

Accuracy of fish-eye lens models

Ciarán Hughes,^{1,2,*} Patrick Denny,² Edward Jones,¹ and Martin Glavin¹

¹Connaught Automotive Research Group, Electrical and Electronic Engineering,
National University of Ireland, Galway, University Road, Galway, Ireland

²Valeo Vision Systems, IDA Business Park, Dunmore Road, Tuam, County Galway, Ireland

*Corresponding author: ciaran.hughes@valeo.com

Received 2 March 2010; revised 11 May 2010; accepted 14 May 2010;
posted 18 May 2010 (Doc. ID 124908); published 7 June 2010

The majority of computer vision applications assumes that the camera adheres to the pinhole camera model. However, most optical systems will introduce undesirable effects. By far, the most evident of these effects is radial lensing, which is particularly noticeable in fish-eye camera systems, where the effect is relatively extreme. Several authors have developed models of fish-eye lenses that can be used to describe the fish-eye displacement. Our aim is to evaluate the accuracy of several of these models. Thus, we present a method by which the lens curve of a fish-eye camera can be extracted using well-founded assumptions and perspective methods. Several of the models from the literature are examined against this empirically derived curve. © 2010 Optical Society of America

OCIS codes: 100.2980, 100.4994, 110.6980, 150.1488.

1. Introduction

The rectilinear pinhole camera model is typically considered the ideal and intuitive model, whereby straight lines in the real world are mapped to straight lines in the image generated by the camera. However, most real optical systems will introduce some undesirable effects, rendering the assumption of the pinhole camera model inaccurate. The most evident of these effects is radial barrel distortion, particularly noticeable in fish-eye camera systems, where the level of this distortion is relatively extreme. This radial distortion causes points on the image plane to be shifted from their ideal position in the rectilinear pinhole camera model, along a radial axis from the principal point in the fish-eye image plane. The visual effect of this displacement in fish-eye optics is that the image will have a higher resolution in the foveal areas, with the resolution decreasing nonlinearly toward the peripheral areas of the image. The considerable advantage of using fish-eye cameras is that a far greater portion of the scene is imaged than with the standard field-of-view (FOV) camera.

In order to examine the accuracy of the models, we present a method by which the radial fish-eye lens curve can be extracted using a set of well-founded assumptions and perspective principles. The lens curve extraction uses a planar calibration grid. The method is nonparametric, and does not assume any particular fish-eye lens model. We use this extracted curve as the basis of a metric by which various fish-eye lens models can be compared.

A. Previous Work

There has been much work done in the area of camera calibration to remove radial lens displacement. Brown [1] described radial distortion using an odd-order polynomial model, and Tsai [2] provided one of the seminal works in modern camera calibration using this model. Many other authors have calibrated cameras using this model to remove the radial lensing effect (e.g., Zhang [3]). However, due to the particularly high levels of radial displacement present in fish-eye cameras, it is generally considered that the odd-order polynomial model cannot sufficiently compensate for the radial displacement present in these cameras. There have been several alternative models developed to deal with fish-eye cameras, including the fish-eye transform (FET) [4], the polynomial fish-eye

0003-6935/10/173338-10\$15.00/0
© 2010 Optical Society of America

transform (PFET) [5], the FOV model [5], and the division model [6,7]. It is these models that we examine in this paper.

On the validation and comparison of fish-eye lens models, there has been little research done, with the exception of work published by Schneider *et al.* [8]. In their work, Schneider *et al.* examine the accuracy of the various fish-eye projection functions using spatial resection and bundle adjustment. However, we do not limit our examination to the fish-eye projection functions, because we also include other fish-eye models in our comparisons.

B. Assumptions

In this paper, we concentrate solely on the accuracy of radial lens models. Thus, we assume *a priori* knowledge of the camera principal point. The principal point can be determined using one of a number of methods [9–11]. We also assume that tangential (decentering) distortion is negligible. There are two primary causes of tangential distortion: inaccurate distortion center estimation, and thin prism distortion. Thin prism distortion arises from imperfections in lens design, manufacturing, and camera assembly, which causes a degree of both radial and tangential distortion [12]. It has been demonstrated that the vast majority of tangential distortion can be compensated for just by using distortion center estimation [13]. Several other researchers have made the assumption that other causes of tangential distortion are negligible [2,3,5,10]. We also assume that there is zero skew (shear) and unit aspect ratio (affinity) [14,15].

When it comes to the fish-eye lens displacement curve, we make only two assumptions regarding its form:

- The lens displacement curve of the camera is a monotonically increasing function, with respect to distance from the principal point.
- The curve approximates a linear function in the foveal areas of the image near the principal point.

The monotonicity of the lens displacement curve is necessary, because it ensures that every point in a scene maps to at most one point on the image plane and preserves the order of points in terms of their distance from the principal point [10]. The second assumption is supported by the fact that for low values of incident angle θ , all of the fish-eye projection functions described in the next section can be approximated as a linear function.

2. Fish-Eye Projection Functions and Models

Rectilinear (pinhole) projection is so called because it preserves the rectilinearity of the projected scene (i.e., straight lines in the scene are projected as straight lines on the image plane). The rectilinear projection mapping function is given as [16]

$$r_u = f \tan(\theta), \quad (1)$$

where f is the distance between the principal point and the image plane, θ is the incident angle (in ra-

dians) of the projected ray to the optical axis of the camera, and r_u is the projected radial distance from the principal point on the image plane [Fig. 1]. However, for wide FOV cameras, under rectilinear projection, the size of the projected image becomes very large, increasing to infinity at a FOV of 180°.

A. Fish-Eye Projection Functions

Fish-eye projection functions are designed such that a greater portion of the scene is projected onto the image sensor on the image plane, at the expense of introducing (often considerable) radial distortion. There are several different fish-eye projection functions [16]. Figure 2 gives representations of each of the types of fish-eye projection as spherical projections.

