

A General Approach for Egomotion Estimation with Omnidirectional Images

Raquel Frizera Vassallo^{†,‡}, José Santos-Victor[†], Hans Jörg Schneebeli[‡]

[†]Instituto de Sistemas e Robótica,
Instituto Superior Técnico,
Av. Rovisco Pais, 1,
1049-001 Lisboa - Portugal.
{raquel,jasv}@isr.ist.utl.pt

[‡]LAI - Laboratório de Automação Inteligente,
Departamento de Engenharia Elétrica,
Universidade Federal do Espírito Santo,
Vitória, ES - Brazil.
{raquel,hans}@ele.ufes.br

Abstract

Computing a camera's ego-motion from an image sequence is easier to accomplish when a spherical retina is used, as opposed to a standard retinal plane. On a spherical field of view both the focus of expansion and contraction are visible, whereas for a planar retina that is not necessarily the case.

Recent research has shown that omnidirectional systems can be used to emulate spherical retinas by mapping image velocity vectors from the omnidirectional image to the spherical retina. That has been done by using the Jacobian of the transformation between the image formation model and the spherical coordinate system. As a consequence, the Jacobian matrix must be derived for each specific omnidirectional camera, to account for the different mirror shapes.

Instead, in this paper we derive the Jacobian matrix using a general projection model, that can describe all single projection center cameras by suitable parameterization. Hence, both the back-projection of an image point to the unit sphere, as well as the mapping of velocities through the transformation Jacobian remains general for all cameras with a single center of projection.

We have conducted a series of experimental tests to illustrate the validity of our approach which lead to encouraging results.

1. Introduction

For many applications in autonomous navigation, the estimation of the observer motion (ego-motion) is a very important step. The ego-motion estimation problem consists of retrieving the observer instantaneous rotation and (direction of) translation while moving through the environment,

taking an image sequence as the input.

Most of the time, the process of ego-motion estimation starts by computing image motion and then using those vectors and properties of the motion field equations to estimate ego-motion. Usual approaches are based on point correspondences [1], optical flow [2] or the so-called direct methods [3, 4].

For the usual cameras that can be modeled as the perspective projection of the 3D structure onto an image plane, ego-motion is difficult to estimate. Even though the information about the observer motion is present in the images, it is very common that the direction of translation (focus of expansion) lies outside their small field of view, which is a challenge for most methods. In addition, different camera motions can produce similar motion fields in the image [5]. A simple example is the case of a camera looking along the Z-axis, where a rotation about the X-axis produces a similar effect to a translation along the Y-axis. Because of the limitations of small field of view images, most ego-motion methods become very sensitive to noise and also to camera orientation.

Instead, if we consider that the observer's retina is spherical, as opposed to planar, ego-motion becomes easier to estimate [6]. For instance, with a hemispherical field of view, either the focus of expansion (FOE) or the focus of contraction (FOC) will be visible in the image. If a whole spherical field of view is available, it is guaranteed that both FOE and FOC are inside the image.

Similarly, by using omnidirectional images, one can benefit from the basic advantages of having a spherical retina for ego-motion estimation. A significant effort has been done in order to design different omnidirectional systems [7, 8] to address different tasks. Since the enlarged field of view significantly improves the robustness of tracking or robot localization, wide-angle images have been

used in many applications of autonomous robot navigation [9, 10, 11] including motion estimation [12, 13].

In the remaining of the paper, we will refer to *planar perspective projection* to indicate the pin-hole camera model with a planar retina and designate the projection to a spherical retina as *spherical projection*.

Gluckman and Nayar [14] showed that good results for ego-motion can be obtained by using omnidirectional images. They start by computing the image motion with a sequence of omnidirectional images. Then, the image motion is mapped to the surface of a unit sphere, as if the camera had a truly spherical retina. This mapping is performed by means of determining the Jacobian of the transformation between spherical projection model and the image formation model of a catadioptric omnidirectional camera. Once the image motion vectors are mapped to the unit sphere, well known ego-motion algorithms can be applied, after small adaptations to the spherical projection model.

