Topological map from only visual orientation information using omnidirectional cameras

L. Puig, J.J. Guerrero and Kostas Daniilidis

***Abstract*— In this paper we present a new way to compute a topological map using only orientation information. We exploit the natural presence of lines in man-made environments in dominant directions. We extract all the image lines present in the scene acquired by an omnidirectional system composed of 6 aligned cameras. From the parallel lines we robustly compute the three dominant directions using vanishing points. With this information we are able to align the camera with respect to the scene and to identify the turns in the trajectory. Assuming a Manhattan world where the changes of heading in the navigation are related by multiples 90 degrees. We also use geometrical image-pair constraints as a tool to identify the visual traversable nodes that compose our topological map. Experiments with an indoor sequence have been performed to validate this approach.**

I. INTRODUCTION

In the field of robotics the representation of the space plays a very important role. In general this representation allows the robot to perform different tasks to interact with the environment. Among these tasks we can mention localization, path planning, navigation, etc. Along the years several representations have been proposed but Kuipers [5] proposes the Spatial Semantic Hierarchy where four levels are considered. The two most used in the literature are the metric and the topological maps. The metric maps are quantitative representations of the environment. This representation usually uses raw data or lines and has some disadvantages. It requires accurate determination of the robot position, it is inefficient for planning, the resolution does not depend on the complexity of the environment [10]. On the other side the topological maps are purely qualitative, and many of its benefits are independent of the accuracy or even the existence of quantitative knowledge of the environment. These characteristics make the topological maps robust to poor odometry and position errors. Topological approaches represent the environment using a graph structure where nodes represent different places in the world and edges denote traversable paths between them [2]. Furthermore, the elements of the topological map are strongly related to the semantics of the environments.

There exist several approaches that deal with the automatic generation of topological maps. The difference between them depends on the method and the sensors used. In [9] they use two laser range finders and one omnidirectional camera. They propose the concept of fingerprints to characterize the places visited by the robot and the partially observable Markov decision processes (POMDP) for global localization. In [2] they try to identify the loop closings. This happens when the robot revisits a place. More specifically, an equivalent sensor reading occurs twice in the sequence. They use a single omnidirectional sensor and the Dempster-Shafer theory to model the uncertainty. In [8] they use a panoramic sensor and two different types of features, the Kanade-Lukas-Tomasi (KLT) and 3D color histograms. The map is modeled as a physics-based mass and spring system. More recently [1] propose an appearance based topological map. They also use an omnidirectional system combined with the planar motion constraint and the epipolar geometry. They use the number of inliers and outliers to define a similarity value that links different locations. These are the edges of the topological map.

In man-made environments the presence of straight lines is common [7]. Moreover, these lines are aligned with the principal orthogonal directions of the world coordinate frame [4]. From the images of parallel lines we can compute the orientation of the camera with respect to the scene. With the relative orientation of the camera and the guaranty that frames are reachable to each other and the temporal ordering constraints given by a video sequence, we decide to explore the creation of a topological map using only orientation information. With the use of lines we overcome some problems of feature-based approaches, such as lack of texture in the scene or the similarity of features, which is commonly observed in long corridors.

In this paper we use an omnidirectional camera composed of 6 aligned wide angle cameras. Even these aligned cameras share a part of their field of view, this shared area is not enough to compute the rotation of the omnidirectional camera from a 3D line-based structure from motion approach. In opposition we exploit the presence of straight lines in man-made environments which are more robust than feature points. From the images of parallel lines we compute the vanishing points which provide the relative orientation of the camera with respect to the scene. We propose to combine the orientation information with the epipolar geometry to build a topological map in an indoor environment. We extract the orientation of each frame with respect to the scene using the vanishing points. The changes of direction are detected when two consecutive frames have a drastic change on their orientation. We use a feature matching, particularly SIFT [6]

Luis Puig and Josechu Guerrero are with the Departamento de Informática e Ingeniería de Sistemas (DIIS) e Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, Zaragoza, Spain {lpuig, jguerrer}@unizar.es

This research has been funded by the Dirección General de Investigación of Spain under project VISPA DPI2009-14664-C02-01.

Kostas Daniilidis is with the GRASP Laboratory, University of Pennsylvania, 3330 Walnut Street, L402, Philadelphia. kostas@cis.upenn.edu

Proceedings of the 2nd. Workshop on Omnidirectional Robot Vision
A workshop of the 2010 IEEE International Conference on Robotics and Automation (ICRA2010)
Anchorage, Alaska, USA, May 7, 2010, ISBN 978-88-95872-02-5
pp. 19-25

features, between these frames to identify the direction of the turn. Finally, the epipolar geometry is used to identify the frames that represent the nodes in the topological map. We compute the essential matrix between every n frames from which we extract the rotation component and compare to that obtained from the visual compass. If they are coherent we recompute the essential matrix with the next frame. The process stops when either the rotations are not coherent or the essential matrix computation fails. We choose the last frame as a new node in the topological map and the process restarts.

