

An Ellipsoid Object Model of the Refraction Surface

Szabolcs Pável^{ab}

^aFaculty of Mathematics and Informatics, Babeş-Bolyai University,
Cluj-Napoca, Romania
szabolcs.pavel@cs.ubbcluj.ro

^bRobert Bosch SRL, Cluj-Napoca, Romania

Abstract

Geometric image distortions appear when cameras register the image from behind a refractive object – e.g. a car windshield. To ensure the reliability of 3D perception algorithms, a distortion model is necessary. The model has to be general enough to capture the variety of possible refractive object geometries. We propose a method where we directly model the refractive media as a thick ellipsoid, and compute the resultant distortions by tracing individual light rays as they refract on the *inner and outer surface* of the object. With this new ellipsoid model provides flexibility and via the model parameters we are able to capture all important factors influencing distortions, namely the curvature of the surfaces, position relative to the camera, and thickness of the refractive material. We test the proposed model on a synthetic dataset, analyzing the advantages and possible failure cases of our method.

Keywords: Image distortions, Calibration, Inverse models

1. Introduction

Video-cameras are preferred sensors for perception in robotics and autonomous driving because of their low cost and high resolution. When vision is used for 3D perception [20, 11], a camera model is employed to associate image pixels with points on objects in the outside world. Camera calibration is the procedure of finding the optimal parameters for the camera model, either through static calibration with predefined calibration patterns, or in an online manner through self-calibration. These camera models and calibration techniques often use a distortion estimation step, where geometric errors in the optic system are corrected.

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Geometric distortions can arise when the camera is placed behind a protective cover – e.g. a windshield of the car. These distortions are affected by the global properties of the object – e.g. position to the camera, curvature of the surface and thickness of the material – as well as irregularities of the surface, resulting in local distortions. In our previous work [15] we modeled the irregularities of the distortion surface using an RBF-network, while assuming that the global properties of the refractive object are known. We constructed the forward model, where knowing the parameters of the camera, the refractive object and the scene we map a pixel to a 3D point in the scene. The forward model was implemented as a fully differentiable raycasting algorithm. Using model inversion and machine learning techniques [2] we estimated the parameters that generated the distortions.

In this work we use a similar methodology as in [15], but this time we address the global properties of the refractive object. We model the surface of the refractive media as an ellipsoid, which is general enough to approximate a large variety of objects on the area seen by the camera. The model is also designed to be composable with the RBF-network model of the local surface. In the following sections we describe the model and the raycasting algorithm. We address the issue of arising symmetries in the distortion estimation process, and we propose a regularization to help the minimization. Finally we test our method on a synthetic dataset.

2. Related Work

Calibration methods can be classified as static calibration methods [21, 24, 22, 16], which use objects with known patterns to provide the highest accuracy, or as self-calibration methods [7, 4, 8, 18], where calibration is done in an online manner during operation, leveraging geometric constraints of the scene. Our method is a static calibration method, as we use images of checkerboard patterns to estimate the distortions.

Different algorithms use specific distortion models with different complexity. The most widely used models consider radial distortions [3, 8, 10], while some cameras, like fish-eye cameras require specific models [7]. More flexible models, e.g. the rational function distortion model [6] are also studied. These models usually consider distortions as a function in pixel space, while we use a physical model of the refractive object to model distortions.

In our work we consider distortions from light refractions. Similar work was done by Agrawal et. al. [1], who analyzed distortions through flat refractive surfaces, using methods of camera calibration in [19, 5]. Morinaka et. al. [14] modeled complex distortions, observed when the camera is placed behind a wine glass or a car windshield using the “raxel” imaging model [9].

Deep learning methods are also used to estimate radial distortions [12, 17], to rectify fish-eye images [23] or to estimate windshield distortions [13] based on a single image.