However in [14], a different Jacobian function must be determined for each distinct omnidirectional system, according to the specific projection model of the camera. Instead, in this work, we present a general Jacobian function that can describe a large set of possible catadioptric omnidirectional cameras.

We rely on the general projection model defined by Geyer and Daniilidis [15]. This model can represent many single projection center systems, including omnidirectional cameras. Using this model, we define a general expression for the Jacobian relating image flow and flow on the spherical retina. The derived Jacobian can be adjusted for different wide-angle cameras simply by changing a pair of parameters.

We have conducted several experiments, where the image motion on omnidirectional images was computed using the tracker of Lucas and Kanade [16, 17]. Then, the omnidirectional flow field is projected to the unit sphere using the proposed general Jacobian function. Then we used the algorithm of Bruss and Horn [18] to estimate ego-motion and illustrate the validity of our approach.

Section 2 presents the image formation model for single projection center cameras, while the Jacobian calculation is described in Section 3. The motion field on a spherical retina and the considered constraint for ego-motion is studied in Section 4. Finally, some of the experimental results and conclusions are presented in the last two sections.

2. The General Projection Model

The Projection Model defined by [15] and studied in [19], represents in an unified manner several single projection center systems, like pin-hole cameras and many omnidirectional cameras based on hyperbolic, elliptical or parabolic mirrors. This model combines a mapping of a 3D

point P to a sphere followed by a projection to the image plane. The center of the sphere lies on the optical axis of the projection to the plane (see Figure 1).

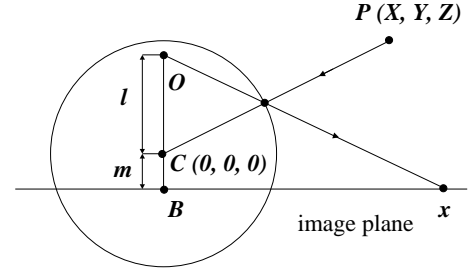


Figure 1. The general projection model for single projection center cameras.

Different camera representations are obtained simply by changing the values of parameters l and m , which represent the normalized distances from the sphere center of the projection model to the projection center O and projection plane. According to the model, the image projection $p(x, y)$ of a 3D point $P(X, Y, Z)$, can be defined by:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{l+m}{lR-Z} \begin{bmatrix} X \\ Y \end{bmatrix} \quad \text{with} \quad R = \sqrt{X^2 + Y^2 + Z^2} \quad (1)$$

It is important to mention that the camera intrinsic parameters, image center and focal length are not considered in the equation above. In the following section, we will see how to inverse project points and velocity vectors from the image plane to the surface of the (unit) spherical retina.

3. From the image plane to the spherical retina

In order to project image flow vectors to the surface of the unit sphere, we need first to know the inverse operation of Equation (1), usually denoted as back-projection, [19].

By back-projecting an image point (x, y) , we obtain a point on the unit sphere $(\hat{X}, \hat{Y}, \hat{Z})$ that represents the direction of the incoming ray from the original 3D point. The back-projection equation is derived from Equation (1), yielding:

$$\begin{bmatrix} \hat{X} \\ \hat{Y} \end{bmatrix} = \frac{l(l+m) + \text{sign}(l+m)\sqrt{(x^2+y^2)(1-l^2) + (l+m)^2}}{x^2+y^2 + (l+m)^2} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\hat{Z} = \pm \sqrt{1 - \hat{X}^2 - \hat{Y}^2} \quad (2)$$

where \hat{Z} becomes negative if $|l+m|/l > \sqrt{x^2+y^2}$ and positive otherwise.

