2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Self-orientation of a hand-held catadioptric system in man-made environments

L. Puig, J. Bermudez, J.J. Guerrero

Abstract— **In central catadioptric systems the 3D lines are projected into conics, actually degenerate conics. In this paper we present a new approach to extract the projected lines corresponding to straight lines in the scene and to compute vanishing points from them. Using the internal calibration and two image points we are able to compute the catadioptric image lines analytically. We exploit the presence of parallel lines in man-made environments to compute the dominant vanishing points in the omnidirectional image. In order to obtain the intersection of two of these conics to compute vanishing points we analyze the self-polar triangle common to this pair. With the information contained in the vanishing points we are able to obtain the self-orientation of a hand-held catadioptric system. This system can be used in a vertical stabilization system required by autonomous navigation or to rectify images required in applications where the vertical orientation of the catadioptric system is assumed. We test our approach performing vertical and full rectifications in real sequences of images.**

I. INTRODUCTION

In recent years the use of catadioptric omnidirectional cameras has increased among the robotics community considerably. The advantages of such systems are its wide field of view and the central single view point property. The former allows to minimize the possibility of fatal occlusions and partial views, helping the tracking of features. The latter allow us to calculate easily the directions of the light rays coming into the camera [1], helping the computation of 3D information from multiple views. In [2] an analysis of this kind of systems is presented and describe those systems which have the single view-point property. Among these we have the hyper-catadioptric system which is composed of an hyperbolic mirror and a perspective camera. In robotics when a catadioptric system is used it is commonly observed that it has a vertical orientation. This is because most robotic platforms used are wheel-based. Under this configuration planar-motion and/or 1D image geometry is assumed which reduces the degrees of freedom (DOF) of the problem [3]. Moreover, in applications where line tracking or line matching is performed this assumption is useful [4], [3]. Besides that there exist robot platforms where the vertical assumption is not satisfied and they require the development of new algorithms to interact with the environment. One of this algorithms can be a self-orientation system to be used for stabilization of a biologically-inspired humanoid robot platform [5]. One of the advantages of the non-vertical configuration is that both the horizontal and vertical vanishing points are present in the image and can be computed by the intersection of parallel lines. In man-made environments we can observe sets of parallel and orthogonal lines and planes that can be exploited to compute the orientation of the system [6]. However, the extraction of lines in catadioptric images becomes extraction of conics. Five collinear image points are required to extract them in the uncalibrated case. However, two points are enough if we take advantage of the internal calibration of the catadioptric system. We call these lines, catadioptric image lines (CILs). Some works have been proposed to deal with this problem. In [7], the space of the equivalent sphere which is the unified domain of central catadioptric sensors combined with the Hough transform is used. In [8] they also use the Hough transform and two parameters on the Gaussian sphere to detect the image lines. The accuracy on the detection of these two approaches depends on the resolution of the Hough transform. The higher the accuracy the more difficult to compute the CILs. In [9] the randomized Hough transform is used to overcome the singularity present in [7], [8] and to speed up the extraction of the conics. This scheme is compared in converge mapping to a RANSAC approach. In [10] an scheme of split and merge is proposed to extract the CILs present in a connected component. These connected components, as in our case, are computed in two steps. The first step consist of detecting the edges using the Canny operator. The second step is a process of chaining which builds the connected components. In contrast to [7], [8], [9] our approach does not use the Hough transform, instead we compute the CIL directly from two image points present in a connected component. Then a RANSAC approach is used to identify the points that belong to this conic. As opposed to [10] we use an estimation of the geometric distance from a point to a conic instead of an algebraic distance. Notice that a connected component can contain more than one CIL and the process has to be repeated until all CILs are extracted.

Once we have extracted the lines in the catadioptric images (CILs) we need to compute the intersection of parallel CILs to extract the vanishing points. In this paper we propose a modification to the computation of the common self-polar triangle [11] in order to compute the intersection between a pair of CILs. Instead of having four intersections points between two general conics we have just two in the case of CILs. When this intersection corresponds to parallel CILs these points are the vanishing points. We compute all the intersection between the CILs present in the image. Then

L. Puig, J. Bermudez and J.J. 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.

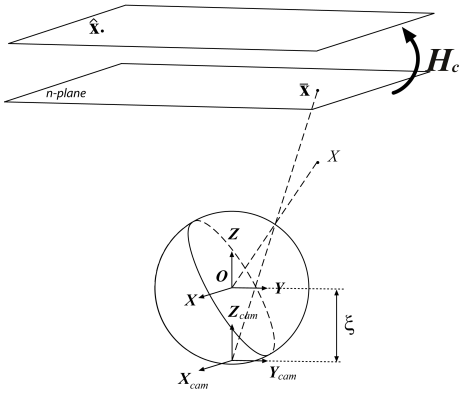


Fig. 1. Sphere camera model.

