Engineering Research Express

PAPER

A monocular wide-field vision system for geolocation with uncertainties in urban scenes

To cite this article: Sebastián I Arroyo *et al* 2020 *Eng. Res. Express* **2** 025041

View the article online for updates and enhancements.

This content was downloaded from IP address 212.115.51.79 on 22/06/2020 at 13:18

Engineering Research Express

PAPER

A monocular wide-field vision system for geolocation with uncertainties in urban scenes

RECEIVED
20 February 2020

REVISED
27 May 2020

ACCEPTED FOR PUBLICATION
10 June 2020

PUBLISHED
22 June 2020

Sebastián I Arroyo[1], **Ulises Bussi**[1,2], **Félix Safar**[1] and **Damián Oliva**[1,2]

[1] Laboratorio de Instrumentación, Automatización y Control, Departamento de Ciencia y Tecnología, Universidad Nacional de Quilmes, Buenos Aires, B1876BXD, Argentina

[2] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina

E-mail: sebastian.arroyo@unq.edu.ar

Keywords: wide-field vision, omnidirectional camera, biologically inspired computer vision, camera calibration, bayesian inference, object geolocation

Abstract

In engineering applications related to video surveillance, the use of monocular omnidirectional cameras would reduce costs and complications associated with infrastructure, installation, synchronization, maintenance and operation of multiple cameras. This makes omnidirectional cameras very useful for transport analysis, a key task of which is to accurately geolocate vehicles and/or pedestrians observed in an ample region. The problem of measuring on the plane was previously solved for monocular central perspective images. However, the problem of determining uncertainties in geolocalization using monocular omnidirectional images, has not been addressed. This problem is not trivial due to the complexity of the image formation models associated with these cameras. The contributions of this work are: (1) The geolocation problem is solved using omnidirectional monocular images through a Bayesian inference approach. (2) The calculation of Bayesian marginalization integrals is simplified through first-order approximations. (3) The accuracy of the estimated positions and uncertainties is shown through Monte Carlo simulations under realistic measurement conditions. (4) The method to geolocate a vehicle's trajectory on a satellite map is applied in an urban setting.

1. Introduction

Currently, central perspective projection cameras (described by the pin-hole model) are used to solve these video-surveillance tasks. These cameras typically have a maximum field of view of approximately $60^\circ \times 60^\circ$, so in order to observe an ample region, it's necessary to use numerous cameras increasing costs and complications associated with infrastructure, installation, synchronization, maintenance and operation. A very common solution is to use pan-tilt-zoom (PTZ) cameras whose direction of observation can be remotely commanded by an operator. However, they have the limitation that they can only observe one region at a time, that is when the operator reorients the camera it loses vision from other regions. Additionally, for the PTZ camera to monitor a broad panorama without being continuously controlled by a human operator requires programming frequent automatic reorientations, generating mechanical friction and reducing the camera's lifespan.

The limitations mentioned above can be solved using omnidirectional cameras (OCs) with a visual field of approximately $360^\circ \times 180^\circ$, allowing the observation of the hemisphere of interest of the scene [1, 2]. There are several sensors which can be used to achieve a wide field vision, such as: synthetic compound eyes, catadioptric and dioptric cameras. Synthetic compound eyes are sensors of reduced size that generally use a set of photodiodes mimicking an ommatidium ordered array [3]. Due to the size and reduced weight of these sensors, they are widely used in small mobile robots; however, given their low resolution they are not used in video surveillance tasks. Catadrioptic cameras use a standard digital camera along with a specially curved reflective surface to increase camera's field of view. [4, 5]. This is a convenient and flexible approach, since the mirror profile can be adapted to achieve greater resolution in certain directions of interest. Nevertheless catadioptric

© 2020 IOP Publishing Ltd

OCs tend to be rather bulky and costly. Finally, the omnidirectional dioptric cameras employ a fisheye lens with a field of view so wide that it extends a few degrees behind the camera. Fisheye lenses resemble the natural underwater phenomenon of how a fish sees a hemispherical upward view from beneath the water, known as Snell's window. At present there are different commercial versions of these cameras that can produce a reasonable angular resolution using high resolution CCD sensors (around 5 mega pixels) [6].

Given the previously mentioned potential advantages of using OCs this work is focused on the development of a wide-field video surveillance for the monitoring of urban scenes. In this application, the objects of interest (for example vehicles and/or pedestrians) are bound to earth's surface due to the gravitational pull [2, 7, 8], and most of the events which need attention or cautionary measures, take place below the horizon. Therefore the objective of this system is to measure the geographical location of objects on the terrestrial plane along with determining the uncertainty of the measurement (figure 1(a)).

The problem of measuring in the world plane from central perspective images, and accurately predicting the uncertainty of these measurements was solved by Criminisi *et al* [9]. They use a homography transformation to map positions from the image to a world plane predicting uncertainty with a first order model, and take into account uncertainties in the image input points and in the homography matrix. They use a linear distortion model and estimate projection parameters minimizing the projection error in the world plane and demonstrate that first order analysis is accurate.

In order to make measurements of the location of objects over a terrestial plane with the OCs, the first step is to model and correct the distortions (figure 1(a)). In this sense, there is a wide bibliography [4, 10, 11] on calibration of catadioptric and dioptric OCs. However, the problem of determining uncertainties in the process of predicting location with OCs has not been addressed. This problem is not trivial due to the complexity of the image formation models associated with these cameras, thus approaching it from a Bayesian perspective is the main contribution of this work.

The Bayesian approach is well suited to formulate both, camera calibration and position estimation problems in explicit probabilistic terms It has been demonstrated for the case of 3D reconstruction by Sundareswara and Schrater [12] that Bayesian prediction marginalizes on the parameters making it less susceptible to statistical fluctuations than the plug-in approach where only the value of the most likely parameter is used. Together with Civera *et al* [13] they use a Bayesian approach to intrinsic and extrinsic calibration in addition to 3D scene estimation. But they rely on multiple view geometry because they use the movement of the camera. Also due to the complexity of the projection models, sampling algorithms are used for the estimation of parameters (calibration) and scene positions.

This paper faces the problem of calibration and prediction with a Bayesian approach with a single static OC already installed in an urban setting and develops a calibration method for localization in the ground plane. The calibration is simple and only requires an operator to manually match some fiducial points in both a satellite image and the OC image. Bayesian marginalizations integrals are simplified by assuming the projection function can be approximated by a first order Taylor series which result in fast calculations. This makes the method suitable for real time localisation with uncertainty.

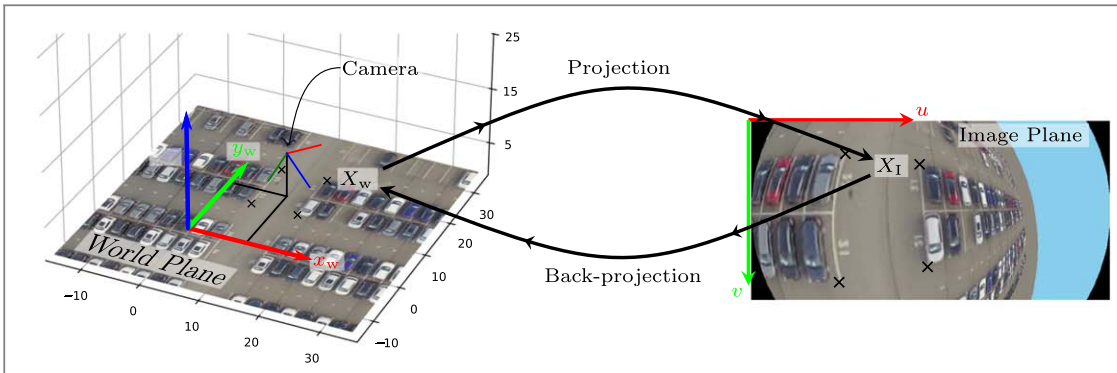
This work is organized as follows: In section 2 the workflow of the proposed method is described, the geometric model of image formation (*projection*) that maps world coordinates to image coordinates, and the inverse (*back-projection*) in which world coordinates from image coordinates can be obtained. In section 3 the Bayesian approach is described in detail, along with its implementation in Python scripts to be applied to real data. In section 4 the linear approximation and the prediction algorithm are shown to be valid estimations through Monte Carlo simulations under realistic conditions of application, later applying this method in real data for the geolocation of vehicle trajectories.

2. Materials and methods

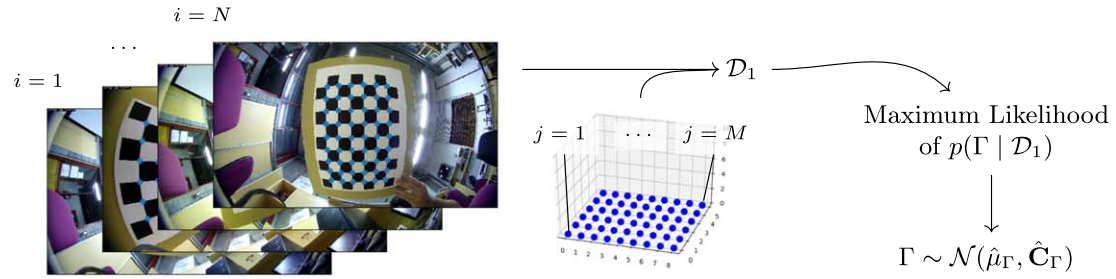
2.1. Workflow of the proposed method

The model of image formation depends on both the OC's intrinsic parameters (such as focal length, radial distortion and CCD optical center, symbolised as Γ) and extrinsic parameters describing the relative pose of the camera to the world (position and orientation, symbolised as Θ) [14, 15]. A *back-projection* function $\mathcal{F}$ mapping coordinates X_{I} to world plane coordinates X_{w} using the camera's parameters Γ and Θ can be calculated (figure 1(a)).

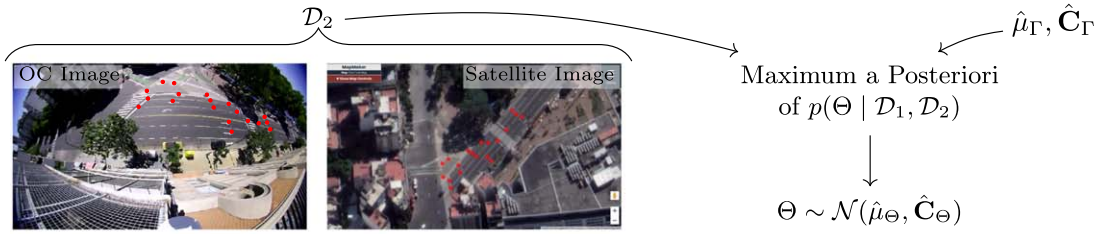
The calibration's first step (figure 1(b)) is performed in the laboratory by detecting multiple corners of a chessboard with known position in the world plane (X_{w}). This process generates an extensive set of training data called $\mathcal{D}_1$ from which an estimate the intrinsic parameters and their uncertainties (denoted as the mean $\hat{\mu}_{\Gamma}$, and the variance $\hat{\mathbf{C}}_{\Gamma}$). After this, it is assumed that the OC is installed in the field observing the region of interest (figure 1(c)). In this setup, a reduced set of world points X'_{w} and their correspondence in the image are observed



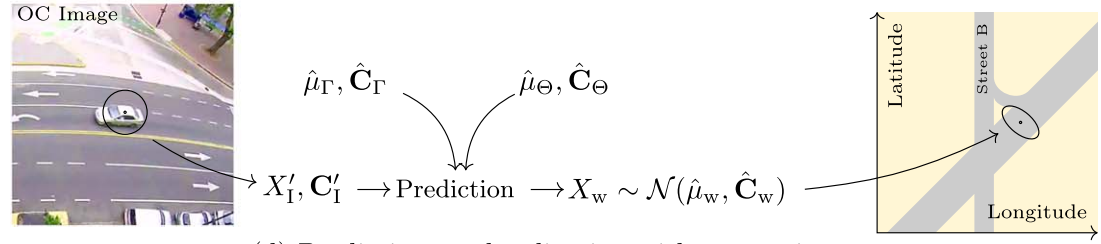
(a) Back-projection.