1. Equidistant Projection Function

In equidistant projection, the radial distance r_d on the image plane is directly proportional to the angle of the incident ray, and is equivalent to the length of the arc segment between the z axis and the projection ray of point P on the sphere [Fig. 2(a)]. Thus, the projection function is

$$r_d = f\theta. \quad (2)$$

To determine the distortion function (i.e., the function that converts a rectilinear point to its equidistant fish-eye equivalent) we solve Eqs. (1) and (2) in terms of θ and equate to get

$$r_d = f \arctan\left(\frac{r_u}{f}\right). \quad (3)$$

The inverse is

$$r_u = f \tan\left(\frac{r_d}{f}\right). \quad (4)$$

These equations describe the conversion between rectilinear image space and the equidistant fish-eye image space, and vice versa.

2. Equisolid Projection Function

In equisolid projection, the projected distance is equivalent to the length of the chord on the projection sphere between the z axis and the projection of point P onto the sphere [Fig. 2(b)]. The projection function is

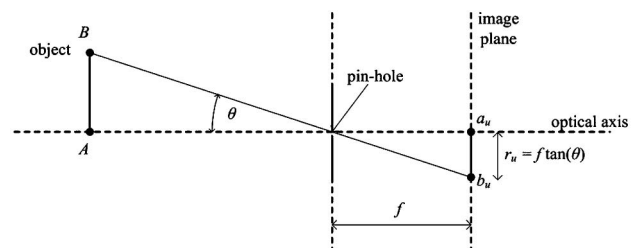


Fig. 1. Rectilinear projection representation.

$$r_d = 2f \sin\left(\frac{\theta}{2}\right). \quad (5)$$

Thus, the equisolid lens distortion function is

$$r_d = 2f \sin\left(\frac{\arctan(r_u/f)}{2}\right). \quad (6)$$

And the inverse is

$$r_u = f \tan\left(2 \arcsin\left(\frac{r_d}{2f}\right)\right). \quad (7)$$

Equisolid projection is also known as equal-area projection, as the ratio of an incident solid angle and its resulting area in an image is constant.

3. Orthographic Projection Function

Orthographic projection is formed from the direct perpendicular projection of the point of the ray intersection with the projection sphere to the image plane [Fig. 2(c)]. The projection function is

$$r_d = f \sin(\theta). \quad (8)$$

The orthographic lens distortion function simplifies to

$$r_d = \frac{r_u}{\left(1 + \frac{r_u^2}{f^2}\right)^{1/2}}. \quad (9)$$

The inverse is a function of similar form:

$$r_u = \frac{r_d}{\left(1 - \frac{r_d^2}{f^2}\right)^{1/2}}. \quad (10)$$

Orthographic projection is not commonly used in fish-eye designs because, as can be seen from Fig. 2(c), points beyond 90° to the optical axis cannot be projected onto the image plane. Additionally, such lenses suffer from greater radial distortion in the extremities of the image than either equidistant or equisolid projections.

It should be noted that the term “orthographic projection” here does not refer to the typical meaning, i.e., we are not referring to the orthographic projection where the point of projection is at infinity (such as is used in, for example, cartography); rather, we use the term to denote the specific spherical projection described in this subsection.

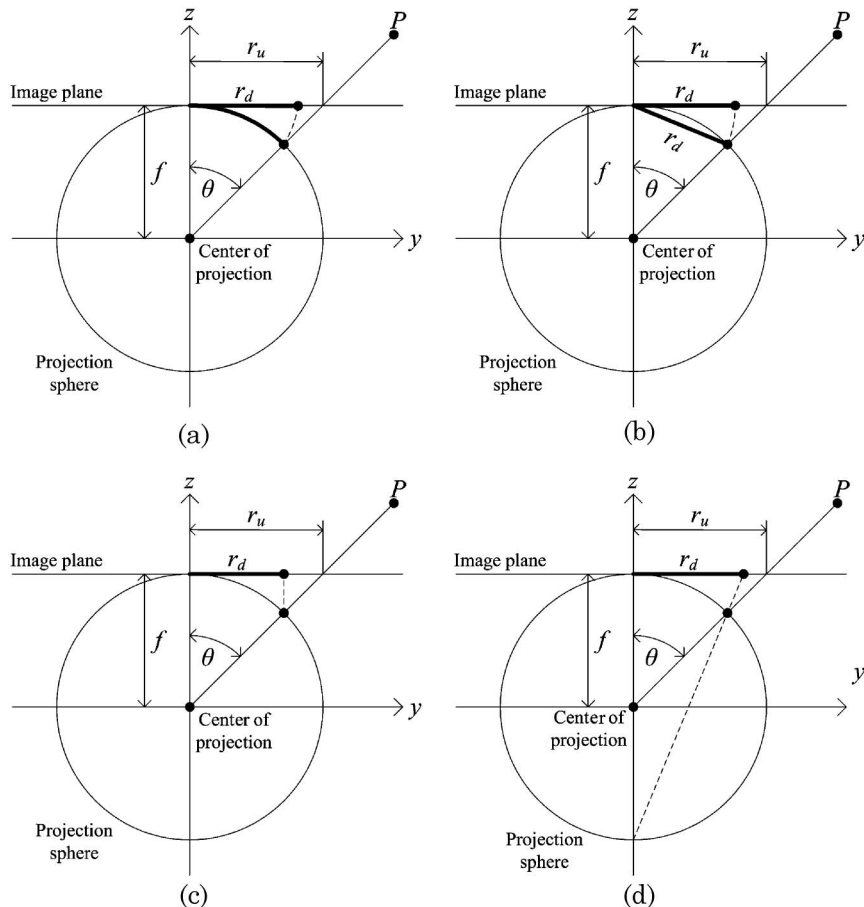


Fig. 2. Fish-eye projection function representations, showing the projection of the point P to the projection sphere and then the reprojection of the point on the projection sphere to the image plane: (a) equidistant, (b) equisolid, (c) orthographic, and (d) stereographic.

