The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

An Original Approach for Automatic Plane Extraction by Omnidirectional Vision

Jean-Charles Bazin, Pierre-Yves Laffont, Inso Kweon, Cédric Demonceaux and Pascal Vasseur

Abstract— **Whereas some methods for plane extraction have been proposed, this problem still remains an open issue due to the complexity of the task. This paper especially focuses on the extraction of points lying on a plane (such as the ground and buildings walls) in sequences acquired by a central omnidirectional camera. Our approach is based on the epipolar constraint for planar scenes (i.e. homography) on a pair of omnidirectional images to detect some interest points belonging to a plane. Our main contribution is the introduction of a new method, called "2-point algorithm for homography", that imposes some constraints on the homography using vanishing point (VP) information. Compared to the widely used DLT (4-point) algorithm, experiments on real data demonstrated that the proposed "2-point algorithm for homography" is more robust to noise and false matching, even when the plane to extract is not dominant in the image. Finally, we show that our system provides key clues for ground segmentation by GrabCut.**

I. Introduction

Image segmentation is a key issue in computer vision. Basically, segmentation aims to partition an image into multiple regions and has been applied for various tasks such as surveillance and object recognition. The goal of this paper is to automatically segment/extract some (usually sparse) feature points lying on a plane, such as the ground and building walls, in omnidirectional images. Extracting these planar points can be applied for Unmanned Aerial Vehicles (UAV) landing, autonomous robot navigation and 3D reconstruction for example. From these sparse points, it is also possible to obtain a dense pixelwise segmentation of the detected planes [1][2]. In this paper, the expressions "plane extraction" and "plane segmentation" are used equivalently and refer to the extraction of these feature points. Independently of the vision system (monocular/stereo, traditional/catadioptric, color/grayscale), plane segmentation is a complicated problem for several reasons: different textures, appearance or luminosity changes (shadow). Most of the existing works for plane extraction are dedicated to ground plane. Previous works can be classified into 4 main categories.

The first category concerns the segmentation in a single image using color and texture information and numerous methods have been proposed [3][4]. These techniques are usually easy to implement but require a color model of the ground and thus cannot be applied in varied environments (indoor/outdoor, bright/dark).

The second category refers to epipolar geometry on monocular image sequence. [5] estimates the homography that leads to the highest number of point inliers by applying RANSAC in an image pair. The plane associated to the best homography is considered the ground plane. However this kind of approaches assume that the ground plane (more generally, the plane to extract) is the dominant plane in the image. It usually leads to a sparse segmentation since only point features of the ground are extracted. [6] computes the dominant optical flow in an image sequence and all the points verifying this flow are considered ground points. However it assumes the ground is the largest region in the image and the camera has a small displacement.

The third category is based on plane detection in 3D data provided by a calibrated stereo camera system. After reconstructing 3D points from a stereo camera system, [7] applies Hough Transform to detect the dominant plane. [8] performs RANSAC using three 3D points to compute the best plane with respect to the number of points that are close to the plane. However these two methods still assume that the ground plane contains the largest number of features. At the contrary, [9] uses an inertial measurement unit (IMU) in order to highly constrain the plane orientation and therefore extract the ground plane more efficiently. However this method requires both a stereo camera and an IMU and, last but not least, the knowledge of a point on the ground plane which is usually manually selected.

The fourth and last category combines epipolar geometry and color information. The method of [7] consists of two steps. First, 3D points are reconstructed from a stereo camera and the dominant plane is extracted. Since points of homogeneous regions cannot be reconstructed in 3D, the second step segments the image using the ground points obtained during the first step as seed points. However this method uses a stereo camera and assumes the ground plane is dominant. At the contrary, [10] first segments the image and then computes homography to detect and merge the regions having coplanar feature points. However it requires a significant number of point features in each initially segmented regions.

It appears that the main limitations of robust existing methods is that they use a calibrated stereo camera system and/or assume the ground is the dominant plane in the image. In this paper, we propose a new method that can overcome these two important drawbacks. Our approach is based on the epipolar constraint for planar scenes (i.e. homography) on a pair of catadioptric/omnidirectional images to detect

J.C. Bazin and I.S. Kweon are with RCV Lab, KAIST, Daejeon, South Korea.

P.Y. Laffont performed this work at RCV Lab, and is now with REVES / INRIA Sophia-Antipolis, France.

C. Demonceaux and P. Vasseur are with MIS laboratory, UPJV, France.

some point features belonging to a plane (such as ground or building walls). To robustly extract the plane when it is not dominant in the image, we propose the *2-point algorithm for homography* which imposes some constraints on the rotation and the normal vector using VP information. The VPs are obtained by previous works on line extraction and VP estimation [11]. Finally, the inliers belonging to the ground plane can be extracted using robust estimators and provide key clues for ground segmentation by GrabCut. This approach has three important advantages. First, it does not require a calibrated stereo camera system. Second, it extracts only the planes verifying the normal provided by the VPs and can robustly extract the ground plane. Third, despite the extra cost of VP estimation, it can run in real-time thanks to the lower number of required RANSAC iterations and a prestage selection.