The rest of the paper is divided as follows. In section II we present in more detail our proposal. In section III we present some experiments with real images where we show the performance of our approach, detecting the turns. Finally in section IV we present the conclusions and future work.

II. Our Approach

In man-made environments the presence of structured data such as lines is common. They are useful in scenes where texture is not enough to acquire classical features, needed for example, to perform a correspondence-based matching, from which position and orientation is obtained. In this kind of environments the use of lines overcomes this problem. In this work we use lines to compute the orientation of the camera with respect to the scene. From only the absolute orientation with respect to the world and the relative two-view orientation of the camera we compute our topological map.

A. Computing the relative orientation

In our approach we compute the relative orientation of the camera with respect to the scene based on the computation of the vanishing points. The vanishing points are computed using a voting scheme. The first step consist of extracting the segments of lines $\mathbf{l}_i$ present in every single image. The Canny extractor is used to extract the edges, then a link function is used to compute the connected components. The two endpoints of one of these connected components are $\mathbf{x}_1, \mathbf{x}_2$. The second step is to compute the putative vanishing points. The line segment passing through the two endpoints is represented by a plane normal of a plane passing through the center of projection and intersecting the image in a line l, such that $\mathbf{l} = \mathbf{x}_1 \times \mathbf{x}_2$. The unit vectors corresponding to the plane normals $\mathbf{l}_i$ can be viewed as points on a unit sphere. The vectors $\mathbf{l}_i$ corresponding to parallel lines in 3D world all lie in one plane. The vanishing direction then corresponds to the plane normal where all these lines lie. Given two lines the common normal is determined by $\mathbf{v}_m = \mathbf{l}_i \times \mathbf{l}_j$. Then, for every pair of plane normals we compute putative vanishing points $\mathbf{v}_m$. The number of total putative vanishing points corresponding to n plane normals are $(n(n-1)/2)$. The third step consist of computing the distance between the putative vanishing points and all the plane normals. Since in the noise free case $\mathbf{l}_i^\mathsf{T}\mathbf{v}_m = 0$, we use this product as a measure of error of the line $\mathbf{l}_i$ pointing in the dominant vanishing direction $\mathbf{v}_m$. The plane normals with a distance smaller than some threshold vote for the corresponding vanishing point. The most voted is chosen as the first vanishing point $\mathbf{v}_i$. Then we look for the second most voted and orthogonal to the first vanishing point $\mathbf{v}_j$. Then we look for the third most voted vanishing point and orthogonal to the other two $\mathbf{v}_k$. If just two vanishing points are computed, the third one is obtained by computing the cross product of the other two $\mathbf{v}_k = \mathbf{v}_i \times \mathbf{v}_j$.

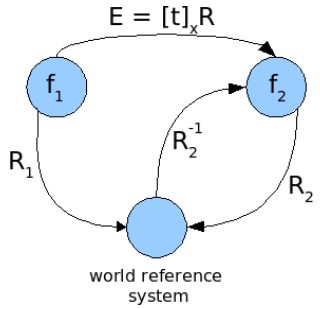


Fig. 1. Relation between the rotation matrices from Essential matrix and from vanishing points.

Since the vanishing points are projections of the vectors associated with three orthogonal directions *i, j, k*, they depend on rotation only. In particular we can write that

$$\mathbf{v}_i = \mathsf{R}e_i \quad \mathbf{v}_j = \mathsf{R}e_j \quad \mathbf{v}_k = \mathsf{R}e_k.$$

From matrix R we are able to extract the orientation angles corresponding to the three main axis (α, β, γ).

B. Topological Map

Topological maps try to simplify the representation of the environment by modelling space using graphs. This representation is suitable when we just need to know if a new place is reachable from our current position. In this case the metric information of the environment is not needed. Depending on the task, topological or metric maps are required to solve different problems. In this section we explain how to construct a topological map from orientation only information. In this work we represent the frames as nodes and edges meaning visual traversality between such frames.