3. Ellipsoid Model

We model the refractive object as the space between two ellipsoids. The two ellipsoids have the same center position $\mathbf{t}_c \in \mathbb{R}^3$ and orientation – represented as an axis-angle rotation $\boldsymbol{\omega} \in \mathbb{R}^3$. The inner ellipsoid has the semi-axes $a, b, c \in \mathbb{R}$. To define the semi-axes of the outer ellipsoid, we add an additional, small thickness $t \in \mathbb{R}$ to each semi-axes of the inner ellipsoid. This way $a + t, b + t, c + t \in \mathbb{R}$ give the semi-axes of the outer ellipsoid. The quantities $\boldsymbol{\theta} = \{\mathbf{t}_c, \boldsymbol{\omega}, a, b, c, t\}$ together form the parameters of the ellipsoid object model.

To make further computation simpler, it is useful to see the ellipsoid as an affine image of a unit sphere centered at the origin. The transformation is defined by a 3×3 matrix \mathbf{A} and the translation vector \mathbf{t}_c , with $\mathbf{A} = R(\boldsymbol{\omega})\text{diag}(a, b, c)$, where $R(\boldsymbol{\omega})$ is a rotation matrix constructed from $\boldsymbol{\omega}$ using Rodrigues’ formula, and $\text{diag}(a, b, c)$ denotes a diagonal matrix with values a, b, c . Using the affine mapping, all necessary operations, including intersection with a ray and surface normal evaluations can be reduced to operations on the unit circle.

The scope of a camera model is to associate pixels with light rays from the outside world. The intersection points between these outgoing light rays and objects (e.g. a checkerboard pattern during calibration) will define what we see on the image. In a distortion-free setup this outgoing light ray coming from the camera center and going through the image pixel – this is the ray given by the pinhole camera model. In our physical model the direction of the original light ray is modulated when it enters or leaves the refractive object – in this work modeled by an ellipsoid. The change of direction is computed using Snell’s law of refraction, and it is a function of the incident ray, the surface normal at the intersection point, and the relative refractive index of the two materials (e.g glass and air). The complete raycasting process is fully differentiable, allowing the gradient based optimization of the ellipsoid model parameters. The method is implemented in the PyTorch framework to leverage the backward mode automatic differentiation for the optimization.

4. Symmetries of the Object Model

The ellipsoid model of the refractive object overparameterizes the image distortions, resulting in arising symmetries of the physical model. The observed distortions are invariant with respect to a set of transformations applied to the object. As a result of the invariance, when we invert the model to estimate parameters of the ellipsoid, the full parameter set is not recoverable without some prior knowledge about the object. While in some cases an estimation of the distortions may be sufficient, reconstructing an approximate 3D model of the object can also be desirable. Identifying the symmetries and properly handling them using regularization techniques is therefore an important step in our method.

To give an intuitive example of these symmetries, we consider a simple 2D case where the refractive object is a thick circle, and we consider two variables: 1) the

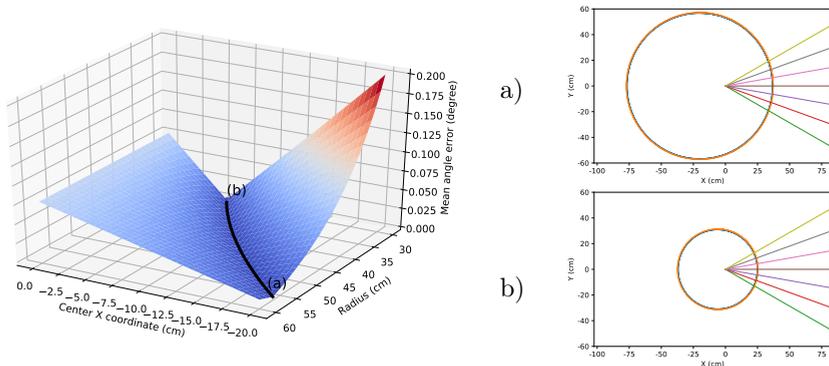


Figure 1: Left: Distortion errors relative to the reference configuration for varying circle center position and radius values. Right: Visualization of two different parameters yielding the same distortions as our reference setup.

relative distance of the circle center to the camera center in the direction of the optical axis; 2) the radius of the circle. This analysis translates well to the center position of the ellipsoid and the length of the semi-axes, the two sets of variables where these problems arise in the 3D case.