4. Stereographic Projection

In stereographic projection, as with the other projection functions, the center of projection of a 3D point to the projection sphere is the center of the projection sphere [Fig. 2(d)]. However, the center of reprojection of that point onto the image plane is the opposite of the tangential point [Fig. 2(d)]. The projection function is

$$r_d = 2f \tan\left(\frac{\theta}{2}\right). \quad (11)$$

Thus, the stereographic lens distortion function is

$$r_d = 2f \tan\left(\frac{\arctan\left(\frac{r_u}{f}\right)}{2}\right). \quad (12)$$

The inverse is

$$r_u = f \tan\left(2 \arctan\left(\frac{r_d}{2f}\right)\right). \quad (13)$$

Through elementary trigonometric properties, Eq. (13) reduces to

$$r_d = \frac{r_u}{1 - \frac{r_u^2}{4f^2}}. \quad (14)$$

This is recognizable as being in the form of the division model, introduced almost simultaneously, and apparently independently, by Bräuer-Burchardt and Voss [6] and Fitzgibbon [7]. The first-order division model is

$$r_d = \frac{r_u}{1 - \lambda r_u^2}. \quad (15)$$

Thus, the stereographic function and the first-order division model amount to the same function, where $\lambda = 1/4f^2$.

B. Fish-Eye Radial Lens Models

Other than the projection functions, several models of fish-eye lenses have been proposed.

1. Polynomial Fish-Eye Transform

A polynomial that uses both odd and even coefficients has been proposed, and has been referred to as the PFET [4,17]:

$$r_d = \sum_{n=1}^{\infty} \kappa_n r_u^n = \kappa_1 r_u + \kappa_2 r_u^2 + \dots + \kappa_n r_u^n + \dots \quad (16)$$

This polynomial model was used as it makes the model independent of the underlying fish-eye mapping function and can take errors in the manufacture of fish-eye lenses into account. It has been suggested that a fifth-order form of the model is adequate to simulate the radial displacement introduced by fish-eye lenses [4].

2. Fish-Eye Transform

A logarithmic function, known as the FET, has also been proposed [4]. The model is described by

$$r_d = s \ln(1 + \lambda r_u), \quad (17)$$

where s is a simple scalar and λ controls the amount of displacement across the image. The inverse of this model is

$$r_u = \frac{\exp(r_d/s) - 1}{\lambda}. \quad (18)$$

3. Field-of-View Model

The FOV model, based on a simple optical model of a fish-eye lens, is described as [5]

$$r_d = \frac{1}{\omega} \arctan\left(2r_u \tan\left(\frac{\omega}{2}\right)\right). \quad (19)$$

The inverse is

$$r_u = \frac{\tan(r_d \omega)}{2 \tan\left(\frac{\omega}{2}\right)}, \quad (20)$$

where ω is the FOV of the camera.

C. Radial Distortion Parameters

Fish-eye lens manufacturers typically attempt to design lenses in which the distortion curves follow one of the projection functions described in Subsection 2.A. However, due to tolerances in the manufacturing process, fish-eye lenses can often deviate from the projection function they are designed to adhere to. To model this potential deviation, it has been proposed that the distortion function can be appended with polynomial elements to account for the deviations of the lens from the projection function [5,8]:

$$\Delta r_d = A_1 r_u^3 + A_2 r_u^5 + A_3 r_u^7, \quad (21)$$

where A_n are the additional radial distortion coefficients. Δr_d is simply added to the distortion function. For example, Eq. (3) becomes

$$r_d = f \arctan\left(\frac{r_u}{f}\right) + \Delta r_d. \quad (22)$$

The basic fish-eye lens model essentially becomes the first-order parameter. We have found in our work that considering coefficients beyond the seventh returns a negligible improvement in the results (this is examined in more detail in Section 4 and Table 4).

3. Lens Curve Acquisition

Here, a method of extracting the radial fish-eye displacement curve is described. This curve describes the radial displacement of points from the rectilinear image plane to the fish-eye image plane for all points at any given radial distance from the principal point. The basis of the method is to find the straight lines

on the rectilinear image plane that correspond to each edge in the fish-eye image plane. Given these straight lines, the rectilinear edge grid that corresponds to the fish-eye checkerboard image can be reconstructed using perspective principles. An overview of the algorithm is as follows:

1. An image of the checkerboard diagram is captured.
2. The edges are separated into their vertical and horizontal edges using the Sobel operator.
3. Using the points of convergence of the two lines nearest to the center in the image, the putative vanishing points are determined.
4. Using basic perspective principles and the putative vanishing points, two values for the slopes of the each line in the image are determined.
5. Ideally, these slope values will be the same for each line, but in practice, there will be error that needs to be minimized. The Levenberg–Marquardt algorithm is used to complete the minimization.
6. The distorted corners are extracted from the original checkerboard image, and the corresponding undistorted corners are determined as the intersection points of the lines extracted in step 5. Thus the radial displacement curve is extracted.
7. The extracted curve is smoothed using the LOESS algorithm to reduce the effect any error in the extraction of the corners.
8. Steps 1 to 6 are repeated a number of times, to improve the accuracy of the extracted curve.