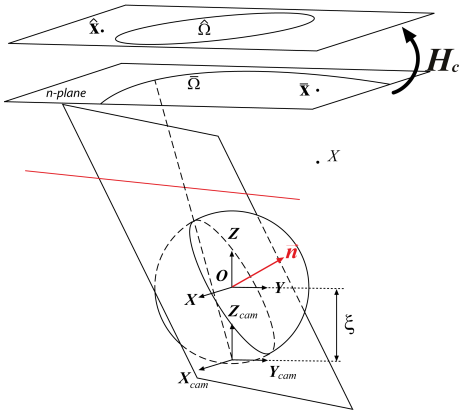


Fig. 2. Projection of a line under the sphere camera model

with a voting approach we robustly determine which ones are the vanishing points. The first vanishing point to compute is the vertical vanishing point (VVP) from which we are able to perform a rectification of the omnidirectional image. With this rectification we obtain an omnidirectional image fitting the vertical assumption and the applications designed with this constraint can be used. Using an analogous process we compute the horizontal vanishing point (HVP). From this HVP we compute the last angle that gives the whole orientation of the catadioptric system.

In section II we explain briefly the sphere camera model and the projection of a 3D line into the catadioptric image. In section III our proposal to extract CILs and its intersection is explained. In section IV the computation of the vanishing point and the rectification process is explained. In section V we show some experiments with the rectification and the orientation computing using images acquired by a walking person with a camera in hand. Finally in section VI we present the conclusions.

II. Sphere Camera Model

Under the sphere camera model [12] all catadioptric systems can be modelled by the unitary sphere and a perspective projection. The projection of a 3D point $\mathbf{X} = (X\ Y\ Z)^{\mathsf{T}}$ into an omnidirectional image point $\hat{\mathbf{x}}$ can be performed as follows (Fig. 1). First, the 3D point is associated with a projective ray $\mathbf{x}$ in the mirror reference system. This is done by P, a conventional projection matrix $\mathbf{x} = \mathsf{P}\mathbf{X}$. We assume the world reference system and the mirror reference system are the same $\mathsf{P} = [\mathtt{I}|\mathbf{0}]$. Second, the 3D ray is projected onto the sphere passing through its center and intersecting in two points $\mathbf{r}_{\pm}$. These points are then projected into an intermediate perspective plane with focal length equal to one, giving the points $\bar{\mathbf{x}}_{\pm}$, one of which is physically true. This step is encoded in the function $\hbar$ (1). The last step is the projection of these points into the omnidirectional image, which is performed by a collineation $\mathsf{H}_c(\hat{\mathbf{x}} = \mathsf{H}_c\bar{\mathbf{x}})$ [13]. Matrix H_c is the combination of the intrinsic parameters of the perspective camera K_c, the rotation between the camera and the mirror R_c and the shape of the mirror. This model considers all central catadioptric cameras, encoded by ξ, which is the distance between the center of the perspective projection and the center of the sphere. $\xi = 0$ for perspective, $\xi = 1$ for para-catadioptric and $0 < \xi < 1$ for hyper-catadioptric.

$$\hbar(\mathbf{x}) = \begin{pmatrix} x \\ y \\ z + \xi\sqrt{x^2 + y^2 + z^2} \end{pmatrix} \tag{1}$$

A. Projection of Lines in Catadioptric Systems

Let $\mathbf{\Pi} = (n_x, n_y, n_z, 0)^{\mathsf{T}}$ a plane defined by a 3D line and the effective view point in the sphere camera model $\mathbf{O}$ (see Fig. 2). The 2D line $\mathbf{n}$ associated to the 3D line by P can be represented as $\mathbf{n} = (n_x, n_y, n_z)^{\mathsf{T}}$. Then, the points $\mathbf{X}$ lying in the 3D line are projected to points $\mathbf{x}$. These points satisfy $\mathbf{n}^{\mathsf{T}}.\mathbf{x} = 0$ and $\mathbf{x} = \hbar^{-1}(\bar{\mathbf{x}})$, so $\mathbf{n}^{\mathsf{T}}.\hbar^{-1}(\bar{\mathbf{x}}) = 0$. As in [13], this equality can be written as

