Visual SLAM with an omnidirectional camera

Alejandro Rituerto
DIIS-I3A
Universidad de Zaragoza
Zaragoza, Spain
Email: aleritu@gmail.com

Luis Puig
DIIS-I3A
Universidad de Zaragoza
Zaragoza, Spain
Email: lpuig@unizar.es

J.J. Guerrero
DIIS-I3A
Universidad de Zaragoza
Zaragoza, Spain
Email: jguerrer@unizar.es

***Abstract*—In this work we integrate the Spherical Camera Model for catadioptric systems in a Visual-SLAM application. The Spherical Camera Model is a projection model that unifies central catadioptric and conventional cameras. To integrate this model into the Extended Kalman Filter-based SLAM we require to linearize the direct and the inverse projection. We have performed an initial experimentation with omnidirectional and conventional real sequences including challenging trajectories. The results confirm that the omnidirectional camera gives much better orientation accuracy improving the estimated camera trajectory.**

***Keywords*-Computer vision systems and applications; Vision sensors.**

I. Introduction

The use of omnidirectional cameras in robotics has increased in the last years. The main reason is their wide FOV. There are many omnidirectional systems, but the most used are the catadioptric systems, compound by a mirror and a camera. These systems have been used in applications such as surveillance [1], robot navigation [2] and 3D reconstruction [3]. In this work we study the application of monocular omnidirectional vision in one of the essential problems of autonomous perception and robotics: Simultaneous Localization and Mapping (SLAM).

SLAM problem [4] appears when an autonomous vehicle does not know neither its position nor the map of its surroundings, and only have partial measurements of the environment to navigate through it. There are many SLAM approaches that use conventional cameras as sensor [5], [6]. However just a few SLAM approaches use omnidirectional systems [7], [8], [9]. In this work we use an Extended Kalman Filter (EKF) algorithm to deal with the SLAM problem.

To obtain metric information from any vision system a projection model is required. Geyer and Daniilidis [10] propose an unified method to model any central catadioptric system. This model was extended by Barreto and Araujo [11] so that it can model central catadioptric systems and conventional cameras, and it is known as the Spherical Camera Model. This method models every system by a unit sphere and a conventional projection. We integrate this projection model into the EKF based SLAM application.

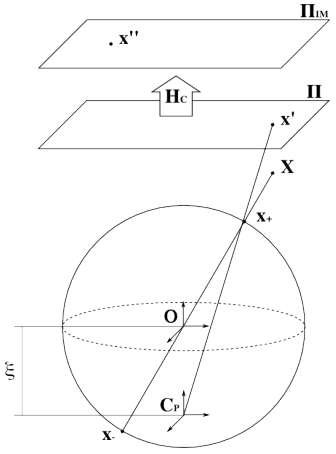


Figure 1. Spherical Camera Model

The wide FOV of the omnidirectional systems is the main advantage over the conventional vision systems. An omnidirectional system can keep tracking of image points in all directions while a conventional system can easily loose tracked points if they are not in front of the camera. In order to prove the advantages of the omnidirectional vision in SLAM we perform an initial comparison between an SLAM system using omnidirectional vision and the same application using conventional vision.

II. The Spherical Camera Model

First at all we present the projection model for the omnidirectional systems presented in [10] and extended in [11]. This model is widely used with omnidirectional vision systems.

The projection of a point in the image is explained as follows (Fig. 1). The scene point $\mathbf{X}$ is referred to the camera coordinates centered in $\mathbf{O}$. First, we compute the projection of the scene point $\mathbf{X}$ to the intersection of the unit sphere and the line joining the point and the center of the sphere $\mathbf{O}$. There are two intersection points, $\mathbf{x}_+$ and $\mathbf{x}_-$, but just one is fisically true. This fisically true point is projected to a virtual projection plane π trough the virtual projection centre $\mathbf{C}_P = (0,\ 0,\ -\xi)^\mathsf{T}$. The new point is $\mathbf{x}'$. These two steps

are coded in one equation:

