Hypercatadioptric Line Images for 3D Orientation and Image Rectification

J. Bermudez-Cameo[a,*], Luis Puig[a], J.J. Guerrero[a]

[a]*Instituto de Investigación en Ingeniería de Aragón-Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza, Spain*

Abstract

In central catadioptric systems 3D lines are projected into conics. In this paper we present a new approach to extract conics in the raw catadioptric image, which correspond to projected straight lines in the scene. Using the internal calibration and two image points we are able to compute analytically these conics which we name hypercatadioptric line images. We obtain the error propagation from the image points to the 3D line projection in function of the calibration parameters. We also perform an exhaustive analysis on the elements that can affect the conic extraction accuracy. Besides that, we exploit the presence of parallel lines in man-made environments to compute the dominant vanishing points (VPs) in the omnidirectional image. In order to obtain the intersection of two of these conics we analyze the self-polar triangle common to this pair. With the information contained in the vanishing points we are able to obtain the 3D orientation of the catadioptric system. This method can be used either 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 use synthetic and real images to test the proposed method. We evaluate the 3D orientation accuracy with a ground truth given by a goniometer and with an inertial measurement unit (IMU). We also test our approach performing vertical and full rectifications in sequences of real images.

1. Introduction

Omnidirectional cameras are devices designed to capture images with a wide field of view. This characteristic introduces a new approach in computer vision minimizing the possibility of fatal occlusions and helping the tracking of features. Among these cameras we find the catadioptric systems, which are a combination of a mirror and a camera. Some of these systems have the single viewpoint constraint. In Baker and Nayar [1] an analysis of central catadioptric systems is presented describing which systems have the single view point property. Among these we have the hypercatadioptric system which is composed of a hyperbolic mirror and a perspective camera. In mobile 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 is assumed simplifying the localization algorithms. In applications where line tracking or line matching is performed [2, 3, 4] this assumption is useful since vertical lines are coplanar with the optical axis becoming straight radial lines in the catadioptric image. When this assumption is not satisfied, vertical lines are projected as conics in the catadioptric image. This situation requires the development of new algorithms capable to deal with these conic projections. Another advantage of the wide field of view in catadioptric systems is that visible segments of projected lines are very large. When we use catadioptric systems in man-made environments we can observe sets of parallel and orthogonal lines. These sets of lines encapsulate geometrical information which we exploit in this work. In particular, vanishing points contain the orientation of the camera with respect to the reference coordinate system defined by the main directions of the environment. However, dealing with line projections in catadioptric images becomes

*Corresponding author

Email addresses: bermudez@unizar.es (J. Bermudez-Cameo), lpuig@unizar.es (Luis Puig), jguerrer@unizar.es (J.J. Guerrero)

extraction of conics. In general, five points are required to determine uniquely a conic. When the internal calibration of the central catadioptric system is known, only two points are needed to compute these conics, which we particularly call Hypercatadioptric Line Images (HLIs).

Line extraction in omnidirectional catadioptric systems has been studied previously. In [5] an exhaustive analysis of line images in paracatadioptric cameras is presented for calibration purposes. In [6], the space of the equivalent sphere which is the unified domain of central catadioptric sensors combined with the Hough transform is used. They also perform tests showing the influence of the calibration parameters in their approach. In [7] Ying and Hu also use the Hough transform and two parameters on the unitary 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 it is to compute the lines. In [8] the randomized Hough transform is used to overcome the singularity present in [6, 7], which speeds up the extraction of conics. This scheme is compared in converge mapping to a RANSAC approach. In [9, 10] an scheme of split and merge is proposed to extract the image lines present in a connected component. The connected components are computed in two steps. The first step consists of detecting the edges using the Canny operator. The second step is a process of chaining which builds the connected components.

A vanishing point (VP) is the point on the image where the projections of two or more parallel 3D lines intersect. This point is the projection of a virtual 3D point located at infinity. That means that coordinates of a VP on the image only changes when the camera is rotated. This property is exploited to obtain the orientation of the camera with respect to a global reference system. The relationship between the orientation of the camera and the vanishing points of perspective images has been deeply studied before[11]. Due to the wide field of view of omnidirectional cameras VPs are commonly located inside the catadioptric image. For this reason VP-based techniques are interesting with omnidirectional cameras. In [10] VPs are used with omnidirectional images to estimate the camera orientation. In [12] vanishing points are used in omnidirectional sequences as orientation feature in an Extended Kalman Filter. A method of expectation maximization is used in [13] to classify and detect VPs from edges on the Gaussian Sphere.

1.1. Contributions

In this paper we present a method to extract HLIs in the raw image in contrast to the approaches presented in [6, 7, 8, 9] which execute the fitting on the unitary sphere. Our approach does not use the Hough transform so the computing of the HLI is made directly from two image points. Extracting the HLIs directly on the image, means that thresholds and fitting parameters are expressed in pixel units. An example of this is the metric used to decide if a point belongs to a conic. Instead of using an algebraic distance we propose an approximation to the geometrical distance on the image. A RANSAC voting scheme is used to extract all the HLIs present in a single connected component. Contrary to [9], the extraction of one HLI is performed in one step. We also perform an exhaustive analysis on the elements that can affect the HLIs extraction accuracy. Since we work on the image it allows measuring the influence and behaviour caused by calibration errors in pixels. This study is applicable to single HLIs in opposition to the study performed in [6] which shows the influence of calibration parameters in a collection of projected lines. The way the results are presented in [6] is conditioned by the Hough transform approach. To complete the analysis we present the propagation error formulas from calibration parameters to the spatial representation of the 3D line projection. Analogous formulation and analysis is performed with respect to the HLI defining points on the image.

Once lines in the catadioptric images have been extracted, vanishing points are estimated by computing the intersection of parallel lines. We propose a modification to the computation of the common self-polar triangle [14] to compute the intersection between a pair of HLIs. The intersection of two HLIs is a particular case in which the number of intersections between conics is reduced from four to two points. When this intersection corresponds to images of parallel lines these points are the vanishing points. We compute all the intersections between the HLIs present in the image. Then with a voting approach we robustly determine which ones are the vanishing points. The Vertical Vanishing Point (VVP) allows us to determine two of the three orientation angles with respect to an absolute reference. Using an analogous process we compute the horizontal vanishing points (HVP)and as consequence the whole orientation of the catadioptric system. This orientation can be used to perform a rectification of the omnidirectional image.

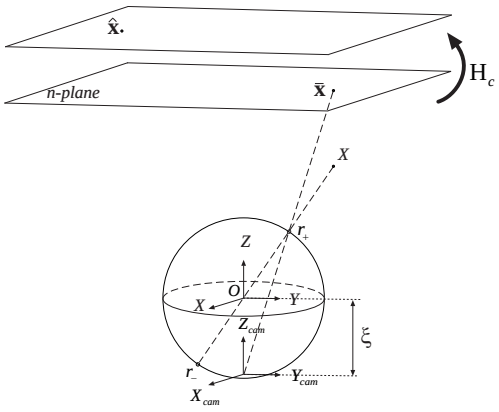


Figure 1: Sphere camera model.

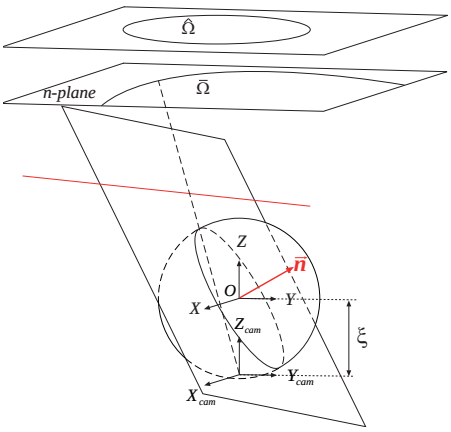


Figure 2: Projection of a line under the sphere camera model.

The rest of the paper is distributed as follows. In section 2 we describe the unified theory to project points and lines in catadioptric systems [15]. In section 3 we present a new approach to describe conics which are projections of lines with two points and we show an algorithm to extract them from a hypercatadioptric image. Section 4 presents an analysis of the influence of calibration on the extraction procedure and an analysis of the uncertainty propagation in the HLI computation. Section 5 describes how to use these conics to estimate vanishing points assuming dominant directions, in order to compute the orientation of the camera. Some experiments are presented in section 6 to validate the proposed method. Finally, we present the conclusions in section 7. Part of this work has been presented in [16] focused in orientation computation and [17] centered in the evaluation of the conic extraction. Here, the proposed method is revised and completed including an analytic study of the error propagation in the computation of the HLI. Several experiments using synthetic and real images were also performed giving a deeper validation of the method.

2. Projections in Central Catadioptric Systems

Under the sphere camera model [15] all central catadioptric systems can be modelled by a projection to the unitary sphere followed by a perspective projection. The projection of a 3D point $\mathbf{X} = (X\ Y\ Z\ 1)^{\mathsf{T}}$ into an omnidirectional image point $\hat{\mathbf{x}}$ can be performed as follows (Fig. 1). First, the 3D point is associated with a 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 H_c [18]. 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 assumed the identity $\mathtt{I}$ and the shape of the mirror M_c. 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 paracatadioptric and $0 < \xi < 1$ for hypercatadioptric.