$$\bar{\mathbf{x}}^{\mathsf{T}}\bar{\Omega}\bar{\mathbf{x}} = 0 \tag{2}$$

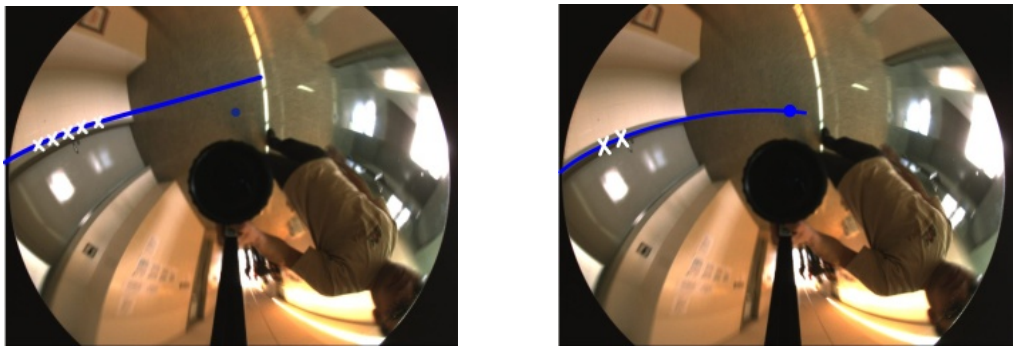
where the image conic is

$$\bar{\Omega} = \begin{pmatrix} n_x^2\left(1-\xi^2\right) - n_z^2\xi^2 & n_x n_y\left(1-\xi^2\right) & n_x n_z \\ n_x n_y\left(1-\xi^2\right) & n_y^2\left(1-\xi^2\right) - n_z^2\xi^2 & n_y n_z \\ n_x n_z & n_y n_z & n_z^2 \end{pmatrix} \tag{3}$$

Notice that $\bar{\Omega}$ is a degenerate conic when the 3D line is coplanar with the optical axis. We exploit this property to perform the detection of catadioptric image lines and to rectify the image.

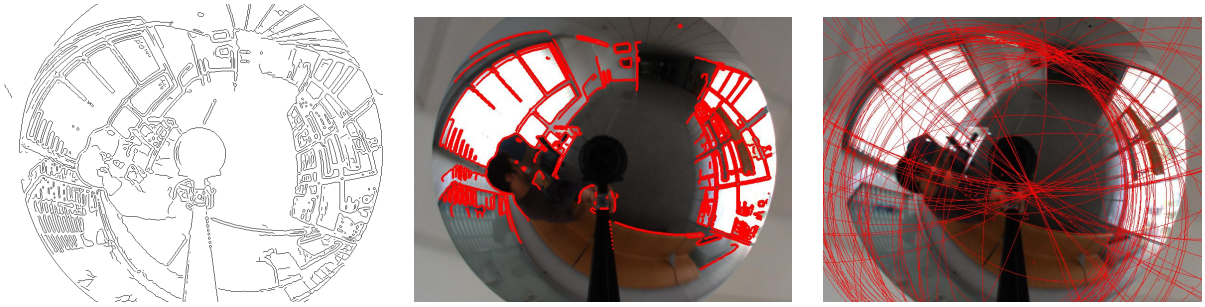
III. Catadioptric Image Lines Computing

In this section we explain the method used to extract the CILs from two image points. As mentioned before in the case of uncalibrated systems we require five points to describe a conic. If these points are not distributed in the whole conic, the estimation is not correctly computed. Another disadvantage of a 5-point approach is the number of parameters. When a robust technique is used, like RANSAC this is quite important, because the number of iterations required hardly increases with the number of parameters of



(a) (b)

Fig. 3. Computing a CIL with (a) using the five point approach. (b) using our approach with only two close points. The central blue point corresponds to the vertical vanishing point.



(a) (b) (c)

Fig. 4. Extraction of image lines (CILs). (a) Canny edge detector result, (b) connected components and (c) CILs extracted.

the model. Our approach overcomes these problems since two points are enough. As we assume the calibrated camera we can describe the conics using only two parameters and the calibration parameters, which allows to extract the CIL from 2 points. We compute the points in the normalized plane $\bar{\mathbf{x}} = (s\dot{x}\ s\dot{y}\ s)^{\mathsf{T}} = (\bar{x}\ \bar{y}\ 1)^{\mathsf{T}}$ using the inverse of matrix H_c

$$\bar{\mathbf{x}} = {\mathsf{H}_c}^{-1}\hat{\mathbf{x}}. \tag{4}$$

Developing (2) and after some algebraic manipulation we obtain

$$\begin{aligned}\left(1-\xi^2\right)\left(n_x\bar{x}+n_y\bar{y}\right)^2 + 2n_z\left(n_x\bar{x}+n_y\bar{y}\right)\ldots \\ +n_z^2\left(1-\xi^2\left(\bar{x}^2+\bar{y}^2\right)\right)=0\end{aligned} \tag{5}$$

simplifying

$$(1-\xi^2 r^2)\beta^2 + 2\beta + (1-\xi^2) = 0 \tag{6}$$

where a change of variable to $\beta = \dfrac{n_z}{n_x\bar{x}+n_y\bar{y}}$ and $r^2 = \bar{x}^2+\bar{y}^2$ is performed.