We consider the reference parameters of circle position of -10 cm and circle radius of 40 cm, and compare all other distortions to the ones measured in this configuration. We express the distortion error relative to the reference setup as the angle between a refracted ray in the reference setup and the examined setup, averaged over multiple incident rays with different initial orientations.

Figure 1 shows the distortion error relative to the reference setup for different parameters. The black notes a set of parameters where the distortion error is less than 10^{-3} degrees. We can observe that we can achieve low distortion errors as long as we adjust both parameters simultaneously in the proper way.

5. Optimization of the Model Parameters

We use the standard setup of static camera calibration to estimate the model parameters. Given a pixel, our forward model finds the 3D point in the scene seen by the camera. In a static calibration setup we use a planar checkerboard pattern as the target object, where the dimensions of the squares are known. Using model inversion – based on a set of images of checkerboard patterns – the parameters of the ellipsoid model can be recovered using gradient based minimization.

Our loss function has two components: a reconstruction and a regularization term. Let \mathbf{I}_i be a calibration image, where $\mathbf{p}_{i,j}$ represents the pixel locations of the j th checkerboard corners on the image. We denote with $f_{\theta,i}(\cdot)$ the raycasting function which takes a pixel and returns a point on the checkerboard object associated

with image \mathbf{I}_i . Let $\mathbf{x}_{i,j}^{cb}$ be the ground-truth world coordinates of the checkerboard corner associated with pixel $\mathbf{p}_{i,j}$. Then the reconstruction error is expressed as the squared error between the estimated and ground-truth corner coordinates:

$$\mathcal{L}_{rec}(\boldsymbol{\theta}) = \sum_{\mathbf{I}_i} \sum_{\mathbf{p}_{i,j}} \|\mathbf{f}_{\boldsymbol{\theta},i}(\mathbf{p}_{i,j}) - \mathbf{x}_{i,j}^{cb}\|^2.$$

In Section 4 we showed that the ellipsoid model overparameterizes the distortions, and the full physical model cannot be recovered without prior knowledge. We include this prior knowledge as a regularization term during the minimization. More specifically, we constrain the distance between the camera center and the point where the principal axis of the camera – also being the Z axis of the camera coordinate system – intersects the inner surface of the ellipsoid. Let $d_Z(\boldsymbol{\theta})$ be the distance between the camera center and the ellipsoid, and β be the expected constant value of this distance. Then the regularization term is a L_2 penalty:

$$\mathcal{L}_{reg}(\boldsymbol{\theta}) = (d_Z(\boldsymbol{\theta}) - \beta)^2.$$

The full loss function is a weighted sum of the two terms:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_{rec}(\boldsymbol{\theta}) + \lambda \mathcal{L}_{reg}(\boldsymbol{\theta}). \quad (5.1)$$

The loss function in Equation 5.1 is minimized using the L-BFGS optimization method. We chose the L-BFGS method as it is a quasi-Newton method, which is both efficient and it does not require a direct evaluation of the Hessian matrix, making the method compatible with backward mode automatic differentiation.

6. Experiments

We tested our method on a synthetic dataset. We generated a set of ground-truth ellipsoid objects, and we used our forward model to render images with checkerboard patterns. The simulated camera mimics the properties of a Raspberry Pi camera module, recording images at 3280×2464 pixel resolution with a horizontal field of view of 62.2 degrees. For each run we used 10 checkerboard patterns placed at random positions and orientations. Each pattern had 8×8 corners on them, therefore the 10 images provided 640 data points in total.

After generating an image dataset, we reinitialized the parameters of the ellipsoid to represent a small circle centered and the origin. We used our optimization method presented in Section 5 to find the optimal parameters, ideally corresponding to the ground-truth ellipsoid used to generate the samples. The minimization method was able to find a model with sub-pixel distortion error in each case. The resulting shape of the ellipsoid however did not always match the ground-truth model.