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

$$\mathsf{H}_c = \underbrace{\begin{pmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathsf{K}_c} \underbrace{\begin{pmatrix} \psi - \xi & 0 & 0 \\ 0 & \xi - \psi & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathsf{M}_c} \tag{2}$$

Let $\mathbf{\Pi} = (n_x, n_y, n_z, 0)^\mathsf{T}$ be a plane defined by a 3D line and the effective view point in the sphere camera model $\mathbf{O}$ (see Fig. 2). The projected 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 [18], this equality can be written as

$$\bar{\mathbf{x}}^\mathsf{T}\bar{\Omega}\bar{\mathbf{x}} = 0 \tag{3}$$

where the image conic is

$$\bar{\Omega} = \begin{pmatrix} n_x^2\left(1-\xi^2\right) - n_z^2\xi^2 & n_x n_y\left(1-\xi^2\right) & n_x n_z \\ n_x n_y\left(1-\xi^2\right) & n_y^2\left(1-\xi^2\right) - n_z^2\xi^2 & n_y n_z \\ n_x n_z & n_y n_z & n_z^2 \end{pmatrix} \tag{4}$$

and the image of the conic in the catadioptric image is

$$\hat{\Omega} = \mathsf{H}_c^{-\mathsf{T}}\bar{\Omega}\mathsf{H}_c^{-1}. \tag{5}$$

Notice that $\bar{\Omega}$ is a degenerate conic when the 3D line is coplanar with the optical axis [18].

3. Computation of Hypercatadioptric Line Images

In this section we present the process used to extract HLIs contained in hypercatadioptric images. First we show the analytical process, which requires only two image points and the mirror parameter. Then we introduce the proposed point-to-conic metric used to decide if a point lies on a conic. Then the whole extraction process is depicted as pseudo-code. Finally we show a comparison of the five-point algorithm with our approach, which distinguishes between conics coming from straight lines and from other projected conics.

3.1. Conic definition using two points

Here we explain the method used to extract the HLIs from two image points. As mentioned before, in the case of uncalibrated systems we require five points to compute a conic. If these points are not well distributed in the whole conic, the estimation is difficult and usually inaccurate. Another disadvantage of a 5-point approach is the number of parameters. When a robust technique like RANSAC is used, this is quite important, because the number of iterations required considerably increases with respect to the number of parameters of the model. Our approach overcomes these problems requiring only two points and the calibration parameters. First we transform the points to the normalized plane $\bar{\mathbf{x}} = (\bar{x}\ \bar{y}\ \bar{z})^\mathsf{T}$ using the inverse of matrix H_c

$$\bar{\mathbf{x}} = \mathsf{H}_c^{-1}\hat{\mathbf{x}}. \tag{6}$$

Developing (3) and after some algebraic manipulation we obtain

$$\left(1-\xi^2\right)\left(n_x\bar{x} + n_y\bar{y}\right)^2 + 2n_z\bar{z}\left(n_x\bar{x} + n_y\bar{y}\right) + n_z^2\left(\bar{z}^2 - \xi^2\left(\bar{x}^2 + \bar{y}^2\right)\right) = 0 \tag{7}$$

simplifying

$$(1-\xi^2)\alpha^2 + 2\alpha + \left(\bar{z}^2 - \xi^2\left(\bar{x}^2 + \bar{y}^2\right)\right) = 0 \tag{8}$$

where a change of variable to $\alpha = \dfrac{n_x\bar{x} + n_y\bar{y}}{n_z}$ is performed.

We can compute α by solving the quadratic equation

$$\alpha = -\frac{\bar{z}}{1-\xi^2} \pm \frac{\xi}{1-\xi^2}\sqrt{\bar{z}^2 + (\bar{x}^2 + \bar{y}^2)(1-\xi^2)} \tag{9}$$

Once we have solved this quadratic equation we can compute the normal $\mathbf{n}$. Consider two points in the normalized plane $\bar{\mathbf{x}}_1 = (\bar{x}_1, \bar{y}_1, \bar{z}_1)^\mathsf{T}$ and $\bar{\mathbf{x}}_2 = (\bar{x}_2, \bar{y}_2, \bar{z}_2)^\mathsf{T}$ lying on the conic we want to extract. From (9) we compute the corresponding α_1 and α_2. Notice that there exist two solutions for α and just one has a physical meaning[1]. Using these parameters we obtain the linear system

$$\begin{pmatrix} \bar{x}_1 & \bar{y}_1 & -\alpha_1 \\ \bar{x}_2 & \bar{y}_2 & -\alpha_2 \end{pmatrix} \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{10}$$

Solving for n_x, n_y and n_z we have

$$n_x = \bar{y}_1\alpha_2 - \bar{y}_2\alpha_1 \qquad n_y = \bar{x}_2\alpha_1 - \bar{x}_1\alpha_2 \quad \text{and} \quad n_z = \bar{x}_2\bar{y}_1 - \bar{x}_1\bar{y}_2. \tag{11}$$

Notice that we have analytically obtained the normal $\mathbf{n}$ that defines the projection plane of the 3D line, therefore the conic $\bar{\Omega}$ described in (4).

3.2. Distance from a point to a conic

In contrast to previous works [6, 7, 8, 9] that deal with points in the unitary sphere, we work in the image plane where conics are computed. In order to know if a point $\mathbf{x}$ lies on a conic Ω, represented as a 6-vector $\mathbf{c} = (c_1, c_2, c_3, c_4, c_5, c_6)$, we need to compute the distance from a point to a conic. Two distances are commonly used to this purpose. An algebraic distance defined by (12) which just gives a value without clear geometrical meaning

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

The other gives the geometric 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 [19]. We propose an approximation to this distance replacing the point-to-conic distance by a point-to-point distance.

Given a conic point $\mathbf{x}_c$ lying on a conic described by the matrix Ω the tangent line to the conic passing through $\mathbf{x}_c$ is the polar line of the point $\mathbf{x}_c$ with respect to the conic Ω.

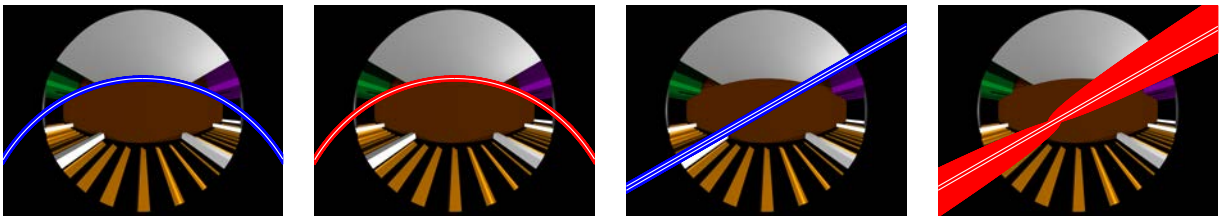
$$s = \Omega\mathbf{x}_c = \begin{pmatrix} c_1 & \frac{c_2}{2} & \frac{c_4}{2} \\ \frac{c_2}{2} & c_3 & \frac{c_5}{2} \\ \frac{c_4}{2} & \frac{c_5}{2} & c_6 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ 1 \end{pmatrix} = \begin{pmatrix} c_1x_c + \frac{c_2}{2}y_c + \frac{c_4}{2} \\ \frac{c_2}{2}x_c + c_3y_c + \frac{c_5}{2} \\ \frac{c_4}{2}x_c + \frac{c_5}{2}y_c + c_6 \end{pmatrix} \tag{13}$$

The perpendicular line to the conic Ω passing through $\mathbf{x}_c$ is defined as

$$l_\perp(\mathbf{x}_c) = \begin{pmatrix} s_2 \\ -s_1 \\ -s_2x_c + s_1y_c \end{pmatrix} \tag{14}$$

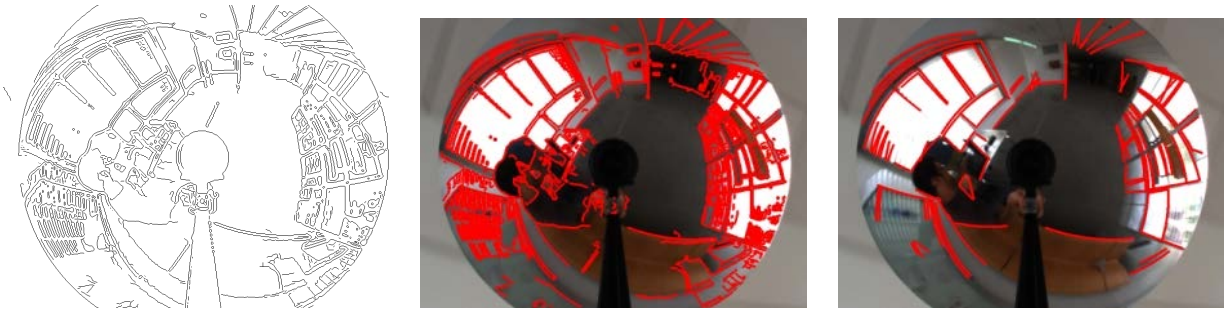
When a point $\mathbf{x}_o$ does not lie on the conic we can compute an estimation to its corresponding perpendicular line using the property that $l_\perp(\mathbf{x}_c) = l_\perp(\mathbf{x}_0 + \Delta\mathbf{x}) \approx l_\perp(\mathbf{x}_0)$.