We can compute β by solving the quadratic equation

$$\beta = -\frac{1}{1-\xi^2 r^2} \pm \frac{\xi}{1-\xi^2 r^2}\sqrt{1+r^2\left(1-\xi^2\right)} \tag{7}$$

Once we have solved this quadratic equation we can compute the normal $\mathbf{n}$. Consider two points in the normalized plane $\bar{\mathbf{x}}_1 = (\bar{x}_1, \bar{y}_1, 1)^{\mathsf{T}}$ and $\bar{\mathbf{x}}_2 = (\bar{x}_2, \bar{y}_2, 1)^{\mathsf{T}}$. From (7) we compute the corresponding β_1 and β_2. Notice that there exist two solutions for β and just one has a physical meaning[1]. Using these parameters we obtain the linear system

$$\begin{pmatrix} \bar{x}_1 & \bar{y}_1 & -\frac{1}{\beta_1} \\ \bar{x}_2 & \bar{y}_2 & -\frac{1}{\beta_2} \end{pmatrix}\begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{8}$$

As $\mathbf{n}$ is orthonormal $n_x^2+n_y^2+n_z^2=1$. Solving for n_x, n_y and n_z we have

$$n_x = \frac{\bar{y}_1/\beta_2 - \bar{y}_2/\beta_1}{\nu} \tag{9}$$

$$n_y = \frac{\bar{x}_2/\beta_1 - \bar{x}_1/\beta_2}{\nu} \tag{10}$$

$$n_z = \frac{\bar{x}_2\bar{y}_1 - \bar{x}_1\bar{y}_2}{\nu} \tag{11}$$

with $\nu = \sqrt{(\bar{x}_2\bar{y}_1 - \bar{x}_1\bar{y}_2)^2 + (\bar{y}_1/\beta_2 - \bar{y}_2/\beta_1)^2 + (\bar{x}_2/\beta_1 - \bar{x}_1/\beta_2)^2}$

Notice that we have analytically computed the normal $\mathbf{n}$ that defines the projection plane of the 3D line (3). In Fig. 3 we show a comparison of the computing of a image line in the uncalibrated case using five points, and the calibrated case using our approach with only two points. In this figure we can observe that our approach obtains a better estimation even with two very close points. We also observe that the distance of the conic to the vanishing point using our 2-point approach is much better that the general 5-point approach.

[1]We have observed that the negative solution is the correct one.

A. Catadioptric Line Images Extraction

Our line extraction proposal can be explained as follows. First we detect the edges using the Canny algorithm. Then the connected pixels are stored in components. For each component we perform a RANSAC approach to detect the CILs present into this component. Two points from the connected component are chosen randomly and the corresponding CIL is computed. The distance from the rest of the points to this CIL is computed. The points with a distance smaller than some threshold vote for this CIL. The process stops when the number of points that has not voted for any conic and the number of points in the component are smaller than a threshold. In Fig. 4 we can observe the three main steps to extract the CILs.

B. Distance from a point to a conic

In contrast to [7], [8], [9], [10] that work with points in the unitary sphere we work in the normalized plane where conics are computed. In order to know if a point $\bar{\mathbf{x}}$ lies on a conic $\mathbf{C}$ we need to compute the distance from a point to a conic. Two distances are commonly used to this purpose. The algebraic distance defined by (12) which just gives an scalar value and the geometric distance which gives the distance from this point to the closest point on the conic. The geometric distance is calculated by solving a 4th order polynomial. This is time consuming and does not allow analytical derivation [14]. We propose an estimation to this distance replacing the point-to-conic distance by a point-to-point distance. Our proposal is based on the gradient of the algebraic distance from a point $\mathbf{x}_c$ to a conic represented as a 6-vector $\mathbf{C} = (c_1, c_2, c_3, c_4, c_5, c_6)$

$$d_{alg} = c_1x^2 + c_2xy + c_3y^2 + c_4x + c_5y + c_6. \tag{12}$$

We define the perpendicular line to a point that lies on the conic $\mathbf{C}$ as

$$\ell_\perp = \mathbf{x}_c + \lambda\tilde{\mathbf{n}}(\mathbf{x}_c) \tag{13}$$

where

$$\tilde{\mathbf{n}}\left(\mathbf{x}_c\right) = \frac{\nabla d_{alg}}{\|\nabla d_{alg}\|} \tag{14}$$

The normal vector $\tilde{\mathbf{n}}$ is computed from the gradient of the algebraic distance.

$$\nabla d_{alg} = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} 2c_1x + c_2y + c_4 \\ c_2x + 2c_3y + c_5 \end{pmatrix} \tag{15}$$

When a point does not lie on the conic $\mathbf{x}_o$ we can compute an estimation to its corresponding perpendicular line using the property that $\tilde{\mathbf{n}}(\mathbf{x}_c) = \tilde{\mathbf{n}}(\mathbf{x}_o + \Delta\mathbf{x}) \approx \tilde{\mathbf{n}}(\mathbf{x}_o)$

$$\ell_{est} = \mathbf{x}_o + \lambda_{est}\tilde{\mathbf{n}}(\mathbf{x}_o) = \begin{pmatrix} x_o + \lambda_{est}\tilde{n}_x(\mathbf{x}_o) \\ y_o + \lambda_{est}\tilde{n}_y(\mathbf{x}_o) \end{pmatrix} \tag{16}$$

To compute λ_{est} we substitute x by $x_o + \lambda_{est}\tilde{n}_x(\mathbf{x}_o)$ and y by $y_o + \lambda_{est}\tilde{n}_y(\mathbf{x}_o)$ in (12), giving a quadratic equation

