging to the lines previously classified.

[1] http://graphics.ethz.ch/~jebazin

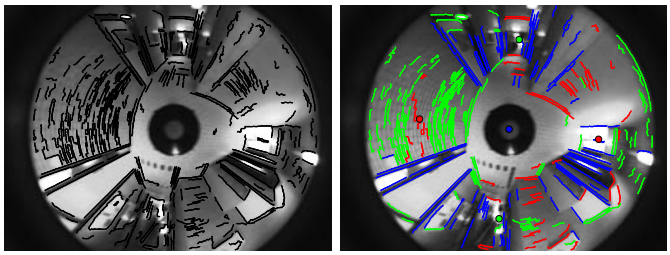


Fig. 2. Left: Lines extracted by Canny Edge detector after pruning step. Right: Same lines grouped in the 3 dominant directions according to our classification. Big dots represent VPs.

A. Selection of Set of Points

The first factor to notice is that information obtained by vertical lines is more robust and less susceptible to noise than horizontal lines, which sometimes are difficult to classify or are not well detected. Furthermore, studying typical images (see examples Fig. 6), we have noticed that in absence of objects most of vertical lines have their origin around the region that define floor-wall boundary, and if objects are present, they are standing over the floor and use to be close to the wall. So, unless these objects are placed all around the room, the origin of the lines that define them still being close to the desired boundary.

With these points of vertical lines we have to generate conics which will define possible floor-wall boundaries. In [9], it has been demonstrated how with a calibrated camera it is possible to define a conic in the image from only two points. If we apply the condition that every line in the image must pass through a vanishing point, just one point belonging to the floor-wall intersection is needed to define a conic in the image being a boundary.

Due to these 2 facts, let us denote as group G_Z the set of points composed by the closest point from each vertical to the center of the image. Additionally, we select a homogeneously distributed set of points from horizontal lines situated at the same height as the points in G_Z. Carrying out this selection for lines in X and Y direction, we obtain two more groups, G_X and G_Y, respectively. This is done in order to remove noisy horizontal segments, such as those found in objects, windows, doors,... and prevent them voting, Fig. 3(left).

B. Generation of Conics

Since we do not know which points of these groups are situated in the floor area, we apply RANSAC to identify the most voted conic, candidate to represent our desired boundary.

As we mentioned before, only two points are needed, one Vanishing Point and one point from the previous sets. Cross product between VP and each of these points $\mathbf{p_i}$ generates a normal vector $\mathbf{n_i}$, which defines a conic $\overline{\Omega}$ finally obtaining $\widehat{\Omega}$ after a projective transformation $\mathsf{H_C}$[10].

$$\mathbf{n_i} = \begin{pmatrix} n_{i_x} \\ n_{i_y} \\ n_{i_z} \end{pmatrix} = \begin{pmatrix} VP_x^S \\ VP_y^S \\ VP_z^S \end{pmatrix} \times \begin{pmatrix} P_{i_x}^S \\ P_{i_y}^S \\ P_{i_z}^S \end{pmatrix} \qquad (1)$$

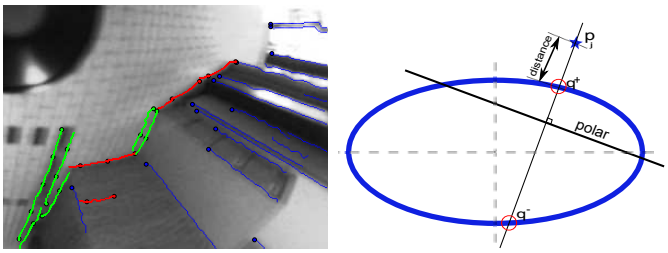


Fig. 3. Left: Selection of points as explained in Section III-A. Right: Graphic explanation for distance measurement between point and conic in Section III-B.

