Line reconstruction using prior knowledge in single non-central view

Jesus Bermudez-Cameo[1]
bermudez@unizar.es
Cédric Demonceaux[2]
cedric.demonceaux@u-bourgogne.fr
Gonzalo Lopez-Nicolas[1]
gonlopez@unizar.es
José J. Guerrero[1]
josechu.guerrero@unizar.es

[1] Instituto de Investigación en Ingeniería de Aragón (I3A)
Universidad de Zaragoza
Zaragoza, Spain

[2] Le2i - Institut Universitaire de Technologie Le Creusot
Université de Bourgogne
Le Creusot, France

Abstract

Line projections in non-central systems contain more geometric information than in central systems. The four degrees of freedom of the 3D line are mapped to the line-image and the 3D line can be theoretically recovered from 4 projecting rays (i.e. line-image points) from a single non-central view. In practice, extraction of line-images is considerably more difficult and the resulting reconstruction is imprecise and sensitive to noise. In this paper we present a minimal solution to recover the geometry of the 3D line from only three line-image points when the line is parallel to a given plane. A second minimal solution allows to recover the 3D line from two points when the direction of the 3D line is known. Both cases can exploit the prior knowledge of the vertical direction in man-made environments to reduce the complexity in line-image extraction and line reconstruction. The formulation based on Plücker lines is integrated in a line-extraction pipeline, which is tested with synthetic and real non-central circular panoramas. In addition, we evaluate the performance of the robust extractor and the accuracy of the proposal in comparison with the unconstrained method.

1 Introduction

Although line fitting in non-central systems provides interesting features, it is still a research open topic with challenging problems to be solved. On the one hand, line projections contain more geometric information because the four degrees of freedom of the 3D line are encapsulated on a single line-image. On the other hand, the 3D reconstruction from a single projection is very sensitive to noise. Additionally, the accuracy of the estimate strongly depends on the effective baseline of the imaging systems, which is usually too small in practice. The complexity of the extraction is related with the minimum required points for fitting the line. Therefore, reducing the number of required points to fit lines in non-central systems can be very interesting in terms of computational time efficiency.

Regarding the algorithms for central systems, the use of additional prior like the Manhattan assumption helps to improve the results [15, 17, 25]. In addition, knowing prior

information from additional sensors like Inertial Measurement Unit (IMU) can reduce the complexity of features extraction [26]. In particular, when fitting line images from non-central images, knowing the gravity direction in man-made settings can be used to reduce the minimal number of required points for line fitting from four to three.

There exist some previous works dealing with point and line projections in non-central systems. For instance, point back-projection models for several non-central catadioptric systems were studied in [28]. However, a closed form solution of the forward projection is not available in most non-central systems. The forward projection of axial non-central catadioptric systems based on quadric mirrors can be obtained as the solution of a polynomial [1]. The non axial case is dealt in [2]. Another example of non-central images comes from moving cameras without common viewpoint. The case of linear pushbroom camera is extensively studied in [13] and the non-central circular panorama is dealt in [18].

The fundamentals for line fitting in non-central systems consist in computing the solution of lines intersecting four generic rays [12, 27, 30]. This theory is exploited in several works dealing with non-central catadioptric systems [6, 7, 11]. Some related works are [5, 16, 29]. In [16] two non-central catadioptric systems are used for lines reconstruction. In non-central cameras not only lines can be recovered from single projections but also other planar curves [29]. In some non-central systems, the line-image encodes the 4 DOFs of the 3D line but also calibration information [5].

Additional constraints can be used to improve the results: for example considering lines orthogonal to the axis of revolution of the system [9, 22] or imposing parallelism and perpendicularity between pairs of lines [4]. Cross-ratio properties [21] or using off-axis systems to avoid degeneracies [8] have been considered. Prior information can also be used to improve the results, for example a prior line-based description of the scene is used in [19, 20] to estimate the pose of a non-central catadioptric system.