To map image velocity vectors to the surface of the spherical retina, we need to calculate the Jacobian of the

back-projection Equation (2). The Jacobian relates temporal derivatives (velocities) in the spherical coordinate system to those in the image frame. It is obtained by differentiating the spherical coordinates $(\hat{X}, \hat{Y}, \hat{Z})$ with respect to the image coordinates (x, y) .

$$J = \begin{bmatrix} \frac{\partial \hat{X}}{\partial x} & \frac{\partial \hat{X}}{\partial y} \\ \frac{\partial \hat{Y}}{\partial x} & \frac{\partial \hat{Y}}{\partial y} \\ \frac{\partial \hat{Z}}{\partial x} & \frac{\partial \hat{Z}}{\partial y} \end{bmatrix} \quad (3)$$

Since the projection model is valid for all cameras with a single projection center, the corresponding Jacobian can be applied to many different omnidirectional cameras (also with a single projection center) simply by changing the values for the parameters l and m in the equations. The partial derivatives for the *generic Jacobian* are listed below and were derived for the case when $(l + m) < 0$:

$$\begin{aligned} \frac{\partial \hat{X}}{\partial x} &= -x^2 \frac{(1-l^2)}{\sqrt{AB}} - 2x^2 \frac{C}{B^2} + \frac{C}{B} \\ \frac{\partial \hat{X}}{\partial y} &= -xy \frac{(1-l^2)}{\sqrt{AB}} - 2xy \frac{C}{B^2} \\ \frac{\partial \hat{Y}}{\partial x} &= \frac{\partial \hat{X}}{\partial y} \\ \frac{\partial \hat{Y}}{\partial y} &= -y^2 \frac{(1-l^2)}{\sqrt{AB}} - 2y^2 \frac{C}{B^2} + \frac{C}{B} \\ \frac{\partial \hat{Z}}{\partial x} &= \text{sign}(\hat{Z}) x \frac{\{[\frac{(1-l^2)}{\sqrt{A}} + 2\frac{C}{B}](x^2 + y^2) - C\}C}{B^2 \sqrt{1 - \frac{C^2}{B^2}(x^2 + y^2)}} \\ \frac{\partial \hat{Z}}{\partial y} &= \text{sign}(\hat{Z}) y \frac{\{[\frac{(1-l^2)}{\sqrt{A}} + 2\frac{C}{B}](x^2 + y^2) - C\}C}{B^2 \sqrt{1 - \frac{C^2}{B^2}(x^2 + y^2)}} \end{aligned} \quad (4)$$

where

$$\begin{aligned} A &= (x^2 + y^2)(1-l^2) + (l+m)^2 \\ B &= x^2 + y^2 + (l+m)^2 \\ C &= l(l+m) - \sqrt{A} \end{aligned}$$

The spherical flow field, U , on the surface of the spherical retina, can be calculated by multiplying the (planar) im-

age flow field by the Jacobian function:

$$U = J \begin{bmatrix} \frac{\partial x}{\partial t} \\ \frac{\partial y}{\partial t} \end{bmatrix}$$

with

$$U = \left[\frac{\partial \hat{X}}{\partial t} \quad \frac{\partial \hat{Y}}{\partial t} \quad \frac{\partial \hat{Z}}{\partial t} \right]^T \quad (5)$$

Having described how to map the image flow computed from an omnidirectional image sequence to the flow that would have been obtained with a spherical retina, we can now apply ego-motion estimation methods to this flow field.

4. Ego-motion from the motion field on a Sphere

As shown in [6], a spherical retina would be advantageous for the computation of ego-motion. The Jacobian matrix presented in the previous section allows us to re-map the (planar) image flow to the surface of a spherical retina.