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

$$\widehat{\mathbf{\Omega_i}} = \mathsf{H_C}^{-t}\overline{\mathbf{\Omega_i}}\mathsf{H_C}^{-1} \quad (3)$$

Now, distance between conic and every point $\mathbf{p_j}$ is computed using an approximation [9] to [11]. We compute the polar line of a point $\mathbf{p_j}$ in the conic $\widehat{\mathbf{\Omega}}$, calculate the perpendicular specifying it lies over $\mathbf{p_j}$. This perpendicular line intersects the conic in two points $\mathbf{q}^+$ and $\mathbf{q}^-$, the minimum Euclidean distance between $\mathbf{p_j}$ and $\mathbf{q}^+$ or $\mathbf{q}^-$ corresponds to distance from point to conic, Fig. 3.

A new normal vector is estimated from the average of all points with minor distance than a threshold, and we iterate the whole process until its convergence (no more points are added). Points voting for this conic are removed from the list, and one of the remaining is chosen to generate an new conic, repeating the procedure and stopping when every point has been assigned.

C. Initial Boundaries Hypothesis

Computers cannot tell from a bunch of raw data how many walls a room is made of, but it is known that the most common indoor places are halls and rooms with similar shape to those shown in Fig. 4. All these geometrical shapes can be depicted by a central square with branches arising from all or some of its faces which at the same time must meet a geometric constraint: Parallel faces have to be one at each side of the imaginary line formed by joining their two corresponding VPs. This comes from the definition of vanishing point as the geometric place where parallel lines appear to converge.

Due to this constraint, the searching algorithm to extract conics (Section III-B) is executed for four different cases, in order to find the first four boundaries (better seen in Fig. 5):

- Boundaries 1 and 3: Are sought using points of G_Z and G_X at each sides of the imaginary line defined by VPs in direction X (Fig. 5 (left)) .
- Boundaries 2 and 4: Are sought using points of G_Z and G_Y at both sides of the imaginary line defined by VPs in direction Y (Fig. 5 (center)) .

Another property is that the four vanishing points define a conic which corresponds to points situated at the same height as the camera, so every point falling within the conic might be on the floor and will be a possible candidate, while the ones out of the conic are automatically eliminated.

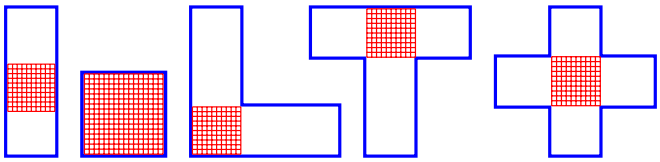


Fig. 4. Most common room/hall shapes (top view). Red grid represents the basic square we are seeking in section III-C.

Remaining points are projected onto the sphere for each of the four cases previously defined, and we proceed to generate conics with these points.

Conics more voted are now selected as possible candidates and the nearest to the center of the image is chosen. We rather chose this closest line to other which might be more voted because it will have chance to be selected in the expansion process (Section III-D), and if chosen now, it would be possible the loss of information. Once the four boundaries have been found, they are combined to conform walls and floor of our first hypothesis (Fig. 5).

D. Hierarchical Expansion Process

Let us denote as B_1, B_2, B_3 and B_4 the four boundaries defined in previous section III-C. The area between those and the end of the image defines four sectors. These sectors may correspond to actual walls or may exist the possibility they can be expanded, understanding $expand$ as replacing the boundary B_i for others which enlarge the area of the first-hypothesis room layout. For each of these sectors we repeat the same method described in Section III-C, obtaining a maximum of 3 new boundaries. Let them be B_i^L, B_i^M and B_i^R in clockwise order as shown in Fig. 7.

When looking for expansion three cases can happen:

- Enough data is available to define the 3 boundaries B_i^M, B_i^L and B_i^R; so there will be expansion in the current sector.
- B_i^M is very close to B_i, this means the most voted wall still being the same and will not be expansion.
- Data only allow us to find 1 or 2 boundaries. This last case can be originated for different situations and should be studied.