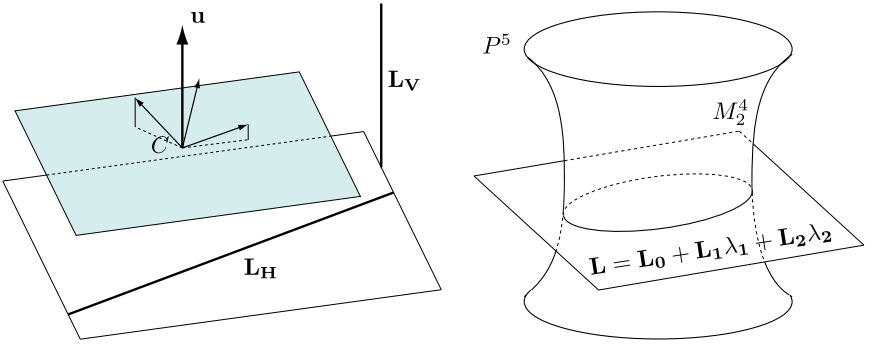
In this paper we present the minimal solution to fit a line parallel to a plane in non-central systems and the minimal solution for fitting a line when its direction is known. These proposals allow using the vertical direction as additional prior knowledge to facilitate the line-image extraction and to increase the accuracy of the metric reconstruction. We evaluate the proposal in terms of accuracy and robustness using extensive simulation. The proposal is integrated in a complete extraction pipeline which is tested with synthetic and real images performing 3D reconstruction from single view.

2 Background

In this Section, we briefly present how to compute a 3D line from a single non-central projection with the notation of [23]. Using Plücker coordinates, the intersection between lines is described by the side operator [14, 23]. A given line $\mathbf{L} = \left(\mathbf{l}^\mathsf{T}, \bar{\mathbf{l}}^\mathsf{T}\right)^\mathsf{T} \in \mathbb{P}^5$ (where $\mathbf{l} \in \mathbb{R}^3$ and $\bar{\mathbf{l}} \in \mathbb{R}^3$) intersects a ray $\Xi \in \mathbb{P}^5$ if

$$side\,(\mathbf{L},\Xi) = \mathbf{L}^\mathsf{T}\mathsf{W}\Xi = \mathbf{l}^\mathsf{T}\xi + \bar{\mathbf{l}}^\mathsf{T}\xi = 0 \quad \text{where} \quad \mathsf{W} = \begin{pmatrix} 0_{3\times3} & I_{3\times3} \\ I_{3\times3} & 0_{3\times3} \end{pmatrix}. \tag{1}$$

Since a 3D line has four degrees of freedom we need at least 4 equations to solve the system for $\mathbf{L}$. If four projection rays from a 3D line provide four independent constraints we can compute the 3D line (see [30] for details). In some degenerated cases four rays do not provide four independent equations [11]: when the projection surface is a plane (the line is



(a) (b)

Figure 1: (a) Euclidean representation of a reference system C with any orientation and a known direction $\mathbf{u}$ in a Manhattan scene. (b) Two dimensional subspace in $\mathbb{P}^5$ and Klein Quadric M_2^4 representing the constraint for being a line in Plücker coordinates.

coplanar with the axis of revolution or coplanar with the plane containing the circle) or when the projection surface is a regulus (i.e. the locus of lines meeting three given skew lines).

3 Line fitting with known vertical direction

Let us consider a non-central imaging system placed in a man-made setting i.e. lines arranged in orthogonal directions (see Fig. 1 (a)). The vertical direction $\mathbf{u}$ corresponds to the direction of vertical lines and it also defines a set of horizontal planes. In this Section we first present the minimal solution to compute a 3D line parallel to a plane and secondly, the minimal solution to compute a 3D line with known direction, both from a single non-central view.

3.1 Computing a 3D line parallel to a plane

If we know that a line is parallel to a given plane $\mathbf{U} = \left(u_0, \mathbf{u}^\mathsf{T}\right)^\mathsf{T}$ with known normal $\mathbf{u}$, the complexity to fit a line projection from a non-central image is reduced from 4 degrees of freedom (DOFs) to 3 DOFs. Given the line $\mathbf{L} = \left(\mathbf{l}^\mathsf{T}, \bar{\mathbf{l}}^\mathsf{T}\right)^\mathsf{T}$ and three skew projection rays $\Xi_k = \left({\xi_k}^\mathsf{T}, {\bar{\xi}_k}^\mathsf{T}\right)^\mathsf{T}$ intersecting the line, the null space of the under determined linear system

$$\Xi_k^\mathsf{T} \mathsf{W} \mathbf{L} = {\bar{\xi}_k}^\mathsf{T} \mathbf{l} + {\xi_k}^\mathsf{T} \bar{\mathbf{l}} = 0 \,, \quad k = 1 \ldots 3 \tag{2}$$

is a subspace of dimension 2 in $\mathbb{P}^5$. A parametrized description of this 2-dimensional subspace is obtained by solving the null space with singular value decomposition. Therefore, the matrix $\mathsf{A} = (\Xi_1, \Xi_2, \Xi_3)^\mathsf{T} \mathsf{W}$ can be written as the product of the three matrices, $\mathsf{A} = \mathsf{U}\Sigma\mathsf{V}$ and taking the three last columns of matrix V, denoted as $\mathbf{L}_0$, $\mathbf{L}_1$ and $\mathbf{L}_2$, we parametrize the null space with $L : \mathbb{P}^2 \mapsto \mathbb{P}^5$ defined as

$$\mathbf{L} = \mathbf{L}_0 + \mathbf{L}_1 \lambda_1 + \mathbf{L}_2 \lambda_2 \,. \tag{3}$$