A. Separating Edges

Given a calibration image of a checkerboard diagram, the horizontal and vertical edges on the checkerboard are identified and independently grouped. After applying a Gaussian smoothing to reduce the impact of any noise, we used an optimized Sobel edge detection to extract the edges [18]. The gradient information from the edge detection was used to separate the edges into their horizontal and vertical sets. Even though in the presence of distortion the gradient of a single line will change over the length of that line, the difference in gradients between the horizontal and vertical lines is still great enough that this separation is possible. Discontinuities in lines caused by corners in the grid are detected using a suitable corner detection method, e.g., [19], and thus corrected. Figures 5(a) and 5(b) show edges extracted from the distorted checkerboard test image, for a typical camera (Sony DSC 3) with a 103° FOV.

B. Determining Putative Vanishing Points

Prior to performing a nonlinear optimization of the vanishing points v_1 and v_2 , a pair of putative vanishing points (or initial guess) is useful. The basis of the putative vanishing point estimation is as follows: if a line is projected to the rectilinear image plane, the nearest point on that line to the principal point is the point of intersection of that line with the perpendicular through the principal point. Because of the assumption that the lens displacement function is

monotonic, the displaced equivalent of the point of intersection will remain the closest point on the projected line to the principal point. Thus, given the projection of a line in the fish-eye image, the slope of the equivalent line in the rectilinear image can be determined by finding the line of minimum distance between the principal point and the fish-eye line. The slope of the rectilinear line is perpendicular to the slope of the line of minimum distance. That is, the slope of the tangent to the fish-eye line at the point of minimum distance to the principal point is equal to the slope of the equivalent rectilinear line. This is demonstrated in Fig. 3.

Additionally, in the region of the principal point, the fish-eye displacement curve can be considered to be linear. Thus, the area around the principal point approximates a rectilinear projection. Therefore, the point on the line in the fish-eye image that is nearest the principal point can be considered coincident with the equivalent point on the line in the rectilinear image. Even if this is not an exactly precise assumption, the result will be a scaled version of the true fish-eye displacement curve, i.e., the characteristics of the curve will remain the same. Thus, taking the two lines nearest the principal point in the fish-eye image, the equivalent rectilinear lines can be constructed from the extracted slopes and points of intersection.

Figures 5(a) and 5(b) show the constructed lines for the horizontal and vertical line sets. The two putative vanishing points can be determined as the intersecting points of these lines. Figure 4(a) shows the determination of the putative vanishing points.

C. Nonlinear Optimization of Vanishing Points

The slopes of the remainder of the lines can be determined in two ways. First, the set of slopes m_i , where $i = 1 \dots n$ is the number of lines, can be determined as

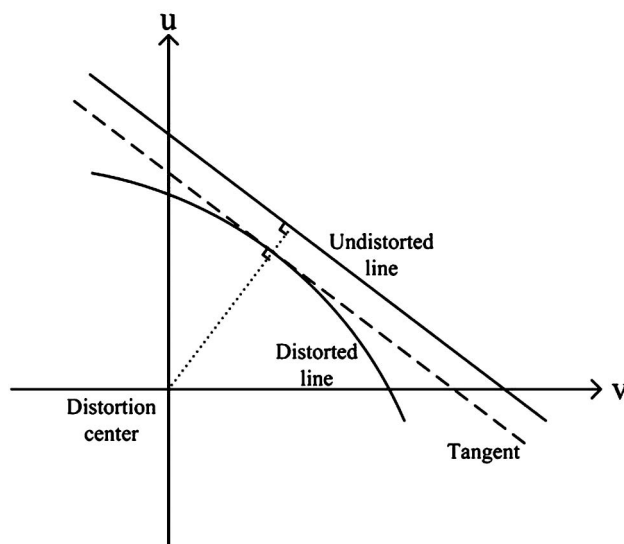


Fig. 3. Point of minimum distance between the distortion center and the distorted line lies on the line perpendicular to the undistorted line through the distortion center.

described in the previous section: the slopes are equal to the slopes of the tangent to the radially displaced line at the point of minimum distance to the principal point. Second, we define the ground line as being parallel to the horizon line and intersecting the principal point. The set of parallel lines will intersect the ground line at equal distances, as shown in Fig. 4(b). Thus, the second set of slopes s_i can be determined.

Theoretically, s_n and m_n should be equal. However, in the presence of noise, there will be deviations that should be minimized. Thus, to achieve the result with least error, the vanishing point must be chosen such that the following error function ξ is minimized, where s and m are given in terms of angles against the x axis in the range $[-\pi/2, \pi/2]$:

$$\xi = \sum_{i=1}^n |\Delta s_i|, \quad (23)$$

where Δs is the acute angle formed by the two lines described by the slopes s and m :

$$\Delta s = \begin{cases} m - s, & |m - s| < \pi/2 \\ \pi - (m - s), & |m - s| > \pi/2 \end{cases} \quad (24)$$

To achieve this, a nonlinear optimization algorithm can be used, such as Levenberg–Marquardt [20]. Thus, the vanishing points are chosen such that the error between the calculations of the slopes using the two methods is minimized. This is repeated for