Figure 2 shows examples for the experiment. We can observe that in examples a), b) and c) the estimated ellipsoid approximates the surface well on the area which

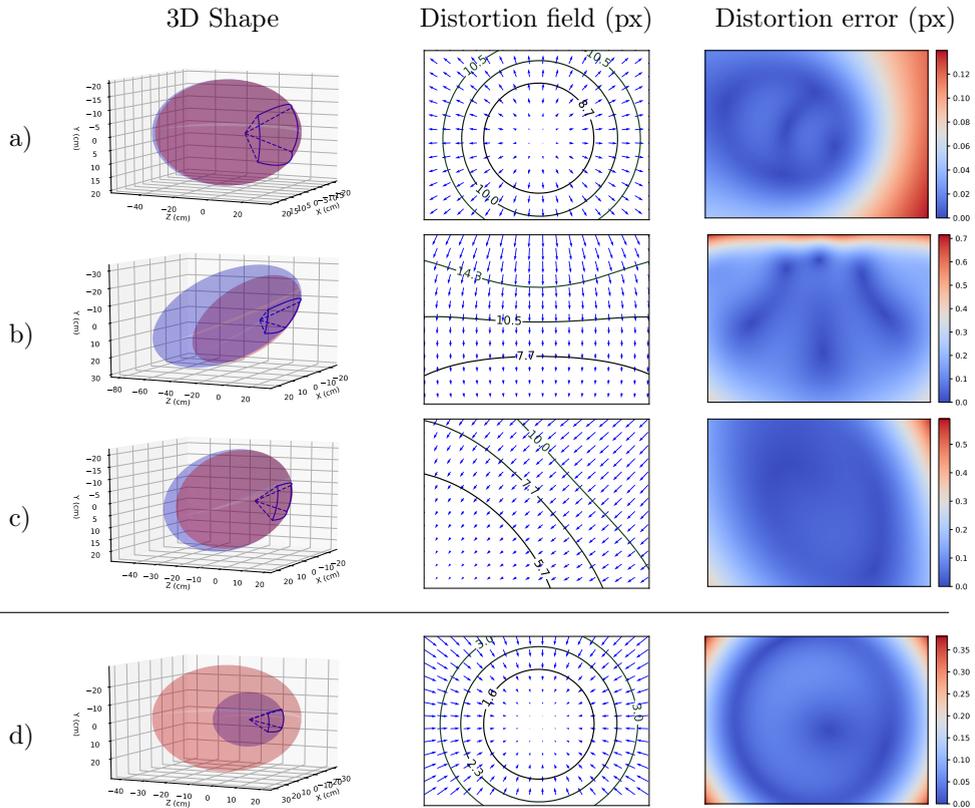


Figure 2: Results of our method on synthetic data. Left column is the predicted (blue) and ground-truth (red) ellipsoid. The blue frame shows the area seen by the camera. Middle column shows the distortion field for the estimated ellipsoid. Last column shows the norm of the distortion error. Example d) is a run without regularization.

is seen by the camera. On the other hand, we can also see that in some case (e.g. in example b) this good local approximation can be achieved without finding the global ground-truth parameters. This suggests, that the exact estimation requires more prior information about the refractive object. Although the predicted ellipsoid is not always correct, the distortion errors are under one pixel in each example, with small variations across the image. Example d) shows a case where because of not using regularization, even if a solution with small distortion error was found, the predicted ellipsoid is not approximating the real surface well, not even at the area seen by the camera.

7. Conclusions

We presented a distortion estimation method for scenarios where the camera is placed behind a refractive object. We modeled the object as an ellipsoid, and used machine learning techniques to estimate the model parameters. We analyzed the possible failure cases, where multiple different ellipsoids result in the same distortions, and proposed a regularization which solves this issue. The method was tested on a synthetic dataset, generated using the forward model of image generation, implemented as a raycasting algorithm. We were able to obtain a close approximation of the object surface on the region seen by the camera.

In our future work we will focus on the validation of the algorithm in a real scenario. Although our method achieves good results in a noise-free setup, application with real cameras and dataset is still an open question. We will also look into the integration of the local model in [15] with the ellipsoid model, resulting in a complete model of distortions through refractive objects.