This subspace is not contained a priori in the Klein quadric M_2^4 and intersects it in a one dimensional curve (see Fig. 1 (b)). To obtain the two parameters λ_1 and λ_2 we need two

additional constraints. First one is the constraint of being a line (usually represented in the $\mathbb{P}^5$ space with the Klein quadric M_2^4) which is expressed by

$$\mathbf{L}^\top \mathbf{W} \mathbf{L} = 0 \,. \tag{4}$$

Given that the 2-dimensional subspace L is not contained a priori in the Klein quadric M_2^4 the solution defined by this constraint lies on a one dimensional curve (see Fig. 1 (b)). Second constraint is that the line is parallel to the plane $\mathbf{U}$ and it is defined by the orthogonality between the direction of the line $\mathbf{l}$ and the vector $\mathbf{u}$ ($\mathbf{u}^\top \mathbf{l} = 0$). Assuming we know $\mathbf{u}$ and solving this constraint for λ_2 we obtain

$$\lambda_2 = -\frac{\mathbf{u}^\top \mathbf{l}_0 + \mathbf{u}^\top \mathbf{l}_1 \lambda_1}{\mathbf{u}^\top \mathbf{l}_2} \tag{5}$$

and substituting it in equation (4) we obtain a quadratic equation,

$$b_1 \lambda_1^2 + 2 b_2 \lambda_1 + b_3 = 0 \quad \text{where} \tag{6}$$

$$\begin{aligned} b_1 &= \beta_{2211} - 2\beta_{1212} + \beta_{1122} \\ b_2 &= \beta_{0122} - \beta_{0212} - \beta_{1202} + \beta_{2201} \\ b_3 &= \beta_{2200} - 2\beta_{0202} + \beta_{0022} \end{aligned}$$

and $\beta_{ijkl} = \mathbf{L}_i{}^\top \mathbf{W} \mathbf{L}_j \mathbf{u}^\top \mathbf{l}_k \mathbf{u}^\top \mathbf{l}_l$. Equation (6) has one unknown λ_1 and two different solutions. From each solution of λ_1 a solution for $\mathbf{L}$ is computed using

$$\mathbf{L} = \left(\mathbf{u}^\top \mathbf{l}_2 \mathbf{L}_0 - \mathbf{u}^\top \mathbf{l}_0 \mathbf{L}_2\right) + \left(\mathbf{u}^\top \mathbf{l}_2 \mathbf{L}_1 - \mathbf{u}^\top \mathbf{l}_1 \mathbf{L}_2\right) \lambda_1 \tag{7}$$

which is obtained by substituting (6) in (3). Consequently, the 3D line has been computed from three independent projection rays intersecting the line and the normal to the plane $\mathbf{u}$. In Section 2 we mention some degenerated cases for 4 rays, note that the 3-points approach can avoid the degeneracy of having the set of defining rays in a regulus because in this case the rays provide 3 independent equations with the additional information of the remaining constraint.

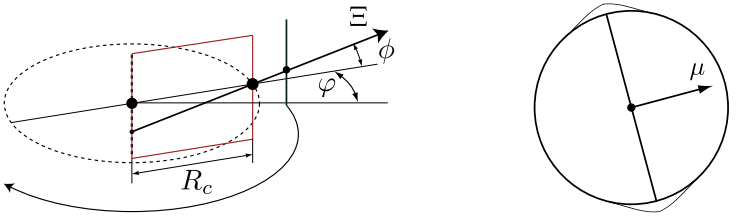
3.2 Computing a 3D line with known direction

When the orientation of a system is known in a Manhattan scene the direction $\mathbf{l}$ of the line $\mathbf{L}$ is known a priori (it is one of the three dominant directions). This means that only two of the four degrees of freedom of the 3D line must be computed and only two projection rays Ξ_k for $k = 1,2$ are needed. As we know the direction $\mathbf{l}$, the intersection between line $\mathbf{L}$ and a ray Ξ_k can be written as

$$\xi_k{}^\top \bar{\mathbf{l}} = -\bar{\xi}_k{}^\top \mathbf{l} \quad k = 1,2 \,. \tag{8}$$