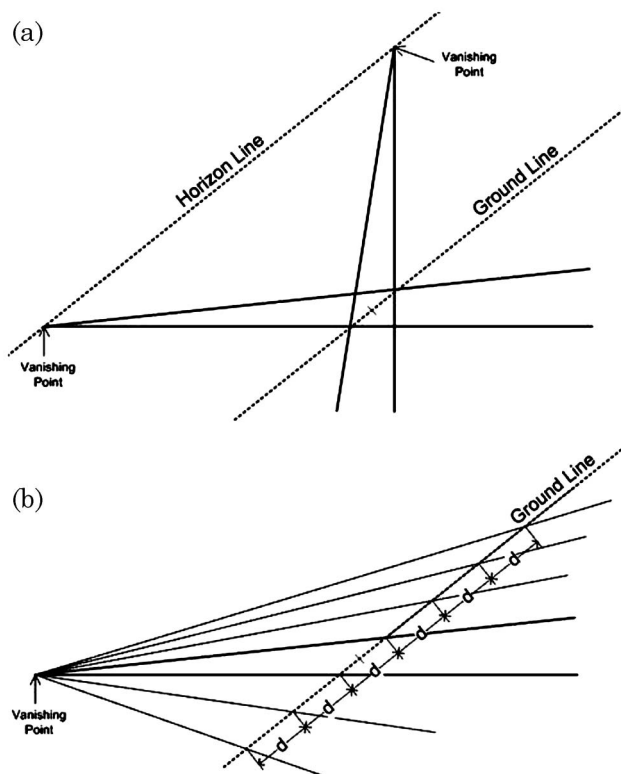


Fig. 4. Two-point perspective: (a) shows how to find the vanishing points, horizon line, and ground line (which is parallel to the horizon line) and (b) shows how parallel lines in 3D space converge at a single point in perspective and cross the ground line at equal distances (marked as d in the figure).

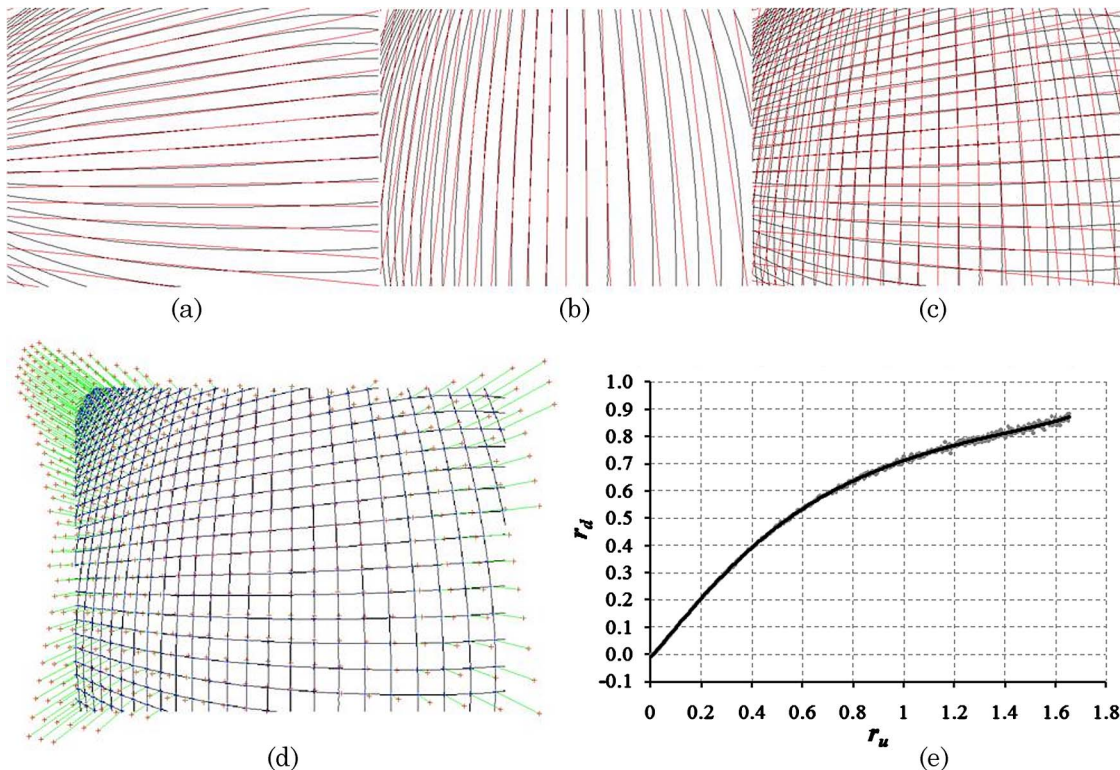


Fig. 5. (Color online) Curves in undistorted space overlaid on the corresponding curves in distorted space for the (a) horizontal lines, (b) vertical lines, and (c) both sets of lines. (d) Shows the corners in the distorted space connected to the corners in the undistorted space and (e) shows the extracted points with locally weighted scatterplot smoothing (LOESS) applied.

both the horizontal and vertical sets of lines, and the two vanishing points are estimated.

D. Estimating Radial Displacement Curve

For each line in the fish-eye image, the equivalent rectilinear line can be recreated, given the vanishing points and the slopes [Figs. 5(a)–5(c)]. From the lines in the fish-eye space, a set of corners P_d can be extracted, using a suitable corner detection method, e.g., [19]. From the set of lines in rectilinear space, a corresponding set of corners P_u can also be extracted. Thus, each corner in P_d has a corresponding corner in P_u , as demonstrated in Fig. 5(d). Each corner in P_u has a rectilinear radial distance r_u , and each corner in P_d has a displaced radial distance r_d , and so a radial displacement curve can be created [Fig. 5(e)]. To improve accuracy, multiple images of the calibration diagram can be used. For the results presented in this paper, a set of ten images was used for each camera.

Additionally, to reduce error in the feature extractions, the resultant curve is smoothed using a quadratic locally weighted scatterplot smoothing (LOESS) algorithm [21]. This regression is used, as it does not assume any underlying function model of the data. If the error were not reduced, the results would include the error in the corner extraction, instead of the result ideally being just the error in the fit of the given model to the extracted distortion curve (Section 4). The corner error reduction is not critical for a comparative examination of the models to a particular fish-eye distortion curve, as the portion of the RMSE introduced by the corner error will remain constant for each of the model fits. However, corner error reduction will result in truer RMSE results for each of the model fits. A span of 20% was used for the LOESS smoothing for the results presented in this paper, as it seemed to remove the majority of the noise without distorting the shape of the underlying distortion curve [Fig. 5(e)].