This paper is organized as follows. First, we introduce omnidirectional vision and the equivalent sphere projection. Then, we present our *2-point algorithm for homography* and robust estimation. Finally, we present some experimental results on real omnidirectional images for ground plane feature extraction and plane segmentation.

II. Omnidirectional Vision and Equivalent Sphere Projection

Omnidirectional vision systems provide a much wider field of view than traditional cameras, typically $360°$ (horizontal) by $180°$ (vertical). Nowadays, several kinds of commercial omnidirectional systems exists, such as Point Grey's Ladybug camera (cf Fig 1.) or 0-360.com's mirror. Among the several advantages provided by omnidirectional vision, we can especially cite the larger amount of information shared between images and the handling of the traditional ambiguity of rotation-translation inherent to traditional cameras.

It has been shown that the projection associated to central omnidirectional cameras is equivalent to a sphere projection. The sphere equivalence provides two important properties for lines (cf [12] and Fig 2). First, a line segment in the world is projected onto a great circle in the equivalent sphere space. A great circle is a plane passing through the sphere center. Second, the projections of parallel lines (i.e. a set of great circles) intersect in 2 antipodal points.

In this paper, we focus on two kinds of omnidirectional vision systems: camera clusters and catadioptric cameras. A camera cluster is a group of synchronized cameras whose pictures are stitched together to build a panoramic image. A famous commercial system is the Ladybug camera by Point Grey and its application to Google Street View [13]. [14][15] considered the separate views acquired by the cameras. We preferred working on the panoramic image since it permits to avoid view discontinuities (e.g. the lines are not cut and the features can be tracked along images). The sphere projection is performed by a linear mapping between the 2D image coordinate values (u, v) and the two spherical angles (α, β) [16].

Catadioptric cameras are composed of a mirror, a lens and a conventional camera. Geyer and Daniilidis have demonstrated the equivalence for the single viewpoint catadioptric system with a two-step projection via a unitary sphere centered on the focus of the mirror (the single viewpoint) [12]. In order to apply the equivalence, it is necessary to know the intrinsic parameters of the camera and some additional mirror parameters which can be estimated by calibration [17].

Fig. 1. Examples of omnidirectional views obtained by Point Grey's Ladybug camera. These images are part of the Google Street View data which is copyrighted by Google.

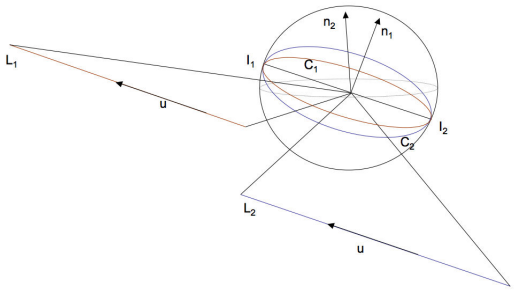

Fig. 2. Equivalent sphere projection: a world line (L_1) is projected onto a great circle (C_1) in the sphere and the projection of parallel lines (L_1 and L_2) intersect in 2 antipodal points (I_1 and I_2).

III. Proposed Method for Automatic Ground Features Extraction

This section aims to answer the following question: how to automatically and robustly extract features on a plane (such as the ground plane or building walls)? An intuitive and widely used method consists in estimating the dominant plane by homography and considering the inliers of the RANSAC procedure as ground features. However it assumes the dominant plane in the image is the ground, which might not be true in practice. Our approach for homography estimation uses vanishing point information in order to constrain the rotation and the ground normal vector. It permits to efficiently extract ground features and also avoid the problem of virtual/tilted plane. First, we introduce the notations used

in the paper and recall homography for calibrated omnidirectional vision. Then we present our proposed *2-point algorithm for homography*.

A. Homography in the equivalent sphere

Let P_w be a world point whose coordinates are $(x_w, y_w, z_w)^T$ in the coordinate frame of the first camera. It is projected onto the associated sphere at $P_s = (x, y, z)^T = \lambda(x_w, y_w, z_w)^T$ where $\lambda = 1/\sqrt{x_w^2 + y_w^2 + z_w^2}$. Subsequently, P_w is represented by $(x'_w, y'_w, z'_w)^T$ in the coordinate frame of the second camera. It is similarly projected onto the associated sphere at $P'_s = (x', y', z')^T = \lambda'(x'_w, y'_w, z'_w)^T$. P_s and P'_s are referred as corresponding spherical points. In [18], homography for traditional perspective cameras has been extended towards calibrated omnidirectional vision and defined as follows:

$$P'_s = \frac{\lambda'}{\lambda} H P_s \text{ where } H = R + \frac{T}{d} N^T \tag{1}$$

where R and T respectively correspond to the rotation and the translation between the two images, N is the unit normal vector of the plane and d is the distance from the center of the sphere to the plane. In the following, we note $\widetilde{T} = T/d$.