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

The next step transforms the virtual plane π in the image plane π_{IM} through a homographic transformation H_C.

$$\mathbf{x}'' = \mathsf{H}_c\mathbf{x}' \quad (2)$$

$$\mathsf{H}_c = \mathsf{K}_c\mathsf{M}_c \quad (3)$$

where K_c includes the camera internal parameters and M_c includes the mirror and system parameters [11]. Finally, the image coordinates are computed by the next function:

$$\mathbf{u} = \begin{pmatrix} u \\ v \end{pmatrix} = f_u(\mathbf{x}'') = \begin{pmatrix} \frac{x''}{z''} \\ \frac{y''}{z''} \end{pmatrix} \quad (4)$$

The parameter of the model ξ define the shape of the mirror we work with. For conventional cameras $\xi = 0$. $\xi = 1$ for central catadioptric systems with parabolic mirror, and $0 < \xi < 1$ with hyperbolic mirror.

III. Localization and Mapping

An autonomous robot must be able to navigate through unknown environments. With SLAM techniques a robot can build a map of the environment and keep track of its pose in this map using noisy measurements of the environment. In Visual SLAM these measurements are characteristic points from the images.

The most used SLAM algorithms are based on the Kalman Filter, a filter that predicts the state of linear systems. As the geometry impose nonlinear functions the Extended Kalman Filter (EKF) [4] is used. The EKF linearize the non-linear relations by approximating them to its first order Taylor series.The EKF algorithm is divided into two parts. In the first part, *Prediction*, the new state of the system $\mathbf{x}_{k+1}$ is estimated from the previous timestep state $\mathbf{x}_k$. The second part of the algorithm, *Update*, uses the measurements of the environment $\mathbf{z}$ to improve this prediction. Every state variable is represented by its mean and its covariance. In our case the system state is coded in $\mathbf{x}_k$.

$$\mathbf{x}_k = (\underbrace{\mathbf{r},\ \mathbf{q},\ \mathbf{V},\ \boldsymbol{\omega}}_{\text{Camera state}},\ \underbrace{x_i,\ y_i,\ z_i,\ \theta_i,\ \phi_i,\ \rho_i}_{\text{Point}})^{\mathsf{T}} \quad (5)$$

where $\mathbf{r}_{(3\times1)}$ is the system pose, $\mathbf{q}_{(4\times1)}$ is the quaternion of the orientation and $\mathbf{V}_{(3\times1)}$ and $\omega_{(3\times1)}$ are the linear and angular velocities. The points are coded by inverse depth [12]. $(x_i,\ y_i,\ z_i)$ is the pose of the camera the first time it saw the 3D point i, θ_i and ϕ_i are the angles that determinate the ray pointing to the 3D point and ρ_i is the inverse depth of the point.

A. The Spherical Camera Model in the EKF

The EKF algorithm needs the first derivative of the measurement equation. As explained before we use the Spherical Camera Model in the measurement equation. So, first at all we have formulated the direct projection derivatives.

- *Jacobian of the Spherical Camera Model*

$$\mathsf{J} = \mathsf{J}_{\mathsf{f_u}}\mathsf{H}_c\mathsf{J}_\hbar \quad (6)$$

$$\mathsf{J}_{\mathsf{f_u}} = \begin{pmatrix} \frac{1}{z''} & 0 & -\frac{x''}{z''^2} \\ 0 & \frac{1}{z''} & -\frac{y''}{z''^2} \end{pmatrix} \quad (7)$$

$$\mathsf{J}_\hbar = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{\xi x}{\rho} & \frac{\xi y}{\rho} & 1+\frac{\xi z}{\rho} \end{pmatrix} \quad (8)$$

where $\rho = \sqrt{x^2+y^2+z^2}$.