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



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

Figure 3: Region defined by a metric threshold of 20 pixels surrounding a conic. (a) Ellipse case using metric distance. (b) Ellipse case using Sampson distance. (c) Hyperbola case using metric distance. (d) Hyperbola case using Sampson distance.



(a) (b) (c)

Figure 4: Extraction of image lines (HLIs). (a) Canny edge detector result, (b) connected components and (c) Extracted HLIs.

$$l_{\perp}\left(\mathbf{x}_0\right) = \begin{pmatrix} s_2\left(\mathbf{x}_0\right) \\ -s_1\left(\mathbf{x}_0\right) \\ -s_2\left(\mathbf{x}_0\right) x_0 + s_1\left(\mathbf{x}_0\right) y_0 \end{pmatrix} \tag{15}$$

With the approximated perpendicular line we compute the intersection between the line and the conic [14].

$$\mathsf{I}_s = \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix} \qquad \mathsf{C}^* = \det\left(\mathsf{C}\right)\mathsf{C}^{-1} \qquad l_{\times} = \begin{pmatrix} 0 & s_2x_0 - s_1y_0 & -s_1 \\ s_1y_0 - s_2x_0 & 0 & -s_2 \\ s_1 & s_2 & 0 \end{pmatrix}$$

$$p^{\pm} = \left(\pm\sqrt{-l'_{\perp}\mathsf{C}^*l_{\perp}}\mathsf{I}_{3x3} + l_{\times}\mathsf{C}\right)\mathsf{I}_s l_{\perp} \tag{16}$$

The approximated distance is the minor distance from both intersections.

Given a conic, a metric threshold with the proposed distance defines a uniform area surrounding the conic (Fig. 3 (a) and Fig. 3 (c)). When using other approaches like Sampson's distance this property is not guaranteed. In Fig. 3 (b-d) we show the area defined by a threshold based on Sampson's distance. We observe that the area is uniform when the conic is an ellipse; however when the conic is a hyperbola the thickness of the area using Sampson's distance is not uniform.

3.3. Hypercatadioptric 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 conics present into this component. Two points from the connected component are selected randomly and the corresponding HLI is computed. The distances from the rest of the points to this conic are computed. The points with a distance smaller than a threshold vote for this HLI. 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 HLIs. A pseudo-code version is presented in Algorithm 1.

Algorithm 1 2-point HLI extraction algorithm

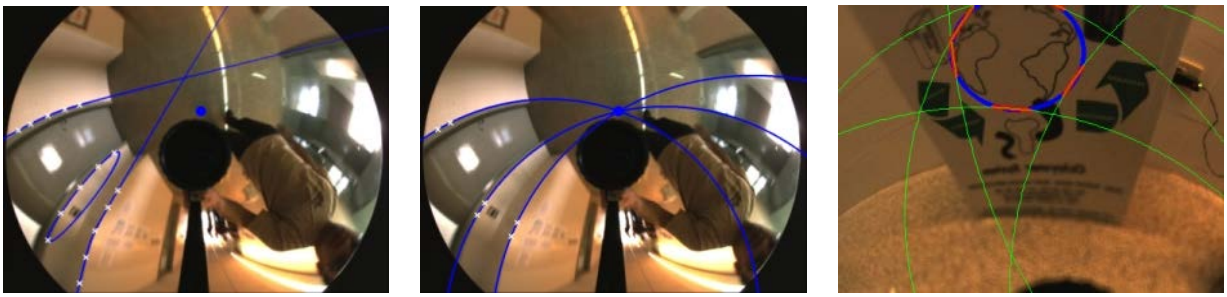
Require: Image
Ensure: Ω_{array}
  $edges = Canny(image)$
  $boundaries = extractboundaries(edges)$
  for $k = 1$ to $nBoundaries$ **do**
    $\mathbf{x} = boundaries(k)$
    $j = 0$
    while $remaining\_points/total\_points > T$ **do**
      $\mathbf{x}_{nplane} = \mathsf{H}_c^{-1}\mathbf{x}$
      for $i = 1$ to $nAttemps$ **do**
        $\mathbf{x}_{random} = rand(\mathbf{x}_{nPlane}, 2)$
        $\Omega_{img}(i) = twoPoints2Conic(\mathbf{x}_{random})$
        $d = dist2conic(\Omega_{img}(i), \mathbf{x})$
        $\mathbf{x}_{vote}(i) = inliers(\mathbf{x}, d, Threshold)$
        $votes(i) = size(\mathbf{x}_{inliers}(i))$
      end for
      $ind_{max} = MaximumVoted(\Omega_{img}, votes)$
      $\Omega = \Omega_{img}(ind_{max})$
      $\mathbf{x}_{inliers} = \mathbf{x}_{vote}(ind_{max})$
      $j = j + 1$
      $\Omega_{array}(j) = \Omega$
      $\mathbf{x} = UpdateRemainingBoundary(\mathbf{x}, \mathbf{x}_{inliers})$
    end while
  end for

Five points vs. Two points

For illustration purposes we show in Fig. 5 (a-b) the extraction of three conics corresponding to vertical lines. We use conic fitting with five points and the extraction of the same HLIs using our 2-point approach. We also show the VVP which is used as a measure of quality. All conics corresponding to vertical lines must intersect this point. When the generic five point approach is used, the shape of the HLIs changes depending on the points used to compute it. It can be either a hyperbola or an ellipse and it can easily change from one to the other. The result is a conic which fits the five selected points but it does not describe the projection of the 3D line. For example, using conics showed in Fig. 5(a) we can not compute the vertical vanishing point by intersection. In the case of the proposed 2-point algorithm the HLI cover with accuracy the points used to compute it, and describe the edge from which the points were extracted. Therefore the three conics intersect in the vanishing point. Another advantage of the 2-point approach is the number of iterations performed inside voting approach. For instance, using a probability $p = 99\%$ of not failing in the random search and 50% of outliers (ε) just 17 iterations are needed to get a result using the proposed approach (2-point) and 146 using the general five point approach. The number of iterations n_r is given by $n_r = \frac{\log(1-p)}{\log(1-(1-\varepsilon)^k)}$.

Discrimination of HLIs from other conics

Since lines present in the scene become conics in hypercatadioptric images, one may think that these conics may represent projections of circles or other conic shapes. Our approach is able to distinguish between conics which are projections of 3D straight lines and the rest of the image conics. To illustrate this, we show in Fig. 5 (c) the behavior of our approach extracting HLIs in hypercatadioptric images. We observe a circular contour whose boundary points are given manually. We apply our extraction algorithm on this



(a) (b) (c)

Figure 5: Computing a HLI 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. (c) Extraction of HLIs over a circular contour. Blue points give the connected component. Red points correspond to the HLIs detected by RANSAC. The corresponding conics to each segment HLI are shown in green.

connected component assuming the contour is composed of HLIs. We can observe how the algorithm is not able to fit the whole circle with a single conic. With our approach we can directly test if a set of image points corresponds to a HLI. The definition of a HLI presented in section 3.1 is not limited to two points. From (10) we observe that each point gives a row of a linear homogeneous system. For n points we have

$$\mathsf{M} = \begin{pmatrix} \bar{x}_1 & \bar{y}_1 & -\alpha_1 \\ \bar{x}_2 & \bar{y}_2 & -\alpha_2 \\ \vdots & \vdots & \vdots \\ \bar{x}_n & \bar{y}_n & -\alpha_n \end{pmatrix} \tag{17}$$

This situation allows fitting a HLI using more than two points. This is very useful in practice when we have more than two pixels defining a conic. At the same time it permits to distinguish HLIs from other conics. The rank of the homogeneous matrix M is two when it represents a HLI.

4. Influence of Calibration and Error Propagation in the Extraction of HLIs

In this section we present an exhaustive analysis of the influence of the calibration parameters of a hypercatadioptric system on the extraction of HLIs. First, we present an analysis of the error propagation from the calibration parameters and the two defining points of a HLI to the normal $\mathbf{n}$ describing the projection plane of the 3D line. Then we show an empirical study of the influence of calibration parameters on the extraction of HLIs. Finally we analyze the influence of the length of the observed conic segment.

4.1. Analytical study of errors propagation

We have performed an analytical study of the propagation errors in the calibration parameters to the normal defining the projection of the line. We also propagate the error in pixels from a pair of defining points of a HLI on the image to its normal in the space. To deal with this uncertainty we use a first order approximation of the formulation.