The most widely used method to compute H is the 4-point algorithm [19][18] (equivalently called DLT in the following). Given 4 point correspondences, DLT computes H by solving an eigenvector problem, using SVD for example.

B. The 2-point Algorithm for Homography

This section aims to estimate $\widetilde{T}$ assuming that R and N are known (section IV will discuss how to obtain them). To deal with scale problem, eq (1) is converted to:

$$P'_s \times H P_s = 0 \tag{2}$$

Using the decomposition of H and mathematical manipulations, eq (2) can be re-written as:

$$P'_s \times \widetilde{T} = -\frac{P'_s \times R P_s}{N^T P_s} \tag{3}$$

For easier derivations, the previous equation is noted as:

$$u \times \widetilde{T} = v \text{ with } u = P'_s \text{ and } v = -\frac{P'_s \times R P_s}{N^T P_s} \tag{4}$$

The system (4) is composed of 3 equations but only two of them are independent. Thus each point correspondence provides two equations. Since $\widetilde{T}$ has 3 DOF, it can be estimated by only 2 point correspondences. By expanding eq (4) for each i^{th} correspondence, we get:

$$\begin{pmatrix} u_2^i \widetilde{T}_3 - u_3^i \widetilde{T}_2 \\ u_3^i \widetilde{T}_1 - u_1^i \widetilde{T}_3 \\ u_1^i \widetilde{T}_2 - u_2^i \widetilde{T}_1 \end{pmatrix} = \begin{pmatrix} v_1^i \\ v_2^i \\ v_3^i \end{pmatrix}$$

where $u^i = (u_1^i, u_2^i, u_3^i)$ and $v^i = (v_1^i, v_2^i, v_3^i)$. Then by factoring with respect to $\widetilde{T}$ (keeping only 2 equations), we obtain $A_i \widetilde{T} = b_i$ where

$$A_i = \begin{pmatrix} 0 & -u_3^i & u_2^i \\ u_3^i & 0 & -u_1^i \end{pmatrix} \text{ and } b_i = \begin{pmatrix} v_1^i \\ v_2^i \end{pmatrix} \tag{5}$$

After stacking two matrices A_i and B_i into A and B, $\widetilde{T}$ can be easily computed by pseudo-inverse:

$$\widetilde{T} = (A^T A)^{-1} A^T B \tag{6}$$

C. Robust Estimation

The quality of the 2-point algorithm depends on the accuracy of R and N. In the following, we study how to handle their noisy estimations.

Accurate *a priori* Information

The experiment section on synthesized data will confirm the intuition that the 2-point algorithm is more robust to data noise than DLT up to a certain level of noise on R and N. It means that 2-point algorithm is preferred when R and N can be precisely estimated. It typically occurs for VP estimation with omnidirectional images and for robotic applications when navigation sensors like gyroscope or compass are used (gravity obtained by these sensors provides the ground normal).

Robust estimator must be used to remove outliers, i.e. false correspondences or correspondences of points not on the ground plane. In our program, the 2-point algorithm is included in RANSAC framework. [19] defines the theoretical minimum number of samples that is required to ensure that at least one sample is free from outliers and an example is depicted in Table I. Compared to the traditional 4-point DLT algorithm, it clearly shows that the 2-point algorithm decreases the theoretical complexity of RANSAC, especially when the percentage of outliers is high. Finally, we apply DLT on the best set of inliers, in order to handle the noisy estimation of R and N and also refine the motion estimation.

sample size	proportion of outliers ϵ in %						
s	5	10	20	25	30	40	50
2	2	3	5	6	7	11	17
4	3	5	9	13	17	34	72

TABLE I

THE NUMBER OF SAMPLES REQUIRED TO ENSURE, WITH A PROBABILITY $p = 0.99$, THAT AT LEAST ONE SAMPLE HAS NO OUTLIERS FOR A GIVEN NUMBER s OF MINIMAL POINTS AND A PROPORTION OF OUTLIERS ϵ. $s = 2$POINTS REFERS TO OUR APPROACH AND $s = 4$POINTS REFERS TO THE 4-POINT DLT ALGORITHM. FROM [19] P119.

Noisy *a priori* Information

In the case, R and N are not precisely estimated, it is important to note that they still provide important information. In our approach (referred as *relaxed RANSAC*), we consider that the noisy estimates can correctly reject the gross outliers, and to deal with noise, we use an inlier threshold relatively large (in our experiments, 5 pixels for image resolution 1280×960). This technique permits to keep the RANSAC speed advantage of the 2-point algorithm.