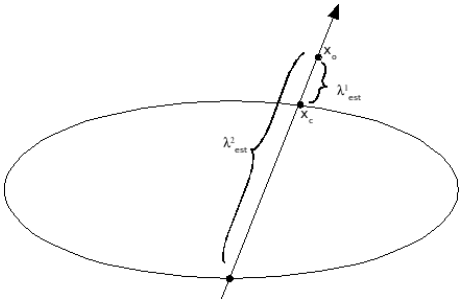


Fig. 5. Approximation to the distance from a point to a conic.

$$\begin{aligned} &\lambda_{est}^2 \underbrace{(c_1\tilde{n}_x^2 + c_2\tilde{n}_x\tilde{n_y} + c_3\tilde{n}_y^2)}_{a} + \\ &\lambda_{est} \underbrace{(2c_1\tilde{n}_x + 2c_3\tilde{n}_y + c_2(x_0\tilde{n}_y + y_0\tilde{n}_x))}_{b} + \\ &\underbrace{c_1x_0^2 + c_2x_0y_0 + c_3y_0^2 + c_4x_0 + c_5y_0 + c_6}_{c} = 0 \end{aligned} \tag{17}$$

We observe that λ_{est} gives the two distances that intersect the conic so, we choose the closest to $\mathbf{x}_o$ as the distance from that point to the conic $d = \|\mathbf{x}_o - \mathbf{x}_c\| = \lambda_{est}$.

IV. Vanishing points and Image Rectification

The vanishing points indicate the intersection of image lines corresponding to parallel lines in the scene. In vertical aligned catadioptric systems, vertical lines are radial lines in the image representation. Their intersection point, the vertical vanishing point (VVP), is located at the image center. When the camera is not vertically aligned, the radial lines become conic curves. In this case, their extraction and then the intersection between them become more difficult to compute. Another consequence is that the VVP moves from the image center. Its new location contains important information about the orientation of the camera with respect to the scene.

A. Intersection of Two CILs Using the Common Self-polar Triangle

In a general configuration, two conics intersect in four points. The intersection of these points define three distinct pair of lines. The intersection of these lines represent the vertices of the self-polar triangle common to a pair of conics [11]. We have studied the particular case where two CILs intersect, which is a degenerate configuration. These degenerate conics intersect in just two points. As we observe in Fig. 6, there exist a line $r(\mu)$ that intersects these two points and the origin of the normalized plane. Our goal is to compute this line and from it to extract the two intersections of the conics that correspond to the two points $\mathbf{P}^+$ and $\mathbf{P}^-$.

Let $\mathbf{n}_1 = (n_{x_1}, n_{y_1}, n_{z_1})^\mathsf{T}$ and $\mathbf{n}_2 = (n_{x_2}, n_{y_2}, n_{z_2})^\mathsf{T}$ two normal vectors representing the projection of two lines in the scene and $\bar{\Omega}_1$ and $\bar{\Omega}_2$ two conics representing the image lines in the normalized plane. The vertices of the self-polar triangle associated to the pencil $\bar{\Omega}(\lambda) = \bar{\Omega}_1 + \lambda\bar{\Omega}_2$ fit the constraint

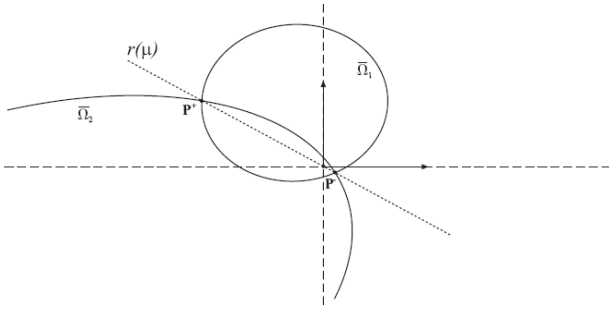


Fig. 6. Intersection of two CILs in the normalized plane.

$$\det(\bar{\Omega}_1 + \lambda\bar{\Omega}_2) = 0. \tag{18}$$

If we develop this constraint we obtain a third order polynomial where just one of the solutions is real and it corresponds to $\lambda_1 = -n_{z1}^2/n_{z2}^2$. So, the null-space of $\bar{\Omega}(\lambda_1) = \bar{\Omega}_1 + \lambda_1\bar{\Omega}_2$ is the line r, expressed in a parametric way as

$$r = \mu \cdot \mathbf{v} = \mu \begin{pmatrix} v_x \\ v_y \end{pmatrix} = \mu \begin{pmatrix} n_{z_2}^2 n_{y_1} n_{z_1} - n_{z_1}^2 n_{y_2} n_{z_2} \\ n_{z_1}^2 n_{x_2} n_{z_2} - n_{z_2}^2 n_{x_1} n_{z_1} \end{pmatrix}. \tag{19}$$

The intersection of this line to both $\bar{\Omega}_1$ and $\bar{\Omega}_2$ gives the two points $\mathbf{P}^+$ and $\mathbf{P}^-$. To obtain them we solve for μ in the following equation

$$\mu^2(c_1 v_x^2 + c_2 v_x v_y + c_3 v_y^2) + \mu(c_4 v_x + c_5 v_y) + c_6 = 0 \tag{20}$$

and substitute in (19).