We first analyze the uncertainty from the calibration parameters to the normal $\mathbf{n}$. For this formulation the calibration parameters of the system are represented in the vector $\kappa = (u_0, v_0, f_x, f_y, s_{skew}, \xi, \psi)^\mathsf{T}$. The corresponding Jacobian is

$$\frac{d\mathbf{n}}{d\kappa} = \Psi \left(\Lambda \Upsilon + \Gamma\right) \tag{18}$$

with

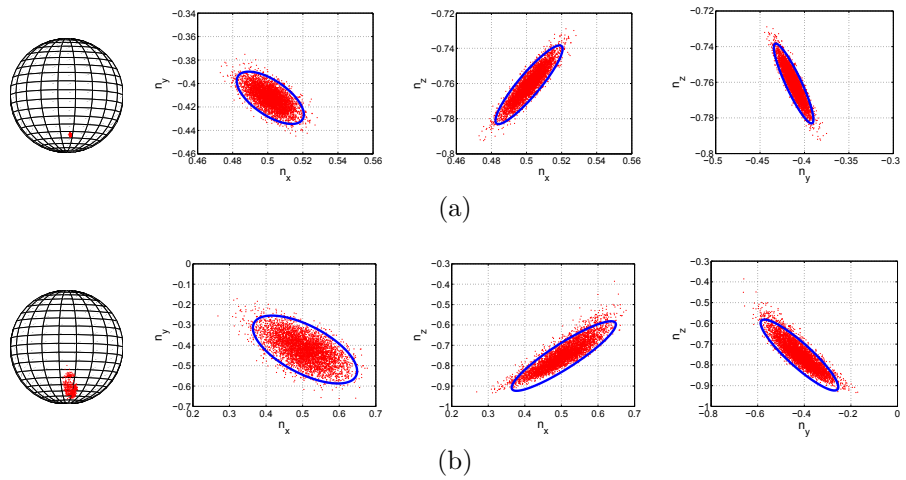


Figure 6: Uncertainty from defining points to normal vector $\mathbf{n}$. Gaussian randomized points around the defining points are used to compute a cloud of normals (red points). The 95% confidence region of the normal is represented from this propagation with an ellipse (in blue). (a) Standard Deviation of 2 pixels. (b) Standard Deviation of 15 pixels.

$$\Psi = \frac{1}{\left(n_x^2 + n_y^2 + n_z^2\right)^{\frac{3}{2}}} \begin{pmatrix} n_y^2 + n_z^2 & -n_x n_y & -n_x n_z \\ -n_x n_y & n_x^2 + n_z^2 & -n_y n_z \\ -n_x n_z & -n_y n_z & n_x^2 + n_y^2 \end{pmatrix}, \tag{19}$$

$$\Gamma = \frac{1}{(1-\xi^2)^2} \begin{pmatrix} \mathsf{0}_{3\times 5} & -\frac{y_2(\lambda_1 - \xi z_1)}{\lambda_1} + \frac{y_1(\lambda_2 - \xi z_2)}{\lambda_2} & \mathsf{0}_{3\times 1} \\ & \frac{x_2(\lambda_1 - \xi z_1)}{\lambda_1} - \frac{x_1(\lambda_2 - \xi z_2)}{\lambda_2} & \\ & 0 & \end{pmatrix} \tag{20}$$

and

$$\Upsilon = \frac{1}{(\xi - \psi)} \begin{pmatrix} \frac{\hat{z}_1}{f_x} & -\frac{s\hat{z}_1}{f_x f_y} & \frac{f_y \Delta x_1 - s\Delta y_1}{f_y f_x^2} & -\frac{s\Delta y_1}{f_y^2 f_x} & \frac{\Delta y_1}{f_y f_x} & \frac{f_y \Delta x_1 - s\Delta y_1}{f_y f_x (\xi - \psi)} & \frac{-f_y \Delta x_1 + s\Delta y_1}{f_y f_x (\xi - \psi)} \\ 0 & -\frac{\hat{z}_1}{f_y} & 0 & \frac{\Delta y_1}{f_y^2} & 0 & -\frac{\Delta y_1}{f_y (\xi - \psi)} & \frac{\Delta y_1}{f_y (\xi - \psi)} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\hat{z}_2}{f_x} & -\frac{s\hat{z}_2}{f_x f_y} & \frac{f_y \Delta x_2 - s\Delta y_2}{f_y f_x^2} & -\frac{s\Delta y_2}{f_y^2 f_x} & \frac{\Delta y_2}{f_y f_x} & \frac{f_y \Delta x_2 - s\Delta y_2}{f_y f_x (\xi - \psi)} & \frac{-f_y \Delta x_2 + s\Delta y_2}{f_y f_x (\xi - \psi)} \\ 0 & -\frac{\hat{z}_2}{f_y} & 0 & -\frac{\Delta y_2}{f_y^2} & 0 & -\frac{\Delta y_2}{f_y (\xi - \psi)} & \frac{\Delta y_2}{f_y (\xi - \psi)} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{21}$$

where $\Delta x_1 = \hat{x}_1 - \hat{z}_1 u_0$, $\Delta x_2 = \hat{x}_2 - \hat{z}_2 u_0$, $\Delta y_1 = \hat{y}_1 - \hat{z}_1 u_0$ and $\Delta y_2 = \hat{y}_2 - \hat{z}_2 u_0$.

We have also obtained the error propagation formulas from the two defining points on the image to the normal $\mathbf{n}$. Let be $\hat{\chi} = (\hat{x}_1, \hat{y}_1, \hat{z}_1, \hat{x}_2, \hat{y}_2, \hat{z}_2)^\intercal$ the corresponding Jacobian is described by the following matrix product

$$\frac{d\mathbf{n}}{d\hat{\chi}} = \Psi \Lambda \begin{pmatrix} \mathsf{H}_c^{-1} & \mathsf{0}_{3\times 3} \\ \mathsf{0}_{3\times 3} & \mathsf{H}_c^{-1} \end{pmatrix} \tag{22}$$

in which

$$\Lambda = \begin{pmatrix} \frac{-\bar{y}_2 \bar{x}_1 \xi}{\lambda_1} & \frac{-\bar{y}_2 \bar{y}_1 \xi}{\lambda_1} + \alpha_2 & \frac{-\bar{y}_2(-1+\xi\bar{z}_1)}{\lambda_1(1-\xi^2)} & \frac{\bar{y}_1 \bar{x}_2 \xi}{\lambda_2} & \frac{\bar{y}_2 \bar{y}_1 \xi}{\lambda_2} - \alpha_1 & \frac{\bar{y}_1(-1+\xi\bar{z}_2)}{\lambda_2(1-\xi^2)} \\ \frac{\bar{x}_2 \bar{x}_1 \xi}{\lambda_1} - \alpha_2 & \frac{\bar{y}_1 \bar{x}_2 \xi}{\lambda_1} & \frac{\bar{x}_2(-1+\xi\bar{z}_1)}{\lambda_1(1-\xi^2)} & \frac{-\bar{x}_2 \bar{x}_1 \xi}{\lambda_2} + \alpha_1 & \frac{-\bar{y}_2 \bar{x}_1 \xi}{\lambda_2} & \frac{-\bar{x}_2(-1+\xi\bar{z}_2)}{\lambda_2(1-\xi^2)} \\ -\bar{y}_2 & \bar{x}_2 & 0 & \bar{y}_1 & -\bar{x}_1 & 0 \end{pmatrix}, \tag{23}$$

where $\lambda_i = \sqrt{\bar{z}_i^2 + (\bar{x}_i^2 + \bar{y}_i^2)(1-\xi^2)}$.

This approximation allows propagating the error from the two defining points in the image plane to the normal $\mathbf{n}$. To evaluate the expressions, in fig. 6 we show the uncertainties of two points defining the HLI. We generated 5000 random values of $\hat{\chi}$ on the image adding a Gaussian variance of two pixels and fifteen pixels (Fig. 6(a) and 6(b), respectively). These points have been mapped to normal vectors using (11). We compare these points with the 95% confidence region we obtain propagating the covariance matrix of $\hat{\chi}$.

4.2. Empirical study of the influence of calibration

In this experiment we consider the following calibration parameters, focal lengths (f_x, f_y), principal point (u_0, v_0) and mirror parameter ξ. We modify each parameter independently inside a certain range from which we select 1000 samples. Then, we compute a specific HLI for each value using its two extreme points. We perform this experiment for two different types of HLIs, one of which is nearly perpendicular to the optical axis and the other which is nearly parallel. To illustrate the results two types of graphics are showed. The first one represents the mean error in pixels of the conic fitting as a function of the calibration parameter. In the second type the horizontal axis represent the variation of the calibration parameter and the vertical axis the angle between the reference normal vector $\mathbf{n}_{ref}$ and the normal vector $\mathbf{n}$ computed with the modified calibration.

- *Focal length:* In Fig. 7(a),(b) we show the corresponding conics when the focal lengths (f_x, f_y) are modified. We observe that these parameters affect their corresponding coordinate of influence x and y. The plot shows that the effect is the inverse for both axis. However, the magnitude is bigger in the case of f_y. Lines with a more horizontal component are more affected by the change on the focal length.

- *Principal point:* In Fig. 7(c),(d) we observe the influence of the principal point (u_0, v_0) on the computation of the HLIs. When these two parameters are modified we observe a displacement proportional to the distortion added to the corresponding parameter on the corresponding axis.

- *Mirror parameter:* In Fig. 7(e) we observe the effect of the mirror parameter ξ on the computation of HLIs. We observe that both lines approximate a straight line when the value of ξ approximates to zero. This is explained because $\xi = 0$ represents the pin-hole model which projects lines to lines. Therefore, the closer a HLI is to a straight line, the less the mirror parameter ξ has an influence.

We observe that the fitting error is bigger for the perpendicular line than for the parallel line, except for errors in v_0. This is explained since the perpendicular lines are transformed in conics with a higher curvature than the parallel ones, which are mapped to straighter lines.