Finally, the radial displacement curve is normalized on both axes by dividing by the maximum image sensor radial distance from the principal point in pixels, to ensure that the curve is independent of image sensor sample resolution.

Table 1. List of Tested Cameras

	Make and Model	FOV
Camera 1	Micron MI-0343 Evaluation Module	170°
Camera 2	OmniVision OV7710 Evaluation Module	178°
Camera 3	OmniVision OV7710 Evaluation Module	170°
Camera 4	Sony DSC C3	103°
Camera 5	Minolta DiIMAGE 7	90°

4. Results

Typically, fish-eye lenses are constructed with the aim of complying with the equidistant and equisolid projection function, and more rarely the orthographic function. The stereographic projection function in lenses is very uncommon, though in the guise of the division model, it has gained popularity in recent literature. In this section we examine the accuracy of the fish-eye lens models over a set of fish-eye cameras. To examine the accuracy for a range of cameras, the radial distortion curve for each camera considered is extracted using a standard checkerboard calibration diagram, as described in the previous section. Then, each fish-eye lens model is fitted to each of the extracted curves, using the Levenberg–Marquardt nonlinear least-mean-squares fit algorithm [20]. The fitting of the model functions to the radial displacement curve is completed using the MATLAB cftool function (MathWorks MATLAB R2009a, Curve Fitting Toolbox). Because of the fact that speed of operation was not a consideration in this implementation, the maximum number of functional evaluations was set to 60,000, the maximum number of iterations to 40,000, and the minimum change of error to 10^{-12} . For each of the models, an appropriate starting point was chosen. For example, for each of the lenses, an approximation of the FOV was known. For the equidistant model, the parameter f could be determined as $f = (\text{FOV}/2)^{-1}$, and the additional radial distortion parameters were initially set to zero.

The RMSE is used to determine the accuracy of the fit. The models and FOVs of the five cameras used are listed in Table 1 (for convenience, the cameras are

Table 2. RMSE of the Functions Fitted to the Distortion Curves Extracted from Each of the Cameras^a

$\times 10^{-3}$	Camera 1	Camera 2	Camera 3	Camera 4	Camera 5
Equidistant	9.156	6.992	9.908	10.202	8.237
+distortion parameters	2.550	4.991	2.827	3.852	3.382
Equisolid	12.764	8.301	13.997	12.811	9.704
+distortion parameters	2.542	4.843	2.818	3.852	3.382
Orthographic	15.884	14.546	17.649	14.443	10.221
+distortion parameters	2.561	5.306	2.858	3.852	3.382
Stereographic	10.107	8.624	10.301	11.201	9.176
+distortion parameters	2.529	4.393	2.775	3.852	3.382
PFET (fifth order)	2.669	6.842	2.921	4.070	3.631
FET	19.465	33.211	21.978	15.528	12.394
+distortion parameters	3.369	8.870	3.571	4.311	3.959
FOV model	21.341	32.496	18.689	13.996	11.348
+distortion parameters	3.554	7.985	3.828	4.455	3.982

^aThe values are in terms of the maximum image sensor radial distance from the principal point in pixels, to ensure that the data presented are independent of image sensor sample resolution.

Table 3. RMSE of Fits to Camera 2 for PFET of Various Orders^a

Order	Third	Fourth	Fifth	Sixth	Seventh
RMSE $\times 10^{-3}$	11.801	11.671	6.842	4.909	2.328

^aThe values are in terms of the maximum image sensor radial distance from the principal point in pixels, to ensure that the data presented is independent of image sensor sample resolution.

referred to as camera 1 to camera 5), while the corresponding errors are detailed in Tables 2 to 5 and Figs. 6 and 7, which are now discussed in detail. The error of such fits can then be used as a measure by which comparisons in terms of accuracy can be made.

It can be seen from Table 2 that the basic equidistant model returns the lowest error of the basic models (i.e., without additional parameters), with the exception of the PFET, for the set of cameras examined. With the additional radial distortion parameters, the error in cameras 1, 3, 4, and 5 are similar for all models, as the additional parameters compensate for any error in the basic model. The basic FET and the FOV models consistently return higher errors than the other models. This is particularly evident with camera 2, which is a strong fish-eye camera, where the errors are significantly higher than the other functions. The fifth-order PFET returns the lowest error for all but the highest FOV cameras,

Table 4. RMSE of Fits to Camera 5 for Equidistant Model with Additional Radial Distortion Parameters of Various Orders

Order	Third	Fifth	Seventh	Ninth	Eleventh
RMSE $\times 10^{-3}$	5.984	4.653	3.382	3.313	3.299

which is expected, as it is the model with the greatest number of parameters.

With camera 2, the error results suggest that allowing the displacement functions to operate as the first-order parameter returns a smaller error than the fifth-order PFET. That is, for this fish-eye camera, a better result can be obtained by first using the displacement functions and then modeling the difference using the additional radial distortion parameters. Conversely, for cameras with lower FOVs, the best results seem to be obtained by using the polynomial model. Naturally, however, adding more parameters to the PFET reduces the error, as shown in Table 3. Table 4 shows that there is a minimal decrease in the returned error if additional radial distortion parameters beyond the seventh order are considered. If an image sensor sample pixel resolution of 640×480 is assumed (and thus a maximum radial distance of 400 pixels), the maximum error for the fits to each of the cameras in terms of pixel error is shown in Table 5.