IV. REAL-TIME ROTATION AND VP ESTIMATION

Several interesting methods have been proposed to estimate the rotation in traditional and omnidirectional images

(readers are invited to refer to [11] for a recent review). Among the existing methods, the line-based approach is the fastest and is selected in our framework. It usually must face 2 issues: first, extracting the lines in the image from edge map, second, clustering the lines to their associated (unknown) vanishing points (VP). In the following, we present our framework to face these 2 issues.

A. Line Extraction and Rotation Estimation

For line detection, we apply [20] which is an extension of the polygonal approximation in sphere space. For self-containment, we explain the main idea of this method. This algorithm starts by detecting edges in the image and building chains of connected edge pixels. Then these chains are projected on the sphere. If a chain verifies the great circle constraint, it is considered a line. Otherwise, it is cut into 2 sub-chains and the procedure iterates until a sub-chain is considered a line or its length is too small. Because of possible edge discontinuity, a line might be decomposed into more than one chain. Therefore a merging step is applied to gather the several sub-chains of a single line and then combine all the gathered data points for accurate line fitting by least-square minimization. This algorithm has 2 important advantages: it can be applied to any central camera (e.g. catadioptric and omnidirectional) and is fast.

For line clustering, the most simple approach is based on the Hough transform [21][22]: the line segments are mapped onto great circles in a histogram representing the sphere surface. VPs are detected as peaks in the histogram, corresponding to areas where several great circles intersect. However this approach might not provide accurate VPs and suffers from the classic defects of the Hough transform such as the importance of parameter sampling and also an expensive computation. In urban scenes, 2 or 3 dominant directions usually exist and the fact that these dominant VPs are orthogonal is often used to constrain the VP estimation (the Manhattan constraint) [23][24][25]. This Hough transform cannot enforce orthogonality of VPs. In [11], we proposed a method that is fast and can impose VP orthogonality. It is based on a top-down approach: a set of rotations (obtained by motion model or heuristic) is considered and the most consistent rotation is selected.

Some experimental results for line extraction and VP estimation are shown in Fig 3 for Google Street Street View omnidirectional images and in the second row of Fig 8 for our catadioptric sequences.

B. Constraints Provided by the Vanishing Points

Rotation and Plane Normal

The VP estimation method provides the K vanishing points (and the plane normal N) and also the relative rotation R between 2 frames, which permits to apply the 2-point algorithm (cf eq (3)). To extract the vertical VP among the 3 VPs, many methods are possible. For example, one may simply choose the one with the highest z coordinate (i.e. the "most vertical" one). Sky detection in outdoor images can also be used, like in [26]. A third method is to simply manually select the vertical VP in the first frame and then track it during the sequence. One may comment that the vertical VP might not be directly observed in the images. If we assume the vertical direction is orthogonal to the ground, which is generally the case in urban environments, the vertical VP can be simply computed by cross-product of the 2 horizontal VPs. Typical results of VP extraction using the proposed method are displayed in Fig 8.

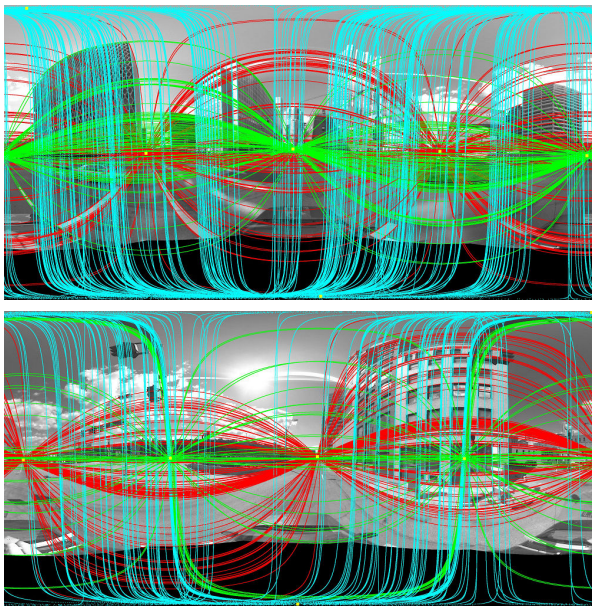

Fig. 3. Line and vanishing points extraction by the proposed algorithm on a Google Street View sequence (same images as Fig 1). Each conic corresponds to a detected line and all parallel lines have the same color. The conics have been enlarged for a better visualization. Additional results are presented in the attached video.

Horizon

The RANSAC procedure is usually applied to all the features of the image. We present a method for greatly reducing both the number of features to test and the proportion of outliers and thus accelerating the RANSAC. It is based on the idea that the ground features lie below the horizon. Therefore, as depicted by Fig 4(a), any ground features P_s in the equivalent sphere must verify the (necessary but not sufficient) condition $P_s \cdot N < 0$ where N is the ground normal obtained from VPs. Thus we apply the RANSAC procedure only to the points verifying this horizon constraint. Figure 4 depicts the projection of the horizon in a catadioptric image using the ground normal vector and the associated mask to remove the points that do not lie below the horizon.