1) Keyframes: The computation of the keyframes is performed in two steps. The first step is to identify the keyframes from only orientation information. The frame where a change of direction, i.e. a turn, is detected and added to the topological map automatically. The change of direction of the motion axis is detected when a change of approximately $\pi/2$ is observed in the magnitude of the angle of the main horizontal axis in two consecutive frames. However, this change can also be observed in other circumstances. To avoid false detection of turns we compute the visual traversality test between the fifth frame prior to and the fifth frame after the possible turn frame. The visual traversality test is explained in detail below. If these two frames are not connected, the turn has been performed and the turn frame is added to the topological map. This vanishing points based

Proceedings of the 2nd. Workshop on Omnidirectional Robot Vision
A workshop of the 2010 IEEE International Conference on Robotics and Automation (ICRA2010)
Anchorage, Alaska, USA, May 7, 2010, ISBN 978-88-95872-02-5
pp. 19-25

approach gives us the existence of a turn of 90° but the projective information does not give the direction of turn. In order to identify the direction of the turn ($+90^\circ$ or -90°) we measure the average motion in pixels of the features in the two analyzed frames. If the features move to the left the motion performed by the camera is to the right and viceversa. When a turn is performed, the direction of motion is assigned to the new one that is detected. This means that the two horizontal axis are switched, an event that is recorded only in the topological map.

In the second step we compute the visually traversable nodes of the topological map. To decide if two frames are connected (visually traversable test), we compute the essential matrix $\mathsf{E} = [\mathbf{t}]_\times \mathsf{R}$ between the frames f_1 and f_2 using a correspondence-based approach. SIFT points are the most used local and robust features and we use them in this work. From the essential matrix we extract the rotation matrix R and translation vector $\mathbf{t}$ as explained in [3]. Then we compare the rotation matrix $\mathbf{R}$ to the one obtained from the combination of the rotation matrix of f_1 and f_2 with respect to the scene, $\mathsf{R}_c = \mathsf{R}_1 \mathsf{R}_2{}^\mathsf{T}$, respectively. (See Fig. 1). If both matrices are congruent and the number of SIFT correspondences is bigger than a threshold, the two frames are connected. In the real world, the interpretation is that the space between the two frames is visually traversable. In this case we increment f_2 by n frames and compute again the essential matrix. When the computing of the essential matrix is not possible, that means the space between f_1 and f_2 is not visually traversable. In this case we add the frame f_2 as a visually traversable node to the topological map. We restart the computation of the essential matrix with the node f_2 as the initial image and look for the next visually traversable node as explained before.

III. Experiments

In this section we present the experiments using an indoor sequence acquired by an omnidirectional camera composed of 6 perspective cameras[1]. We use a total of 17,000 frames (3400 by each camera, the top sensor is not used). The trajectory performed involves several turns inside a building. The calibration of the individual cameras and the alignment matrices of the cameras with respect to the camera head coordinate frame are given by the manufacturer.

A. Orientation computation

As we mentioned before the first step of our approach is to compute the vanishing points. In Fig. 2 we observe the extraction and classification of the lines from the cameras used by the omnidirectional system. Each color represents a direction in the scene. From the three vanishing points obtained we extract the rotation angles for each axis. As the camera is moving on a platform an almost planar motion is performed. The motion around the z-axis is the main motion observed through the trajectory.

In Fig. 3 we show the computation of the rotation angle corresponding to the main axis of the omnidirectional camera

[1]Ladybug 2 http://www.ptgrey.com/

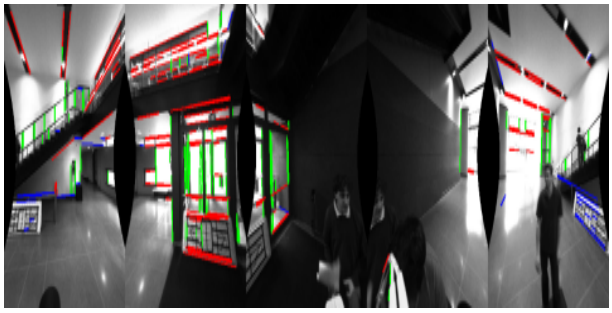


Fig. 2. Lines corresponding to the computed vanishing points.