4.3. Influence of observed length of the HLI

In the extraction process of the HLIs using RANSAC, two points in a connected component are randomly selected and a HLI is computed. Then it is used to obtain all the points that belong to such HLI. In this experiment we want to observe the influence of the distance between these two defining points on the accuracy of the extracted HLI. As in previous experiment we observe that the influence of calibration errors is stronger in perpendicular lines to the optical axis. In this experiment we perform simulations using this type of line. We try different line lengths, from a few degrees to the longest theoretical line which has an observed extension close to 180°. We also modify the mirror parameter to observe its behavior. As we expected, we observe that the longer the line is, the better the extraction of the HLI is. In Fig. 8(a) we observe such behavior with a mirror parameter of $\xi = 0.75$. We also observe that when we approach the maximum observed lenght of the line the error starts increasing, showing the limit of the longest line that should be used to extract the corresponding HLI. In Fig. 8(b) we repeat the experiment, this time using a mirror parameter of $\xi = 0.95$ close to a parabolic mirror ($\xi = 1$). We observe that the error increase a lot when we are closer to both the longest line and the parabolic mirror. The para-catadioptric system case has been studied in [18] and it is simpler than the hypercatadioptric analyzed here.

With the information obtained from this experiment we can infer that we should select the two more distant points to extract their corresponding HLI. However, there exist a upper limit that depends on the

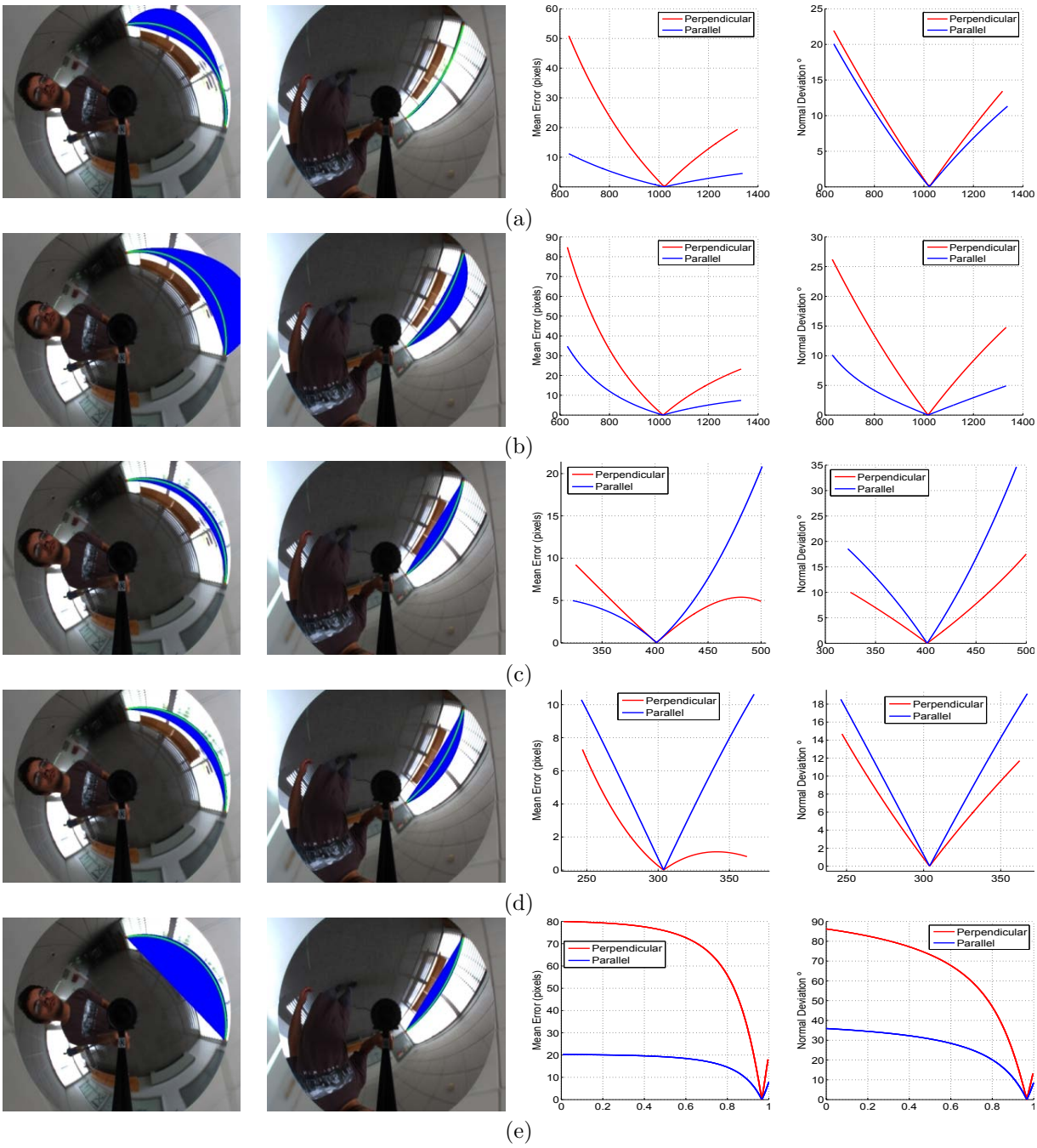


Figure 7: Mean error in pixels of the conic fitting varying the calibration parameters. Angle between reference normal and normal when varying the calibration parameters. (a,b) Focal length, f_x, f_y. (c,d) Principal point, u_0, v_0. (e) Mirror parameter ξ.

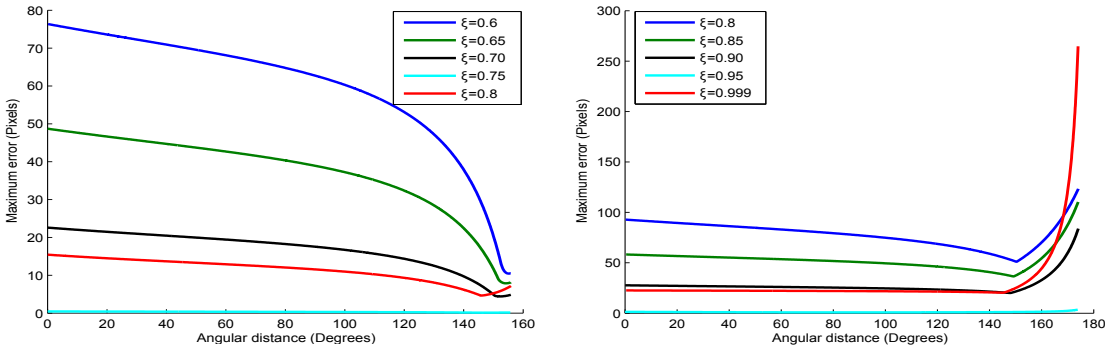


Figure 8: Maximum error in pixels as a function of the extension (in radians) of the observed HLI for mirror parameters (a) $\xi = 0.75$ and (b) $\xi = 0.95$.

length of the HLI and the mirror parameter. In practice it is very unlikely to observe a line of such length. Since we use a RANSAC approach, whose principle is to search in the space of solutions minimizing the error, this problematic situation is automatically avoided in practice.

5. 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 catadioptric image. Their intersection point, the vertical vanishing point (VVP), is located close to the image center. When the camera is not vertically aligned, the vertical lines become conic curves as we explained before. One consequence is that the VVP displaces from the image center. Its new location contains important information about the orientation of the camera with respect to the scene.

5.1. Intersection of two HLIs using the common self-polar triangle

In a general configuration, two conics intersect in four points (Fig. 9). The union of couples 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 [14]. We have studied the particular case where two HLIs intersect, which is a degenerate configuration, since they intersect in two points. As we observe in Fig. 10, there exist a line l 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 $\mathbf{n}_2 = (n_{x_2}, n_{y_2}, n_{z_2})^\mathsf{T}$ be 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 $\bar{\Omega}(\lambda) = \bar{\Omega}_1 + \lambda\bar{\Omega}_2$ satisfy the constraint

$$\det(\bar{\Omega}_1 + \lambda\bar{\Omega}_2) = 0. \tag{24}$$

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 l, expressed in a parametric way as

$$l = \mu \cdot \mathbf{v} = \mu \begin{pmatrix} v_x \\ v_y \end{pmatrix} = \mu \begin{pmatrix} n_{z_2}^2 n_{y_1} n_{z_1} - n_{z_1}^2 n_{y_2} n_{z_2} \\ n_{z_1}^2 n_{x_2} n_{z_2} - n_{z_2}^2 n_{x_1} n_{z_1} \end{pmatrix}. \tag{25}$$

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

$$\mu^2(c_1 v_x^2 + c_2 v_x v_y + c_3 v_y^2) + \mu(c_4 v_x + c_5 v_y) + c_6 = 0 \tag{26}$$

and substitute in (25).

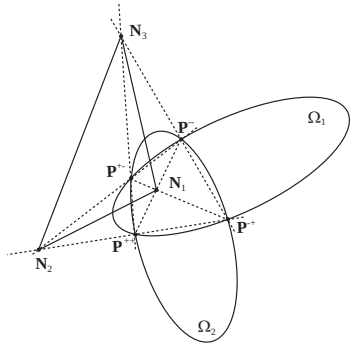


Figure 9: Intersection of two generic conics.