V. Experimental Results

This section presents experimental results of ground feature extraction by the VP-constrained homography and then ground plane segmentation. We processed two kinds of omnidirectional images: acquired by a catadioptric system and a camera cluster. The catadioptric system is composed of a paraboloid mirror manufactured by Panosmart and a Sony DFW-SX910 camera. The image resolution is 1280x960. The

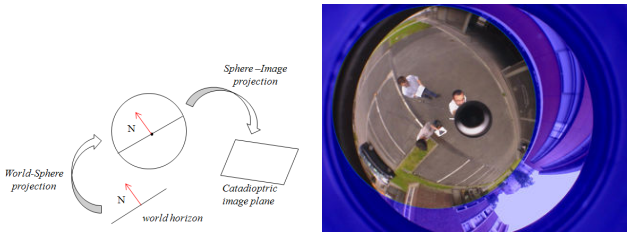

Fig. 4. Left: the normal vector N of the horizon plane in the equivalent sphere is similar to the normal vector of the horizon plane in the world. Right: projection of the horizon from the ground normal vector provided by the vertical VP for horizon constraint.

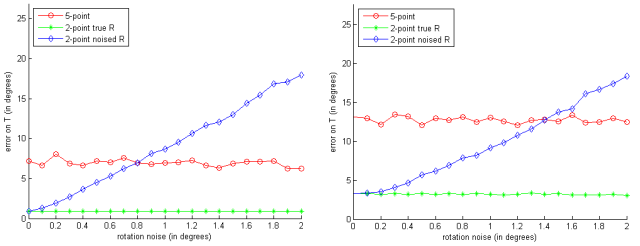

Fig. 5. Comparison of mean error in degrees (y axis) for the estimation of the translation direction on synthesized catadioptric images by the 4-point and the proposed 2-point algorithms, with respect to rotation noise (mean=0 and increasing std on x-axis). Data points have been noised by a gaussian distribution of mean=0 and std=1 pixel (left figure) and 3 pixels (right figure).

catadioptric system is calibrated using the toolbox based on [17]. The camera cluster is Point Grey's Ladybug camera. It is composed of six 1024x768 color cameras with small overlap of their field of view. Five cameras are positioned in a horizontal ring to capture side-view images, and one is located on the top to take top-view images. Calibration information is provided by Point Grey and contains all the intrinsic and relative extrinsic parameters of all six cameras. The advanced Point Grey acquisition program is able to build a panoramic image, in real-time, by stitching the 6 images using the calibration information. The panoramic image size is 1664x832. [14][15] considered the separate images and projected the points on the equivalent sphere by multiplying the inverse calibration matrix and the inverse rotation matrix in each camera. On the contrary, we worked on the panoramic image and performed a linear mapping onto the sphere[16], as explained in section II.

A. Synthesized Data

To analyze the dependency of our 2-point algorithm to data noise, we synthesized 1000 catadioptric images composed of 100 matching points and we have applied to their coordinates a gaussian noise. In a first series of experiments, the ground truth rotation was directly used for the computation of T by the 2-point algorithm. To simulate the fact that the rotation cannot be perfectly estimated from the images, we have performed a second series of experiments where the 3 rotation angles have been corrupted by a gaussian noise of mean=0 and an increasing std=$[0°, \ldots, 2°]$. For comparison, we have also applied the 4-point algorithm to compute the homography matrix. Figure 5 depicts the mean error in degrees for the estimation of the translation direction. As the 4-point algorithm and the 2-point algorithm with true rotation do not depend on the rotation noise, their error is constant. It can be noticed that the error of the 2-point algorithm with noised rotation linearly increases with the level of rotation noise. It also shows that the 2-point algorithm is more efficient than the 4-point algorithm when the rotation is corrupted up to a certain level of noise, especially when the noise of data points increases. It is a very important result because it demonstrates that the constraint provided by accurate *a priori* motion estimation and the fact that only 2 points are required permits to obtain a more robust estimation.

B. Real Catadioptric Videos

VP extraction

As discussed in section IV, we respectively applied [20] and [11] for line extraction and rotation estimation. This procedure runs in about 10-15ms per frame. Examples of VP extraction obtained by our proposed method in real catadioptric images are displayed in Fig 8 (2^{nd} row) and 6.