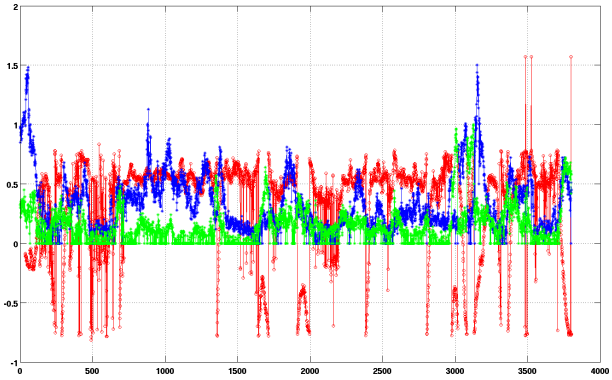
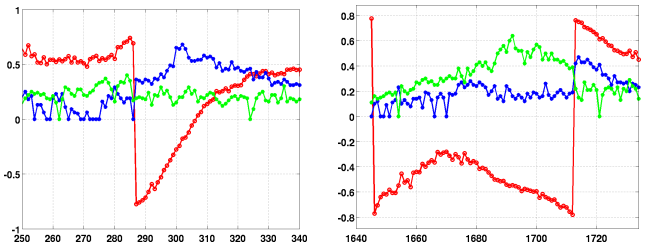


Fig. 3. Rotation angle of the x-axis through the whole sequence in radians (red). Number of lines scaled by 0.01, supporting the two horizontal directions of the camera. x-axis (blue) and y-axis (green). The horizontal axis of the plot represents the frame number (we recommend to see the color version of the paper).

and the number of lines (scaled by 100) supporting the two main directions of this system. We observe changes of magnitude of $\pi/2$ in consecutive frames. This displacement indicates a possible change on the axis of motion. A few examples of these possible turns are the frames 287, 686, 1357, 3018 and 3081. We analyze two particular cases, given by frames 287 and 1645. In the first case (Fig. 4(a)), we observe an orientation change of 1.46 radians. In this case a turn has been performed, from left to right. We observe that the number of lines supporting the main axis decreases (blue) while the number of lines supporting the other non-



(a) (b)

Fig. 4. (a) Turn performed in the trajectory. (b) Change of orientation with respect to the scene. Corresponding rotation angle of the x-axis in radians (red) to the frame number indicated in the horizontal axis of the plot. In blue, number of lines supporting the x-axis vanishing direction. In green, number of lines supporting the y-axis vanishing direction. Both numbers are scaled by 0.01.

Proceedings of the 2nd. Workshop on Omnidirectional Robot Vision
A workshop of the 2010 IEEE International Conference on Robotics and Automation (ICRA2010)
Anchorage, Alaska, USA, May 7, 2010, ISBN 978-88-95872-02-5
pp. 19-25

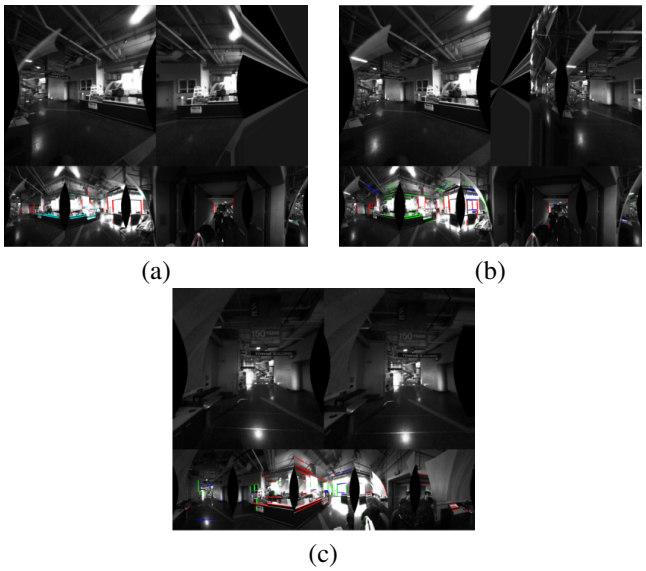


Fig. 5. Change of direction corresponding to a turn detected in two consecutive frames (a) and (b). (c) frame aligned to the new scene, the turn has been completed (0.064 radians).

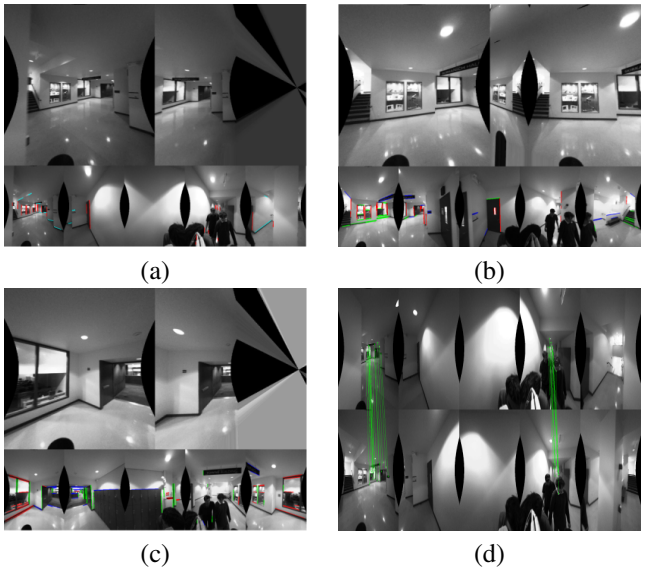


Fig. 6. Change of direction that does not correspond to a turn. (a) frame 1645. (b) frame 1672. (c) frame 1713. (d) Visual traversality test between frames 1640 and 1650.