Table 5. Maximum Error of Various Models Fitted to Various Cameras, in Terms of Pixels (Assuming an Image Sensor Pixel Sample Resolution of 640×480)

	Camera 1	Camera 2	Camera 3	Camera 4	Camera 5
Equidistant	3.7496	2.81	3.9892	4.1284	3.3352
+distortion parameters	1.0204	1.9948	1.13	1.5412	1.3528
Equisolid	5.1244	3.3376	5.6216	5.1328	3.9044
+distortion parameters	1.018	1.9388	1.1248	1.5412	1.3528
Orthographic	6.3728	5.8852	7.0952	5.7848	4.1444
+distortion parameters	1.0124	2.1228	1.1424	1.5412	1.3528
Stereographic	4.0532	3.4752	4.1372	4.4852	3.6804
+distortion parameters	1.0124	1.7588	1.11	1.5412	1.3528
PFET (fifth order)	1.0684	2.736	1.1696	1.6268	1.4532
FET	7.8044	13.3568	8.8452	6.2324	4.9648
+distortion parameters	1.3504	3.5448	1.4308	1.726	1.582
FOV model	8.5656	12.9736	7.452	5.5804	4.5532
+distortion parameters	1.4204	3.1948	1.53	1.7812	1.5928

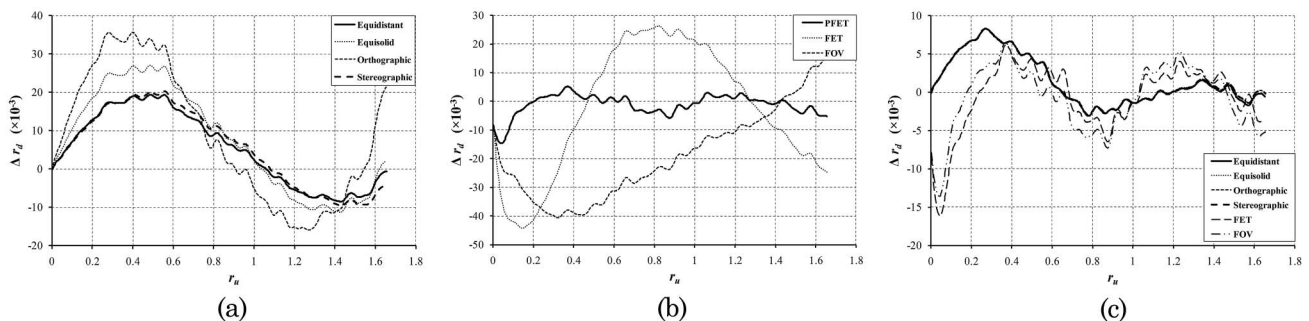


Fig. 6. Residuals after the fitting of each of the models to camera 1: (a) equidistant, equisolid, orthographic, and stereographic projection functions, (b) PFET, FET, and FOV models, and (c) all of the models with the additional radial distortion parameters included. In (c), the projection functions with the additional parameters are almost coincident. Note the change in scales between the graphs.

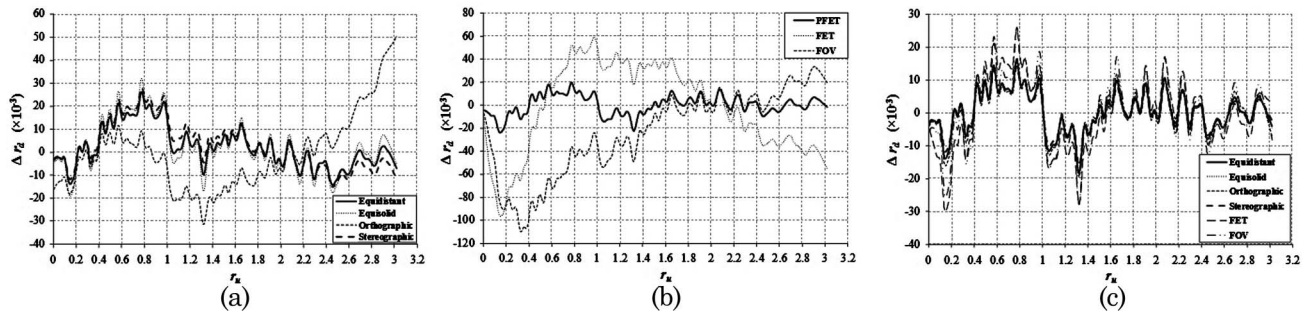


Fig. 7. Residuals after the fitting of each of the models to camera 2: (a) equidistant, equisolid, orthographic, and stereographic projection functions, (b) PFET, FET, and FOV models, and (c) all of the models with the additional radial distortion parameters included. In (c), the projection functions with the additional parameters are almost coincident. Note the change in scales between the graphs.

Figures 6 and 7 show the residuals of the various model fits for camera one and camera two, respectively. For camera 1, the residuals of the projection functions (equidistant, equisolid, orthographic, and stereographic) show similar form, but different amplitudes. In correlation with the results in Table 2, the equidistant and stereographic functions display the lowest amplitude in the residuals. The PFET, FET, and FOV models show very different residual forms, with the PFET being closest to zero. Finally, the functions that include the additional radial distortion parameters are shown in Fig. 6(c). The equidistant, equisolid, and stereographic errors with the additional parameters are indiscernible from one another (and are under the trace of the equidistant model), while the orthographic plus additional parameters error shows a slight difference from the other projection function errors at high values of r_u . The FOV and FET model errors with the additional parameters are significantly different from the other projection functions.

In all of the graphs in Fig. 6, the noise is of similar amplitude and location. This indicates that the source of the noise is in the extracted radial displacement curve and, thus, the corner extraction described in Section 3 and is independent of the model that is fitted. The fact that the amplitude of the noise is considerably smaller than the overall residual amplitude also indicates that the residuals for the model fits are primarily based on the adequacy of the model rather than noise in the corner extraction.