Figure 6 shows a typical example of motion estimation on a Google Street View sequence obtained from the estimation of rotation and car holonomic constraints. Given the rotation obtained by the VPs, the car position at time t, noted $(p_x(t), p_y(t))$, can be estimated by holonomic constraints:

$$p_x(t) = p_x(t-1) + d_t * cos(\theta_t + \Delta_{\theta_t}/2) \quad (7)$$

$$p_y(t) = p_y(t-1) + d_t * sin(\theta_t + \Delta\theta_t/2) \quad (8)$$

where d_t is the distance traveled between $t-1$ and t, θ_t is the rotation at time t and $\Delta_{\theta_t} = \theta_t - \theta_{t-1}$. Initially $p_x(t) = p_y(t) = 0$. In our case, since the distance d_t is known only up-to-scale, we set $d_t = 1$ for every t (i.e. constant velocity). This sequence contains about 1000 frames and covers around 900 meters. This kind of motion estimation technique (rotation + holonomic constraints) provides only a rough trajectory estimation. For example, since $d = 1$ it explains some non-overlapping streets. However it permits to (1) interpret and verify the estimated rotation angles more easily and also (2) reflects the path structure. For example, we can note that the estimated trajectory contains orthogonal and parallel parts, which corresponds to the structure of the scene. Therefore the estimated VPs can be safely inserted in the proposed 2-point algorithm for homography.

Ground features extraction

From the VPs, we obtain the ground normal and compute back the relative rotation. Feature points have been extracted and matched by SIFT algorithm [27]. To deal with outliers, we performed robust estimation and we present the results obtained by the relaxed RANSAC of the 2-point algorithm (cf section III-C). The first row of Figure 8 depicts the inliers and outliers obtained by the traditional DLT. It shows that DLT often classifies points wrongly and detects a virtual

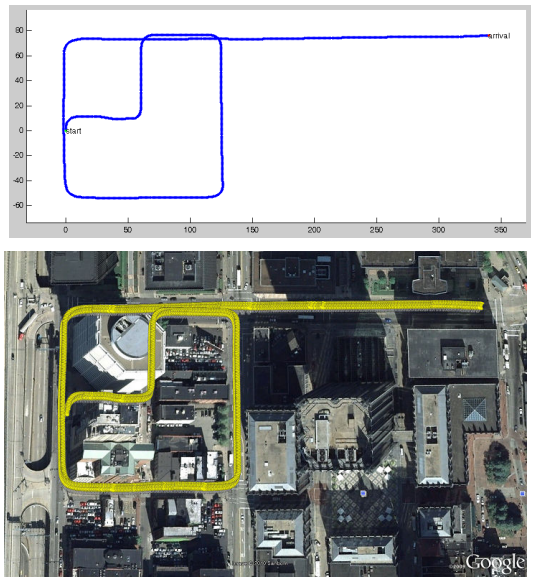

Fig. 6. Motion estimation (top) on a part of a Google Street View sequence from the estimation of rotation and car holonomic constraints. The ground truth trajectory provided by Google is displayed in (bottom).

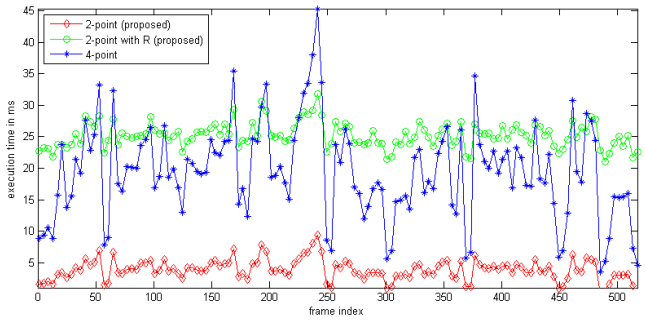

Fig. 7. Comparison of execution time in ms for the 2-point algorithms (with and without VP estimation) and DLT algorithm using the minimum number of theoretical RANSAC iterations for 50% outliers (best seen in color).

plane. On the contrary, our proposed framework (2^{nd} and 3^{rd} rows) manages to successfully extract the ground features, which demonstrates the robustness of our approach. Fig 7 compares the execution time of the relaxed RANSAC for the 2-point algorithm (noted 2ptRR) and the RANSAC for the traditional 4-point algorithm (noted 4ptRR). In average, 2ptRR runs a little bit slower than 4ptRR for this sequence. However 2ptRR still runs in real time and provides much more robust results than 4ptRR. Data analysis has shown that 2ptRR is faster than 4ptRR when the number of features is high, because of the time consuming RANSAC iterations. Finally, it is worthwhile to note that the 2ptRR without the rotation estimation step is extremely fast (about 5ms in average). It could represent an interesting tool for robotic platforms equipped with IMU or gyroscope for example. Readers are invited to refer to the attached video and the authors' website for additional results.