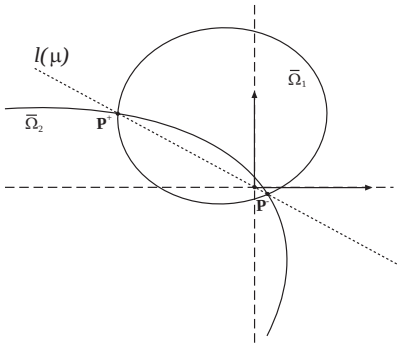


Figure 10: Intersection of two conics in the normalized plane.

5.2. Vertical vanishing point (VVP)

We use a classic algorithm to detect the VVP. Let m be the number of putative vertical HLIs detected in the omnidirectional image and let $\mathbf{n}_i$ their corresponding representation in the normalized plane. For every pair of HLIs (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 HLIs 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.

5.3. Horizontal vanishing point (HVP)

Once the VVP is extracted we can exploit several properties to compute the horizontal vanishing point (HVP). The VVP $\bar{\mathbf{x}}_{VVP} = \left(\bar{x}_{v_x}, \bar{x}_{v_y}\right)$ defines a separation between vertical lines and potential horizontal lines. The VVP defines the plane of the horizon,which is projected into the horizon conic, defined as:

$$\Omega_v = \begin{pmatrix} \zeta^2\bar{x}_{v_x}^2\left(1-\xi^2\right) - \left(\zeta-\xi\right)^2\xi^2 & \zeta^2\bar{x}_{v_x}\bar{x}_{v_y}\left(1-\xi^2\right) & \zeta\bar{x}_{v_x}\left(\zeta-\xi\right) \\ \zeta^2\bar{x}_{v_x}\bar{x}_{v_y}\left(1-\xi^2\right) & \zeta^2\bar{x}_{v_y}^2\left(1-\xi^2\right) - \left(\zeta-\xi\right)^2\xi^2 & \zeta\bar{x}_{v_y}\left(\zeta-\xi\right) \\ \zeta\bar{x}_{v_x}\left(\zeta-\xi\right) & \zeta\bar{x}_{v_y}\left(\zeta-\xi\right) & \left(\zeta-\xi\right)^2 \end{pmatrix}, \tag{27}$$

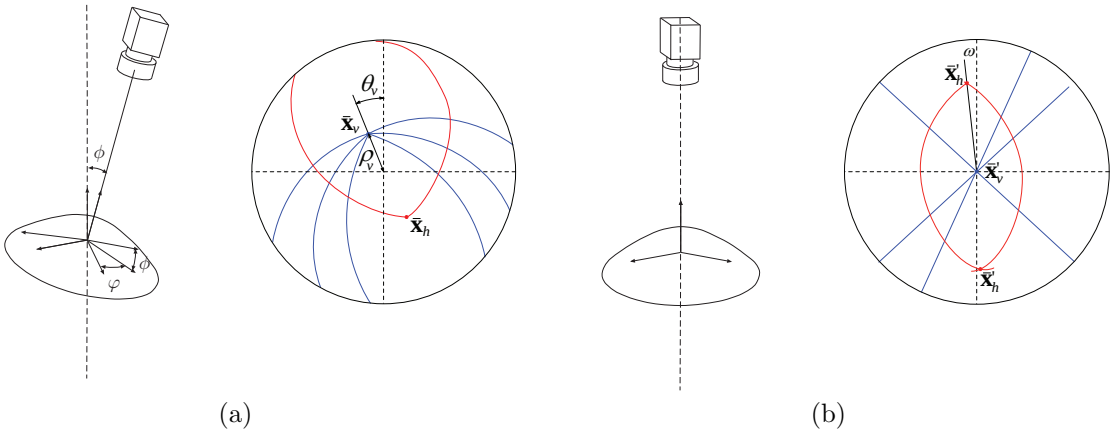
where $\zeta = \frac{\xi+\sqrt{1+(1-\xi^2)\left(\bar{x}_{v_x}^2+\bar{x}_{v_y}^2\right)}}{\bar{x}_{v_x}^2+\bar{x}_{v_y}^2+1}$. Therefore the HVP must lie on this conic.

By introducing these constraints we reduce the search space and we impose the main directions to be perpendicular. Notice that using a single image without any assumptions it is impossible to distinguish between the VVP and the HVP. However, some prior knowledge is easy to have in practice.

5.4. Image rectification

Here we explain the relation between the VVP computed in the normalized plane and the whole orientation of the catadioptric system. Having an absolute reference system defined by the main directions, the orientation of the camera with respect to this system is described by three angles (ϕ φ ω). Two of them are obtained from the VVP. Writing the VVP in polar coordinates $\bar{\mathbf{x}}_{vp} = (\rho_v, \theta_v)^\mathsf{T}$ (Fig. 11(a)) we observe that there exist a relation between the angle θ_v and the angle φ representing the rotation of the catadioptric system around the z-axis (28). The negative angle is produced by the mirror effect which inverts the catadioptric image.

$$\varphi = -\theta_v \tag{28}$$



(a) (b)

Figure 11: (a) Camera Reference: The VVP location is related with orientation angles ϕ and φ. (b) Vertical Reference: After rotating the camera around the x-axis an angle $-\phi$ the VVP is in the principal point. In this reference the HVP defines the orientation angle ω. Each vertical 3D line is projected on a radial straight line.

We observe that the component ρ_v is intrinsically related to the angle ϕ and the mirror parameter ξ of the catadioptric system. Since angles ϕ and φ are independent, we consider the case where $\varphi = 0$ (Fig. 11(b)). Using (25) and (26) with a pair of parallel HLIs in polar coordinates we compute the following relationship

$$\rho_v = -\frac{\sin\phi}{\cos\phi \pm \xi}, \tag{29}$$

selecting geometrically compatible solutions, ϕ can be isolated resulting in:

$$\phi = \operatorname{atan2}(1, \rho_v) + \arccos\left(\frac{-\rho_v \xi}{\sqrt{\rho_v^2 + 1}}\right). \tag{30}$$

With ϕ and φ angles any image can be transformed to a vertical reference in which VVP lies on the image center (Fig. 11(c)). We perform the vertical rectification in two steps. The first step undoes a horizontal rotation according to angle φ ($-\theta_v$ in the image) to an arbitrary central reference (see Fig. 11(b)). The second step consist of translating the VVP to the image center through a rotation around the vertical axis by the angle ϕ, which is computed (30) from ρ_v (in the image) and the mirror parameter ξ (see Fig. 11(c)). This procedure is performed by the following equation

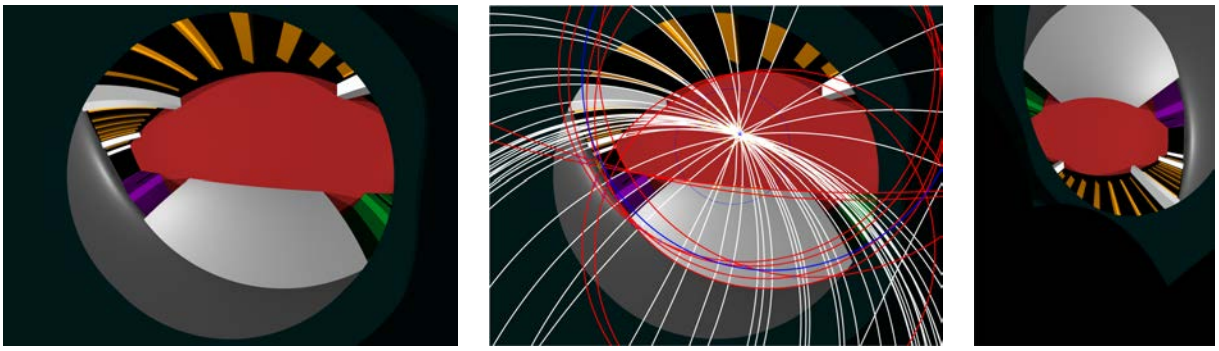
$$\bar{\mathbf{x}}'_{\mathbf{h}} = \hbar\left(rot_z(\varphi)\, rot_x(-\phi)\, rot_z(-\varphi)\, \hbar^{-1}(\bar{\mathbf{x}}_{\mathbf{h}})\right) \tag{31}$$

Once we have rectified the HVP to this reference we can compute the full orientation and perform the full rectification of the catadioptric image. We compute its polar coordinates from which we only require the angle component ω. With this angle we translate it to a central position. Since a horizontal rotation was performed in the previous step, the position of the HVP has been modified and we have to undo such transformation. From these angles we construct a rotation matrix R which relates the absolute reference system with the camera reference system

$$\mathbf{X}_{cam} = \mathsf{R}\mathbf{X}_{abs}, \quad \text{where} \qquad \mathsf{R} = rot_z(\varphi) rot_x(\phi) rot_z(\omega - \varphi) \tag{32}$$

6. Experiments

In this section we present experiments focused on testing the accuracy of the proposed algorithm. We use synthetic and real images as well as image sequences acquired with a hand-held hypercatadioptric system



(a) (b) (c)

Figure 12: Orientation accuracy test with synthetic images: (a) Original Image, (b) VP extraction, (c) Rectified image using the orientation.

Ref.	0º	5º	10º	15º	20º	25º	30º	35º	40º	45º	50º	55º	60º
Our Method	0.45	5.27	10.11	15.07	19.90	24.58	30.09	35.54	40.41	45.35	50.06	55.01	59.97
Bazin's Method [10]	0.12	5.10	9.80	14.56	19.77	24.76	29.79	34.73	39.77	45.02	49.71	54.74	59.75

Table 1: ϕ angle extraction accuracy test on simulated scenario.

and a hypercatadioptric system mounted on a helmet.