References

- [1] AGRAWAL, A., RAMALINGAM, S., TAGUCHI, Y., AND CHARI, V. A theory of multi-layer flat refractive geometry. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2012), IEEE, pp. 3346–3353.
- [2] BISHOP, C. *Pattern Recognition and Machine Learning*. Springer-Verlag, 2006.
- [3] BROWN, D. C. Decentering distortion of lenses. *Photogrammetric Engineering and Remote Sensing* 32, 3 (1966), 444–462.
- [4] CEFALU, A., HAALA, N., AND FRITSCH, D. Structureless bundle adjustment with self-calibration using accumulated constraints. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* 3, 3 (2016), 3–9.
- [5] CHARI, V., AND STURM, P. Multiple-view geometry of the refractive plane. In *BMVC 2009-20th British Machine Vision Conference* (2009), The British Machine Vision Association (BMVA), pp. 1–11.
- [6] CLAUS, D., AND FITZGIBBON, A. W. A rational function lens distortion model for general cameras. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (2005), vol. 1, IEEE, pp. 213–219.
- [7] DEVERNAY, F., AND FAUGERAS, O. Straight lines have to be straight. *Machine vision and applications* 13, 1 (2001), 14–24.
- [8] FITZGIBBON, A. W. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001* (2001), vol. 1, IEEE, pp. I–125–I–132.
- [9] GROSSBERG, M. D., AND NAYAR, S. K. The raxel imaging model and ray-based calibration. *International Journal of Computer Vision* 61, 2 (2005), 119–137.
- [10] HARTLEY, R., AND KANG, S. B. Parameter-free radial distortion correction with center of distortion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 8 (2007), 1309–1321.

- [11] HARTLEY, R., AND ZISSERMAN, A. *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [12] LOPEZ, M., MARI, R., GARGALLO, P., KUANG, Y., GONZALEZ-JIMENEZ, J., AND HARO, G. Deep single image camera calibration with radial distortion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 11817–11825.
- [13] LŐRINCZ, S.-B., PÁVEL, S., AND CSATÓ, L. Single view distortion correction using semantic guidance. In *2019 International Joint Conference on Neural Networks (IJCNN)* (2019), IEEE, pp. 1–6.
- [14] MORINAKA, S., SAKAUE, F., SATO, J., ISHIMARU, K., AND KAWASAKI, N. 3D reconstruction under light ray distortion from parametric focal cameras. *Pattern Recognition Letters* (11 2018), 16–54.
- [15] PÁVEL, S., SÁNDOR, C., AND CSATÓ, L. Distortion estimation through explicit modeling of the refractive surface. In *International Conference on Artificial Neural Networks* (2019), Springer, pp. 17–28.
- [16] PRESCOTT, B., AND MCLEAN, G. Line-based correction of radial lens distortion. *Graphical Models and Image Processing* 59, 1 (1997), 39–47.
- [17] RONG, J., HUANG, S., SHANG, Z., AND YING, X. Radial lens distortion correction using convolutional neural networks trained with synthesized images. In *Asian Conference on Computer Vision* (2016), Springer, pp. 35–49.
- [18] STEIN, G. P. Lens distortion calibration using point correspondences. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1997), IEEE, pp. 602–608.
- [19] STURM, P., AND RAMALINGAM, S. A generic concept for camera calibration. In *European Conference on Computer Vision* (2004), T. Pajdla and J. Matas, Eds., Springer, Springer Berlin Heidelberg, pp. 1–13.
- [20] SZELISKI, R. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.
- [21] TSAI, R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation* 3, 4 (1987), 323–344.
- [22] WANG, A., QIU, T., AND SHAO, L. A simple method of radial distortion correction with centre of distortion estimation. *Journal of Mathematical Imaging and Vision* 35, 3 (2009), 165–172.
- [23] YIN, X., WANG, X., YU, J., ZHANG, M., FUA, P., AND TAO, D. Fisheyerecnet: A multi-context collaborative deep network for fisheye image rectification. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2018), pp. 469–484.
- [24] ZHANG, Z. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence* 22 (2000), 1330–1334.