C. Segmentation

GrabCut is an efficient technique to segment some objects or regions in an image (cf [2] for details). In the original Grabcut paper, the user only provides an "incomplete labeling" (rectangle around the foreground region) to indicate a background region and no extra information is needed to specify the foreground region. To obtain better results, additional user editing could indicate definitely foreground or background regions. In our framework, all these user inputs are given automatically. To substitute the incomplete user-labeling, a mask is calculated from the horizon constraint (cf Fig 4). This mask is similar with the rectangle of the grabcut: the outside area is definitely true background (i.e. not ground plane) and the inside area might contain background and foreground). For the additional user editing inside the mask, our framework uses the inliers of the 2-point algorithm as data definitively foreground (i.e. ground plane). Even if the inliers seem relatively sparse, experiments have shown that it is often enough for Grabcut framework. Segmentation results are depicted in the 4^{th} row of Fig 8. The entire parking plane is correctly segmented. The main difference with true segmentation is that the persons or a very small part of building/car can be partially mislabeled. If needed, Grabcut could be easily replaced by an another seed-based segmentation method and additional constraints (e.g. color histograms) could be incorporated. Our contribution is to provide features on a plane, not an entire segmented plane. However, the results by the current method are still very satisfying.

VI. Conclusion

This paper faced the problem of automatic plane extraction in omnidirectional videos. We developed an original complete framework that provides robust results and runs in real-time. Our main contribution is the so-called *2-point algorithm for homography*. The key idea is to impose some constraints on the homography based on VP information. Experiments on real data have demonstrated that the widely used DLT algorithm often suffers from the virtual plane problem and also fails to extract a plane when it is not the dominant one in the image. On the contrary, our proposed *2-point algorithm for homography* manages to filter out the planes from VP information and robustly extract ground features during the experiments. Finally, we have also shown that the inliers obtained by our framework provide key clues for ground segmentation by GrabCut.

References

[1] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI'00)*, 2000.

[2] C. Rother, V. Kolmogorov, and A. Blake. "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23:309–314, 2004.

[3] D. Santosh, S. Achar, and C. V. Jawahar. Autonomous image-based exploration for mobile robot navigation. In *ICRA'08*.

[4] S. Lenser and M. Veloso. Visual sonar: Fast obstacle avoidance using monocular vision. In *IROS'03*.