To use the obtained spherical motion field for ego-motion estimation, we still need to define the spherical motion field equation and show how it relates to the ego-motion parameters. For a camera moving with linear and angular velocities, $T = (T_x, T_y, T_z)$ and $\Omega = (\omega_x, \omega_y, \omega_z)$, the velocity of a 3D point P relative to the camera is given by:

$$\dot{P} = -T - \Omega \times P \quad (6)$$

The projection, \hat{P} , on the unit sphere of the point P can be computed using the spherical projection equation:

$$\hat{P} = \frac{P}{\|P\|} \quad (7)$$

If we differentiate Equation (7) with respect to time and use Equation (6) to substitute \dot{P} in the resulting expression, we get the equation of the spherical retina motion field:

$$U(\hat{P}) = \frac{1}{\|\hat{P}\|} ((T \cdot \hat{P})\hat{P} - T) - \Omega \times \hat{P} \quad (8)$$

This equation describes the velocity U at point \hat{P} on the unit sphere as a function of the camera motion (T, Ω) and depth, $\|P\|$. Hence, our goal is to estimate Ω and T , from observations of the spherical motion field U .

As the depth $\|P_i\|$ for each projected point \hat{P}_i on the unit sphere is not usually known, Equation (8) cannot be used directly, which makes the problem harder. When the linear velocity is different from zero, it is possible to re-write Equation (8) to remove the depth dependence and use this new formulation to estimate ego-motion.

In [14], several well know ego-motion algorithms designed for planar perspective projection were adapted to spherical projection. In this paper, we focus our attention to one of those methods to illustrate the validity of our approach. We used the Bruss and Horn [18] algorithm and the depth dependency on the spherical motion field Equation (8) is removed by taking the cross product with \hat{P} and the dot product with T :

$$T \cdot (\hat{P} \times (U + (\Omega \times \hat{P}))) = 0 \quad (9)$$

Ego-motion is estimated through an iterative process. From Equation (9), a set of velocity vectors on the unit sphere are used to find a least squares estimate of Ω as a function of T . Then the result is substituted back into (9), resulting in a non-linear constraint on T . The direction of T is computed by non-linear minimization considering $\|T\| = 1$. The method used is basically the one described in [20].

However, Equation (9) can only be used when the camera translation T is different to zero. For pure rotation we must resort to the original Equation (8). In this case, the dependence on depth disappears because of the absence of translation and Ω can be obtained by a simple least square estimation.

5. Experiments and Results

We have proposed a general approach for ego-motion with omnidirectional images. We adopted a projection model valid for all single center of projection cameras to develop a general expression for the Jacobian that projects (planar) image vectors onto the unit sphere. Then, motion field on the spherical retina is used to calculate the camera motion. In this section we show some experimental results that demonstrate the utility of our approach.

The robot used in our experiments was a TRC Labmate (see Figure 2-a). It is a differential-driven robot equipped with an on board computer (Pentium II MMX - 350 MHz - 160 RAM). The vision system consists of an omnidirectional catadioptric system, composed by the camera pointing upwards to a spherical mirror. This system is mounted on top of the mobile robot with its axis coincident to the platform's rotation axis (see Figure 2-b).

Even though the camera with the spherical mirror does not have a single projection center system, we have shown in [19] that our sensor can be well approximated by the projection model presented in the Section 2. The parameters values were identified to $l = 1.81521$ and $m = -6.8666$.

The reference coordinate frame is defined by the camera position. The Z -axis is coincident with the optical axis and

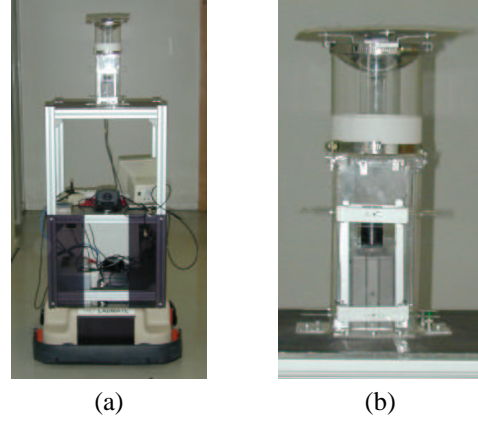


Figure 2. The robot (a) and the catadioptric vision system (b).

points upwards, the X -axis points to the robot backward direction and the Y -axis points to the robot right hand side. Thus, according to the adopted reference the forward motion of the robot corresponds to a negative velocity in the X direction.