By including the Plücker constraint $\mathbf{l}^\top \bar{\mathbf{l}} = 0$, we can compose a linear system of three equations and, solving for $\bar{\mathbf{l}}$, we obtain

$$\bar{\mathbf{l}} = \begin{pmatrix} \xi_1{}^\top \\ \xi_2{}^\top \\ \mathbf{l}^\top \end{pmatrix}^{-1} \begin{pmatrix} -\bar{\xi}_1{}^\top \mathbf{l} \\ -\bar{\xi}_2{}^\top \mathbf{l} \\ 0 \end{pmatrix} . \tag{9}$$



(a) (b)

Figure 2: (a) Non-central panorama projection model: Any projection ray Ξ corresponding to a pixel (characterized by angles φ and ϕ) intersects the vertical axis and a circle of radius R_c. (b) Mixture of Gaussians based on direction: The parameter of the distribution μ represents the orthogonal vector to a given direction $\mathbf{x}_i$ in a given plane.

When more than two rays are available the pseudo inverse can be used.

Notice that in this case we are fitting an element of the $\mathbb{P}^5$ space with more than four equations. Therefore the obtained solution has five independent dimensions and it does not perfectly fit the Plücker condition because it does not have to necessarily be a 3D line.

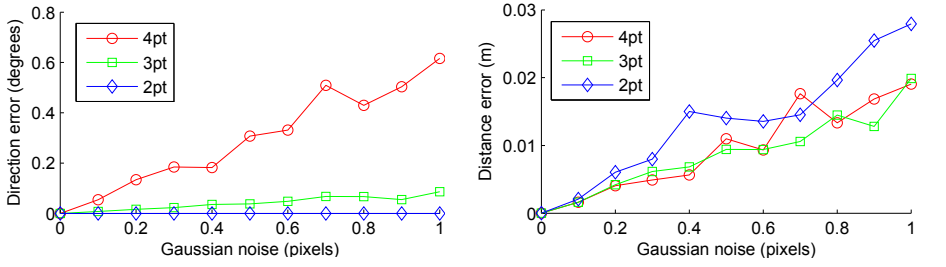
4 Line reconstruction from single non-central view

In this section, we present a method for line-image extraction in non-central images. The procedure is split in two steps. During the first step we assume a soft-Manhattan scene (1 dominant direction and the rest of the lines on the plane orthogonal to this direction), whereas for the rest we assume hard-Manhattan (3 orthogonal dominant directions). We assume that the vertical direction $\mathbf{u}$ is known and coincident with the direction of the gravity (e.g. it is obtained from a IMU system) but the imaging system is not aligned with this direction (see Fig. 1 (a)).

The lines composing the scene can be horizontal (e.g. $\mathbf{L}_H$) with direction orthogonal to $\mathbf{u}$ or vertical (e.g. $\mathbf{L}_V$) with direction coincident to $\mathbf{u}$. Horizontal lines are fitted using the 3-points method presented in Section 3.1 and vertical lines with the 2-points method presented in Section 3.2. The proposal has been applied to non-central circular panoramas [3, 13] (see Fig. 2 (a)) but it can be adapted to any other type of non-central system.

For automatic extraction, we extract edges from the image using Canny detector. To avoid false edges we first filter the image using the Rolling Guidance Filter [31]. These edges are then stored in connected components. From each component line-images are extracted using a greedy RANSAC approach which follows the pre-filter scheme described in PROSAC [10] and USAC [24]. To evaluate the quality of the points we use a measure of the effective baseline of a set of rays as described in [3]. This allows generating hypotheses more likely to be valid. Then, from each set of 3 points, three hypotheses are computed: two assuming the line is orthogonal to the vertical direction (Section 3.1) and one assuming the line is parallel to the vertical direction (Section 3.2). Only one of these solutions has enough supporting points on the image, hence we can store the three candidate solutions as hypotheses and use the voting scheme to select one of them. Using the distance presented in [3] we test each hypothesis. The most voted line-projection is selected, obtaining the 3D coordinates of the line and its classification as horizontal or vertical.

Line directions are elements of the Euclidean space $\mathbb{E}^3$. When being horizontal they are all contained in the same plane. If we assume that horizontal lines are arranged in dominant



(a) (b)

Figure 3: Comparison of the accuracy in line fitting with Gaussian noise in the line-image points without error in prior information. (a) Direction error (deg). (b) Distance error (m).

directions we can first cluster the horizontal lines in two classes along the unidimensional space defined by a unitary circle on the horizontal plane (See Fig. 2 (b)). For clustering them we use an algorithm based on mixture of Gaussians. Consider the following distribution

$$P(\mathbf{x}_i|c,\theta_c) = \frac{1}{\sigma_c\sqrt{2\pi}} e^{-\frac{\arcsin\left(\mu_c{}^{\mathsf{T}}\mathbf{x}_i\right)^2}{2\sigma_c^2}} \tag{10}$$

to describe the conditioned probability of the normalized direction vector $\mathbf{x}_i$ given that the label of the direction is c and the distribution parameters are $\theta_c = (\mu_c, \sigma_c)$ where μ_c represents the mean of the orthogonal space to the corresponding dominant direction. This distribution properly represents the direction considering both senses and both vanishing points simultaneously (see Fig. 2 (b)).