Figure 7 shows the residuals of the model fits to camera 2. Again, in Fig. 7(c), the equidistant, equisolid, stereographic, and orthographic projection functions are practically coincident and are all graphed under the trace of the equidistant function. The FOV and FET model errors follow a similar form, but are discernible from the others. The results show that noise is a larger contributing factor to the residual than the previous camera. The increase in noise is likely due to the fact that, with increasing radial lens displacement, the magnitude and shape of the corners in the calibration image will vary significantly. Additionally, in the extremities of the calibration image, there will be a reduced spatial resolution. These factors mean that the corners are extracted with greater error, which is reflected in the noise

in the graphs of the residuals. However, even though the noise is a considerable factor for the residuals in camera 2, the dominant factor is still the adequacy of the models.

5. Conclusions

In this paper, we have examined several fish-eye lens models and have compared each model with the extracted radial displacement curves of several different FOV cameras. The comparison was based upon quantitative differences between the models. While the choice of model is dependent upon the implementation constraints and the requirements for the final output image, this paper has discussed ways in which this choice can be objectively made.

To achieve this comparison, we presented a non-parametric, model-independent method by which the radial displacement curve of a given camera can be extracted. This method was based only on some well-established assumptions about the shape and form of radial displacement curves. We used this curve as the basis of a metric to compare the various radial lens models.

Depending on the manufacturing process, some particularly inexpensive lens elements can cause a deviation in their displacement curve from the mapping function they were designed to adhere to. To model this deviation, additional polynomial distortion parameters can be added to the radial displacement functions. The results presented here show a significant improvement when these parameters are used as described in Section 2.C, with up to a fivefold decrease in the returned error over the standard fish-eye functions.

This paper has examined the various fish-eye models, using a heuristic optimization procedure to determine the parameters of the models from the extracted radial displacement curves. It has not examined the numerous calibration methods that exist to determine the parameters of the models, though it is possible the method presented within this paper could be extended to provide an objective comparison of fish-eye calibration methods.

A potential issue with the proposed method lies in the use of the Sobel operator to extract edges in the fish-eye image, which displays variable spatial resolution across the image. In particular, this will cause

the biggest issue in the periphery of the image, which is arguably the most important region for describing the fish-eye function of a given camera. This was evident in the results for camera 2 shown in Fig. 7. Future work would address this issue; perhaps via the adaptation of a scale-invariant feature transform [22] for this particular application (which may also overcome orientation and approximate affine transforms of features in these regions).

This research is funded by Enterprise Ireland and Valeo Vision Systems (formerly Connaught Electronics Limited) under the Enterprise Ireland Innovation Partnerships Scheme (grant IP/2004/0244).

References

1. D. C. Brown, "Decentering distortion of lenses," *Photograph. Eng.* **32**, 444–462 (1966).
2. R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE Trans. Robot. Automat.* **3**, 323–344 (1987).
3. Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 1330–1334 (2000).
4. A. Basu and S. Licardie, "Alternative models for fish-eye lenses," *Pattern Recogn. Lett.* **16**, 433–441 (1995).
5. F. Devernay and O. Faugeras, "Straight lines have to be straight: automatic calibration and removal of distortion from scenes of structured environments," *Mach. Vis. Appl.* **13**, 14–24 (2001).
6. C. Bräuer-Burchardt and K. Voss, "A new algorithm to correct fish-eye- and strong wide-angle-lens-distortion from single images," in *Proceedings of the IEEE International Conference on Image Processing* (IEEE, 2001), pp. 225–228.
7. A. W. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2001), pp. 125–132.
8. D. Schneider, E. Schwalbe, and H.-G. Maas, "Validation of geometric models for fisheye lenses," *ISPRS J. Photogramm. Remote Sens.* **64**, 259–266 (2009).
9. C. Hughes, R. McFeely, P. Denny, M. Glavin, and E. Jones, "Equidistant ($f\theta$) fish-eye perspective with application in distortion centre estimation," *Image Vis. Comput.* **28**, 538–551 (2010).
10. R. I. Hartley and S. B. Kang, "Parameter-free radial distortion correction with center of distortion estimation," *IEEE Trans. Pattern Anal. Mach. Intell.* **29**, 1309–1321 (2007).
11. K. V. Asari, "Design of an efficient VLSI architecture for non-linear spatial warping of wide-angle camera images," *J. Syst. Architect.* **50**, 743–755 (2004).
12. J. Weng, P. Cohen, and M. Herniou, "Camera calibration with distortion models and accuracy evaluation," *IEEE Trans. Pattern Anal. Mach. Intell.* **14**, 965–980 (1992).
13. G. P. Stein, "Internal camera calibration using rotation and geometric shapes," M.S. thesis (MIT, 1993).
14. R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. (Cambridge U. Press, 2004).
15. G. Xu, J. Terai, and H.-Y. Shum, "A linear algorithm for camera self-calibration, motion and structure recovery for multi-planar scenes from two perspective images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2000), pp. 474–479.
16. K. Miyamoto, "Fish eye lens," *J. Opt. Soc. Am.* **54**, 1060–1061 (1964).
17. S. Shah and J. K. Aggarwal, "Intrinsic parameter calibration procedure for a (high-distortion) fish-eye lens camera with distortion model and accuracy estimation," *Pattern Recogn.* **29**, 1775–1788 (1996).
18. B. B. Jähne, *Digital Image Processing*, 5th ed. (Springer-Verlag, 2002), Chap. 12.
19. Z. Wang, W. Wu, X. Xu, and D. Xue, "Recognition and location of the internal corners of planar checkerboard calibration pattern image," *Appl. Math. Comput.* **185**, 894–906 (2007).
20. D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *SIAM J. Appl. Math.* **11**, 431–441 (1963).
21. W. S. Cleveland and S. J. Devlin, "Locally weighted regression: an approach to regression analysis by local fitting," *J. Am. Stat. Assoc.* **83**, 596–610 (1988).
22. D. G. Lowe, "Object recognition from local scale-invariant features," *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, 1999), pp. 1150–1157.