(b) Intrinsic calibration with chessboard.



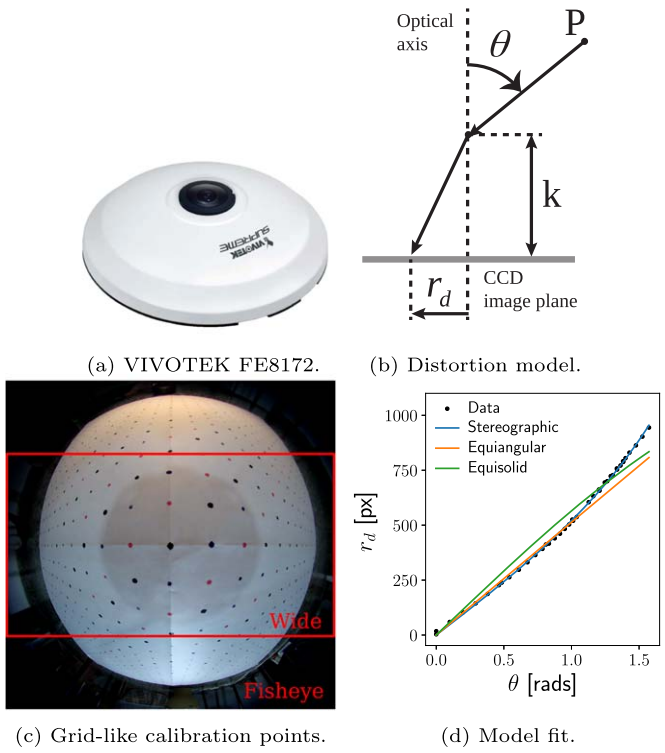
(c) Extrinsic calibration with fiducial points.



(d) Prediction, geolocalization with uncertainty.

Figure 1. Proposed method for geolocation with uncertainties from images obtained with an omnidirectional camera. (a) The back-projection mapping $X_w = \mathcal{F}(X_I, \Gamma, \Theta)$ calculates the position on the world plane X_w, given the position in the image X_I, the intrinsic parameters Γ and extrinsic parameters Θ. (b) Intrinsic calibration. N images of a known chessboard are acquired and the corners in each image are automatically detected. This process generates the dataset $\mathcal{D}_1$ that allows to calculate the mean and variance of the intrinsic parameters, $\hat{\mu}_\Gamma$, $\hat{\mathbf{C}}_\Gamma$ through maximum likelihood estimation. (c) Extrinsic calibration. On the left is showed an image of the plane of interest acquired with the OC. On the right, a satellite image of the same area. A few points that can be detected in both images (red dots) are identified. With this dataset, denoted $\mathcal{D}_2$, (and additional *a priori* information about the OC's pose, provided by the camera installer) the extrinsic parameters associated with the position and orientation of the camera $\hat{\mu}_\Theta$, $\hat{\mathbf{C}}_\Theta$ are estimated by Maximum a Posteriori estimation. (d) Geolocation of objects. An object of interest is detected in the OC image (X_I', with $\mathbf{C}_I'$ uncertainty). This information is combined with the estimates of the calibration parameters to predict the object position in the world reference frame and the associated uncertainty, $\hat{\mu}_w$, $\hat{\mathbf{C}}_w$.

(denoted as the calibration dataset $\mathcal{D}_2$) which allow to estimate mean and variance of pose ($\hat{\mu}_\Theta$, $\hat{\mathbf{C}}_\Theta$). Those calculations associated with the estimation of model parameters $\{\Gamma, \Theta\}$ are performed offline. After this process the objective is to project the coordinates of a new detection in the image (X_I') to the world plane ($\hat{\mu}_w$) and their uncertainties ($\hat{\mathbf{C}}_w$), which is intended to be computed online, see figure 1(d). Formally, the estimation of the probability density function(PDF) $p(X_w|X_I', \mathbf{C}_I', \mathcal{D}_1, \mathcal{D}_2)$ (the probability of position in the map X_w given the measurement in the image X_I', $\mathbf{C}_I'$ and the calibration data) is performed. The linear propagation of



(a) VIVOTEK FE8172. (b) Distortion model.

(c) Grid-like calibration points. (d) Model fit.

Figure 2. (a) Fisheye camera VIVOTEK FE8172 used in this work. (b) Intrinsic model for the fisheye camera. (c) Image obtained with the OC camera of a grid-like set of calibration points. Notice the strong distortions in the image [1]. The red box indicates the region observed by the OC camera in 'Wide' mode (instead of full fisheye mode). (d) Relationship between r_d distance versus θ incidence angle for three intrinsic models [10, 11]: equiangular, equisolid and stereographic, the latter showing the best fit.

uncertainties approach is taken, since it is computationally inexpensive and also accurate, as further demonstrated.

2.2. Camera

In this work a fisheye IP camera VIVOTEK FE8172 (figure 2(a)) is used. It has a field of view (FOV) of $360° \times 183°$ allowing for the observation of a full hemisphere, this camera is compact and easily connected over Ethernet port. A simplified intrinsic model for this OC is shown in figure 2(b). It is observed that an incident ray from the point P with an angle θ with respect to the optical axis, is refracted in the projection center of the camera forming an exit angle different from the one of entrance, and projecting into the image plane (corresponding to the CCD sensor) at a distance r_d from the projection center. The model of fisheye cameras is specified by defining a relationship between r_d and θ, this being in general strongly non linear. This differs from the model of central perspective projection cameras, for which input and output angles are the same (pin-hole model). In a previous work [1], a calibration of the VIVOTEK FE8172 camera was performed showing that the stereographic model is a good characterisation of lens' distortion (figure 2(d)). It is given by the relation $r_d = k \tan(\theta/2)$, where k is the central distance. At full resolution, 1920×1920, radial distortion parameter obtained was $k = 952.16$ px.

In order to improve system's performance in terms of framerate, the resolution is set to 1600×900 and set the camera to 'wide' mode obtaining a FOV of 183° horizontally and about 120° vertically. This doesn't hinder performance because the FOV of full fisheye mode covers an area so extense that a large part of the image corresponds to uninteresting regions while full resolution mode reduces camera's framerate with little visual information added with respect to 1600×900.

2.3. Stereographic projection model

A projection model describes the path of a light ray that originates from a 3D world position (see figure 3) as it passes through the camera lens and hits the CCD chip incrementing image pixel's intensity measure(figure 3). This process is broken down in two steps [1, 15, 16]. First, the 3D position of the light source X_w is rotated and translated to the camera's frame of reference yielding X_c. Information about distance to the camera is eliminated, only direction of arrival of light's ray with respect to the camera matters, yielding X_h a 2D vector. Second, it goes through a function that models lens distortion and projection to the CCD chip, yielding X_I. Given that third dimension is lost, it is not possible in general to back-project X_w from X_I, but with the hypothesis that the light source is at ground level plane the back-projection function can be solved.

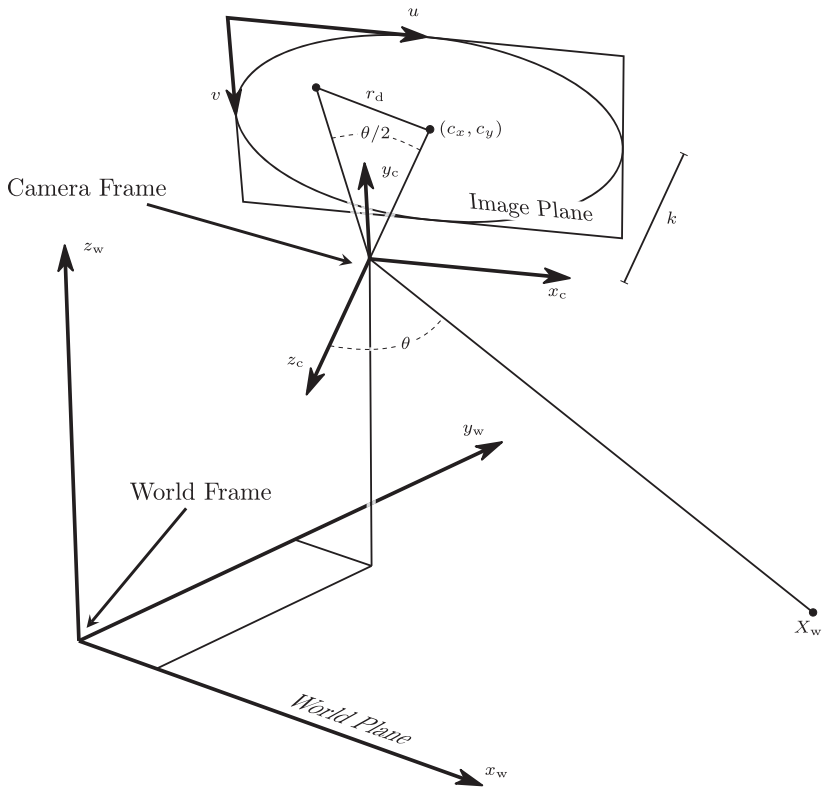


Figure 3. Image formation model when the OC camera observes the X_w point on the world plane ($z_w = 0$). The incident ray forms an angle θ with the optical axis of the camera (which corresponds the axis z of the camera's frame of reference). This ray is refracted according to the stereographic model (see figure 2(b)) which projects on the CCD sensor.

There are several proposed models of optical distortion [10, 17], in [1] it is shown that the stereographic model is a good description for the OC used (figure 2(d)) and is easy to fit having only one parameter describing distortion. Although there are more general distortion models, they have more parameters than needed for this case [18]. In the rest of this section the projection and back-projection function is formulated following OpenCV's projection model.

The first step is to transform the world position to the frame of reference of the camera. The world coordinates of an object $X_w = [x_w, y_w, z_w]^\top$, expressed in the reference frame of the world, are rotated and translated to the frame of reference of the camera, in symbols

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \mathbf{R}(r_x, r_y, r_z) \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}. \tag{1}$$

The rotation matrix **R** is calculated from three parameters that express the orientation, following OpenCV's convention r_x, r_y, r_z are the components of the Rodrigues vector [19, 20]. These six parameters that describe the rotation and translation are the *extrinsic parameters*, the concatenation of the Rodrigues vector and translation vector: $\Theta \equiv [r_x, r_y, r_z, t_x, t_y, t_z]^\top$. Information about distance to the camera is eliminated by projecting to the image plane placed at $z_c = 1$. The now bidimensional coordinates $X_h \equiv [x_h, y_h]^\top$ are

$$X_h = \begin{bmatrix} x_h \\ y_h \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} / z_c. \tag{2}$$

Optical distortion is applied to X_h. The stereographic model only deals with radial distortion, assuming cylindrical symmetry. The radius in the image plane (defined as follows in equation (3)) is used to calculate the polar angle between light ray and the optical axis (equation (4)). The stereographic model applies a nonlinear distortion on it (equation (5)), here it introduces the only non-linear parameter k that scales to pixels units. In symbols all this steps are:

$$r_h = \sqrt{x_h^2 + y_h^2}, \tag{3}$$

$$\theta = \arctan(r_h), \tag{4}$$

$$r_d = k \tan(\theta/2), \tag{5}$$

$$X_{\mathrm{I}} = \frac{r_d}{r_h} X_{\mathrm{h}} + \begin{bmatrix} c_x \\ c_y \end{bmatrix}. \tag{6}$$

Conventionally the origin of pixel coordinates is located at the top left corner of the image, thus the coordinates are appropriately displaced by $[c_x,\ c_y]^\top$ to obtain $X_{\mathrm{I}} = [u,\ v]^\top$. The parameters that depend solely on the camera and describe the optical distortion are *intrinsic parameters* $\Gamma = [c_x,\ c_y,\ k]^\top$.