To initialize new features we need to estimate the covariance of the new feature from the uncertainty of the image point. Through the inverse projection of the model is possible to estimate the 3D ray where an image point lies. The estimation of the covariance of a new feature can be done through the jacobian of the inverse projection. The inverse projection starts with the image coordinates $\mathbf{u} = (u,\ v)^{\mathsf{T}}$. The point $\mathbf{x}''$ is $\mathbf{x}'' = (u,\ v,\ 1)^{\mathsf{T}}$. The equations of the inverse model are

$$\mathbf{x}' = {\mathsf{H}_c}^{-1}\mathbf{x}'' \quad (9)$$

$$\mathbf{x} = \hbar(\mathbf{x}')^{-1} = \begin{pmatrix} x' \\ y' \\ z' - \frac{\xi(x'^2+y'^2+z'^2)}{\xi z'+\chi} \end{pmatrix} \quad (10)$$

where $\chi = \sqrt{(1-\xi^2)(x'^2+y'^2)+z'^2}$.

So, secondly we formulate the inverse projection derivatives.

- *Jacobian of the Spherical Camera Model inverse projection*

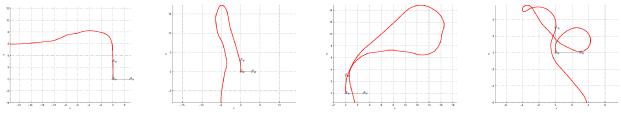
$$\mathsf{J} = {\mathsf{H}_c}^{-1}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{\partial z}{\partial x'} & \frac{\partial z}{\partial y'} & \frac{\partial z}{\partial z'} \end{pmatrix} \quad (11)$$

$$\frac{\partial z}{\partial x'} = -\frac{2\xi x'(\xi z'+\chi) - \frac{(1-\xi^2)\xi x'\rho^2}{\chi}}{(\xi z'+\chi)^2} \quad (12)$$

$$\frac{\partial z}{\partial y'} = -\frac{2\xi y'(\xi z'+\chi) - \frac{(1-\xi^2)\xi y'\rho^2}{\chi}}{(\xi z'+\chi)^2} \quad (13)$$

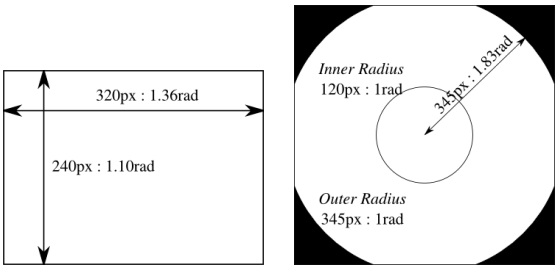
$$\frac{\partial z}{\partial z'} = -\frac{2\xi z'(\xi z'+\chi) - (\xi+\frac{z'}{\chi})\xi\rho^2}{(\xi z'+\chi)^2} \quad (14)$$

where $\chi = \sqrt{(1-\xi^2)(x'^2+y'^2)+z'^2}$ and $\rho^2 = x'^2 + y'^2 + z'^2$.



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

Figure 2. Selected trajectories



Conventional image Omnidirectional image

Figure 3. Angular resolution of the vision systems

IV. Experiments with real images

In this section we present the experiments performed with a Visual SLAM[1] application using conventional and omnidirectional images. The measurement points are codified by inverse depth parametrization [12]. The Spherical Model is used as measurement equation and combined with SIFT [13] features as measurement points. In both type of images we use a SIFT-based matching approach. We match the features in the next frame located inside an uncertainty ellipse considering a 95% of confidence.

We use two image sequences provided by The Rawseeds Project[2]. They were acquired with a hyper-catadioptric camera and with a conventional camera. We have calibrated the hyper-catadioptric camera [14] for better results. The ground truth is given by an improved odometry. To better analyze the behavior of these two approaches we have split the whole trajectory in partial trajectories which result hard to follow for the conventional SLAM. We choose four trajectories to perform the comparison (Fig. 2). The number of frames for the omnidirectional sequences goes from 400 frames for the first trajectory to 850 frames for the third one. The conventional sequences are compound by two times the number of frames of the omnidirectional ones in order to avoid the trajectory difficulties of the conventional vision.