Third case is present when we lack of data, caused by lines not detected or by an occluded corner (Fig. 7(left)). Both cases imply expanding floor area but results are completely different, therefore care must be taken.

If missing boundary is a lateral (B_i^L, B_i^R) or lateral plus middle (B_i^M), and points from well-detected boundaries only fall at one side from the VP within the current sector, this is due to an occluded corner. Thus the missing border is defined as a radial line through the center of the image and the point, belonging to the well-detected boundary, whose angle is the closest to the angle defined by the VP.

On the other hand, if previous conditions are not satisfied, we assume some line was not detected, hence if the missing boundary is any of B_i^L or B_i^R, it will be defined as the

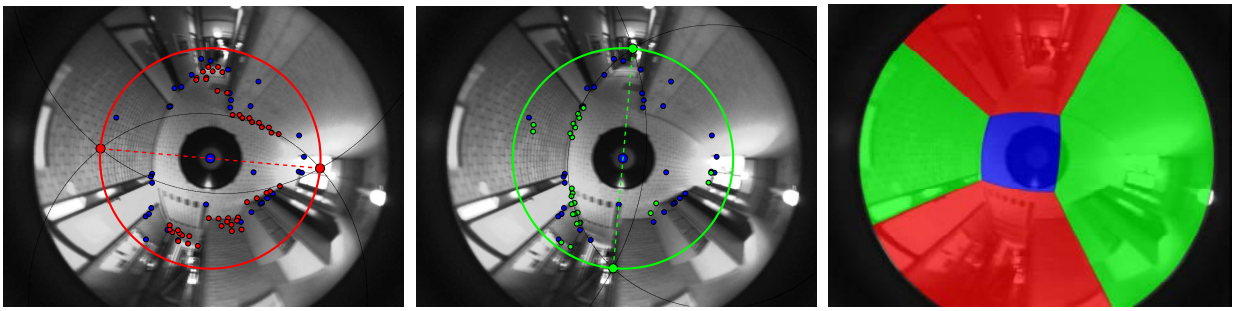


Fig. 5. First two images show points from groups G_Z, G_X and G_Y under constraints exposed in section. III-C, where blue, red and green dots correspond to G_Z, G_X and G_Y points respectively. Dashed red and green lines are the imaginary lines, going through the VPs, which divide the image in 2 parts. Finally, black conics represent the most voted boundaries for each case. Right image shows the result of combining those boundaries to generate the first hypothesis.

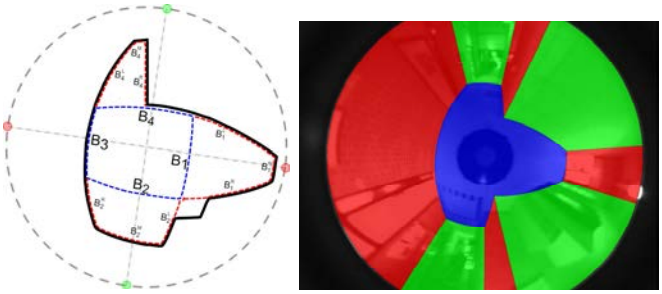


Fig. 7. Left: Synthetic example depicting the possible cases (B_1 and B_2 are expandable regions, B_3 will not be expanded, and B_4 corresponds to an occluded corner). Black line represents the actual room boundaries, first hypothesis in dashed blue, and final expansions in dashed red. Right: final result of a real example.

resulting conic passing through its corresponding VP and the last point belonging to B_i^M. By contrast, if the missing edge is B_i^M, we consider it should be no expansion except if there are a relevant number of voting points in the lateral boundaries B_i^L and B_i^R.

IV. Results

Our experiments have been performed using Matlab, running at 3 sec per frame, and with the dataset COGNIRON composed of indoor images (768×1024) with a wide variety of rooms. These images were taken by a camera with a hyperbolic mirror spotted on a mobile robot. The calibration of the camera is also available online [12].