OpenCV's formulation of the projection model is followed, except for the specific form of the radial distortion function. This makes it fairly easy to later extend the procedure here explained to OpenCV's distortion models.

2.4. Stereographic back-projection

Since the goal is to predict the real world position of objects, the model of image formation from the previous section must be inverted. To map image positions into world positions the model of image formation is followed reversing every step so that a position in the image X_{I} can be transformed to a position in the physical world X_{w}. The back-projection is referred as the function $\mathcal{F}$ that maps $[u,\ v]^\top$ to $[x_{\mathrm{w}},\ y_{\mathrm{w}}]^\top$ using the intrinsic parameters $[c_x,\ c_y,\ k]^\top$ and extrinsic parameters $[r_x,\ r_y,\ r_z,\ t_x,\ t_y,\ t_z]^\top$, in other words $\mathcal{F}(X_{\mathrm{I}},\ \Theta,\ \Gamma) \rightarrow X_{\mathrm{w}}$. The calculations are shown in algorithm 1.

Algorithm 1. Back-projection function

1: **function** $\mathcal{F}$ $[u,\ v]$, $[c_x,\ c_y,\ k]$, $[r_x,\ r_y,\ r_z,\ t_x,\ t_y,\ t_z]$
   Intrinsic correction:
2: $r_{\mathrm{d}} \leftarrow \sqrt{(u - c_x)^2 + (v - c_y)^2}$
3: $\theta \leftarrow 2 \arctan(r_{\mathrm{d}} \,/\, k)$
4: $r_{\mathrm{h}} \leftarrow \tan\theta$
5: $x_{\mathrm{h}} \leftarrow x_{\mathrm{d}}\ r_{\mathrm{h}} \,/\, r_{\mathrm{d}}$
6: $y_{\mathrm{h}} \leftarrow y_{\mathrm{d}}\ r_{\mathrm{h}} \,/\, r_{\mathrm{d}}$
   Extrinsic projection:
7: $\mathbf{R} \leftarrow$ Rodrigues$(r_x,\ r_y,\ r_z)$ ▷ Rodrigues rotation formula
8:
9: $[t_1,\ t_2,\ t_3] \leftarrow \mathbf{R}^\top \cdot [t_x,\ t_y,\ t_z]^\top$ ▷ Rotate translation vector
10:
11: $z_{\mathrm{c}} \leftarrow t_3 \,/\, (R_{13}\ x_{\mathrm{h}} + R_{23}\ y_{\mathrm{h}} + R_{33})$ ▷ Solve collinearity equations
12: $x_{\mathrm{w}} \leftarrow z_{\mathrm{c}}\ (R_{11}\ x_{\mathrm{h}} + R_{21}\ y_{\mathrm{h}} + R_{31}) - t_1$
13: $y_{\mathrm{w}} \leftarrow z_{\mathrm{c}}\ (R_{12}\ x_{\mathrm{h}} + R_{22}\ y_{\mathrm{h}} + R_{32}) - t_2$
14: **return** $x_{\mathrm{w}},\ y_{\mathrm{w}}$
15: **end function**

First, in line 2 the image position is displaced to be expressed in reference to the optical center $[c_x,\ c_y]^\top$. The radius with respect to the optical center in pixels is used to calculate the angle of arrival of the light ray using parameter k in line 3. The tangent of this angle is the radius in homogeneous coordinates (see line 4), which by simple proportionality serves to calculate X_{h} as shown in line 5.

$[x_{\mathrm{h}},\ y_{\mathrm{h}}]^\top$ is a 2D vector on the plane $z_{\mathrm{c}} = 1$ in the frame of reference of the camera. To project to the world frame of reference a 3D position needs to be calculated. The missing 3rd dimensional information (distance of the object to the camera) is made up as in [7] with a reasonable hypothesis in the context of traffic monitoring: the object of interest is on the ground. Thus far $[x_{\mathrm{h}},\ y_{\mathrm{h}}]^\top$ is known and solving for $[x_{\mathrm{w}},\ y_{\mathrm{w}}]^\top$ gives $[x_{\mathrm{h}},\ y_{\mathrm{h}}]^\top$, the parametrization of the pose (R_{ij} is the element i, j of the rotation matrix $\mathbf{R}$) and the hypothesis $z_{\mathrm{w}} = 0$. Working with equations (1) and (2) yields the well known collinearity equations. The solution is shown in lines 9 through 13.

2.5. Previous work on intrinsic calibration and python libraries

The most widely used camera calibration procedure is based in Zhang [21], Bouguet [22] for its ease of use. The calibration procedure usually is: print a chessboard-like pattern and attach it to a planar surface. Take about 10 images of the pattern in different positions with respect to the camera. Detect chessboard corner points in the image automatically. Feed the detected corner points and its corresponding planar coordinates to the algorithm, it will return the distortion parameters of the camera and the rotation-translation of the chessboard in each image.

OpenCV [16] is an open source computer vision library that has been largely adopted as the primary development tool by the community of researchers and developers in computer vision [23]. It includes solutions

for camera calibration [21, 22] and camera pose estimation [24] for a variety of optical distortion models including OCs. Among the solutions it provides, it estimates distortion parameters, extrinsic parameters and perspective transformation. As it will be explained in section 5 this paper's contribution can be added to OpenCV to yield a more complete treatment of uncertainties.

3. Bayesian approach to calibration

In this section an approach to camera calibration from a Bayesian perspective is proposed. The starting point is the general expression of the predictive distribution. From there, the two step calibration process is deduced: intrinsic and extrinsic calibration; in both cases the posterior probability of the parameters given the calibration data is estimated. With the estimated posteriors on the parameters, the predictive distribution is aproximated as a linear propagation of uncertainty.

The predictive distribution of the world position of an object is conditioned on a measurement on the image and on previous data, $p(X_w|X_I', \mathbf{C}_I', \mathcal{D}_1, \mathcal{D}_2)$ [25, 26] (see figure 1(d)). Following the standard procedures of camera calibration, previous data is separated in two, intrinsic calibration data, $\mathcal{D}_1$, and extrinsic calibration data, $\mathcal{D}_2$. The new measurement corresponds to the detection of an object in the image, X_I'. This detection process gives a position in the image but also must report some quantification of the uncertainty of detection. It will be denoted by a covariance matrix $\mathbf{C}_I'$ that is considered to come directly from the detection algorithm.

It follows that the predictive PDF can be expanded as

$$p(X_w|X_I', \mathbf{C}_I', \mathcal{D}_1, \mathcal{D}_2) = \int p(X_w|X_I, \Gamma, \Theta)\ p(X_I|X_I', \mathbf{C}_I')\ p(\Gamma|\mathcal{D}_1)\ p(\Theta|\mathcal{D}_2, \mathcal{D}_1)\ dX_I\ d\Gamma\ d\Theta. \tag{7}$$

There are four terms in the integrand:

- The first term is the Dirac delta function on the back-projection, $p(X_w|X_I, \Gamma, \Theta) = \delta[X_w - \mathcal{F}(X_I, \Gamma, \Theta)]$.
- The second term describes the PDF of the random varible that represents the position in the image, assumed normal given a noisy measurement parameterised by X_I', $\mathbf{C}_I'$. In symbols $p(X_I|X_I', \mathbf{C}_I') = \mathcal{N}(X_I|X_I', \mathbf{C}_I')$.
- The third term $p(\Gamma|\mathcal{D}_1)$ is the posterior probability of the intrinsic parameters given the intrinsic calibration data. It will be addressed in section 3.1. The result is the estimation of the mean and variance of said PDF, assumed normal; that is, $p(\Gamma|\mathcal{D}_1) = \mathcal{N}(\Gamma|\hat{\mu}_\Gamma, \hat{\mathbf{C}}_\Gamma)$.
- The fourth term $p(\Theta|\mathcal{D}_2, \mathcal{D}_1)$ is the posterior probability of the pose of the camera given the extrinsic calibration data, and the intrinsic calibration as well. This is because the extrinsic calibration requires the results of the intrisic calibration as will be explained in section 3.2. Again, the estimated posterior is a normal distribution $p(\Theta|\mathcal{D}_2, \mathcal{D}_1) = \mathcal{N}(\Theta|\hat{\mu}_\Theta, \hat{\mathbf{C}}_\Theta)$.

Replacing with the normal PDFs that will be estimated in the following pages yields

$$p(X_w|X_I', \mathbf{C}_I', \mathcal{D}_1, \mathcal{D}_2) = \int \delta[X_w - \mathcal{F}(X_I, \Gamma, \Theta)]\ \mathcal{N}(X_I|X_I', \mathbf{C}_I')\ \mathcal{N}(\Gamma|\hat{\mu}_\Gamma, \hat{\mathbf{C}}_\Gamma)\ \mathcal{N}(\Theta|\hat{\mu}_\Theta, \hat{\mathbf{C}}_\Theta)\ dX_I\ d\Gamma\ d\Theta. \tag{8}$$

Figure 4 shows a graphical representation of the calculation of the predictive distribution for the hypothetical case in which variables X_I, X_w were one-dimensional.

It is important to note that, even though all the PDFs in the integrand were approximated to normal distributions, the integral is still hard to evaluate due to the non-linearity of $\mathcal{F}$. The integral could be solved using expensive computational strategies, but as explained above the goal is to perform this calculation online. Linearising $\mathcal{F}$ around X_I', $\hat{\mu}_\Gamma$, $\hat{\mu}_\Theta$ reduces the calculation to a simple linear combination of mutually independent normal random vectors [27],

$$p(X_w|X_I', \mathbf{C}_I', \mathcal{D}_1, \mathcal{D}_2) = \mathcal{N}(X_w|\hat{\mu}_w, \hat{\mathbf{C}}_w) \tag{9}$$

$$\text{where} \quad \hat{\mu}_w = \mathcal{F}(X_I', \hat{\mu}_\Gamma, \hat{\mu}_\Theta), \tag{10}$$

$$\text{and} \quad \hat{\mathbf{C}}_w = \mathbf{J}^{\mathcal{F}}_{X_I} \mathbf{C}_I' \mathbf{J}^{\mathcal{F}\top}_{X_I} + \mathbf{J}^{\mathcal{F}}_{\Gamma} \hat{\mathbf{C}}_\Gamma \mathbf{J}^{\mathcal{F}\top}_{\Gamma} + \mathbf{J}^{\mathcal{F}}_{\Theta} \hat{\mathbf{C}}_\Theta \mathbf{J}^{\mathcal{F}\top}_{\Theta}. \tag{11}$$

In words, the approximately normal PDF of the predicted position in the world has a mean that is a direct evaluation of the back-projection function on the means of the detected image position and the parameters; and a covariance that combines the uncertainty on the detection and parameters through the Jacobian $\mathbf{J}$.