6.1. Orientation accuracy

This group of experiments is designed to evaluate the accuracy of the orientation of the camera computed from vanishing points. Initially, we use synthetic images to avoid setting and calibration errors. We also use real images which are acquired with a hypercatadioptric system. We accurately move this system with a precision head composed of a goniometer, from which we obtain the ground truth orientation of the system.

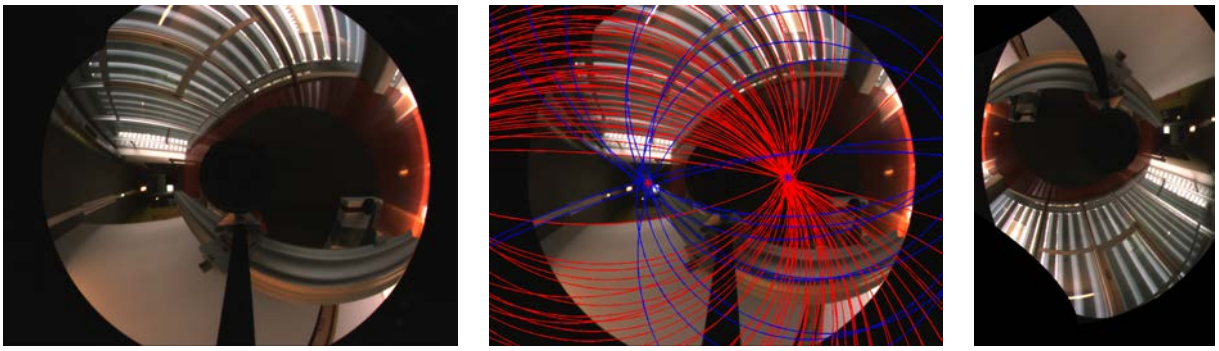
Synthetic images

The goal of this experiment is to estimate the orientation errors with the proposed method avoiding the calibration influence. We use a synthetic environment based on a real scenario composed of elements like doors and columns which define main directions. The catadioptric system is modelled by a perspective camera and a hyperbolic mirror. To test the proposed method thirteen images have been generated from 0º to 60º using a ray-tracing software[2] introducing a consecutive variation of five degrees in ϕ angle. In Table 1 we show the results obtained with the proposed method and the method presented by Bazin et al. in [10]. The obtained errors are in average smaller than 0.25 degrees. There is no significant difference between our results and Bazin's method. Fig. 12 shows an example of the vanishing points extraction and full image rectification from a synthetic image generated from an angle $\phi = 40$º.

Real images

In this experiment we test the accuracy of the algorithm in a real environment. The camera is attached to a precision head[3] which provides the ground truth. Images have been acquired with a hypercatadioptric system[4]. To evaluate the influence of the lines density and quality, the experiment has been repeated in two different scenarios. Scenario A is a corridor where main directions are well defined, the number of vertical

[2]http://www.povray.org
[3]UTR 80, Manufactured by Newport with goniometer of 1/60 º resolution
[4]http://neovision.cz



(a) (b) (c)

Figure 13: Orientation accuracy test with real images in scenario A: (a) Original Image, (b) VP extraction, (c) Rectified image.

Reference	0º	5º	10º	15º	20º	25º	30º	35º	40º	45º	50º	55º	60º
Scenario A Mean	0.63	4.76	10.87	15.30	20.15	25.21	30.06	34.88	39.97	44.93	49.93	54.42	59.22
Scenario A Std. Dev.	0.153	0.227	0.417	0.200	0.038	0.021	0.045	0.043	0.132	0.083	0.031	0.036	0.008
Scenario B Mean	-0.32	5.52	11.28	15.83	21.57	25.45	29.75	35.20	39.30	44.38	49.28	53.62	59.13
Scenario B Std. Dev.	0.226	0.162	0.167	0.086	0.300	0.038	0.382	0.468	0.185	0.386	0.345	0.249	1.115

Table 2: ϕ angle extraction accuracy test on real stages.

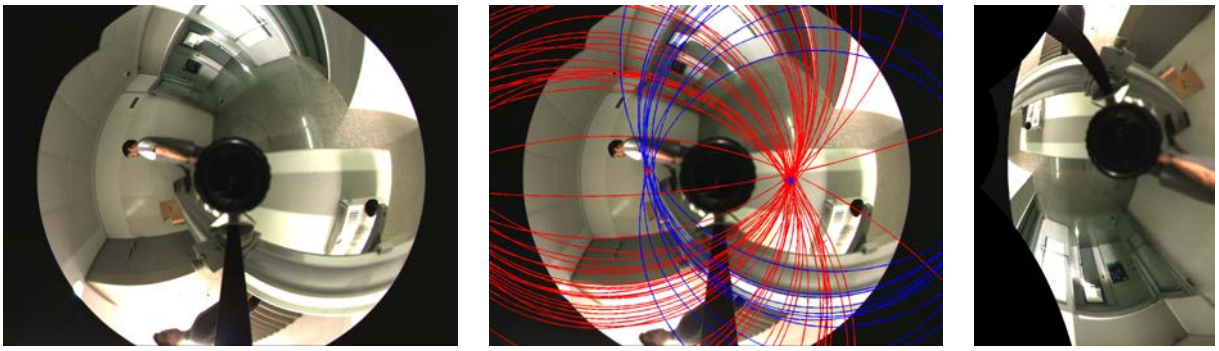
lines is high, floor and walls have different colors and without reflections. Scenario B is a small hall in which the number of vertical lines is low, the color of the constructive elements is uniform and there exist reflections on the floor.

We also use thirteen images from 0º to 60º which have been acquired introducing an increment of five degrees in ϕ angle and the algorithm has been executed five times for each image. In Table 2 we show the result from the different two scenarios. The first row is the reference angle given by the goniometer. Next rows are the mean and standard deviation of the orientation calculation for each scenario. Since we have not a perfectly aligned system with respect to the vertical we assume an offset setup error for the complete experiment.

Mean errors are 0.31 degrees for well-conditioned scenario A and 0.80 degrees for scenario B. Maximum errors are 0.87 degrees and 1.57 degrees respectively. These results are better than maximum error of 2 degrees presented by Magnier et al. in [20]. Results obtained in scenario A, which have more and longer lines, are considerably more accurate than results from scenario B. Therefore as expected, the number of HLIs and their quality have influence in angle accuracy. Spurious HLIs extracted from non-line objects, like people, also introduce uncertainty into the extraction process. We observe that the inclusion of one parameter radial distortion in the projection model improve the accuracy. We observe the high influence of radial distortion and principal point in angle ϕ accuracy. The relationship between angle ϕ and the distance from principal point to the VVP, which can be seen in (29), explains this tendency. In Fig. 13 and 14 we show an example of the vanishing points extraction from two real images taken in the two considered scenarios. The system has an inclination of 40º. Full orientation extracted from vertical and horizontal vanishing point is used to rectify the original image.

6.2. Rectification of image sequences

In order to evaluate the system in real environments the proposed method has been tested with image sequences. This application consist of a 3D orientation computation in which the extracted orientation is used to rectify the image sequence. The result is an oriented image sequence.



(a) (b) (c)

Figure 14: Orientation accuracy test with real images scenario B: (a) Original Image, (b) VP extraction, (c) Rectified image.

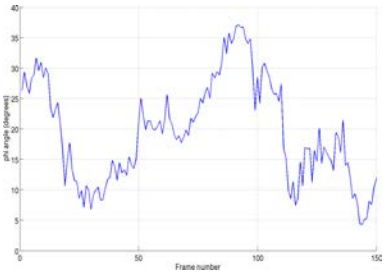


Figure 15: Elevation angle ϕ of the catadioptric system. Frames 1-150.

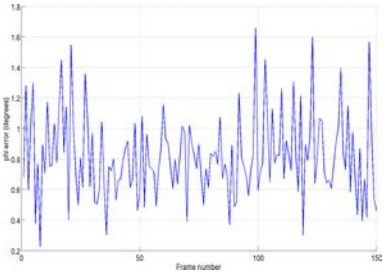


Figure 16: Angular deviation of the vertical lines after rectification, in degrees.

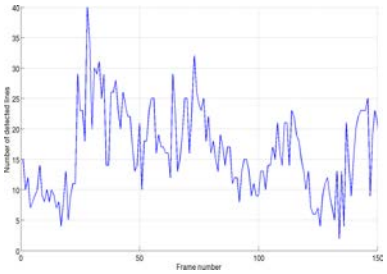


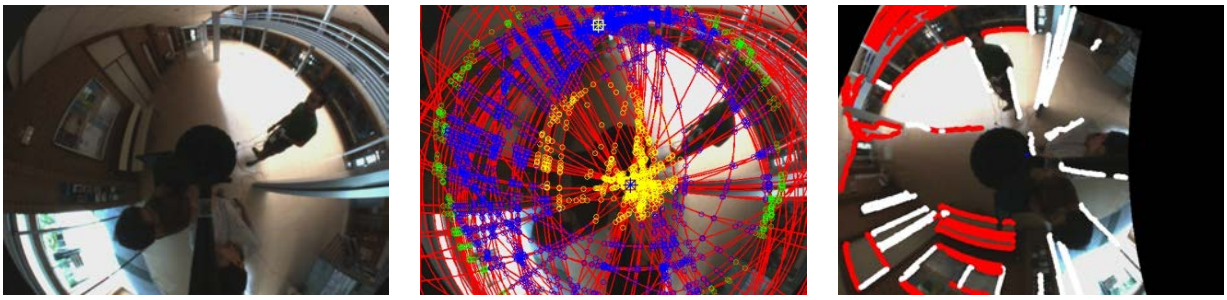
Figure 17: Number of vertical lines present in each frame of sequence 1.