The first experiment corresponds to pure translational motion. The robot moved along the X -axis in the negative direction with a velocity of 50 mm/s. Images were taken during motion and the image flow was calculated with a tracker based on Lucas and Kanade approach [17]. A pair of omnidirectional images used by the tracker and the resulting motion field are shown in Figure 3.

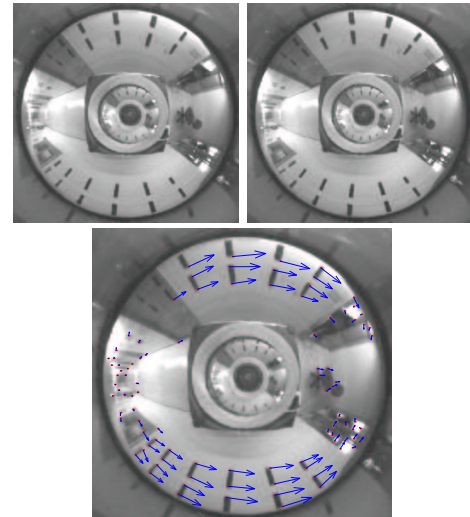


Figure 3. Two images used for computing image flow (top), selected points and vectors obtained by the tracker (bottom).

The (planar) image vectors were projected onto the unit sphere using the general expression for the Jacobian defined in Section 3. The camera motion was computed using the Bruss and Horn ego-motion algorithm. The error in the direction of translation and rotation are calculated as shown in Equation (10). As translation can only be recovered up to a scale factor, only its direction is considered for the error assessment, as we always keep $\|T\| = 1$:

$$\begin{aligned} e_t &= \arccos(\hat{T} \cdot T) \\ e_\Omega &= \hat{\Omega} - \Omega \end{aligned} \quad (10)$$

where T and Ω stand for the values obtained by odometry readings and \hat{T} and $\hat{\Omega}$ are the estimated translation and rotation.

The obtained spherical motion field, expressed in (m/s) , and the calculated ego-motion are shown in Figure 4. Vectors are scaled proportionally to their real size in order to fit the graph. Table 1 shows the estimated values for the translation direction (\hat{T}), the rotation vector ($\hat{\Omega}$) and the corresponding errors.

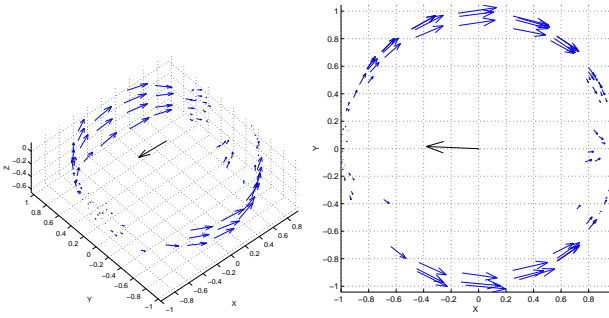


Figure 4. A 3D view and a top view of the projected motion field on the unit sphere in the camera reference frame. Motion vectors are measured in (m/s) . The vector in the center indicates the estimated direction of translation. The rotation vector is not seen because it is almost zero. All vectors are scaled to fit the graph.

For a second experiment, the robot was moved combining translation and rotation: a negative translation along the X -axis (50 mm/s) and a positive rotation around the Z -axis ($\approx 2^\circ/s$). Results are presented in Figure 5 and Table 2.

In these experiments, the estimation errors both for the angular speed estimate and the direction of translation are quite small, thus validating our approach.