We show some of our results in different kind of indoor situations Fig. 6. First two examples correspond to T and L shape halls (like the ones shown in Fig. 4), walls are not too saturated with objects so the result is accurate. In this second example we also observe an occluded corner at the superior part of the image. Third picture is taken in a room where walls are made of glass (top and bottom of the image); due to these walls very bright areas appear in the scene, but we still achieve a good approximation of its structure.

Forth case shows 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.

	Image1	Image2	Image3	Image4	Image5
Precision	0.973	0.984	0.896	0.964	0.904
Recall	0.887	0.969	0.992	0.937	0.878
F_1	0.928	0.977	0.942	0.950	0.891

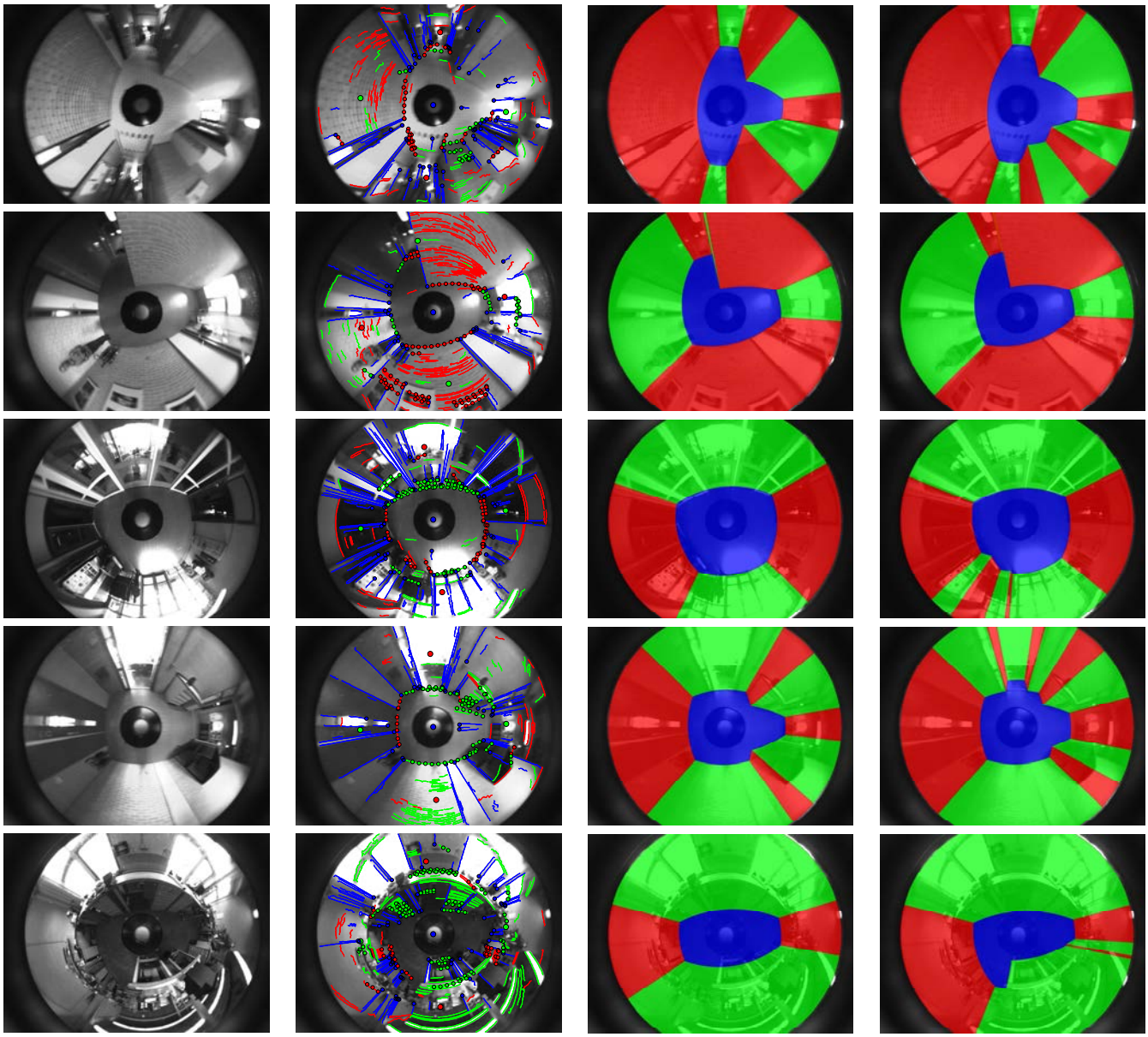
TABLE I
Performance values obtained for images of Fig. 6