vertical orthogonal axis increases (green) and at the frame where the turn is detected a switch is performed. In Fig. 5 we show the two frames where the turn was detected and a frame where the camera is aligned to the new position in the scene, i. e. when the orientation angle of this frame with respect to the scene is close to zero. In this particular case the turn has been performed in approximately 20 frames. To detect the turns performed in the opposite direction (right-to-left), we observe the changes of orientation of magnitude close to $\pi/2$. From negative to positive values.

The second case analyzed corresponds to the 1645 frame (Fig. 4(b)) where a change of 1.53 radians has been observed. In this case the radical change on the orientation does not correspond to a turn. The angle between the main axis of the camera reference system and the motion axis is bigger than $\pi/4$. In this case, the camera is pointing to a different direction from where the motion is performed. A characteristic of this behavior is that the camera is not aligned to the scene, i.e. its orientation is not close to zero. The end of this phase is identified when there is a change in the orientation from a negative value to a positive one with a magnitude close to $\pi/2$ (see Fig. 4(b) frame 1713). In this case we also observe how the number of lines supporting the two non-vertical directions switches when the radical change of orientation is observed. The start, middle and ending steps of this phase can be observed in Fig. 6(a)(b) and (c). Fig. 6(d) shows the SIFT correspondences between the frames 1640 and 1650. The number of correspondences indicates that the two frames are connected. Therefore this change on the orientation does not correspond to a turn.

B. Drawbacks of the approach

It is possible that the lines present in the scene are not aligned with the principal orthogonal directions of the world coordinate frame. In this case the estimation of the vanishing points is not reliable. We can observe this situation in Fig. 3, from frame 200 to frame 283. An example of the lines detected in this situation can be observed in Fig. 7.

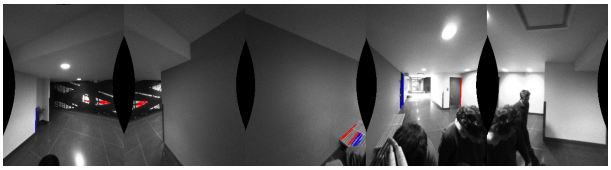


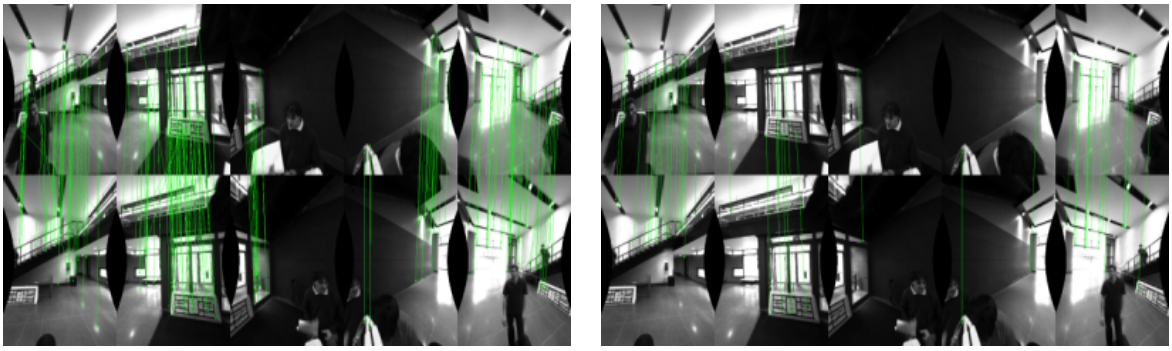
Fig. 7. Bad estimation of the vanishing points in frame 200.

C. Building Topological Map

In order to build a topological map we need to identify the keyframes that represent the nodes. Above we explain how to identify the turn frames. These frames are added to the topological map. The rest of keyframes are computed using an essential matrix to identify the visual traversable frames. The essential matrix is computed using a feature based approach. We extract the SIFT points from two frames. To have a better distribution of the points we use a bucketing in every image. We match these well distributed points to have the first putative correspondences (see Fig. 8(a)). The presence of outliers is inevitable. In that order we use a robust approach with a geometric constraint to avoid the outliers Fig. 8(b). With eight correspondences we compute the essential matrix. From which we extract the rotation matrix.