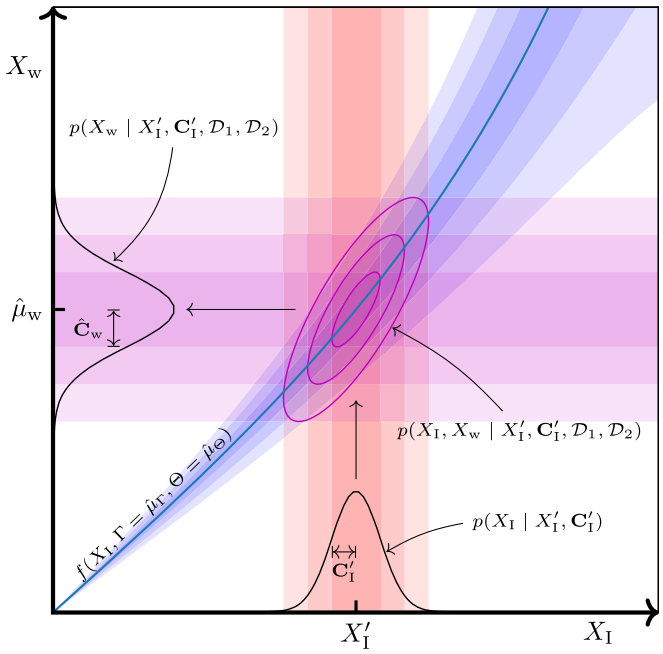


Figure 4. Unidimensional diagram to illustrate how the PDF associated to the detected input $p(X_I|X_I', \mathbf{C}_I')$ is combined with the PDF of the calibration (posterior density of the parameters given the data, $p(\Gamma, \Theta|\mathcal{D}_1, \mathcal{D}_2)$) in blue to yield the PDF in the output $p(X_w|\hat{\mu}_w, \hat{\mathbf{C}}_w, \mathcal{D}_1, \mathcal{D}_2)$.

3.1. Intrinsic calibration

Let $\mathcal{D}_1 = \{(X_w'^{(i,j)}, X_I'^{(i,j)}, \mathbf{C}_I'^{(i,j)})\}_{i,j}$ be the data set for intrinsic calibration where each tuple i, j consist of world coordinates $X_w'^{(i,j)}$ and is corresponding projection in the image $X_I'^{(i,j)}$ (see figure 1(b)), index $i \in [1, N]$ denoting the different images and $j \in [1, M]$ each corner in the chessboard calibration pattern. Detections might not have the same accuracy so $\mathbf{C}_I'^{(i,j)}$ is included in the list of calibration data.

All the parameters for this data set are $\Omega = \{\Gamma, \{\Theta^{(i)}\}_i\}$, where $\{\Theta^{(i)}\}_i$ is the list of extrinsic parameters of every picture. The posterior density of the parameters given the data

$$p(\Omega|\mathcal{D}_1) = \frac{p(\mathcal{D}_1|\Omega)\, p(\Omega)}{p(\mathcal{D}_1)}. \tag{12}$$

Since at this point there is no prior information on Ω the posterior can be equated to the likelihood, which in turn is the product of the probability of each data tuple given the parameters. In symbols

$$p(\Omega|\mathcal{D}_1) \propto \prod_{i,j} p(X_w'^{(i,j)}, X_I'^{(i,j)}, \mathbf{C}_I'^{(i,j)}|\Gamma, \Theta^{(i)}). \tag{13}$$

Every term in the productorial is the probability of measured data $^{(i,j)}$ conditioned on the parameters. Taking $p(X_w'^{(i,j)}, X_I'^{(i,j)}, \mathbf{C}_I'^{(i,j)}|\Gamma, \Theta^{(i)})$ and applying the definition of conditional probability to leave $X_I'^{(i,j)}$, $\mathbf{C}_I'^{(i,j)}$ on the right side of the conditional quickly leads to

$$p(\Omega|\mathcal{D}_1) \propto \prod_{i,j} \mathcal{N}(X_w'^{(i,j)}|\hat{\mu}_w^{(i,j)}, \hat{\mathbf{C}}_w^{(i,j)}) \tag{14}$$

$$\text{where} \quad \hat{\mu}_w^{(i,j)} = \mathcal{F}(X_I'^{(i,j)}, \Gamma, \Theta^{(i)}), \tag{15}$$

$$\text{and} \quad \hat{\mathbf{C}}_w^{(i,j)} = \mathbf{J}_{X_I}^{\mathcal{F}} \mathbf{C}_I'^{(i,j)} \mathbf{J}_{X_I}^{\mathcal{F}\top}. \tag{16}$$

Which looks like a simplified version of equation (9) because there is no PDF on Γ, $\Theta^{(i)}$, but it arised under a similar procedure. The probability of $\Omega|\mathcal{D}_1$ in equation (13) can be evaluated numerically for some value of Ω, it requires the calibration data and to compute $\mathcal{F}$ and its derivative with respect to X_I as shown in equation (14). Methods like Metropolis-Hastings [28] can estimate the mean and variance of $\Omega|\mathcal{D}_1$. But recall that out of $\Omega = \{\Gamma, \{\Theta^{(i)}\}_i\}$ the camera positions with respect to the calibration pattern are of no use later on, the objective of the intrinsic calibration is Γ only because the optical distortion is a constant intrinsic to the camera. The estimation of $\{\Theta^{(i)}\}_i$ is ancillary. Marginalizing with respect to $\{\Theta^{(i)}\}_i$ is trivial under the reasonable assumption that $\Omega|\mathcal{D}_1$ is approximately normal and Γ and $\{\Theta^{(i)}\}_i$ are independent. Intrinsic calibration results in the estimation of

$$p(\Gamma|\mathcal{D}_1) = \mathcal{N}(\Gamma|\hat{\mu}_\Gamma, \hat{\mathbf{C}}_\Gamma). \tag{17}$$

where $\hat{\mu}_\Gamma$, $\hat{\mathbf{C}}_\Gamma$ are the components of the mean and variance of $\Omega|\mathcal{D}_1$ (obtained by Metropolis-Hastings) that correspond to Γ.

3.2. Extrinsic calibration

After intrinsic calibration in controlled conditions, where the PDF of Γ was estimated, the camera is set up in some urban location pointing to some zone of interest (figure 1(c)). Calibration data is now a set denoted as $\mathcal{D}_2$ of M points on the real world and its associated image coordinates $\{(X_{\mathrm{w}}'^{(j)}, X_{\mathrm{I}}'^{(j)}, \mathbf{C}_{\mathrm{I}}'^{(j)})\}_j$ with $j \in [1, M]$. The extrinsic calibration is the procedure to estimate mean and variance of the camera pose Θ.

By the law of total probability and assuming independence between Θ and $\mathcal{D}_1$ and between Γ and $\mathcal{D}_2$ the posterior on Θ is

$$p(\Theta|\mathcal{D}_2, \mathcal{D}_1) = \int_\Gamma p(\Theta|\mathcal{D}_2, \Gamma)\, p(\Gamma|\mathcal{D}_1)\mathrm{d}\Gamma \tag{18}$$

where $p(\Gamma|\mathcal{D}_1) = \mathcal{N}(\Gamma|\hat{\mu}_\Gamma, \hat{\mathbf{C}}_\Gamma)$ (equation (17)). By Bayes' rule $p(\Theta|\mathcal{D}_2, \Gamma) \propto p(\mathcal{D}_2|\Theta, \Gamma)p(\Theta|\Gamma)$ but as Θ and Γ are independent $p(\Theta|\Gamma) = p(\Theta)$. The posterior distribution of Θ is

$$p(\Theta|\mathcal{D}_2, \mathcal{D}_1) \propto p(\Theta) \int_\Gamma p(\mathcal{D}_2|\Theta, \Gamma)\, p(\Gamma|\mathcal{D}_1)\mathrm{d}\Gamma. \tag{19}$$

Notice that a non flat prior on the camera pose is allowed $p(\Theta)$, as this is a physical magnitude for which there might be some information after installation, unlike the camera intrinsic parameters that depend on the model, which might be quite obscure to elucidate.

As in equation (14) the likelihood $p(\mathcal{D}_2|\Theta, \Gamma)$ can be calculated as the product of the likelihood of each data tuple resulting in

$$\begin{aligned} p(\Theta|\mathcal{D}_2, \mathcal{D}_1) &\propto p(\Theta)\prod_j \mathcal{N}(X_{\mathrm{w}}'^{(j)}|\hat{\mu}_{\mathrm{w}}^{(j)}, \hat{\mathbf{C}}_{\mathrm{w}}^{(j)}), \\ \text{where} \quad \hat{\mu}_{\mathrm{w}}^{(j)} &= \mathcal{F}(X_{\mathrm{I}}'^{(j)}, \Theta, \hat{\mu}_\Gamma), \\ \text{and} \quad \hat{\mathbf{C}}_{\mathrm{w}}^{(j)} &= \mathbf{J}_{X_{\mathrm{I}}}^{\mathcal{F}} \mathbf{C}_{\mathrm{I}}'^{(j)} \mathbf{J}_{X_{\mathrm{I}}}^{\mathcal{F}\,T} + \mathbf{J}_\Gamma^{\mathcal{F}} \hat{\mathbf{C}}_\Gamma \mathbf{J}_\Gamma^{\mathcal{F}T}. \end{aligned} \tag{20}$$

Now the derivative of $\mathcal{F}$ respect to Γ is also required and computing $\mathbf{J}_\Gamma^{\mathcal{F}}$. Numerical methods, again, can estimate the mean $\hat{\mu}_\Gamma$ and variance $\hat{\mathbf{C}}_\Gamma$ of $\Theta|\mathcal{D}_2, \mathcal{D}_1$ from equation (20), such that

$$p(\Theta|\mathcal{D}_2, \mathcal{D}_1) = \mathcal{N}(\Theta|\hat{\mu}_\Gamma, \hat{\mathbf{C}}_\Gamma). \tag{21}$$

3.3. Summing up calibration and prediction

In brief, the procedure is as follows. Taking images of a calibration pattern in the laboratory as in figure 1(b) produces the intrinsic calibration data $\mathcal{D}_1$ and the result of calibration is to parameterize the posterior PDF of the intrinsic parameters $p(\Gamma|\mathcal{D}_1)$ with a mean $\hat{\mu}_\Gamma$ and variance $\hat{\mathbf{C}}_\Gamma$. This is done computing the posterior via equation (14) (refer to section 4 for more details) and standard methods of numerical integration like Metropolis-Hastings. When the camera is finally installed in its final position the extrinsic calibration points can be extracted $\mathcal{D}_2$ (figure 1(c)) that are used to estimate the mean and variance $\hat{\mu}_\Theta$, $\hat{\mathbf{C}}_\Theta$ of the posterior PDF $p(\Theta|\mathcal{D}_1, \mathcal{D}_2)$ (computed as shown in equation (20)). This completes the calibration. With a new detection of a vehicle X_{I}', $\mathbf{C}_{\mathrm{I}}'$ the predicted PDF in the world frame of reference is calculated with equation (9) as ilustrated in figure 1(d). This prediction can be performed online since the computational cost is negligible.