B. Vertical Vanishing Point (VVP)

We use a classic algorithm to detect the VVP. Let m be the number of putative vertical CILs detected in the omnidirectional image and let $\mathbf{n}_i$ their corresponding representation in the normalized plane. For every pair of CILs (there is a total of $m(m-1)/2$ pairs), we compute their intersection as explained above. Then for each line $\mathbf{n}_i$ we compute the distance to these points. If the line is parallel to that pair of CILs the distance is smaller than a threshold and then that line votes that possible VVP. The most voted point is considered the VVP. A refinement of the estimation can be performed using the p lines that voted for the VVP. This refinement can be performed using singular value decomposition to solve a linear system, followed by an optimization process to improve the accuracy. As these steps also increase the computational cost, we decide to avoid them in our final implementation.

C. Image Rectification

Here we explain the relation between the VVP computed in the normalized plane and the orientation of the catadioptric system. Writing the VVP in polar coordinates $\bar{\mathbf{x}}_{vp} = (\rho_{vp}, \theta_{vp})^\mathsf{T}$ (see Fig. 7(d)) we observe that there exist a relation between the angle θ_{vp} and the angle ψ representing the rotation of the catadioptric system around the z-axis (21). The negative angle is produced by the mirror effect which inverts the catadioptric image.

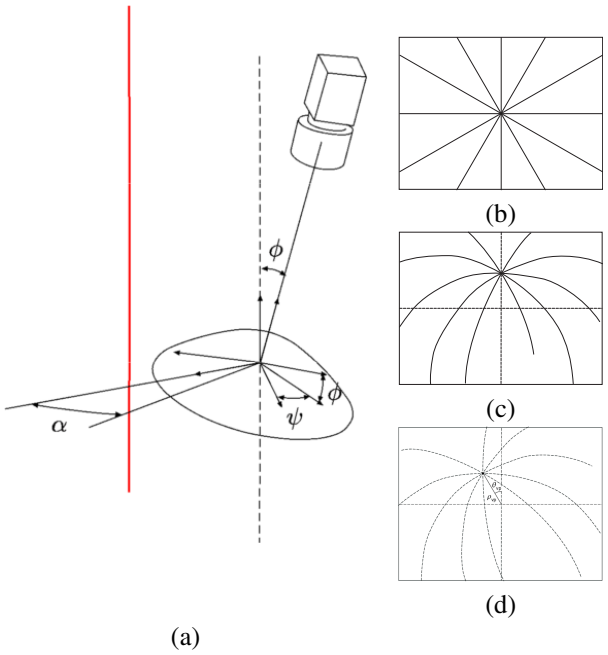


Fig. 7. (a) Configuration of the catadioptric system in a hand-held situation. (b) The VVP is in the center of the image. (c) The VVP moves in the vertical axis when the camera rotates around the x-axis. (d) The VVP rotates around the image center when the camera rotates around the z-axis.

$$\psi = -\theta_{vp} \tag{21}$$

We observed that the component ρ_{vp} is intrinsically related to the rotation angle ϕ and the mirror parameter ξ of the catadioptric system. Since angles ϕ and ψ are independent, we consider the case where $\psi = 0$ (see Fig. 7(c)). Using (19) and (20) with a pair of parallel CILs in polar coordinates we compute the following relationship

$$\rho_{vp} = -\frac{\sin\phi}{\cos\phi \pm \xi}. \tag{22}$$

A lookup table can be built to speed up the computing of the vertical orientation. An analogous process is performed to detect the horizontal vanishing point. With the information provided by this point we are able to compute the full orientation of the catadioptric system.

V. Experiments

In this section we present some experiments rectifying real images. We acquire two image sequences with a calibrated hand-held hyper-catadioptric system[2]. The calibration was performed using [15]. The process to extract the vanishing points and to perform the rectification can be summarized as follows:

1) The edges are detected by the Canny operator.
2) The process to construct the connected components is performed.
3) A RANSAC approach is performed for each connected component to extract all CILs present on it.
4) All CIL intersections are computed and the vanishing points are estimated.

[2]http://www.neovision.cz/

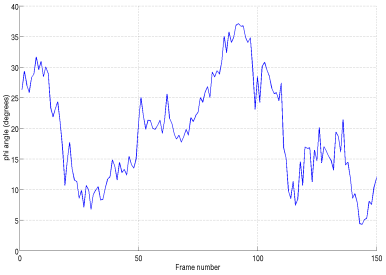


Fig. 8. Elevation angle ϕ of the catadioptric system. Frames 1-150.

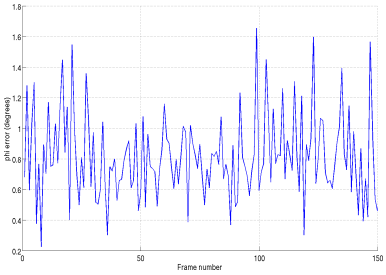


Fig. 9. Angular deviation of the vertical lines after rectification, in degrees.