Before making a valid comparison we evaluate the angular resolution of the cameras. The angular resolution of both systems is equivalent (see Fig. 3). All values are close to 200 pixels per radian.

Using monocular vision we cannot estimate the real scale of the 3D points. So, we decide to use the orientation angle to compare the results since camera translation and 3D coordinates of points are affected by scale. The estimated results are compared to those given by the odometry. This scale problem has been solved in the literature using stereo vision or using additional sensors.

Fig. 4 shows the SLAM results for the four selected trajectories. The dashed black lines correspond to the odometry ground truth, the blue line is the estimation with the omnidirectional camera and the red line the estimation with the conventional camera. Fig. 4(a), 4(b) and 4(c) shows the trajectory and orientation results for the three first trajectories, respectively. In the three cases the omnidirectional estimation of the trajectory follows the real one. The conventional trajectory also follows the real trajectory, but the solution is worse than the omnidirectional one. The orientation results for the omnidirectional camera are better than the conventional camera results, the values of the mean of the absolute error for the omnidirectional camera are lower that the conventional camera ones. Fig. 4(d) shows the trajectory and the orientation results for the fourth trajectory. This trajectory includes many turns. However, the estimation with the omnidirectional camera performs all the turns correctly. In this case we can see clearly the scale change in the same estimation. With the conventional camera the estimated trajectory do not follow the real movement. In the case of the orientation the errors are bigger than in trajectory four. For the omnidirectional system the mean of the absolute orientation error is 0.29 radians and for the conventional camera is 0.62 radians.

Table I shows the mean of the absolute orientation error for every trajectory and the two vision systems. These errors have been estimated with the SLAM orientation and the odometry orientation. For every trajectory the error with the omnidirectional camera is smaller that with the conventional camera, although we use double number of frames with that system. The mean absolute error for the omnidirectional system is 0.10 radians and for the conventional system is 0.29 radians.

Table I
Mean Absolute Error (MAE) of orientation in radians

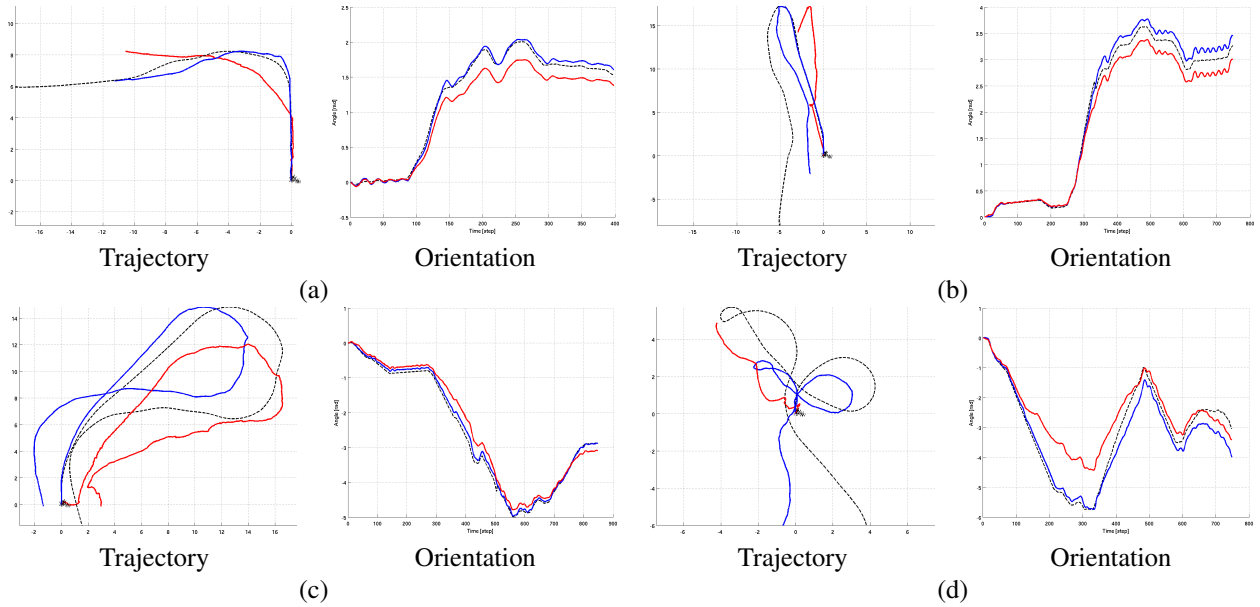
Trajectory	Omnidirectional	Conventional
1	0.04	0.16
2	0.11	0.15
3	0.07	0.22
4	0.29	0.62
MAE	0.10	0.29