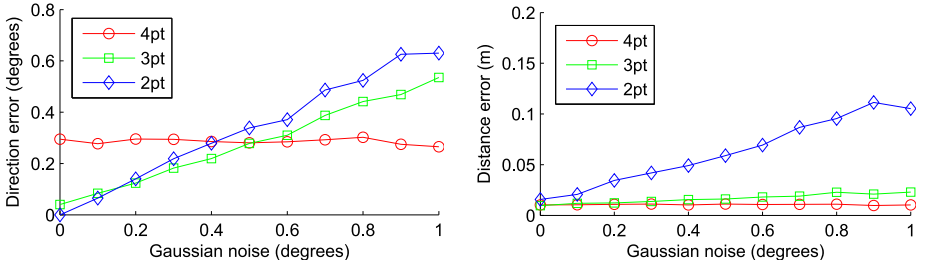
The initial solution is computed from two overlapped histograms. The posterior probability of each sample is estimated using the Bayes rule. For clustering the directions in the two classes we maximize the likelihood for obtaining the parameters $\theta = (\theta_1, \theta_2)$ of the two dominant orthogonal directions. After the clustering, the segments supporting horizontal lines are classified in two dominant directions, we obtain the rotation of the system with respect the scene, and 3D lines can be recomputed using the 2-points approach presented in Section 3.2. Once the clustering is performed, we exploit the connectivity of the extracted segments to generate hypotheses of connection. The connectivity map is built by composing a set of local Delaunay triangulations computed on the tangent map of the torus defined by the back-projection model.

5 Evaluation and experiments

In this Section we present a set of experiments to evaluate the performance of the proposed method. First, we evaluate the accuracy in 3D line fitting. Second, we test the robustness of the extraction proposal and finally we show examples of line extractions in synthetic and real non-central images.

5.1 Accuracy and robustness evaluation

We present a comparison in terms of accuracy of the 4 points algorithm (without constraints), the 3 points algorithm (assuming the line is parallel to a plane), and the 2 points algorithm



(a) (b)

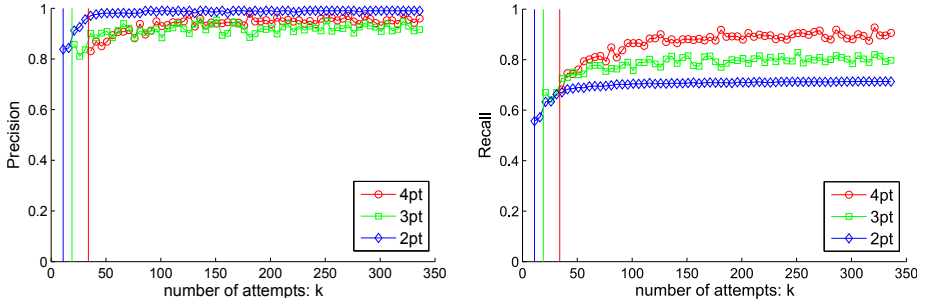
Figure 4: Comparison of the accuracy in line fitting with Gaussian noise in prior information (with σ from 0 to 1 degrees) and Gaussian noise in line-image points with $\sigma = 0.5$ pixels. (a) Direction error (deg). (b) Distance error (m).

(assuming the direction of the line is known). The system is a non-central panoramic imaging system with radius $R_c = 0.5$, field of view of 180×360 deg and resolution of 2048×4096 pixels. For comparing the accuracy we have simulated the projection of 100 random 3D lines and we have added Gaussian noise to the line-image points. Then the lines are fitted using the three approaches. For the first approach (4pt), 4 points of the line-image are used to fit the line (these points are selected for maximizing the effective baseline). For the second approach (3pt), 3 points of the line-image and the orientation of a plane parallel to the line are used for fitting the line. In this case, we also add Gaussian noise to the orientation of the plane (and again the points of the line-image are selected to maximize the effective baseline). For the third approach (2pt) the solution is computed from two points of the line-image but assuming that the direction of the line is known and having a Gaussian noise on the line direction. In Fig. 3 and Fig. 4, we represent the estimated accuracy according to two errors:

- the direction error between the estimated 3D line and the ground-truth line computed as $\varepsilon_\varphi = \arccos\left(\mathbf{l}^{\mathsf{T}}\mathbf{l}_{GT}\right)$.
- the depth error between the estimated 3D line and the ground-truth line computed as $\varepsilon_d = \left|\|\bar{\mathbf{l}}\| - \|\bar{\mathbf{l}}_{GT}\|\right|$.