In the final experiment, the robot moves with a pure rotation. As mentioned before, for this special case when translation is zero $T(0, 0, 0)$, Equation (9) cannot be applied. A least square estimate for Ω is obtained using the original

Translation	$T (T_x \ T_y \ T_z)$
odometry	$T = [-1 \ 0 \ 0]$
estimated	$\hat{T} = [-0.9990 \ 0.0425 \ 0.0109]$
error	$e_t = 2.5176^\circ$
Rotation	$\hat{\Omega} (\omega_x \ \omega_y \ \omega_z) (^\circ/s)$
odometry	$\Omega = [0 \ 0 \ 0]$
estimated	$\hat{\Omega} = [0.0024 \ -0.0397 \ 0.0281]$
error	$e_\Omega = [0.0024 \ -0.0397 \ 0.0281]$
	$\ e_\Omega\ = 0.0486$

Table 1. Ego-motion values and errors for pure translation.

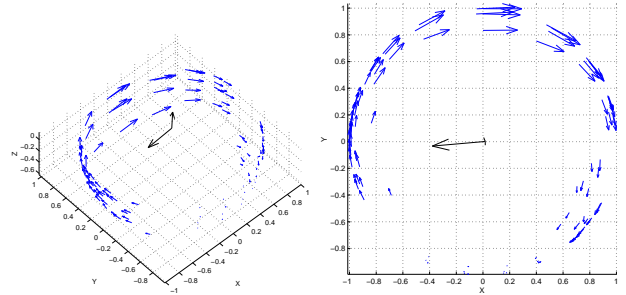
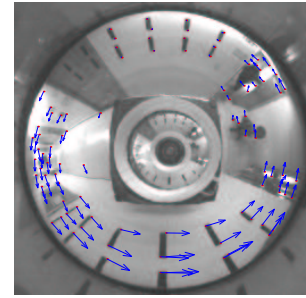


Figure 5. Results for combined camera motion - translation along the X -axis and rotation around the Z -axis.

Translation	$T (T_x \ T_y \ T_z)$
odometry	$T = [-0.9922 \ -0.1244 \ 0]$
estimated	$\hat{T} = [-0.9953 \ -0.0876 \ 0.0408]$
error	$e_t = 3.1540^\circ$
Rotation	$\hat{\Omega} (\omega_x \ \omega_y \ \omega_z) (^\circ/s)$
odometry	$\Omega = [0 \ 0 \ 1.9652]$
estimated	$\hat{\Omega} = [0.1261 \ 0.0286 \ 2.0168]$
error	$e_\Omega = [0.1261 \ 0.0286 \ 0.0516]$
	$\ e_\Omega\ = 0.1392$

Table 2. Ego-motion values and errors for the combined motion.

Equation (8) which is no longer depended of depth. The results for the pure rotation around the Z -axis ($\approx 5^\circ/s$) is shown in Figure 6 and Table 3.

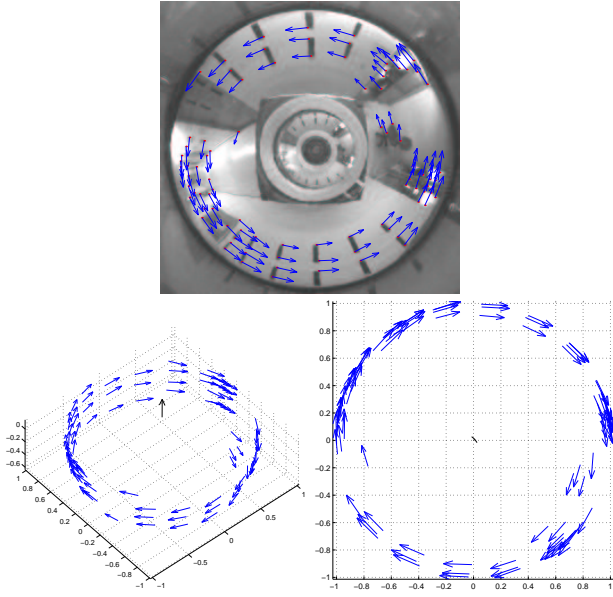


Figure 6. Results for pure rotation around the Z -axis.