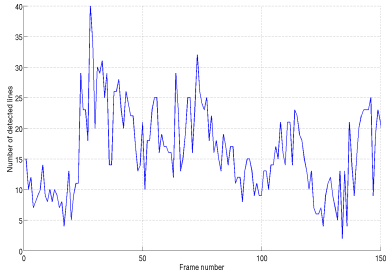


Fig. 10. Number of vertical lines present in each frame of sequence 1.

5) The vertical correction is performed using the VVP.
6) The full rectification is performed using the HVP.

In the first experiment we perform the vertical rectification on the first 150 frames of the first sequence. In Fig. 8 we show the ϕ angle computed. This angle describes the elevation angle of the catadioptric system through the sequence. To measure the accuracy of our approach we compute the verticality of the lines present in every frame. This process consist of building the corresponding panoramic image from the omnidirectional image. Then all points belonging to a vertical line are used to compute the line equation. Finally we measure the angle deviation of this computed line with a true vertical line. The average error computed was $0.72°$ with a maximum error of $1.63°$ and a standard deviation of $0.70°$. In Fig. 9 we show the average error of the angle deviation of all the vertical lines present in each frame of the sequence 1 after the rectification. We observe that this error is related to the number of vertical CILs present in the catadioptric image (Fig. 10). The more the number of vertical CILs present in the frame the better the estimation of the vanishing points and consequently a better rectification of the image. In Fig. 11 we observe the vertical rectification using only the information provided by the VVP[3].

In the second experiment we compute the full rectification of the second sequence using the vertical and the horizontal vanishing points. In Fig. 12 we can observe how the two vanishing points are computed. Once the rectification is computed we align the images to the reference system given by the vanishing points, i.e., the scene reference system. This allows to see how the only movement present in the sequence is a translation (see Fig. 13).

VI. Conclusions

We have presented a new way to extract lines in omnidirectional images generated by a calibrated catadioptric system. We use just two points lying on the CIL and an approximation to the geometric distance from a point to a conic. We also show how to compute the intersection of two image lines based on the common self-polar triangle. To show the effectiveness of this approach we perform experiments with real images. We perform the vertical rectification in order to create images where applications that require the vertical constraint can be used. We also compute the orientation of a hand-held hyper-catadioptric system through a video sequence. In this case the tracking of just one feature will be needed to compute the translation of the catadioptric system.

[3]See video attachment where the rectification is performed for the 150 frames.

References

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

[2] S. Baker and S. Nayar, "A theory of single-viewpoint catadioptric image formation," *Int. J. Comput. Vision*, vol. 35(2), pp. 175–196, 1999.

[3] J. J. Guerrero, A. C. Murillo, and C. Sagües, "Localization and matching using the planar trifocal tensor with bearing-only data," *IEEE Transactions on Robotics*, vol. 24(2), pp. 494–501, 2008.

[4] Y. Mezouar, H. Abdelkader, P. Martinet, and F. Chaumette, "Central catadioptric visual servoing from 3d straight lines," in *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 1, Sept.-2 Oct. 2004, pp. 343–348.

[5] R. Miyauchi, N. Shiroma, and F. Matsuno, "Development of omnidirectional image stabilization system using camera posture information," in *Robotics and Biomimetics, 2007. ROBIO 2007. IEEE International Conference on*, Dec. 2007, pp. 920–925.

[6] J. Kosecka and W. Zhang, "Video compass," in *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part IV*. London, UK: Springer-Verlag, 2002, pp. 476–490.

[7] P. Vasseur and E. M. Mouaddib, "Central catadioptric line detection," in *British Machine Vision Conference*, September 2004.

[8] X. Ying and Z. Hu, "Catadioptric line features detection using hough transform," in *ICPR 2004. Proceedings of the 17th International Conference on*, vol. 4, Aug. 2004, pp. 839–842.

[9] C. Mei and E. Malis, "Fast central catadioptric line extraction, estimation, tracking and structure from motion," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, Oct. 2006, pp. 4774–4779.

[10] J. C. Bazin, C. Demonceaux, and P. Vasseur, "Fast central catadioptric line extraction," in *IbPRIA '07: Proceedings of the 3rd Iberian conference on Pattern Recognition and Image Analysis, Part II*, 2007, pp. 25–32.