As commented in previous sections, it is necessary to introduce additional constraints to distinguish between VVP and HVP. When we deal with image sequences it is possible to track the vanishing points. The vanishing points detected in the first frame are defined as reference. The tracking is performed with previous VPs inside a confidence region. This region is described as a covariance matrix which depends on the hypothesized angular velocity. Three sequences have been acquired using two different catadioptric systems. Sequences 1 and 2 have been acquired with a hand-held hypercatadioptric system[5]. Sequence 3 has been acquired with a hypercatadioptric system mounted on a helmet (see Fig. 20(a)).

Image sequence 1

In this experiment we only perform the vertical rectification of a sequence. In Fig.15 we show the ϕ angle computed for the first 150 frames of sequence 1. 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.16 we show the average error of the angle deviation of all the vertical lines present in each frame of sequence 1 after rectification. We observe that this error is related to the number of vertical HLIs present in the catadioptric image (Fig. 17). The more the number of vertical HLIs present in the frame the better the estimation of the vanishing points, and consequently a better rectification of the image.

[5]http://neovision.cz



(a) (b) (c)

Figure 18: Example of full image rectification. (a) Frame 1 of the sequence 2. (b) 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. (c) Full-rectified image. The vertical HLIs are shown in white and the horizontal ones in red.

Frame 107 Frame 242

Figure 19: 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.

Image sequence 2

In this experiment we compute the full rectification of the second sequence. In Fig. 18 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 translation (Fig. 19).

Helmet camera and IMU comparison

In this experiment we measure the orientation of a head during a human walk. An inertial measurement unit (IMU) has been coupled to the helmet to compare the computed angle with the angle given by the IMU. In Fig. 20(b) we show an image of the sequence and many conics lying on the vanishing points. Fig. 20(c) shows the comparison between the angle given by the IMU and the algorithm during the image sequence.

7. Conclusions

We have presented a new method for the extraction of conics which are projections of 3D lines in omnidirectional images generated by a calibrated hypercatadioptric system. We propose an approach that requires only two points and the calibration of the system to extract them and use it with a metric distance from point to conic in an extraction algorithm. We analyze the uncertainty propagation using the proposed approach showing the influence of each particular calibration parameter and the two defining points. We develop the common self-polar triangle approach to the particular case of HLIs intersection. With a voting approach we select the VPs from the intersections of the HLIs. Orientation of the hypercatadioptric system is estimated relating the location of the vanishing points on the image with the orientation angles in an man-made structured environment. To show the effectiveness of this method we perform experiments with synthetic and real images comparing our proposal to similar approaches. In particular we test the accuracy

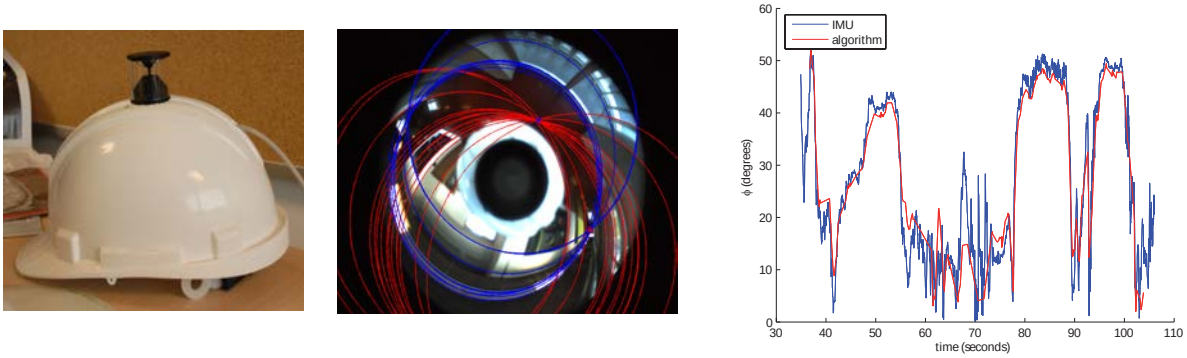


Figure 20: Image sequence acquired with the omni-camera helmet: (a) Acquisition helmet, (b) Vanishing Point Extraction, (c) Comparison between angle ϕ given by the accelerometer and the vision based algorithm.

of the orientation estimation by using static images, obtaining mean orientation errors of less than 1 degree. We also show the behaviour of the method dealing with image sequences. We compare our algorithm with the orientation data extracted from an IMU. Using the orientation we perform the vertical rectification of the sequence in order to create images where applications that require the vertical constraint can be used.

Acknowledgments

This work was supported by the Spanish project VISPA DPI2009-14664-C02-01 and FEDER funds. First author was supported by the FPU program AP2010-3849. The final publication is available at Elsevier via http://dx.doi.org/10.1016/j.robot.2012.02.008

References

[1] S. Baker, S. Nayar, A theory of single-viewpoint catadioptric image formation, Int. J. Comput. Vision 35(2) (1999) 175–196.
[2] Y. Mezouar, H. Abdelkader, P. Martinet, F. Chaumette, Central catadioptric visual servoing from 3D straight lines, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 , vol. 1, 343–348, 2004.
[3] J. J. Guerrero, A. C. Murillo, C. Sagües, Localization and Matching using the Planar Trifocal Tensor with Bearing-only Data, IEEE Transactions on Robotics 24(2) (2008) 494–501.
[4] D. Scaramuzza, A. Martinelli, R. Siegwart, A Robust Descriptor for Tracking Vertical Lines in Omnidirectional Images and its Use in Mobile Robotics, International Journal of Robotics Research 28 (2) (2009) 149–171, special Issue on Field and Service Robotics.
[5] J. P. Barreto, H. Araujo, Fitting conics to paracatadioptric projections of lines, Computer Vision and Image Understanding 101 (3) (2006) 151 – 165.
[6] P. Vasseur, E. M. Mouaddib, Central Catadioptric Line Detection, in: British Machine Vision Conference, 2004.
[7] X. Ying, Z. Hu, Catadioptric line features detection using Hough transform, in: ICPR 2004. Proceedings of the 17th International Conference on, vol. 4, ISSN 1051-4651, 839–842, 2004.
[8] C. Mei, E. Malis, Fast central catadioptric line extraction, estimation, tracking and structure from motion, in: Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on, 4774–4779, 2006.
[9] J. C. Bazin, C. Demonceaux, P. Vasseur, Fast Central Catadioptric Line Extraction, in: IbPRIA '07: Proceedings of the 3rd Iberian conference on Pattern Recognition and Image Analysis, Part II, 25–32, 2007.
[10] J. C. Bazin, C. Demonceaux, P. Vasseur, I. Kweon, Motion estimation by decoupling rotation and translation in catadioptric vision., Computer Vision and Image Understanding 114 (2) (2010) 254–273, .
[11] R. I. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press,, 2000.
[12] M. Bosse, R. Rikoski, J. Leonard, S. Teller, Vanishing points and 3D lines from omnidirectional video, in: International Conference on Image Processing. Proceedings. , 513 – 516 vol.3, 2002.
[13] M. Antone, S. Teller, Automatic recovery of relative camera rotations for urban scenes, in: IEEE Conference on Computer Vision and Pattern Recognition. Proceedings., 282 –289 vol.2, 2000.
[14] J. Barreto, General Central Projection Systems: Modeling, Calibration and Visual Servoing, Ph.D. thesis, 2003.
[15] C. Geyer, K. Daniilidis, A Unifying Theory for Central Panoramic Systems and Practical Applications, in: European Conference on Computer Vision (ECCV) (2), 445–461, 2000.

[16] L. Puig, J. Bermudez, J. J. Guerrero, Self-orientation of a hand-held catadioptric system in man-made environments, 2010 IEEE International Conference on Robotics and Automation (ICRA) pp:2549-2555, Anchorage, Alaska May .

[17] J. Bermudez, L. Puig, J. J. Guerrero, Line extraction in central hyper-catadioptric systems, OMNIVIS - 10th Workshop on Omnidirectional Vision, Camera Networks and Nonclassical Cameras , pp8:1-7,June, 2010 .

[18] J. P. Barreto, H. Araujo, Geometric Properties of Central Catadioptric Line Images and Their Application in Calibration, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (8) (2005) 1327–1333.

[19] P. Sturm, P. Gargallo, Conic fitting using the geometric distance, in: Proceedings of the Asian Conference on Computer Vision, Tokyo, Japan, Springer, 2007.

[20] B. Magnier, F. Comby, O. Strauss, J. Triboulet, C. Demonceaux, Highly Specific Pose Estimation with a Catadioptric Omnidirectional Camera, in: In IEEE Int. Conf. on Imaging Systems and Techniques (IST'10), Thessaloniki, Grece, 2010.