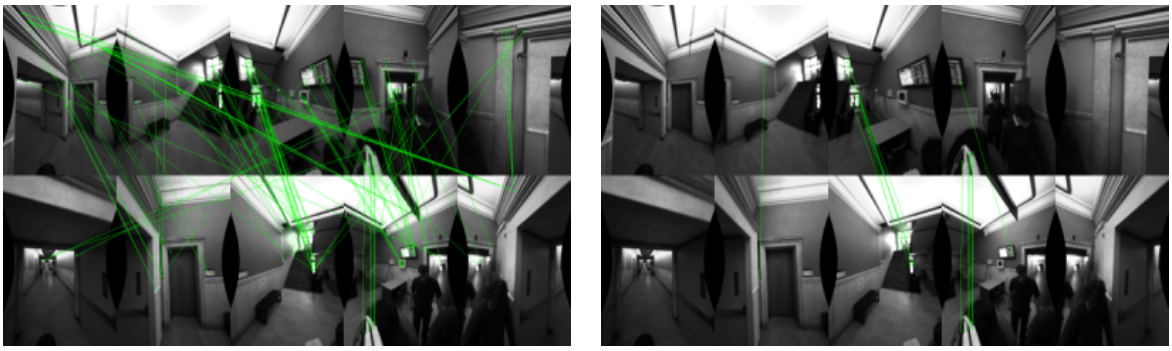
From the rotation matrices corresponding to the two frames we extract the rotation angles corresponding to each axis. If the difference between the angles is greater than a predetermined threshold or the number of matches is less than 10, these frames are not connected. In Fig. 9 we observe two non-connected frames. In this case the second frame is selected as a keyframe and it is added to the topological map.

Proceedings of the 2nd. Workshop on Omnidirectional Robot Vision
A workshop of the 2010 IEEE International Conference on Robotics and Automation (ICRA2010)
Anchorage, Alaska, USA, May 7, 2010, ISBN 978-88-95872-02-5
pp. 19-25



(a) (b)

Fig. 8. Connected frames. Frames 35 and 45. (a) Putative matches. (b) Robust matches.



(a) (b)

Fig. 9. Non-Connected frames. Frames 675 and 685. (a) Putative matches. (b) Robust matches.

When the frames are connected we recompute the essential matrix with the next frame and repeat this action until we find a couple of non-connected frames. Fig. 8 shows two frames that are connected.

As we mentioned before we detect the turns in the trajectory (Fig. 5). To identify the direction of the turn we verify the motion of the features between the two frames where the turn was detected.

An example of the topological map created by our approach can be observed in Fig. 10. The red circles are the detected turns and the green ones are the visually traversable frames computed by the epipolar geometry. The distance between the nodes is related to the number of frames that separates them. We also observe the corresponding orientation of the frame with respect to the scene. We take the orientation of the first frame as the world reference system. The trajectory corresponds to a loop. The initial and final keyframes are not close since we do not have the correct scale to plot the map. We observe that the initial and final frames have a similar orientation. The blue circles represent areas where a change of orientation is detected but it does not correspond to a turn (see Fig. 4(b)). As we observe the direction of the motion is not changed in these cases.

From the constructed topological map and the orientation information we observe that it is possible to detect corridors and to perform loop-closure.

1) Corridor detection: Assuming a continuous motion from the image sequence. If turns are not detected. This means the orientation of the omnidirectional system is contained inside of certain range. Moreover, if the number of lines supporting the non-vertical dominant direction is clearly bigger than the rest. With this information we can infer that the camera is traversing a corridor. In Fig. 11(a) we observe how the angle (in red) is inside the range $(0.4, 0.7)$ radians. It starts at frame 716 and finishes at frame 1357. The number of lines supporting this dominant direction (blue line) is clearly superior to the rest (green line). In Fig. 11(b,c) we show the initial and the final frames where the corridor is detected.

2) Loop-Closure detection: In order to detect the loop-closure we compute the essential matrix between the actual keyframe and all the previous keyframes stored in the topological map. A loop-closure is detected when the two keyframes have the same orientation and the number of correspondences between the two frames is bigger that a previously defined threshold. In the sequence used just one loop is present. In Fig. 12 we observe frame 287 (-0.77 radians) and frame 3365 (-0.74 radians) that besides having a similar orientation they have 46 correspondences validated by the epipolar geometry.