Last scene correspond to a room with many objects, colors are very dark, which makes difficult line extraction at some areas. At the same time, most of the longest detected lines fall over the objects, what might lead to misclassify wall-floor boundaries, but as we can see, we are still achieving good results.

Comparing results from our algorithm with their respective ground truth, we define as true positives (tp) the number of pixels both have in common, false positives (fp) the number of pixels identified as floor by our method but do not correspond to the floor in the ground truth, and false negatives (fn) the number of pixels identified as not floor when they result to be floor in the ground truth. With these values we compute **precision** ($\frac{tp}{tp+fp}$), **recall** ($\frac{tp}{tp+fn}$) and $\mathbf{F_1}$ ($\frac{2\,precision\,recall}{precision+recall}$) for several images, Table I.

V. Conclusion and Future Work

We have proposed a new method to extract vanishing points from an omnidirectional image and to perform line classification. Since for the database of images the Z axis from camera and world reference were aligned, our proposal have been designed with this constraint, but we believe this can be extended for the general case where the alignment of axis is not known. We also have proposed a new simple and robust method for scene layout recovery and we have shown its performance in experimental results. This is useful in many applications, since knowing where floor-wall boundary are and the height of the camera, we can known exactly where every point of the scene is. Currently we are working into spread out this method for a whole sequence of images in order to improve accuracy in those images with possible misclassifications.



(a) (b) (c) (d)

Fig. 6. Examples of experimental results obtained for five different images. (a)Input images. (b) Line classification and extracted points which vote for boundary selection. (c) Output images by our algorithm. (d) Ground truth, manually labeled.

REFERENCES

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

[2] D. Lee, M. Hebert, and T. Kanade, "Geometric reasoning for single image structure recovery," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 2136–2143.

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

[4] G. Tsai, C. Xu, J. Liu, and B. Kuipers, "Real-time indoor scene understanding using bayesian filtering with motion cues," in *ICCV*, 2011, pp. 121–128.

[5] N. D. Ozisik, G. López-Nicolás, and J. J. Guerrero, "Scene structure recovery from a single omnidirectional image," in *ICCV Workshops*, 2011, pp. 359–366.

[6] J. C. Bazin, I. Kweon, C. Demonceaux, and 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.

[7] J. Bermudez, L. Puig, and J. J. Guerrero, "Line extraction in central hyper-catadioptric systems," in *OMNIVIS*, 2010.

[8] C. Geyer and K. Daniilidis, "A unifying theory for central panoramic systems and practical applications," in *ECCV (2)*, 2000, pp. 445–461.

[9] J. Bermudez-Cameo, L. Puig, and J. J. Guerrero, "Hypercatadioptric line images for 3d orientation and image rectification," *Robotics and Autonomous Systems*, vol. 60, no. 6, pp. 755–768, 2012.

[10] J. Barreto, "General central projection systems: Modeling, calibration and visual servoing," Ph.D. dissertation, 2003.

[11] P. Sturm and P. Gargallo, "Conic fitting using the geometric distance," in *Proceedings of the Asian Conference on Computer Vision, Tokyo, Japan*, 2007, pp. 784–795.

[12] Z. Zivkovic, O. Booij, and B. Krose, "From images to rooms," *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 411–418, 2007.