Rotation	$\Omega (\omega_x \ \omega_y \ \omega_z) (^\circ/s)$
odometry	$\Omega = [0 \ 0 \ 4.7498]$
estimated	$\hat{\Omega} = [0.1261 \ 0.1146 \ 3.8560]$
error	$e_\Omega = [0.1261 \ 0.1146 \ -0.8938]$ $\ e_\Omega \ = 0.9099$

Table 3. Estimated value and error for pure rotation.

Without knowing the camera motion in advance, it is difficult to select amongst the two available models: the combined motion model - Equation(9) - or the pure rotation model - Equation(8) with $T = 0$.

One possibility is to apply first the pure rotation model to estimate a (pure) rotation. The quality of the fitting can be assessed by looking at the residuals, r , between the observed flow and the reconstructed flow of the pure rotation flow model:

$$r_i = U_i + \hat{\Omega} \times \hat{P} \quad (11)$$

where $\hat{\Omega}$ is the angular velocity estimate.

The mean residual vector, $\bar{r} = \text{mean}(r_i)$ and standard deviation $\sigma_r = \text{trace}(\text{cov}(r_i))$, are used as performance indexes of the model fitting procedure. In fact, if we use a pure rotation model, when the camera is undergoing a pure

translation or a combined motion, one can expect that the residuals will be biased and present a significant standard deviation, when compared to the usage of the correct model.

Choosing \bar{r} and σ_r as decision variables, a threshold could be defined for model selection. When \bar{r} and σ_r exceed the defined limits, the combined motion model - Equation(9)- should be used.

To illustrate this idea we applied the pure rotation model - Equation (8) - to the three examples presented in the previous experiments. Table 4 shows that the values for \bar{r} and σ_r are much larger for the pure translation and combined motion cases than for the pure rotation case. That observation shows the applied model does only fit the third test set. A threshold could be defined somewhere between the observed values. A numerous set of trials and measures should be done from the motions of interest to define the those limits, with statistical significance.

Flow error (m/s)	Pure Translation
Mean	$\bar{r} = [0.0151 \ 0.0010 \ -0.0004]$
Standard deviation	$\ \bar{r} \ = 0.0151$ $\sigma_r = 0.0234$
Flow error (m/s)	Combined Motion
Mean	$\bar{r} = [0.0152 \ 0.0002 \ 0.0004]$
Standard deviation	$\ \bar{r} \ = 0.0152$ $\sigma_r = 0.0235$
Flow error (m/s)	Pure Rotation
Mean	$\bar{r} = [-0.0007 \ -0.0003 \ -0.0023]$
Standard deviation	$\ \bar{r} \ = 0.0024$ $\sigma_r = 0.0079$

Table 4. Mean and standard deviation values for the residual between the reconstructed spherical motion field (considering the pure rotation model) and the observed motion. The lowest values occur for the pure rotation case.

6. Conclusion

We have addressed the problem of ego-motion estimation from omnidirectional images. The original motivation started from the observation that the observer motion parameters are easier to estimate when using a spherical retina than with a planar one.

Previous research has shown how to project image velocity vectors computed from omnidirectional images to the surface of a (unit) spherical retina. Then existing ego-motion algorithms designed for planar images could be adapted to the spherical motion field. In [14], the transformation between (planar) image vectors to spherical vectors

is achieved by means of a Jacobian matrix which is specific of the used omnidirectional system. Thus every time a different wide-angle imaging system is used, a new equation for the Jacobian must be derived.

To overcome this limitation, we proposed a general expression for the Jacobian that can be used for many different omnidirectional cameras, simply by setting two parameters. We rely on a general image formation model, valid for all single center of projection cameras. Therefore, the equation of the back-projection to the unit sphere and the Jacobian remain general for many different systems.

We demonstrate the validity of our approach by using the Jacobian to project image velocity vectors from omnidirectional images to the surface of the unit sphere. Then, an ego-motion algorithm is used to estimate camera motion.