Fig. 3 shows the influence of image error without noise in the prior information. The abscissa axis corresponds to the Gaussian noise added to the image points from $\sigma = 0$ pixels to $\sigma = 1$ pixel. As expected, the error is always bigger for the unconstrained method (4pt). Notice that when the direction is known (2pt) the direction is not really estimated and the plotted error in Fig. 3 (a) is always zero.

In Fig. 4, Gaussian fixed error of $\sigma = 0.5$ pixels is added to the line-image points and we introduce a variation in the Gaussian error added to the prior information: the prior known plane for the 3-points case and the prior known direction for the 2-points case. The abscissa axis corresponds to the Gaussian noise added to the prior information from $\sigma = 0$ degrees to $\sigma = 1$ degree. Notice that in this case the error when using the 4-points approach is almost constant because we are introducing a variation in the prior noise which does not affect the unconstrained method, and it is only affected by noise on the line-image points ($\sigma = 0.5$ pixels). As expected the worst results are for the 2-points method which is the one using more prior information. Notice that when the vertical direction is known up to 0.5deg (typical accuracy of a commercial IMU) our proposals (3pt, 2pt) outperform the unconstrained approach (4pt).



(a) (b)

Figure 5: Precision and recall of the robust approach depending on the number of attempts for a ratio of inliers $\omega = 0.6$. (a) Precision, (b) Recall.

We also evaluate the robustness of the method presented in Section 4. We simulate the projection of 100 random lines with two different kind of noise. The projected points have Gaussian noise with $\sigma = 0.5 pixels$ representing the projection error. In addition, we have added spurious points following a uniform distribution. We denote the original points as inliers and the spurious points as outliers. The ratio between the generated inliers and outliers is denoted as ω and its fixed to 0.6. Not all the inlier points have the same influence when fitting the model because we first pre-filter the hypothesis with the baseline criterion. In Fig. 5 we show the results of precision and recall, where $precision = \frac{true\ positives}{true\ positives + false\ positives}$ and $recall = \frac{true\ positives}{true\ positives + false\ negatives}$. The vertical lines represent the value for the number of attempts with a probability of being successful $P = 0.99$.

Due to the nature of robust extraction, where a single false positive can condition the whole fitting, we are interested in having a high precision to the detriment of the recall. As expected, the number of attempts needed for an acceptable precision is inferior for the constrained methods although a little higher than the theoretical value.

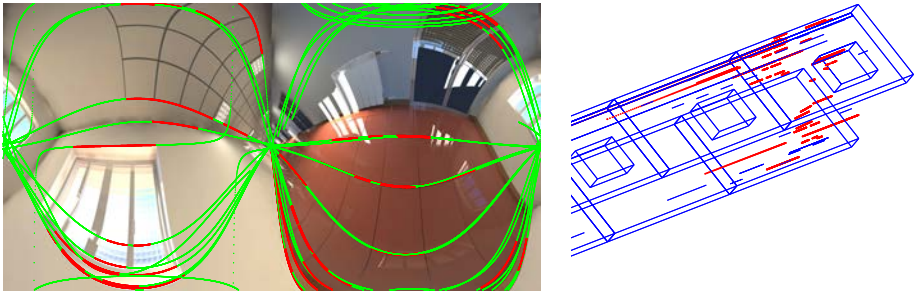
5.2 Line extraction with synthetic and real images

In this section we present the results of the pipeline presented in Section 4 obtained from synthetic and real images of a Manhattan scenario. The synthetic images have been generated with the raytracing software MegaPov [1] which allows defining parametric non-central cameras. The scenario is a modification of a publicly available synthetic scenario [2]. The synthetic panoramas have resolution of 4096×2048 pixels and the radius of the generation circle is $R_c = 0.5m$. In Fig. 6 we show the results of extracted line-images of the first horizontal dominant direction. The line projections are depicted in green and the supporting points in red (see Fig. 6 (a)). In the 3D view Fig. 6 (b) the ground truth is shown in blue and the extracted 3D segments in red.