4. Results

In this section it is shown that the linear approximation for uncertainty propagation delivers significant accuracy when compared to a more proper but computationally intensive nonlinear Monte Carlo estimation. The two-step calibration and prediction are applied to simulated data cases: first generate data of realistic chessboard pictures for the intrinsic calibration and a total of six final camera installation positions and orientations for the extrinsic calibration. Then use real chessboard data obtained in controlled conditions to estimate the intrinsic parameters; the camera was installed at a testing site and calibration points were manually obtained from images to estimate the extrinsic parameters. Finally, the uncertainty of the predicted world positions for a vehicle detected within the video sequence is shown.

As a pattern for intrinsic calibration a 37 cm long chessboard with 9×6 interior corners was used. $N = 33$ pictures were taken and then applied OpenCV corner detector as shown in figure 5(a). These pictures were taken to cover the field of view, as suggested by Fraser [18], the detected corners are shown in figure 5(b). OpenCV's

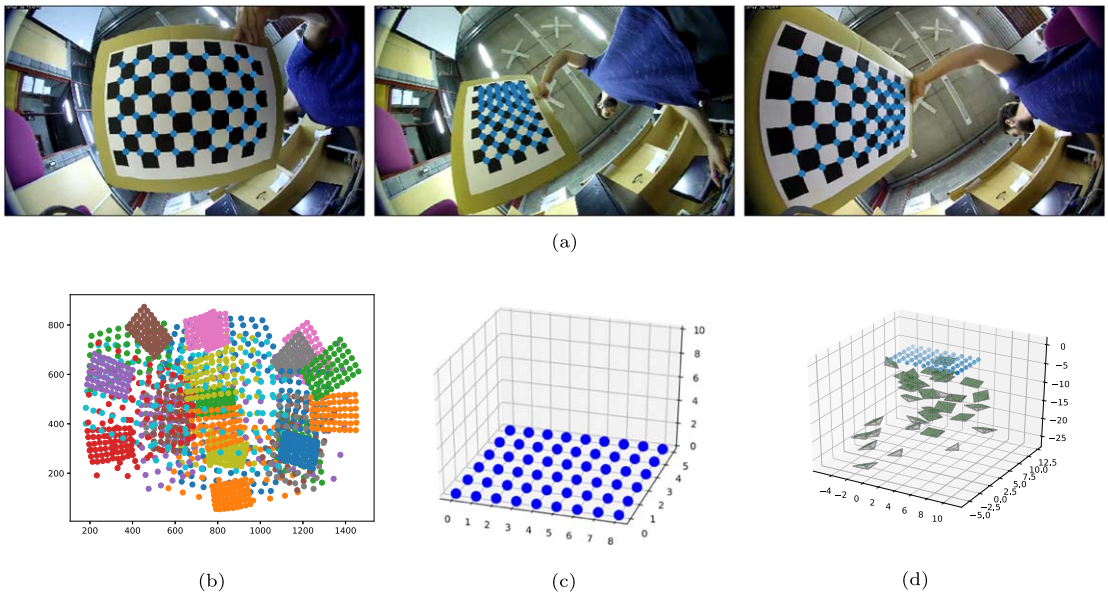


Figure 5. (a) Three out of the 33 images of the calibration pattern, a 9 × 6 chessboard. (b) An overlap of all detected corners is shown. (c) The 3D coordinates of the corners in the world frame of reference. (d) The 33 poses of the camera that OpenCV calibrates along with the intrinsic parameters of the camera (with a different model than the author's).

calibrateCamera function takes the detected corners and their corresponding positions in 3D (figure 5(c)) and returns 33 camera poses shown in figure 5(d). The detected chessboard corners and the estimated camera poses are used either as initial conditions for the sampling algorithms or as ground truth to generate synthetic data, as explained in the following subsections.

Both the acquisition of video/images and off-line data processing were carried out in a desktop computer running under Linux operating system using Python [29] scripts with the aid of the libraries NumPy [30], SciPy [31], Matplotlib [32], OpenCV [16] and the Spyder IDE [33]. As OpenCV implements the calibration algorithms of Bouguet [22] it was adopted as starting point for calculations, and for general image manipulation. The library PyMC3 [34] was used for Monte Carlo simulations.

4.1. Comparing linear approximation with Monte Carlo

In this section the first order approximation of the propagation function against a Monte Carlo (MC) evaluation of the nonlinear mapping are compared. The heart of the stereographic model is a highly nonlinear radial distortion function because it must conform to the severe optical distortion that characterizes the OC. And more generally, any image formation model includes a perspective projection that is strongly nonlinear in the camera pose so it is not at all evident that the linear approximation would hold in practice.

In a similar fashion as done by Criminisi *et al* [35] a population of tuples $(X_\mathrm{I},\ \Gamma,\ \Theta)$ that follow normal distributions are generated. Each variable then $X_\mathrm{I} \sim \mathcal{N}(\mu_\mathrm{I},\ \mathbf{C}_\mathrm{I})$, $\Gamma \sim \mathcal{N}(\mu_\Gamma,\ \mathbf{C}_\Gamma)$ and $\Theta \sim \mathcal{N}(\mu_\Theta,\ \mathbf{C}_\Theta)$. The resulting set of points will be compared with the parameterized normal PDF obtained by linear propagation of the normal distributions where the points were drawn from. If the linear approximation is valid then the mean and covariance of the MC particles will be close to the propagated mean and covariance.

The data gathered for the intrinsic calibration is a useful source of realistic image coordinates. Instead of arbitrarily defining a number of poses that imitate chessboard calibration data it was preferred to borrow some camera poses associated to a real data set as it covers a reasonable range of positions. Taking the 33 OpenCV's estimated camera poses as the set of $\{\Theta^{(i)}\}_i$; along with the 3D world positions of the chessboard corners and intrinsic values Γ that arise from author's previous work with the camera and the understanding of the stereographic model. Define $\Gamma = [800,\ 465,\ 800]^\top$. The center of optical distortion is at the center of the image (hence c_x and c_y are taken from the image size) and the optical distortion parameter k is half the width of the image because it is interpreted as the radius when the incoming light ray is perpendicular to the optical axis.

To generate the population of samples for MC the image corners are first fabricated. Using equations (1)–(6) to project the 54 chessboard world coordinates $X_\mathrm{w}^{(i,j)}$ to image coordinates for each pose. Ending up with 33 sets of 54 pixel coordinates $\mu_\mathrm{I}^{(i,j)}$. The detected image coordinates are determined with 1 pixel standard deviation (therefore $\mathbf{C}_\mathrm{I} = \mathbf{I}_2$). It is considered reasonable (and this is later confirmed empirically) that the intrinsic and extrinsic parameters have been determined to about three significant digits, that is the standard deviation is 10^{-3} of the parameter value, hence defining $\mathbf{C}_\Gamma,\ \mathbf{C}_\Theta$.

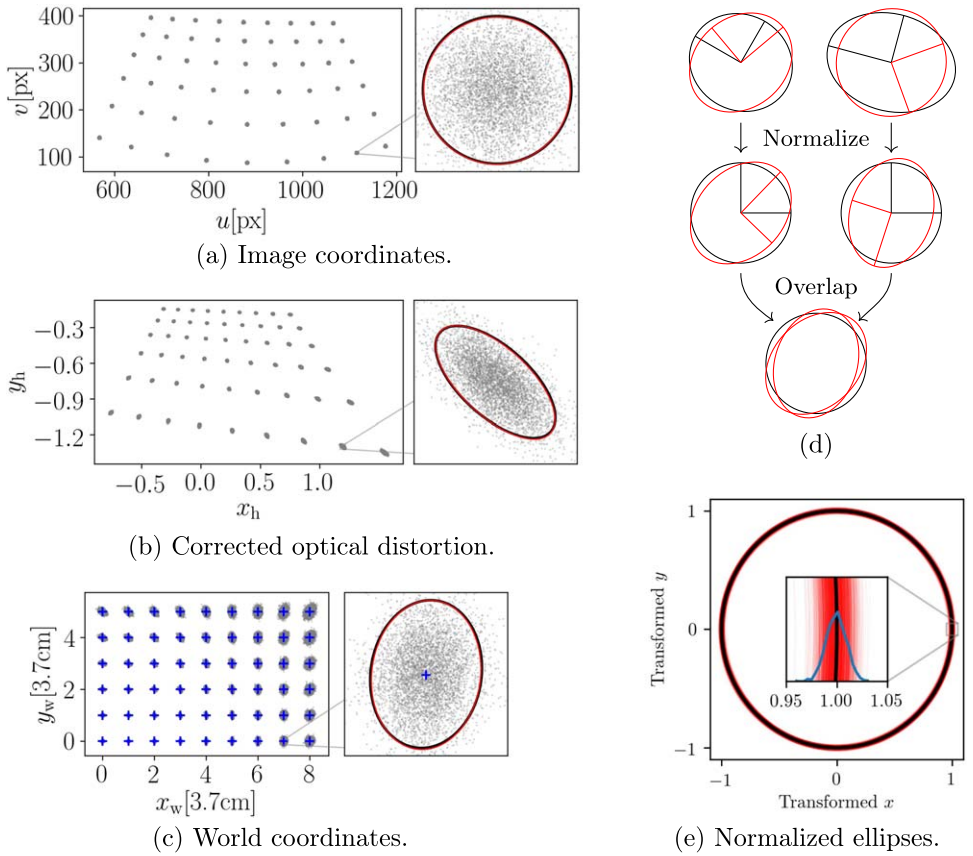


Figure 6. (a)–(c) The comparison of MC and linear error propagation, on the inset on the right the 5000 dots are the MC population, the black line is their covariance ellipse of 90% confidence to be compared with the covariance ellipse obtained via linear propagation of uncertainty. On the left, the MC populations of the 54 calibration points. (a) Notice the curvature due to optical distortion. (b) The optical distortion has been corrected. (c) The propagation to the world plane, the covariance ellipses of MC and the predicted ones are virtually indistinguishable. (d) Normalizing so that the 54 × 33 MC covariances become the identity matrix puts all linearly predicted covariances on equal footing for comparison. (e) Normalised covariance ellipses form a thin halo around the unit circle, showing that for all cases the discrepancy between MC and linear propagation is very small.

Next step is to generate the set of $N_{MC} = 5000$ Monte Carlo samples, drawing image detection positions, intrinsic and extrinsic parameters according to the means and covariances just mentioned. The N_{MC} tuples are then fed to the back-projection function. Not only the final outcome in world coordinates are examined but also the intermediate coordinates X_h, similar to a central perspective camera without distortion.

In figure 6(a) the image coordinates sampled from a Gaussian distribution, 5000 samples for each corner detected, the zoomed inset on the right shows the comparison between the samples for a detected corner, the covariance ellipses of 90% probability, both estimated from the Monte Carlo samples (in black) and theoretical first order analysis (in red). In figure 6(b) the same samples were corrected for intrinsic distortion (lines 2–5 of algorithm 1) and it can be seen that the ellipses have become elongated in the radial direction, also, the difference between MC and the linear approximation has been accentuated due to the linearising error in the radial direction. In figure 6(c) the perspective projection is performed (lines 11–13). The uncertainty in the six pose parameters adds more uncertainty but the estimations of covariances from Monte Carlo and linear propagation are indistinguishable.