We also discussed the problem of choosing amongst two models for ego-motion estimation. One model considers rotational motion only, while the second encompasses simultaneous translation and rotation. The selection criterion is based on the mean and standard deviation of the residuals between the reconstructed motion field and the observed one. The results obtained were quite encouraging.

References

- [1] F. Lustman, O. Faugeras, and G. Toscani, "Motion and structure from motion from point and line matches," *Proc. of First Int'l Conf. Computer Vision*, June 1987.
- [2] Heeger D. J. and Jepson A. D., "Subspace methods for recovering rigid motion," *IJCV*, vol. 7, no. 2, pp. 95–117, 1992.
- [3] C. Fermuller and Y. Aloimonos, "Direct perception of three-dimensional motion from patterns of visual motion," *Science*, vol. 270, pp. 1973–1976, December 1995.
- [4] C. Silva and J. Santos-Victor, "Robust egomotion estimation from the normal flow using search subspaces," *IEEE Transactions on PAMI*, vol. 19, no. 9, pp. 1026–1034, September 1997.
- [5] K. Danidellis and H.-H. Nagel, "The coupling of rotation and translation in motion estimation of planar surfaces," In *IEEE Conf. on Computer Vision and Pattern Recognition*, 1993.
- [6] R.C. Nelson and J. Aloimonos, "Finding motion parameters from spherical motion fields (or the advantages of having eyes in the back of your head)," *Biological Cybernetics*, vol. 58, pp. 261–273, 1988.
- [7] S. Baker and S. K. Nayar, "A theory of catadioptric image formation," In *Proc. of 6th International Conference on Computer Vision - ICCV98*, 1998.
- [8] S. Baker and S. K. Nayar, "A theory of single-viewpoint catadioptric image formation," *International Journal of Computer Vision*, vol. 32, no. 2, pp. 175–196, 1999.
- [9] N. Winters, J. Gaspar, G. Lacey, and J. Santos-Victor, "Omni-directional vision for robot navigation," *IEEE Workshop on Omnidirectional Vision (held with CVPR)*, June 2000.
- [10] Y. Matsumoto, K. Ikeda, M. Inaba, and H. Inoue, "Exploration and navigation in corridor environment based on omni-view sequence," In *Proc. of IROS 2000*, 2000.
- [11] D. Coombs, M. Herman, T. Hong, and M. Nashman, "Real-time obstacle avoidance using central flow divergence and peripheral flow," *Proc. of ICCV'95*, 1995.
- [12] J.W. Lee, S. You, and U. Neumann, "Large motion estimation for omnidirectional vision," *Proc. of IEEE Workshop on Omnidirectional Vision*, June 2000.
- [13] T. Svoboda, T. Pajdla, and Vaclav Hlavac, "Motion estimation using central panoramic cameras," In *IEEE Conf. on Intelligent Vehicles*, October 1998.
- [14] J. Gluckman and S. K. Nayar, "Ego-motion and omnidirectional cameras," In *Proc. of 6th International Conference on Computer Vision - ICCV98*, 1998.
- [15] C. Geyer and K. Daniilidis, "Equivalence of catadioptric projections and mappings of the sphere," In *Proc. of IEEE Workshops on Omnidirectional Vision*, 2000.
- [16] J. Shi and C. Tomasi, "Good features to track," In *Proc. of CVPR94*, 1994.
- [17] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision.," *IJCAI'81*, 1981.
- [18] A. Bruss and B. K. P. Horn, "Passive navigation," *Computer Vision, Graphics and Image Processing*, vol. 21, pp. 276–283, 1983.
- [19] J. Gaspar, E. Grossmann, and J. Santos-Victor, "Interactive reconstruction from an omnidirectional image," In *Proc. of the 9th International Symposium on Intelligent Robotic Systems - SIRS2001*, 2001.
- [20] T.Y. Tian, C. Tomasi, and D.J. Heeger, "Comparison of approaches to egomotion computation," *Proc. of CVPR'96*, 1996.