We have also tested the proposal with real non-central images. For that we use the approximation described in [4]. The non-central panorama has a size of 4096×2048 pixels and it has been composed using a rotating off-axis fisheye camera (uEye UI-148xSE-C with lens Lensagon CF5M1414) with radius $R_c = 0.77m$. This panorama has been built stitching a stream of 222 frames with resolution of 32×1920 where each image cover a step size of 1.62

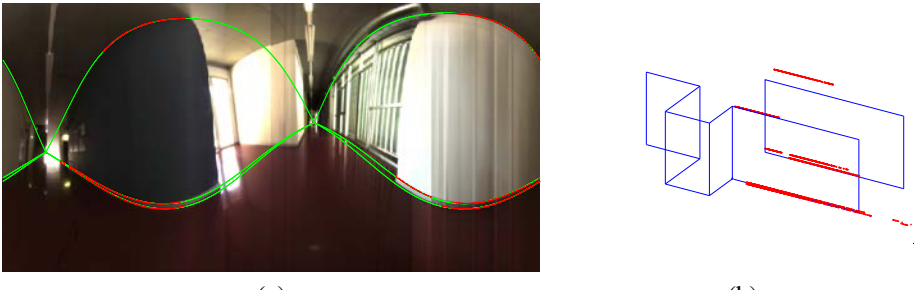
[1] megapov.inetart.net

[2] http://hof.povray.org/office-13.html, The Office - Jaime Vives Piqueres, 2004



(a) (b)

Figure 6: Extraction and reconstruction example from synthetic non-central panorama: (a) Extracted lines following the main direction over a non-central panorama (green for the parametric line and red for supporting points). (b) 3D view of the ground truth (in blue) and the reconstruction of the 3D line segments (in red).



(a) (b)

Figure 7: Extraction and reconstruction example from real non-central panorama: (a) Extracted lines following the main direction over a non-central panorama (green for the parametric line and red for supporting points). (b) 3D view of the ground truth (in blue) and the reconstruction of the 3D segments (in red)

degrees. The estimation of the vertical direction has been obtained from an IMU mounted on the camera. According to the IMU the horizontal plane in the camera reference is slanted 9.06 degrees. In Fig.7 we show an example of line-image extraction correctly extracted in a non-central panorama (the rest of the extracted lines have been omitted for facilitating the presentation). The line projections are depicted in green and the supporting points in red (see Fig. 7(a)). In Fig.7 (b) we show the extracted 3D segments in red and in blue the ground truth manually defined from measures taken with a laser electronic distance meter (EDM). The dimensions of the section of the corridor measured with the EDM are 3.72 m high and 3.18 m width, and the measures obtained from the metric reconstruction from the panorama are 3.70m height and 3.63m width.

6 Conclusions

In this paper, we present the minimal solution for computing a 3D line parallel to a given plane in non-central images. This approach can exploit prior information, e.g. from IMUs in Manhattan settings, to reduce the complexity of line-image extraction also improving the accuracy. In addition, we solve the minimal solution for computing a 3D line in non-central

cameras when the direction of this line is known. Both approaches are integrated in a pipeline for line extraction and reconstruction. The accuracy and robustness of the proposals are evaluated in simulation, concluding that the proposals outperform the unconstrained algorithm taking into account typical accuracy in prior information (standard commercial IMUs error is around 0.5 deg). Finally, the extraction method is tested with synthetic and real images.

Acknowledgement

This work was supported by Projects DPI2014-61792-EXP and DPI2015-65962-R (MINECO / FEDER, UE). First author was supported by the FPU program AP2010-3849.

References

[1] Amit Agrawal, Yuichi Taguchi, and Srikumar Ramalingam. Analytical forward projection for axial non-central dioptric and catadioptric cameras. In *European Conference on Computer Vision (ECCV)*, pages 129–143, 2010.

[2] Amit Agrawal, Yuichi Taguchi, and Srikumar Ramalingam. Beyond alhazen's problem: Analytical projection model for non-central catadioptric cameras with quadric mirrors. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2993–3000, 2011.

[3] Jesus Bermudez-Cameo. *Line-Projection in Omnidirectional Vision: Modelling, extraction and calibration in central and non-central cameras*. PhD thesis, Universidad de Zaragoza, 2016.

[4] Jesus Bermudez-Cameo, Joao P. Barreto, Gonzalo Lopez-Nicolas, and Jose. J Guerrero. Minimal solution for computing pairs of lines in non-central cameras. In *The 12th Asian Conference on Computer Vision (ACCV)*, 2014.

[5] Jesus Bermudez-Cameo, Gonzalo Lopez-Nicolas, and Jose J. Guerrero. Line-images in cone mirror catadioptric systems. In *22th International Conference on Pattern Recognition (ICPR)*, pages 2083–2088, 2014.

[6] Vincenzo Caglioti and Simone Gasparini. On the localization of straight lines in 3D space from single 2D images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 1129–1134, 2005.