3) Traversable area detection: In order to analyze the situation of the keyframes with respect to the scene we compute the position of the lines present on the scene. We observe that the position of lines perpendicular to the main axis of the omnidirectional camera indicate if it is possible to move in that direction. We divided the space in four big areas corresponding to the main directions, up, down, left and

Proceedings of the 2nd. Workshop on Omnidirectional Robot Vision
A workshop of the 2010 IEEE International Conference on Robotics and Automation (ICRA2010)
Anchorage, Alaska, USA, May 7, 2010, ISBN 978-88-95872-02-5
pp. 19-25

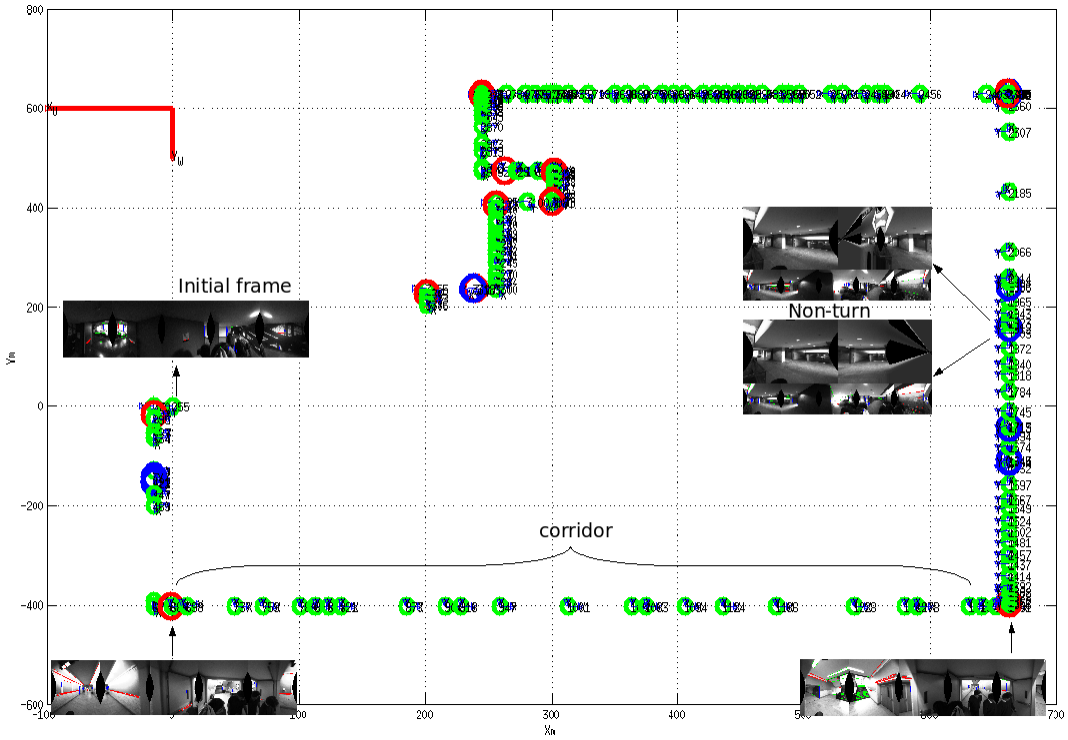


Fig. 10. Topological map of the indoor sequence.

right. We count the number of lines inside each area for each dominant direction. In Fig. 13 we show an example where the number of perpendicular lines (green) in the left cell indicates that it is possible to move into that direction. As we can see in Fig. 13(a), those lines are pointing to the corridor where the camera come from. Therefore, it is actually possible to move in that direction.

IV. Conclusions

We have presented a new approach to compute a topological map in an indoors environment using an omndirectional camera. We exploit the presence of straight lines to compute the vanishing points, from which we estimate the orientation of the camera with respect to the scene. From only orientation information we are able to detect turns in the trajectory and with an initial reference system we are able to construct a coherent topological map. We use geometrical constraints, particularly the essential matrix, to select the keyframes that represent the nodes in the topological map. The nodes are visually traversable to each other. We performed experiments with a sequence of real images in an indoor environment. We observe that the use of only orientation information and the epipolar geometry as a tool give a coherent construction of a topological map.

References

[1] O. Booij, B. Terwijn, Z. Zivkovic, and B. Krose. Navigation using an appearance based topological map. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3927–3932, April 2007.

[2] Toon Goedemé, Tinne Tuytelaars, and Luc J. Van Gool. Visual topological map building in self-similar environments. In *ICINCO-RA*, pages 3–9, 2006.

[3] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.

[4] Jana Kosecka and Wei Zhang. Video compass. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part IV*, pages 476–490, London, UK, 2002.

[5] Benjamin Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233, 1999.