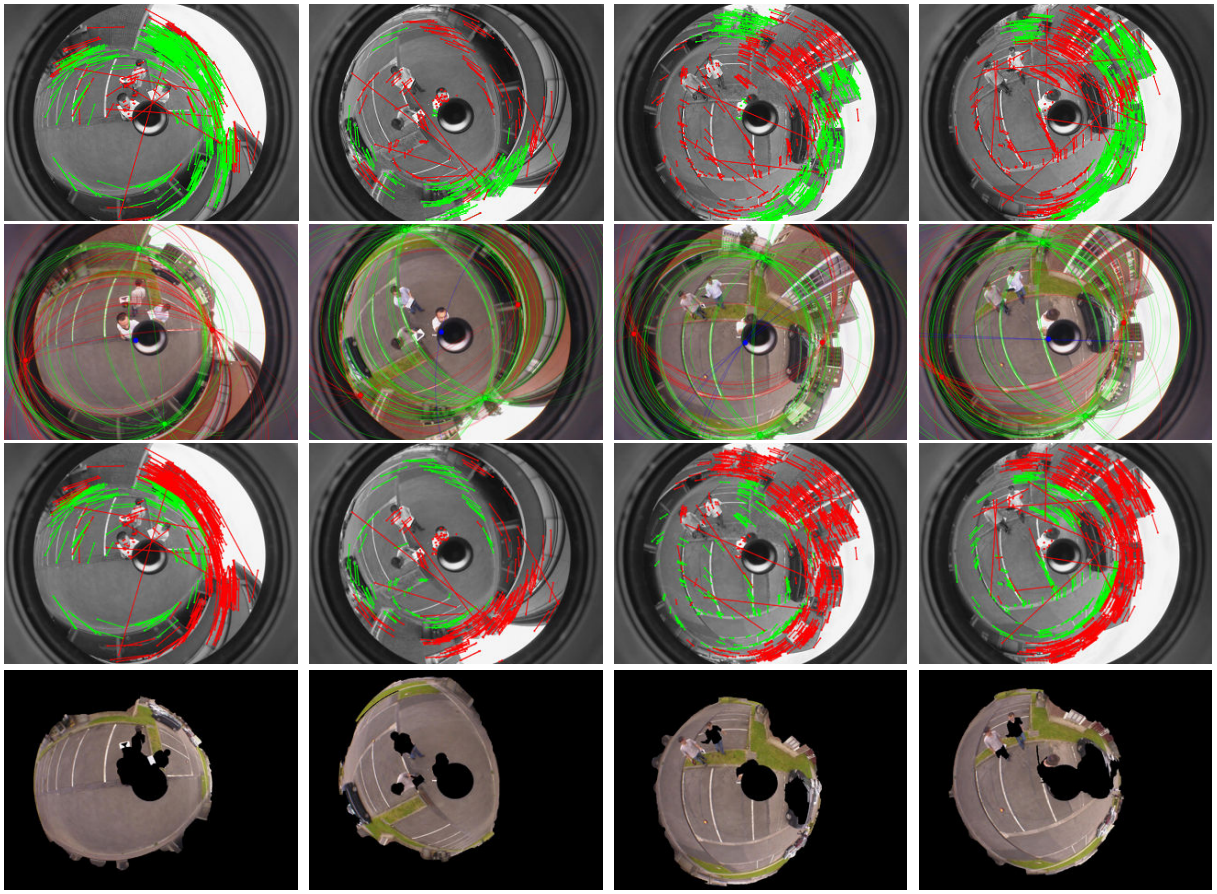

Fig. 8. Intermediate steps and final ground plane segmentation. 1^{st} row: inliers (points on the plane, in green) and outliers (points not on the plane or wrong matches, in red) obtained with the traditional DLT algorithm (images drawn in gray to enhance the feature points). Many inliers are wrongly classified and lie on a virtual plane. 2^{nd} row: vanishing point extraction (same color legend as Fig 3). 3^{rd} row: inliers and outliers obtained by the proposed *2-point algorithm for homography*, which outperforms the performance of the traditional DLT (shown in 1^{st} row). 4^{th} row: final segmentation by GrabCut (best seen in color). Additional results are presented in the **attached video**.

[5] J. Zhou and B. Li. Robust ground plane detection with normalized homography in monocular sequences from a robot platform. In *IEEE International Conference on Image Processing (ICIP'06)*.

[6] N. Ohnishi and A. Imiya. Model-based plane-segmentation using optical flow and dominant plane. In *MIRAGE*, volume 4418 of *Lecture Notes in Computer Science*, pages 295–306. Springer, 2007.

[7] A. Dankers, N. Barnes, and A. Zelinsky. Active vision for road scene awareness. In *IEEE Intelligent Vehicles Symposium (IVS'05)*, 2005.

[8] K. Konolige, M. Agrawal, R. C. Bolles, C. Cowan, M. Fischler, and B. P. Gerkey. Outdoor mapping and navigation using stereo vision. In *Proc. of the International Symposium on Experimental Robotics (ISER)*, 2006.

[9] J. Lobo and J. Dias. Ground plane detection using visual and inertial data fusion. In *IROS'98*.

[10] B. Liang and N. Pears. Ground plane segmentation from multiple visual cues. In *International Conference on Image and Graphics (ICIG'02)*, 2002.

[11] J. C. Bazin, I.S. 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 *IROS'08*.

[12] C. Geyer and K. Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision (IJCV'01)*, 45(3):223–243.

[13] Google street view: accessed from Google Map `http://maps.google.com/`.

[14] J.-H. Kim, R. Hartley, J.M Frahm, and M. Pollefeys. Visual odometry for non-overlapping views using second-order cone programming. In *ACCV'07*.

[15] J.-H. Kim, H. Li, and R. Hartley. Motion estimation for multi-camera systems using global optimization. *CVPR'08*.

[16] A. Banno and K. Ikeuchi. Omnidirectional texturing based on robust 3d registration through euclidean reconstruction from two spherical images. *Computer Vision and Image Understanding (CVIU'09)*, 2009.

[17] J. P. Barreto and H. Araujo. Geometric properties of central catadioptric line images and their application in calibration. *PAMI'05*.

[18] C. Mei, S. Benhimane, E. Malis, and P. Rives. Homography-based tracking for central catadioptric cameras. In *IROS'06*.

[19] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[20] J. C. Bazin, I.S. Kweon, C. Demonceaux, and P. Vasseur. Rectangle extraction in catadioptric images. In *ICCV Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras (OMNIVIS'07)*.

[21] S. T. Barnard. Interpreting perspective image. *Artificial Intelligence Journal*, 21(4):435–462, 1983.

[22] L. Quan and R. Mohr. Determining perspective structures using hierarchical hough transform. *Pattern Recognition Letters*, 9:279–286, 1989.

[23] P. Denis, J. H. Elder, and F. J. Estrada. Efficient edge-based methods for estimating manhattan frames in urban imagery. In *Proceedings of the European Conference on Computer Vision (ECCV'08)*, 2008.

[24] R. Cipolla, T. Drummond, and D. Robertson. Camera calibration from vanishing points in images of architectural scenes. In *British Machine Vision Conference (BMVC'99)*, volume II, pages 382–392, 1999.

[25] J. Kosecka and W. Zhang. Video compass. In *Proceedings of European Conference on Computer Vision (ECCV'02)*, pages 657–673, 2002.

[26] C. Demonceaux, P. Vasseur, and C. Pégard. UAV attitude computation by omnidirectional vision in urban environment. In *ICRA'07*.

[27] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision (IJCV'03)*, volume 20, pages 91–110, 2003.