[7] Vincenzo Caglioti, Simone Gasparini, and Pierluigi Taddei. Methods for space line localization from single catadioptric images: new proposals and comparisons. In *11th International Conference on Computer Vision (ICCV)*, pages 1–6, 2007.

[8] Vincenzo Caglioti, Pierluigi Taddei, Giacomo Boracchi, Simone Gasparini, and Alessandro Giusti. Single-image calibration of off-axis catadioptric cameras using lines. In *International Conference on Computer Vision (ICCV)*, pages 1–6, 2007.

[9] Wang Chen, Irene Cheng, Zihui Xiong, Anup Basu, and Maojun Zhang. A 2-point algorithm for 3D reconstruction of horizontal lines from a single omni-directional image. *Pattern Recognition Letters*, 32(3):524–531, 2011.

[10] Ondrej Chum and Jiri Matas. Matching with prosac-progressive sample consensus. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 220–226, 2005.

[11] Simone Gasparini and Vincenzo Caglioti. Line localization from single catadioptric images. *International Journal of Computer Vision*, 94(3):361–374, 2011.

[12] Phillip Griffiths and Joseph Harris. *Principles of algebraic geometry*, volume 52. John Wiley & Sons, 2011.

[13] Rajiv Gupta and Richard I Hartley. Linear pushbroom cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(9):963–975, 1997.

[14] Kenichi Kanatani. *Understanding geometric algebra: Hamilton, Grassmann and Clifford for Computer Vision and Graphics*. 2015.

[15] Jana Kosecka and Wei Zhang. Video compass. In *ECCV '02: 7th European Conference on Computer Vision*, pages 476–490. Springer, 2002.

[16] Douglas Lanman, Megan Wachs, Gabriel Taubin, and Fernando Cukierman. Reconstructing a 3D line from a single catadioptric image. In *Third International Symposium on 3D Data Processing, Visualization, and Transmission*, pages 89–96, 2006.

[17] David C Lee, Martial Hebert, and Takeo Kanade. Geometric reasoning for single image structure recovery. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2136–2143, 2009.

[18] Marc Menem and Tomás Pajdla. Constraints on perspective images and circular panoramas. In *British Machine Vision Conference (BMVC)*, pages 1–10, 2004.

[19] Pedro Miraldo and Helder Araujo. Planar pose estimation for general cameras using known 3d lines. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4234–4240, 2014.

[20] Pedro Miraldo, Helder Araujo, and Nuno Goncalves. Pose estimation for general cameras using lines. *IEEE Transactions on Cybernetics*, (99):1–1, 2015.

[21] Luis Perdigoto and Helder Araujo. Reconstruction of 3D lines from a single axial catadioptric image using cross-ratio. In *21th International Conference on Pattern Recognition (ICPR)*, pages 857–860, 2012.

[22] Carlo Pinciroli, Andrea Bonarini, and Matteo Matteucci. Robust detection of 3D scene horizontal and vertical lines in conical catadioptric sensors. In *Proceedings of the 6th workshop on omnidirectional vision, camera networks and non-classical cameras (OMNIVIS)*, 2005.

[23] Helmut Pottmann and Johannes Wallner. *Computational line geometry*. Springer, 2001.

[24] Rahul Raguram, Ondrej Chum, Marc Pollefeys, Jiri Matas, and Jan-Michael Frahm. Usac: a universal framework for random sample consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):2022–2038, 2013.

[25] Srikumar Ramalingam and Matthew Brand. Lifting 3D manhattan lines from a single image. In *IEEE International Conference on Computer Vision (ICCV)*, pages 497–504, 2013.

[26] Olivier Saurer, Pascal Vasseur, Rémi Boutteau, Cédric Demonceaux, Marc Pollefeys, and Friedrich Fraundorfer. Homography based egomotion estimation with a common direction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[27] John Greenlees Semple and Geoffrey Thomas Kneebone. *Algebraic projective geometry*. Oxford University Press, 1998.

[28] Rahul Swaminathan, Michael D Grossberg, and Shree K Nayar. Non-single viewpoint catadioptric cameras: Geometry and analysis. *International Journal of Computer Vision*, 66(3):211–229, 2006.

[29] Rahul Swaminathan, Ao Wu, and Haoyuan Dong. Depth from distortions. In *The 8th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras-OMNIVIS*, 2008.

[30] Seth Teller and Michael Hohmeyer. Determining the lines through four lines. *Journal of graphics tools*, 4(3):11–22, 1999.

[31] Qi Zhang, Xiaoyong Shen, Li Xu, and Jiaya Jia. Rolling guidance filter. In *European Conference on Computer Vision–ECCV*, pages 815–830. Springer, 2014.