V. Conclusions

We have developed a visual SLAM application based on the Extended Kalman Filter that can use catadioptric omnidirectional cameras and perspective cameras. To do this we have formulated the Spherical Camera Model Jacobians needed by the EKF. With that new application we have tested different image sequences of a hyper-catadioptric system

[1] http://www.robots.ox.ac.uk/~SSS06/Website/Practicals/~SSS06.Prac2.MonocularSLAM.tar.gz

[2] http://www.rawseeds.org



Blue line: Omnidirectional camera estimation; *Red line*: Conventional SLAM estimation; *Dashed black line*: Odometry values;

Figure 4. SLAM results for the selected trajectories. (a) shows the results for the first trajectory, (b) for the second trajectory, (c)) for the third trajectory and (d) shows results for the fourth trajectory. Each figure is compound by the trajectory and orientation plots.

and a perspective camera. Besides we have compared the results obtained of this testing. This comparison has shown the superiority of the omnidirectional systems in monocular visual SLAM.

ACKNOWLEDGMENT

This work has been supported by project DPI2009-14664-C02-01. Also thanks to the I3A fellowship program and J. Civera and J.M. Montiel for their basic SLAM tool and fruitful discussions.

REFERENCES

[1] G. Scotti, L. Marcenaro, C. Coelho, F. Selvaggi, and C. Regazzoni, "Dual camera intelligent sensor for high definition 360 degrees surveillance," *Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 250–257, 2005.

[2] J. S. Chahl and M. V. Srinivasan, "A complete panoramic vision system, incorporating imaging, ranging, and three dimensional navigation," in *OMNIVIS*, 2000, pp. 104–111.

[3] M. Lhuillier, "Toward flexible 3d modeling using a catadioptric camera," in *Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[4] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[5] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007.

[6] M. Pupilli and A. Calway, "Real-time visual slam with resilience to erratic motion," in *Computer Vision and Pattern Recognition*, vol. 1, June 2006, pp. 1244–1249.

[7] F.-S. Huang and K.-T. Song, "Vision slam using omnidirectional visual scan matching," in *IEEE/RSJ IROS 2008*, Sept. 2008, pp. 1588–1593.

[8] D. Scaramuzza and R. Siegwart, "Appearance guided monocular omnidirectional visual odometry for outdoor ground vehicles," IEEE Transactions on Robotics, Special Issue on Visual SLAM, October 2008.

[9] J.-P. Tardif, Y. Pavlidis, and K. Daniilidis, "Monocular visual odometry in urban environments using an omnidirectional camera," in *IEEE/RSJ IROS 2008*, September 2008, pp. 2531–2538.

[10] C. Geyer and K. Daniilidis, "A unifying theory for central panoramic systems and practical applications," in *ECCV (2)*, 2000, pp. 445–461.

[11] J. P. Barreto and H. Araujo, "Issues on the geometry of central catadioptric image formation," in *Computer Vision and Pattern Recognition*, 2001, pp. 422–427.

[12] J. Civera, A. J. Davison, and J. M. M. Montiel, "Inverse depth parametrization for monocular slam," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 932–945, October 2008.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 2004.

[14] Y. Bastanlar, L. Puig, P. Sturm, J. J. Guerrero, and J. Barreto, "Dlt-like calibration of central catadoptric cameras," in *Proceedings of the Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras, Marseille, France*, October 2008.