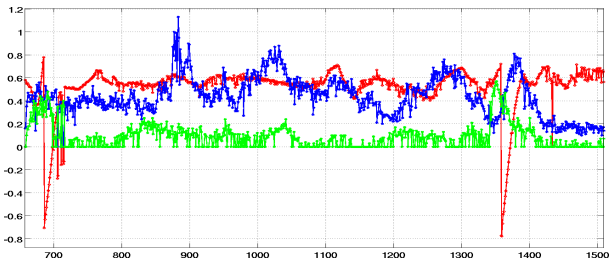
[6] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2004.

[7] A. C. Murillo, C. Sagues, J. J. Guerrero, T. Goedemé, T. Tuytelaars, and L. Van Gool. From omnidirectional images to hierarchical localization. *Robotics and Autonomous Systems*, 55(5):372–382, 2007.

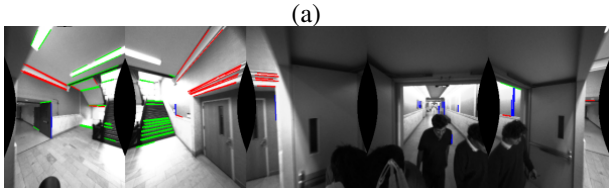
[8] Paul E. Rybski, Franziska Zacharias, Jean-Francois Lett, Osama Masoud, Maria Gini, and Nikolaos Papanikolopoulos. Using visual features to build topological maps of indoor environments. In *In Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, pages 850–855, 2003.

[9] Adriana Tapus, Nicola Tomatis, and Roland Siegwart. Topological global localization and mapping with fingerprints and uncertainty. In *ISER*, pages 99–111, 2004.

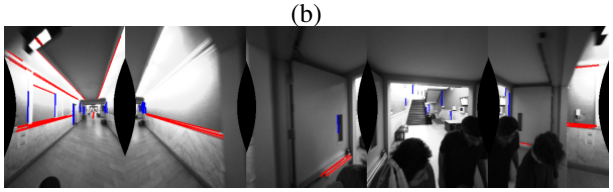
[10] Sebastian Thrun and Arno Bücken. Learning maps for indoor mobile robot navigation. *Artificial Intelligence*, 99:21–71, 1998.



(a)



(b)



(c)

Fig. 11. Corridor detection. (a) Detection in the sequence showing the number of lines scaled by 0.01, supporting the two main directions of the camera. x-axis (blue) and y-axis (green). (b) Initial frame. (c) Final frame.

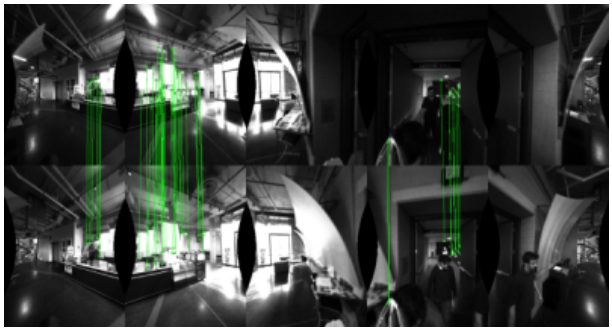
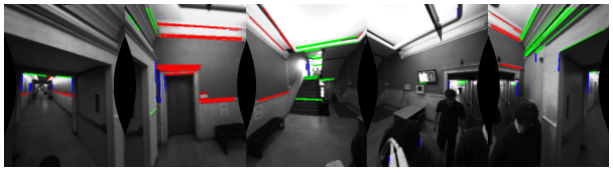
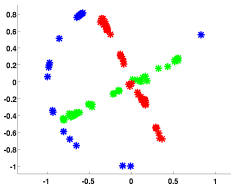


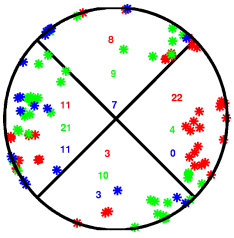
Fig. 12. Frames 287 and 3365 where the loop-closure is detected.



(a)



(b)



(c)

Fig. 13. Identification of traversable areas. (a) Frame 690 with lines supporting the dominant directions $x-$axis (red), $y-$axis (green) and$z-$axis (blue). (b) Space of lines represented by their normals and point to the corresponding vanishing points. (c) Lines spread on the four displacement areas (up, right, down and left).

Proceedings of the 2nd. Workshop on Omnidirectional Robot Vision
A workshop of the 2010 IEEE International Conference on Robotics and Automation (ICRA2010)
Anchorage, Alaska, USA, May 7, 2010, ISBN 978-88-95872-02-5
pp. 19-25