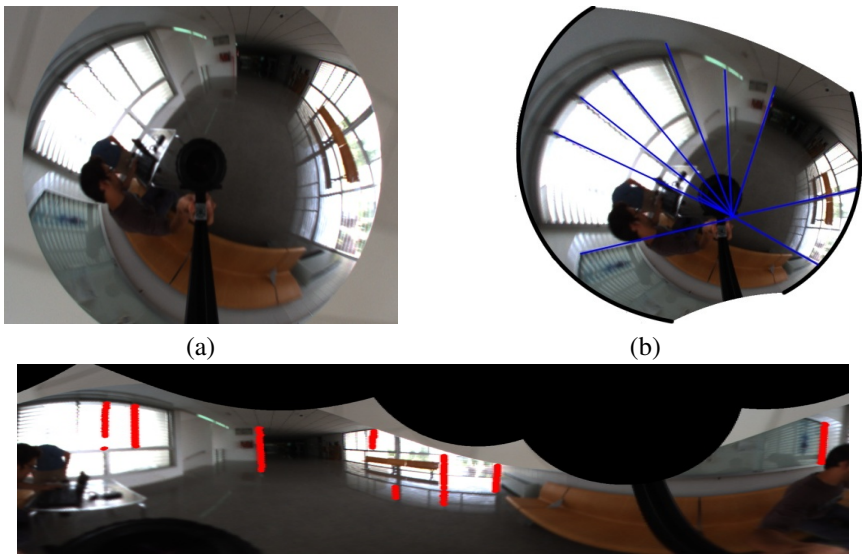
[11] J. Barreto, "General central projection systems: Modeling, calibration and visual servoing," Ph.D. dissertation, 2003.

[12] C. Geyer and K. Daniilidis, "A unifying theory for central panoramic systems and practical applications," in *ECCV (2)*, 2000, pp. 445–461. [Online]. Available: citeseer.ist.psu.edu/geyer00unifying.html

[13] J. P. Barreto and H. Araujo, "Geometric properties of central catadioptric line images and their application in calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1327–1333, 2005.

[14] P. Sturm and P. Gargallo, "Conic fitting using the geometric distance," in *Proceedings of the Asian Conference on Computer Vision, Tokyo, Japan*. Springer, 2007. [Online]. Available: http://perception.inrialpes.fr/Publications/2007/SG07

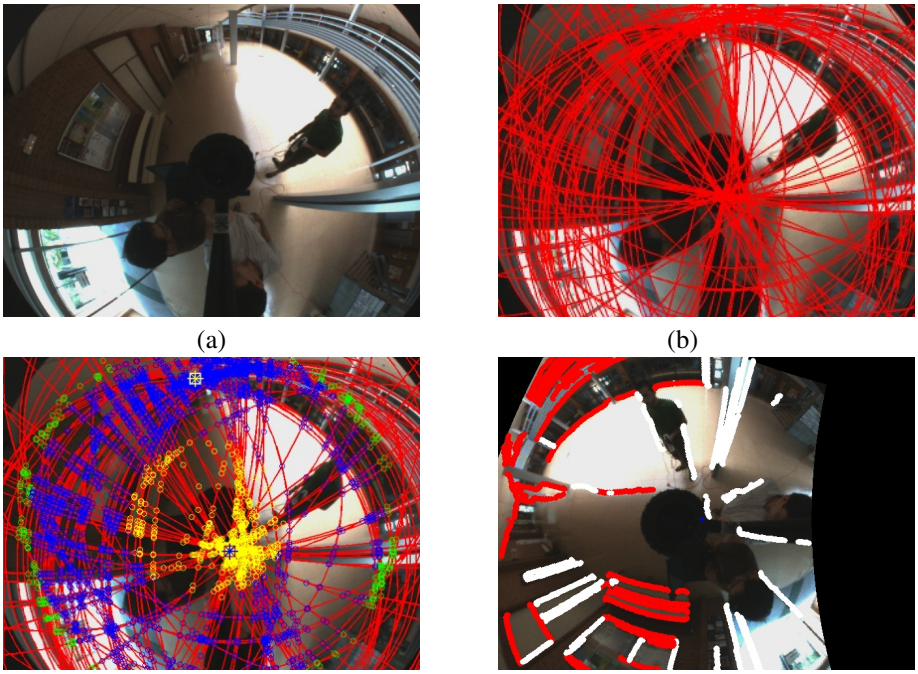
[15] C. Mei and P. Rives, "Single viewpoint omnidirectional camera calibration from planar grids," in *ICPR*, 2007, pp. 3945–3950.



(a) (b)

(c)

Fig. 11. Example of vertical image rectified. (a) Original omnidirectional image. (b) Rectified image with the vertical CILs passing through the VVP. (c) Panoramic representation showing the vertical lines in red.



(a) (b)

(c) (d)

Fig. 12. Example of full image rectification. (a) Frame 1 of the sequence 2. (b) Conic extraction using our approach. (c) Putative vertical and horizontal vanishing points. The yellow circles represent the putative vertical vanishing points. The blue ones the putative horizontal vanishing points and the green ones are the intersections points that cannot be consider either vertical or horizontal vanishing points. The white square is the estimated HVP and the black one is the VVP. (d) Full-rectified image. The vertical CILs are shown in white and the horizontal ones in red. See color version.

Frame 107 Frame 242

Fig. 13. Panoramic representation of several full-rectified frames. Vertical lines are shown in white and horizontal ones in red. The horizontal vanishing point is aligned to the image center.