There are 33 × 54 calibration points, each of them was used to produce a pair of prediction PDFs. One by linearly propagating uncertainties and a second one by fitting a Gaussian distribution to the MC back-projected samples. To visually assess the similarity between the two PDFs for all 1782 calibration points, the covariance ellipse associated to the PDF obtained by first order propagation are transformed to a new base where its corresponding numerical MC counterpart becomes the unitary normal distribution with zero mean. Ilustrated in 6(d), then subtract the center of the ellipse from linear propagation and apply a change of base such that this ellipse becomes a unitary circle. Plotting the transformed first order covariance ellipses in figure 6(e) in red lines and as reference the MC covariance circle in red showing all the red ellipses superimposed result in a blue halo around the reference circle.

Table 1. Comparison of true intrinsic values and estimation from samples drawn from the posterior probability distribution. Each sample is a 201 dimensional vector, only the three components of the intrinsic parameters are utilized.

	True Value	Samples Mean ($\hat{\mu}_\Gamma$)	Samples SD
c_x	800	800.001	0.05
c_y	452	451.999	0.06
k	800	799.999	0.13

4.2. Intrinsic calibration with synthetic data

To test the intrinsic calibration the posterior probability distribution of the three intrinsic parameters of the camera is estimated. first and second moment. Figure 5 shows the 33 camera poses with respect to the checkboard points and all corner detections in one single image.

The 3 intrinsic parameters and the 33 $\times$ 6 extrinsic parameters (6 per image) form a multivariate random vector of 201 components. The probability of the vector is evaluated as shown in equation (13). There were drawed 442 chains of 50 samples with Differential Evolution Metropolis (DEM) [34, 36]. The starting values for the chains were defined ad hoc to minimize the burn-in period. The histograms of the samples from this section and the ones to follow were unimodal and bell shaped.

$$\hat{\mathbf{C}}_\Gamma = \begin{bmatrix} 0.00247 & 0.000\,20 & -0.00030 \\ 0.00020 & 0.003\,79 & 0.00123 \\ -0.00030 & 0.001\,23 & 0.01793 \end{bmatrix} \tag{22}$$

Table 1 compares the true values of the parameters with the estimations from the samples, the disparity is in the sixth significant digit, it is due to the statistical fluctuation of the artificially added detection noise (standard deviation of 1 pixel). This shows that the expectation of the posterior probability is a good estimator of the true parameter values. The variance of the samples comes from the width of the dispersion of said noise, the more uncertain the detection in the image the less informative the posterior.

4.3. Intrinsic calibration with real data

To estimate the intrinsic parameters of the OC it is followed the same procedure as above with experimental data, the detected corners in the 33 chessboard images (not the ones artificially generated assuming known distortion parameters and camera poses).

Before sampling a standard non linear optimization function to get better seed values is used. The extrinsic parameters given by OpenCV's calibrateCamera and the intrinsic parameters used as ground truth in the previous section result in a back-projection of the corners that show significant discrepancies to the true chessboard positions. To provide DEM with better initial values for sampling, a standard non linear optimization routine from Scipy [31] that brings the back-projections closer to their target is used. Minimizing the error function associated with the posterior on the parameters (equation (13)). The back-projection with the values from synthetic chessboard (the initial guess) are shown in the left panel of figure 7 and in the right panel the back-projection with the optimized parameters. OpenCV estimates the parameters minimizing the projection (in image) error, that's why they are bad estimates for back-projection. The clear improvement in fitting drastically cuts down the burn-in period when sampling.

Assuming a 1 pixel error in corner detection, the mean and variance of 500 chains of 2000 samples are

$$\begin{aligned} \hat{\mu}_\Gamma &= [816.45,\ 472.64,\ 795.19], \\ \hat{\mathbf{C}}_\Gamma &= \begin{bmatrix} 0.628 & -0.051 & 0.016 \\ -0.051 & 0.640 & 0.367 \\ 0.016 & 0.367 & 3.746 \end{bmatrix}. \end{aligned} \tag{23}$$

Notice that the variance is much greater than the one estimated for the simulated intrinsic calibration because it now accounts for the error in the model.

4.4. Extrinsic calibration and predictions with synthetic data

Following section 4.2 where the calibration for a simulated camera was solved, placement of the same camera is simulated in an urban environment to perform the extrinsic calibration and test the algorithm with plausible ad hoc camera poses.

The main interest is to test the calibration in a set of realistic conditions in the context of monitoring of vehicles and pedestrian in urban scenes. Camera height above ground {7.5 m, 15 m}; its optical axis forming and angle with respect to vertical: {0°, 30°, 60°}; and 20 calibration points, in total encompassing 6 situations.

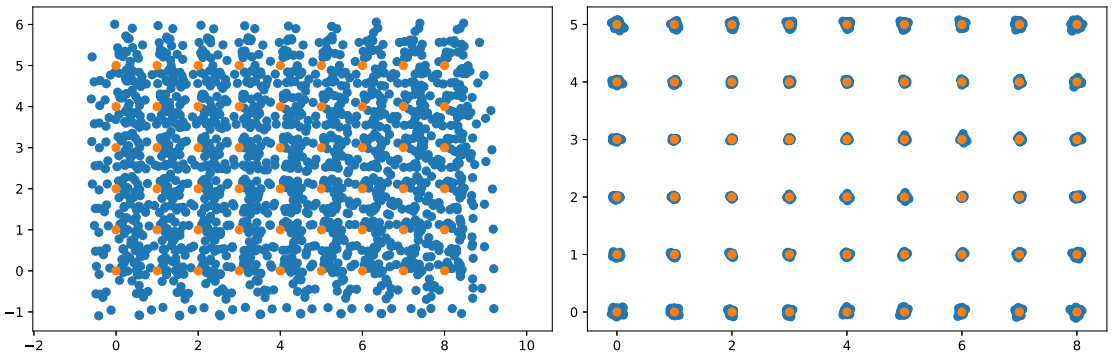


Figure 7. Initial values of distortion parameters and the 33 poses are obtained by simple non linear optimization of the back-projection error. Red dots are calibration pattern in the world frame of reference, in blue the back-projection of the detected chessboard corners. On the right, after the non-linear optimization, the back-projection has improved significantly with respect to before the optimization (left).

Points on the z $=$ 0 plane are in region of 50 m radius such that are evenly distributed in the observed image, half of them will be used to calibrate the pose of the camera and the other half for testing. That is, they are all projected to image coordinates, ten world-image pairs will be used to estimate the pose. Then prediction of the world coordinates of the ten unused image detections to be compared to their corresponding world coordinates is performed.

With the calibration points and the estimation of intrinsic parameters previously obtained, the calibration procedure to sample the six dimensional pose space is applied. In every case 30 chains of 1000 MC samples are drawn. The means and variances of the six sets of samples are used to back-project the synthetic image detections and their uncertainty to the corresponding georeferenced positions. Figure 8 shows the projected ellipses on the world reference frame, the size of the ellipses and the error with respect to the true position has been magnified by a factor of 10 to make the disparity visible in figure 8. The projected uncertainty is smaller for positions closer to the camera and also the ellipses are less elongated because those regions hold a better view factor, as the projected point gets further away from the camera the uncertainty grows, specially in the radial direction due to the perspective effect. To visualize all the projections errors it is linearly transformed each projection error to the space where the projected covariance becomes the identity matrix as in section 4.1. In figure 8(c) all the calibration points have been transformed in this way, for reference the circle of 90% probability is drawn.

Table 2 reports the root mean squared deviation between the real world positions and the back-projections of the calibration points and prediction test points. The prediction error on testing points is always greater than the error on calibration points and both are in the order of 10^{-1}m.

4.5. Extrinsic calibration and prediction with real data

Following section 4.3 calibration points are used to estimate the camera pose in a real world situation and geolocate the trajectory of a vehicle.

The camera was placed 15.7 $\pm$ 0.2 m above the ground, this is the *a priori* information used for calibration. Manually defining $M = 19$ calibration points that consist of corresponding pairs of image and latitude-longitude coordinates. The terrain where the experiment took place is even and horizontal, so that the assumption $z_w = 0$ holds. Also this facilitates the conversion of the world coordinates to and from different representations (degrees of latitude-longitude, pixels inside a satellite image, meters) using a simple scaling factor. The point on the floor directly below the *a priori* position of the camera was defined as the coordinate origin (0 m, 0 m) of the ground plane. Detections in the image were assigned 1 pixel of standard deviation. Figure 9 shows the image calibration points and its corresponding latitude-longitude points. The trajectory of a car as it traverses the field of view of the camera is shown in figure 9(a) and this detections have 1 pixel of standard deviation, they correspond to a feature of the car close to the ground.

Using the estimation of intrinsic parameters from section 4.3 and the *a priori* information, Differential Evolution Metropolis returned 60 chains of 9500 samples of 6-D rotation-traslation vectors. The mean and variance of the samples are

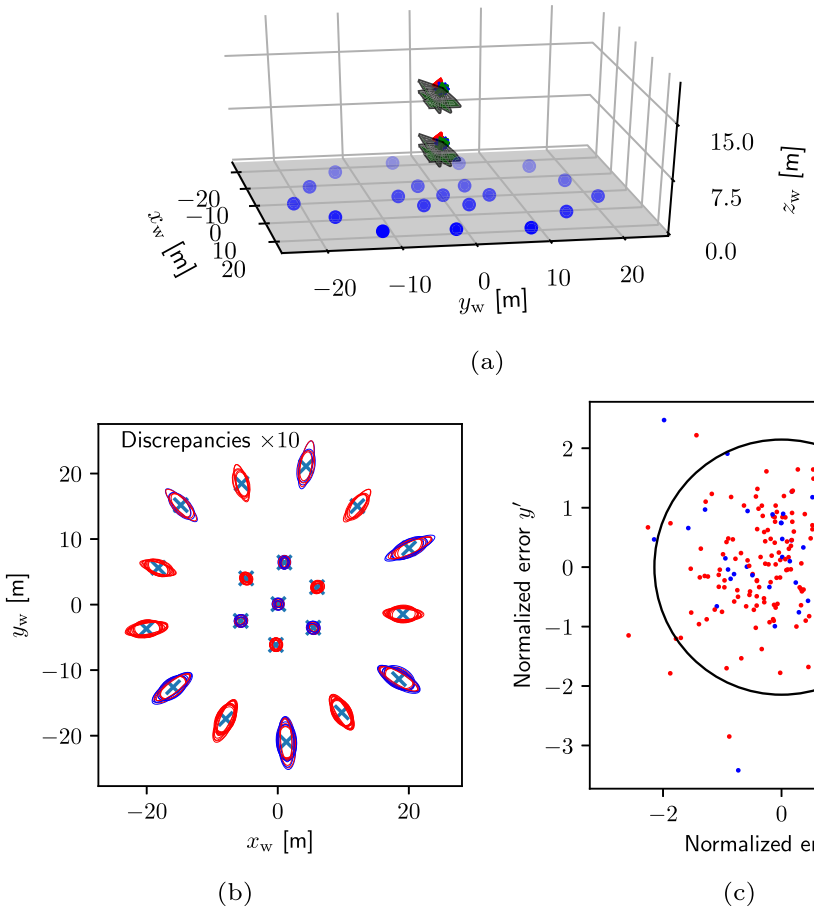


Figure 8. (a) The synthetic calibration points and six cases of camera poses. (b) The back-projected points in the world plane, the confidence ellipses (six per calibration point) have been magnified ×10 to make them visible, in red if the point was used for extrinsic calibration and in blue if it was used for prediction. To compare all back-projections, in (c) each calibration point is transformed to the space where the propagated confidence ellipse is the 90% confidence circle.

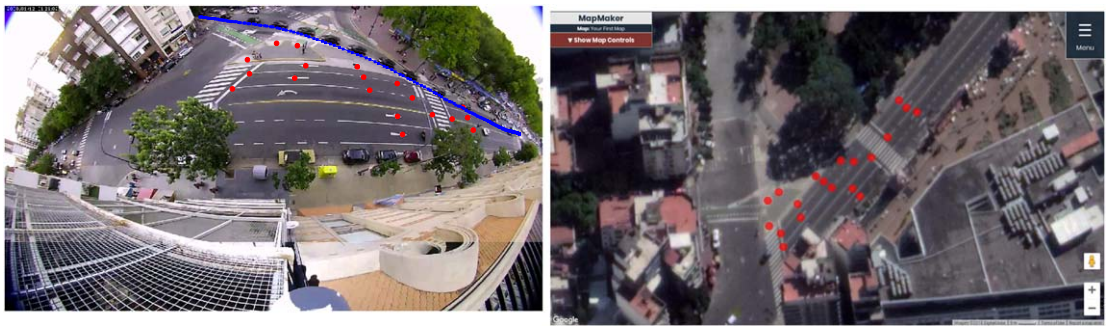
Table 2. For each combination of orientation angle and height there is a number of visible calibration and prediction test points (It's less than ten when the fiduciary points fall out the field of view of the camera). The root mean squared deviation is reported for both training and test points.

Only calibrating with 10 points					
ang[°]	h[m]	N_{Train}	N_{Test}	Train RMSD[m]	Test RMSD[m]
0	7.5	10	10	0.076	0.136
0	15	10	10	0.083	0.103
30	7.5	8	8	0.096	0.078
30	15	10	10	0.070	0.113
60	7.5	7	6	0.126	0.131
60	15	8	8	0.057	0.119

$$\hat{\mu}_\Theta = [2.7441,\ 1.1450,\ -0.1767,\ -3.2703,\ -2.2972,\ 17.6410],$$

$$\hat{\mathbf{C}}_\Theta = \begin{bmatrix} 4.1 & 0.1 & 1.0 & -14.4 & -56.9 & 52.2 \\ 0.1 & 2.7 & -0.6 & 47.7 & -1.0 & 26.3 \\ 1.0 & -0.6 & 1.8 & -42.5 & -14.3 & -3.5 \\ -14.4 & 47.7 & -42.5 & 1554.9 & 183.6 & 532.1 \\ -56.9 & -1.0 & -14.3 & 183.6 & 1027.9 & -921.9 \\ 52.2 & 26.3 & -3.5 & 532.1 & -921.9 & 1306.9 \end{bmatrix} \times 10^{-5}. \quad (24)$$

Where the rotation component of $\hat{\mu}_\Theta$ (first three elements) is in radians and the translation component is in meters, the standard deviation of the former being ∼0.3° and ∼0.1 m of the latter. $\hat{\mu}_\Gamma$ is the rotation-translation parameters of the world reference frame from the point of view of the camera, the position of the camera in the



(a) OC image. (b) Satellite image.

Figure 9. Extrinsic calibration with real data. Red dots are calibration points, in blue the trace of a car.

world reference frame is calculated by $-\mathbf{R}(\hat{r}_x, \hat{r}_y, \hat{r}_z)^\top \cdot [\hat{t}_x, \hat{t}_y, \hat{t}_y]^\top$. It yields a height of 17.0 m with a std of 0.14 m, in the order of the actual height above ground.

The predicted car trace in world coordinates has an uncertainty that combines the estimated uncertainties of the intrinsic parameters, extrinsic parameters and image detection. In figure 10 the blue ellipses are the 90% probability regions, drawn every few back-projected detections of the car (red dots). The effect that the perspective projection has on the propagation of uncertainty has two components, one being the distance to the camera that magnifies the uncertainty, the reverse of an inverse-square law. In the inset of figure 10 it is empirically shown that the area of the 90% confidence ellipse is proportional to the square of the distance to the camera. The second component is the view factor of the back-projected point with respect to the camera that stretches the ellipse in a direction radial to the closest point to the camera. In this case the optical distortion and view factor tend to elongate the confidence ellipses in approximately the same direction, that is why the ellipses are so stretched. The smallest area of the 90% probability region is 3.15 m^2, when the car is closest to the camera, and increases with the square of the distance as shown in the lower inset of figure 10.

5. Conclusion and discussion

Wide-field vision systems (based on synthetic compound eyes or omnidirectional cameras) are currently being incorporated to engineering applications related to terrestrial and aerial mobile robotics. Despite the advantages mentioned in section 1, OCs are not widely used in video surveillance applications in urban environments; where the traditional solution is still the installation of many cameras (fixed or PTZ type) each with reduced visual field.

The main limitation of using the OC's in this type of applications is the strong distortions introduced in the image. Beyond this limitation (resolved by correcting the distortions computationally [4, 14]) the use of fisheye cameras has the advantage of observing a complete hemisphere of the scene at all times. This is very useful in transport-related applications in which the movement of vehicles or pedestrians in wide regions must be analyzed (for example in convoluted road intersections, see figure 10) [2]. In addition, the evaluation of geo localization uncertainties are needed for estimation algorithms based on Bayesian filters (Kalman filter, particle filter, etc) used for motion analysis and prediction, tracking and decision making on vehicular traffic violations. For these reasons, this work studies the use of a monocular omnidirectional camera to geolocate objects solving the calibration and prediction problems from a Bayesian perspective.

5.1. Bayesian approach to camera calibration

Camera calibration is a critical part of any photogrammetric system. The Bayesian approach is well suited to formulate both calibration and prediction problem in explicit probabilistic terms, and to incorporate *a priori* information about the camera and/or its installation pose.

Sundareswara and Schrater [12] demonstrated that the Bayesian prediction is less susceptible to statistical fluctuations than maximum likelihood estimation. Their work follows similar ideas to the present one but with critical differences. Sundareswara and Schrater [12] use a pin hole model (not dealing with severe distortions), they calibrate in one step (instead of two) with several views of the object of interest (here a single monocular view is assumed), estimating the posterior probability of the parameters and the reconstruction at the same time (here intrinsic calibration must be done prior to the installation of the camera). The result of this is a population of samples of the parameters that is later averaged, for marginalization, during 3D reconstruction (here

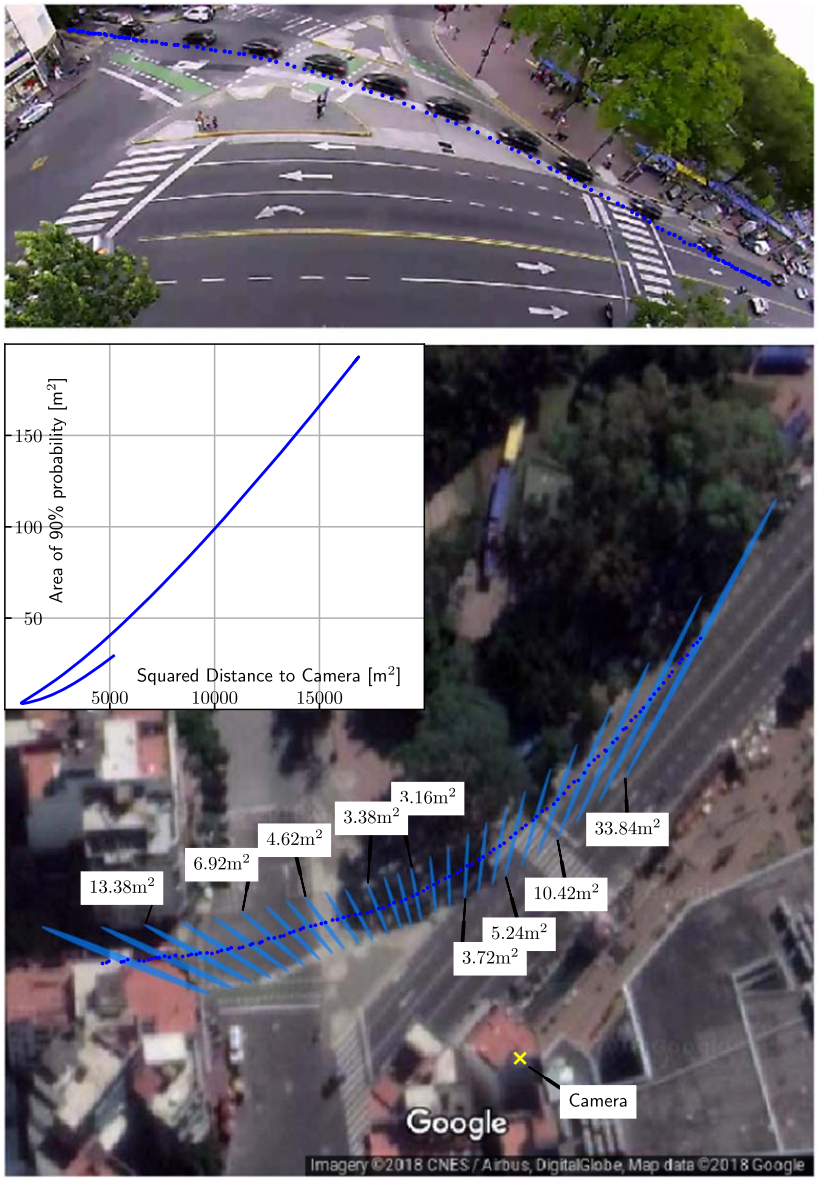


Figure 10. (Top) The detected car positions, they were back-projected to the world reference frame (main panel), the propagated uncertainty is drawn in blue as 90% confidence covariance ellipses. The smallest ellipse has an area of 3.2 m^2 and happens when the car is closest to the camera. The inset shows that the area of the ellipses are proportional to the square of the distance to the camera.

calibration means to estimate a mean and a covariance; prediction as linear propagation automatically incorporates marginalization).

The methodology proposed in this work is designed for vehicle motion analysis applications in urban environments and consists of two calibration steps and a computationally efficient method for position prediction. The first step is very similar to standard camera calibration techniques and estimates the posterior PDF of the optical distortion parameters within the laboratory.

The second step is specific to the proposed back-projection function and estimates the posterior of the extrinsic parameters. In this case, the Bayesian approach allows for the introduction of *a priori* information about the camera pose provided by the installer: in the case of very few calibration points the prior should decrease the uncertainty of calibration, and also eliminate the ambiguity of multiple solutions that are typical of symmetric calibration rigs [37].

The posterior distributions of the parameters given the data are estimated with Differential Evolution Metropolis. The population of samples obtained showed that the distribution was uni-modal and bell shaped. This observation opens the possibility to replace this method with a non linear optimization to get the most probable value of the parameter and Laplace approximation to estimate the variance, which has a lower computational cost [28].

In the prediction step, propagation of uncertainty assumes first order approximation and it is shown that the assumption holds even if the camera has a severe optical distortion by Monte Carlo simulations in figure 6. The Jacobians for the propagation were calculated using the chain rule. Also the propagation of uncertainty can be improved by accounting for higher moments of the PDF and higher orders of Taylor expansion if the higher order derivatives of the back projection function were available. Mekid and Vaja [38] derive the expression of the propagation of up to fourth moment (including skewness and kurtosis) through a Taylor series truncated at third order for the case of 2D random vectors. This could be implemented as methods of automatic differentiation became available for high level programming languages [39].

This work assumes perfect measurement of world coordinates X_w and that the fiducial points are perfectly on the ground plane, meaning that $z_w = 0$ exactly. This are the only variables not treated as a random, they are treated as exact measurements. But uncertainty on z_w could reasonably arise from two factors: the fiducial points being selected on objects slightly out of the ground plane (on a road hump or bump or on the sidewalk) and deviations of the observed surface from the assumed plane model. Errors in both variables will increase the uncertainty of the estimated extrinsic parameters Θ; and following from equation (9) this will increase the uncertainty in geolocation. Expanding the model to include uncertainty of X_w and z_w would complete the Bayesian formulation. This could done easily by the theorem of marginalisation of normal PDFs [28]. Also it is important to note that the retro projection model can be expanded for models of the ground other than the horizontal plane, including curved surfaces such as quadrics.

5.2. Results of the method in simulations and real data

The simulated calibrations showed that the intrinsic parameters were estimated with high accuracy and that the extrinsic calibration predicts world positions that agree perfectly with the propagated confidence ellipse (figure 8). The ellipses are smaller if projected closer to the camera and if the view factor is small they become stretched in the radial direction, both effects tend to be more pronounced as the point projected on the horizontal plane is further away from the camera.

Calibrating with real data, the intrinsic parameters are estimated with a standard deviation of around one thousand of the estimated value (equation (23)). The increase with respect to the simulated case is because real data does not perfectly follow the proposed model. The extrinsic calibration returns the camera pose with an uncertainty of less than 1° for orientation and 10^{-1}m for position (equation (24)). Predicting the world position of a car is shown in figure 10 as 90% confidence ellipses the area of the ellipse is proportional to the squared distance of the vehicle to the camera, which is the expected behavior of a perspective projection. Accuracy can be improved with more accurate calibration points and possibly by expanding the model to describe the curvature of the ground surface and optical distortion at finer level of detail.

5.3. Relationship between the proposed method and the OpenCV library

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library which includes a comprehensive set of both classic and state-of-the-art computer vision algorithms [16] and is widely used by the artificial vision system developer community. For this reason, this work follows OpenCV's formulation for the calibration model except for the specific function that models radial distortion. This leaves open the possibility to include later other distortion models. The adoption of the stereographic model is an appropriate description of the optical distortion for the camera utilized in this work [1]. The back-projection to world coordinates is solved analytically (algoritm 1) assuming that the object is on the horizontal $z_w = 0$ plane.

OpenCV provides functionality that is similar to the solutions here proposed but with an incomplete treatment of uncertainty. The function `calibrateCamera` estimates intrinsic parameters by minimizing the projection error in a least squares estimator fashion [40] following Zhang [21], Bouguet [22]. It also computes the Jacobian of the projected image coordinates with respect to the parameters but not with the purpose of uncertainty propagation, it is used during the global optimization of camera calibration. It returns a vector of standard deviations of the parameters by an inverse propagation of sorts: it multiplies the unbiased estimator of the projected variance by the Moore-Penrose inverse of the Jacobian. There is no treatment of interacting terms in the covariance, it assumes the parameters are uncorrelated. This work calibrations show (equations (23) and 24) covariance matrices with non negligible interaction terms clearly meaning that the presented approach can contribute to improve OpenCV's methods. Also `calibrateCamera` does not take into account the uncertainty of the detected corners. The function `solvePnP` solves for the pose of an object given corresponding 3D-2D points and `warpPerspective` can map image coordinates to a world plane provided the right transformation matrix; both without treatment of uncertainty.

In sum, this work deals with a set of topics relevant to engineering applications of wide field vision systems. The algorithm developed fulfills the function of predicting the position in a map with correct quantification of position uncertainty, thus functioning as a position sensor.

Acknowledgments

The authors thank Dr Inés Samengo for comments on the manuscript.

Funding:

This work was supported by Universidad Nacional de Quilmes, project *Sistemas Autónomos Basados en Inteligenica Artificial*, under *Programa I + D UNQ EXPTE 1303/19*. Sebastián I. Arroyo and Ulises Bussi acknowledge the PhD scholarships received from Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET).

ORCID iDs

Sebastián I Arroyo https://orcid.org/0000-0001-9696-0513
Ulises Bussi https://orcid.org/0000-0002-0299-2651
Félix Safar https://orcid.org/0000-0002-5223-3469
Damián Oliva https://orcid.org/0000-0001-9714-4112

References

[1] Stanganelli D, Oliva D E, Noblia M and Safar F 2014 Calibración de una cámara fisheye comercial con el modelo unificado para la observación de objetos múltiples *2014 IEEE Biennial Congress of Argentina (ARGENCON)* (IEEE) (https://doi.org/10.1109/argencon.2014.6868487)
[2] Wang W, Gee T, Price J and Qi H 2015 Real time multi-vehicle tracking and counting at intersections from a fisheye camera *2015 IEEE Winter Conf. on Applications of Computer Vision* (IEEE) 17–24
[3] Floreano D *et al* 2013 Miniature curved artificial compound eyes *Proc. of the National Academy of Sciences* **110** 9267–72
[4] Chahl J S and Srinivasan M V 1997 Reflective surfaces for panoramic imaging *Appl. Opt.* **36** 8275–85
[5] Junior J M, Tommaselli A M G and Moraes M V A 2016 Calibration of a catadioptric omnidirectional vision system with conic mirror *ISPRS J. Photogramm. Remote Sens.* **113** 97–105
[6] VIVOTEK. Product Brochure, IP Surveillance Solutions 2018 http://download.vivotek.com/downloadfile/downloads/brochure/brochure_en.pdf
[7] Klinger T, Rottensteiner F and Heipke C 2017 Probabilistic multi-person localisation and tracking in image sequences *ISPRS J. Photogramm. Remote Sens.* **127** 73–88
[8] Peng P, Tian Y, Wang Y, Li J and Huang T 2015 Robust multiple cameras pedestrian detection with multi-view bayesian network *Pattern Recognit.* **48** 1760–72
[9] Criminisi A, Reid I and Zisserman A 1999 A plane measuring device *Image Vision Comput.* **17** 625–34
[10] Geyer C and Daniilidis K 2000 A unifying theory for central panoramic systems and practical implications *European Conf. on Computer Vision (ECCV)* pp 445–61978-3-540-67686-7
[11] Ying X and Hu Z 2004 Can we consider central catadioptric cameras and fisheye cameras within a unified imaging model *European Conf. on Computer Vision* (Berlin: Springer) pp 442–55
[12] Sundareswara R and Schrater P R 2005 Bayesian modelling of camera calibration and reconstruction *Fifth International Conf. on 3-D Digital Imaging and Modeling (3DIM'05)* pp 394–401
[13] Civera J, Bueno D R, Davison A J and Montiel J M M 2009 Camera self-calibration for sequential bayesian structure from motion *2009 IEEE International Conf. on Robotics and Automation* (IEEE) pp 403–8
[14] Corke P 2011 *Robotics, Vision and Control—Fundamental algorithms in MATLAB* (Berlin: Springer) (https://doi.org/10.1007/978-3-540-73958-6_2)
[15] Ponce J and Forsyth D 2012 *Computer Vision: A Modern Approach* (One Lake Street, Upper Saddle River, New Jersey 07458: Pearson Education) 9780136085928
[16] Bradski G and Kaehler A 2008 *Learning OpenCv: Computer Vision With The Opencv Library* 1st (Sebastopol, California: O'Reilly Media) 9780596516130
[17] Wong W K, ShenPua W, Loo C K and Lim W S 2011 A study of different unwarping methods for omnidirectional imaging *2011 {IEEE} International Conference on Signal and Image Processing Applications ({ICSIPA})* (IEEE) pp 433–8
[18] Fraser C S 2013 Automatic camera calibration in close range photogrammetry *Photogrammetric Engineering & Remote Sensing* **79** 381–8
[19] Murray R M, Li Z and Sastry S S 1994 *A Mathematical Introduction to Robotic Manipulation* 29 (Boca Raton, FL: CRC Press) 9780849379819
[20] Valdenebro A G 2016 Visualizing rotations and composition of rotations with Rodrigues' vector *Eur. J. Phys.* **37** 065001
[21] Zhang Z 2000 A flexible new technique for camera calibration *IEEE Trans. Pattern Anal. Mach. Intell.* **22** 1330–4
[22] Bouguet J-Y 2004 Camera calibration toolbox for matlab http://www.vision.caltech.edu/bouguetj/calib_doc/index.html
[23] Laganière R 2014 *OpenCV Computer Vision Application Programming Cookbook* 2nd edn (Birmingham, UK: Packt Publishing Ltd) 1782161481
[24] Ke T and Roumeliotis S I 2017 An efficient algebraic solution to the perspective-three-point problem *2017 IEEE Conf. on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017* pp 4618–26

[25] Gelman A, Carlin J B, Stern H S and Rubin D B 2004 *Bayesian Data Analysis* (*Texts in Statistical Science*) 2nd edn (Boca Raton, London: Chapman nd Hall/CRC) 9781584883883
[26] Wilson A G 2020 The case for Bayesian deep learning *arXiv* **2001.10995** - https://cims.nyu.edu/~andrewgw/caseforbdl.pdf
[27] Taboga M 2010 *Lectures on Probability and Statistics* https://statlect.com
[28] Bishop C M 2006 *Pattern Recognition and Machine Learning. Information Science and Statistics* (*Information science and statistics*) (Berlin/Heidelberg, Germany: Springer Science + Business Media) 9780387310732
[29] Van Rossum G and Drake F L 2009 *Python 3 Reference Manual* (Scotts Valley, CA: CreateSpace) 1441412697
[30] Oliphant T E 2006 *A Guide to NumPy* 1 (USA: Trelgol Publishing) https://ecs.wgtn.ac.nz/foswiki/pub/Support/ManualPagesAndDocumentation/numpybook.pdf
[31] Virtanen P 2020 Scipy: open source scientific tools for python *Nature Methods* **17** 261–72
[32] Hunter J D 2007 Matplotlib: A 2d graphics environment *Comput. Sci. Eng.* **9** 90–5
[33] Spyder Project Contributors 2019 Spyder-Documentation https://www.spyder-ide.org/
[34] Salvatier J, Wiecki T V and Fonnesbeck C 2016 Probabilistic programming in python using pymc3 *PeerJ Computer Science* **2** e55
[35] Criminisi A, Reid I and Zisserman A 2000 Single view metrology *Int. J. Comput. Vision* **40** 123–48
[36] Ter Braak C J F 2006 A Markov Chain Monte Carlo version of the genetic algorithm Differential Evolution: Easy Bayesian computing for real parameter spaces *Stat. Comput.* **16** 239–49
[37] Cai S and Zang Z 2013 A deformed chessboard pattern for automatic camera calibration *2013 International Conf. on Advanced ICT and Education (ICAICTE-13)* (Atlantis Press) (https://doi.org/10.2991/icaicte.2013.117)
[38] Mekid S and Vaja D 2008 Propagation of uncertainty: expressions of second and third order uncertainty with third and fourth moments *Measurement* **41** 600–9
[39] Revels J, Lubin M and Papamarkou T 2016 Forward-mode automatic differentiation in julia arXiv:1607.07892
[40] van de Geer S A 2005 *Least Squares Estimation* (Atlanta, Georgia, U.S.: American Cancer Society) 9780470013199 (https://doi.org/10.1002/0470013192.bsa199)